COMPARACIÓN DEL DESEMPEÑO DE UNA METAHEURÍSTICA BASADA EN BÚSQUEDA ARMÓNICA Y EL ALGORITMO GENÉTICO ADAPTATIVO PARA EL ANÁLISIS DE PATRONES DE DISPERSIÓN

Trabajo de grado presentado como requisito para optar al título de:
INGENIERO FÍSICO

DIEGO FELIPE RAMÍREZ CHÁVEZ
STIBEL ALEJANDRO CAMAYO MUÑOZ

Director:
DIEGO FERNANDO CORAL CORAL

Codirector:
CARLOS ALBERTO COBOS LOZADA

UNIVERSIDAD DEL CAUCA
FACULTAD DE CIENCIAS NATURALES, EXACTAS Y DE LA EDUCACIÓN
DEPARTAMENTO DE FÍSICA.
INGENIERÍA FÍSICA
POPAYÁN
2024

COMPARACIÓN DEL DESEMPEÑO DE UNA METAHEURÍSTICA BASADA EN BÚSQUEDA ARMÓNICA Y EL ALGORITMO GENÉTICO ADAPTATIVO PARA EL ANÁLISIS DE PATRONES DE DISPERSIÓN

DIEGO FELIPE RAMÍREZ CHÁVEZ
STIBEL ALEJANDRO CAMAYO MUÑOZ

Tesis de Trabajo de Grado presentada a la Facultad de Ciencias Naturales, Exactas y de la Educación de la Universidad del Cauca para la obtención del Título de Ingeniero Físico

Director:
Dr. Diego Fernando Coral Coral

Codirector:
Dr. Carlos Alberto Cobos Lozada

Popayán, 2024.

Nota de Aceptación

_____

_____

_____

_____

_____
Dr. Diego Fernando Coral Coral
Director

_____
Dr. Carlos Alberto Cobos Lozada
Codirector

_____
Dra. Alejandra Isabel Guerrero Duymovic
Jurado

_____
Dr. Camilo Sánchez Ferreira
Jurado

Fecha de sustentación: Popayán, Junio 12 de 2024.

Agradecimientos

A la Universidad del Cauca y a la educación pública.

A los profesores Diego Fernando Coral y Carlos Alberto Cobos por su indispensable paciencia, disposición, asesoría y enseñanza constante, que nos permitieron transformar una idea en este trabajo de investigación.

A las profesoras y profesores del Departamento de Física.

Al grupo de investigación de Ciencia y Tecnología de Materiales Cerámicos CYTEMAC y el grupo de investigación y desarrollo en Tecnologías de la Información GTI de la Universidad del Cauca.

A cada uno de nuestros compañeros y amigos que hicimos a lo largo de la carrera y del trabajo de grado.

Diego Felipe Ramírez Chávez

Quiero agradecer a mi familia y a mi novia Jennifer por todo su amor y apoyo; y por siempre cuidarme.

Agradezco en particular a mí mamá Luz Mary que con su amor y paciencia nunca ha dejado de creer en mis capacidades y potencial, y me ha empujado a ser la persona que soy ahora.

A mi compañero de investigación y amigo Stibel, por su compromiso para con este trabajo de grado.

A mis amigos del colegio.

Stibel Alejandro Camayo Muñoz

Este logro está dedicado a mi familia, en especial a mi madre Martha Muñoz y mi padre Fernando Camayo, por su constante esfuerzo, apoyo incondicional y amor incansable a lo largo de todo este proceso. Su aliento y sabiduría han sido fundamentales para llegar hasta aquí.

A mi colega y amigo Diego, por su valeroso esfuerzo, compañerismo y dedicación que hoy nos ha permitido culminar con éxito este trabajo de grado

Gracias.

Tabla de Contenido

Lista de Tablas

Lista de Figuras

I. Resumen

La caracterización de materiales es un área de investigación de suma importancia en la actualidad, muchas de las técnicas de caracterización se apoyan de análisis computacional y hacen uso de algoritmos de optimización que buscan hacer un mejor uso de los recursos computaciones que les permita obtener mejores resultados en tiempos razonables. Mejoras en estos algoritmos se puede ver reflejadas en caracterizaciones más eficientes. Por lo anterior esta investigación se centró en modificar el algoritmo de optimización utilizado por la herramienta Computational Reverse-Engineering Analysis for Scattering Experiments (CREASE), en busca de mejorar su desempeño; esta herramienta hace un análisis morfológico de muestras a partir de perfiles de dispersión de ángulo pequeño (SAS) utilizando un modelo de Rigid-Body y el algoritmo genético (GA) de adaptación dinámica como su algoritmo de optimización.

El objetivo fue evaluar el desempeño de CREASE al sustituir el GA por una metaheurística basada en búsqueda armónica (Harmony Search, HS), específicamente el Nobel Global Harmony Search (NGHS), en el análisis de perfiles SAS de soluciones de baja concentración vesículas ensambladas a partir de polímeros anfifílicos. Se eligió NGHS tras evaluar cinco metaheurísticas basadas en HS, destacando NGHS por su velocidad de convergencia y bajo error (SSE) alcanzado. Se realizaron ajustes de sus hiperparámetros y se evaluaron estrategias para prevenir su convergencia prematura.

Los resultados mostraron que NGHS mantuvo la misma precisión que el GA, pero con mayor eficiencia, logrando soluciones de calidad similar con solo una sexta, y en algunos casos una décima parte, del número de evaluaciones del GA. Permitiendo con el CREASE realizar análisis de perfiles SAS con menos de la mitad de las evaluaciones, ahorrando recursos computacionales y facilitando análisis más exhaustivos.

Además, NGHS abordó algunas falencias del proceso de optimización del GA y, con solo tres hiperparámetros frente a los once del GA, facilita su uso y adaptación a diferentes tipos de muestras para usuarios con poca experiencia en optimización.

II. Introducción

Las técnicas Small Angle Scattering (SAS) son técnicas de análisis estándar bien establecidas, útiles para estudiar la estructura de la materia y sus interacciones; son especialmente utilizadas para el análisis de estructuras no periódicas de tamaño coloidal, con escala desde unos 10 Å hasta varios miles de Å [1], [2]. SAS permite obtener información sobre la muestra analizada, específicamente su morfología, dimensiones y estado de agregación o empaquetamiento, cuyas aplicaciones son de importancia en diversas ramas de la ciencia y la tecnología, como la física de la materia condensada, la biología molecular, la biofísica, la ciencia de los polímeros y la metalurgia [3], [4], [5], [6], [7], [8]. Dicha información sobre la muestra debe ser extraída a partir del patrón de dispersión observado, para lo cual se usan modelos fisicomatemáticos establecidos y metodologías apoyadas en procesamiento computacional [9], [10], [11].

En ciencia de los materiales, las técnicas SAS más usadas son Small angle X-ray scattering (SAXS) y Small Angle Neutron Scattering (SANS). Las técnicas SAS se basan en el principio físico de dispersión de ondas por la materia donde un haz primario de radiación de rayos X (SAXS) o neutrones (SANS) se hace incidir sobre el objeto de estudio, el cual dispersa de forma elástica la radiación produciendo un patrón de dispersión bidimensional, patrón SAS de ahora en adelante, que puede medirse usando cámaras CCD o gas electron multiplier detectors [4], y que representa la intensidad dispersada (I) en función del vector de onda ($\boldsymbol{Q}$). Es usual obtener perfiles SAS 1D, perfiles SAS de ahora en adelante, a partir de promediar el patrón SAS; esto permite mejorar la calidad estadística de los datos y en algunos casos facilita el análisis de los resultados [12], por ejemplo, en el caso de muestras anisótropa la información morfológica de su patrón SAS puede representarse completamente en un perfil SAS, que representa la intensidad dispersada (I) en función de la magnitud vector de onda ($Q = \|\boldsymbol{Q}\|$).

El método 'Computational Reverse-Engineering Analysis for Scattering Experiments' (CREASE) [13] es una herramienta desarrollada recientemente por el laboratorio de investigación de la Profesora Arthi Jayaraman en la Universidad de Delaware, útil para analizar perfiles SAS, que usa la metodología de mínimos cuadrados ponderados para obtener información estructural clave a partir de uno o varios perfiles SAXS y/o SANS de una muestra, $I_{exp}(Q)$, e incluso hacer una estructura 3D representativa de la misma, útil para análisis adicionales [14]. CREASE tiene capacidad para analizar distintos tipos de muestras, como soluciones y mezclas de nanopartículas de baja y alta concentración, y es fácilmente adaptable para nuevos tipos de muestras [3], [10], [13]. Recientemente desarrollaron una versión del CREASE, conocida como CREASE-2D, capaz de analizar patrones SAS [15], útil para el análisis de muestra isotrópas. En este estudio se trabajó sobre el CREASE.

El modelo que usa CREASE es basado en el modelado Rigid-Body que consiste en hacer una reconstrucción 3D de la muestra o parte de esta, modelándola como una conformación espacial de subunidades, los Rigid-Bodies, reconstrucción a partir de la cual calcula su perfil SAS $I_{comp}(Q)$ y lo compara con el perfil SAS experimental de la muestra $I_{\exp}(Q)$; este enfoque tiene la particularidad de tener un coste computacional considerable ya que el cálculo de $I_{comp}(Q)$ requiere de considerables operaciones aritméticas aunque en algunos casos se ha logrado sustituir exitosamente con el uso de modelos de Machine Learning (ML) [3], [16]. La reconstrucción 3D se va modificando de forma que $I_{comp}(Q)$ se acerque lo más posible a $I_{exp}(Q)$. A diferencia de otras metodologías que usan el modelado Rigid Body que trabajan sobre la posición de cada una de las subunidades como ad-initio [9] o Montecarlo Inverso [17], CREASE utiliza parámetros estructurales a partir de los cuales ubica las subunidades logrando bajar la dimensionalidad del problema considerablemente, por ejemplo, como se ve en la Figura 6, para el caso de una solución de baja concentración de vesículas ensambladas a partir de polímeros anfifílicos algunos de sus parámetros podrían ser el radio de núcleo, grosor de sus capas y la dispersión de estas dimensiones [10]. En la práctica la herramienta CREASE ha demostrado desempeñarse muy bien para el análisis de perfiles SAS [3], [10], [14].

Para la búsqueda de los parámetros estructurales, que consiste básicamente de un proceso de optimización en el que se busca minimizar la diferencia entre $I_{comp}(Q)$ e $I_{exp}(Q)$, el CREASE hace uso del algoritmo genético (Genetic Algorithm, GA) de adaptación dinámica [10], [18]. Esta metaheurística es un tipo de algoritmo de optimización basado en población particularmente en la teoría evolución de Darwin; en este enfoque, una población de soluciones candidatas evoluciona a lo largo de varias generaciones para encontrar la mejor solución posible [10], [19]. Este GA trabaja sobre una codificación binaria de los parámetros estructurales, en el que el número de bits usados para cada parámetro es el

mismo y lo determina el usuario. La cantidad de hipeparámetros del GA que el usuario debe establecer para el análisis es de 10.

Esta etapa de optimización interna es una fase crítica en el rendimiento del CREASE ya que determina la cantidad de $I_{comp}(Q)$ calculados necesarios para encontrar los parámetros estructurales óptimos, por lo que mejorar dicha optimización permitiría disminuir dicha cantidad acelerando el análisis y disminuyendo su coste computacional. Así, el equipo del CREASE ha implementado una versión en la que apoyan el proceso de optimización del GA con el uso de redes neuronales artificiales en el análisis de bloque de polímeros anfifílicos, logrando en algunos casos una mejora en la velocidad de convergencia y en otros de la precisión [16].

La selección de la metaheurística apropiada para un problema específico es una parte crucial del diseño de los algoritmos. Esto implica un proceso de experimentación y ajuste de la metaheurística seleccionada que conduzca a soluciones óptimas. Con esto, la búsqueda de la metaheurística adecuada que mejore algoritmos o herramientas, tanto en eficiencia como en precisión, continúa siendo un área de investigación importante en el campo de la ciencia y la ingeniería. Ejemplo de ello son los múltiples estudios recientes que han logrado mejorar procesos científicos e ingenieriles cambiando su enfoque de optimización [20], [21], [22], [23]. En estos estudios en particular se ha evaluado el desempeño del GA contra la metaheurística búsqueda armónica (Harmony Search, HS) en diversos casos, revelando un desempeño superior del HS sobre el GA tanto en términos de tiempo como de precisión en la búsqueda de soluciones, posicionando al HS como una potencial alternativa al GA en herramientas científicas e ingenieriles.

En la referencia [24], se compara el rendimiento del GA con el HS, sobre funciones unimodales (funciones con un único óptimo) y multimodales (funciones con múltiples óptimos locales), revelando una destacada superioridad del HS sobre el GA en todas las funciones de prueba usadas para la investigación. Vale la pena aclarar que el desempeño de una metaheurística debe ser evaluado en cada problema de forma específica, pues no es posible asegurar a priori su correcto desempeño en un problema en particular, como establece el teorema de *No Free Lunch* (NFL) [25].

El HS es un tipo de algoritmo de optimización basado en población, inspirado en el proceso de improvisación musical, un concepto simple, resultando en una fácil implementación, pocos hiperparámetros, y fácil integración a otras metaheurísticas [26], [27]. Cabe resaltar que, así como el GA, el HS cuenta con múltiples variantes, entre las más conocidas, se encuentran: Improve Harmony Search (IHS), Global-Best Harmony Search (GHS), Novel Global-best Harmony Search (NGHS), Self-adaptative Global-best Harmony Search (SGHS) [26], [27].

II.1. Definición del problema

El uso de la herramienta CREASE para el análisis de perfiles SAS requiere considerable tiempo de cómputo, lo que puede limitar su alcance especialmente en entornos que no cuenten con recursos computacionales superlativos. Además, los resultados reportados en su uso para el análisis de soluciones de baja concentración de vesículas ensambladas a partir de polímeros anfifílicos presentan una varianza significativa entre los resultados de analizar el mismo perfil SAS múltiples veces [10].

Una fase crítica en el rendimiento del CREASE reside en la etapa de optimización interna. Como se mencionó previamente, para dicha tarea hace uso del GA, que se encuentra implementado en su versión binaria y con hiperparámetros adaptables. Varios estudios [19], [20], [21], [22], [23], [24], han comparado el rendimiento del GA y sus variantes contra el HS en diversos espacios de búsqueda; revelando un mejor desempeño para el HS y limitaciones del GA en la resolución de problemas no lineales sin restricciones, con variables de diseño continuas, como es el caso del espacio de búsqueda de CREASE.

Por lo anterior en este trabajo se plantea la siguiente pregunta problema:

¿Cuál es el impacto en el desempeño (eficiencia y precisión) de la herramienta CREASE al sustituir el algoritmo genético (GA) de adaptación dinámica por una metaheurística basada en búsqueda armónica (HS), en el análisis de perfiles de dispersión de bajos ángulos (SAS) de sistemas nanoparticulados de baja concentración?

Esta investigación se desarrolló sobre el sistema nanoparticulado de soluciones de baja concentración de vesículas ensambladas a partir de polímeros anfifílicos, ya que es la *shape* más ampliamente documentada en el repositorio del CREASE.

II.2. Objetivos

II.2.A. Objetivo General

Evaluar el desempeño (eficiencia y precisión) de la herramienta "Computational Reverse Engineering Analysis for Scattering Experiments" CREASE al sustituir el algoritmo genético (GA) de adaptación dinámica por una metaheurística basada en búsqueda armónica (HS) en el análisis de perfiles de dispersión de bajos ángulos (SAS) de sistemas nanoparticulados de baja concentración.

II.2.B. Objetivos Específicos

1. Proponer una variante de HS para abordar el problema de optimización de la herramienta CREASE.
2. Adaptar el CREASE para usar la variante propuesta de HS y su uso en un contexto donde se pueda almacenar datos del proceso de optimización para su posterior análisis.
3. Comparar el rendimiento de la variante HS propuesta frente a GA en 4 benchmarks, usando test estadísticos no paramétricos.

Como resultado de esta investigación, se propuso la metaheurística NGHS basada en HS como una alternativa al GA en la herramienta CREASE. Al evaluar el desempeño del CREASE sustituyendo el GA por el NGHS, este mostró una precisión similar, pero con mayor eficiencia, logrando en promedio soluciones con menor error (SSE) utilizando solo una sexta parte de las evaluaciones, y en algunos casos hasta una doceava parte. Con esta metaheurística se consiguió que el CREASE realice el análisis de perfiles SAS con menos de la mitad de las evaluaciones requeridas por el GA, ahorrando tiempo de ejecución y permitiendo un uso más eficiente de los recursos computacionales para otros fines o análisis más exhaustivos. Además, NGHS aborda dos falencias detectadas en el GA y, con solo dos hiperparámetros frente a los diez del GA, facilita su uso para usuarios con poca experiencia en optimización, así como su adaptabilidad a la variedad de muestras que CREASE puede analizar.

Durante la realización de esta investigación, parte de los resultados fueron presentados en modalidad de póster en los congresos internacionales: "41st International Conference on Vacuum Ultraviolet and X-ray Physics" (VUVX 2023) y "31st International Materials Research Congress" (IMRC) realizados en Brasil y México, respectivamente. Además, se realizó una presentación oral en el "VII Congreso Nacional de Ingeniería Física y 2nd Applied Physics, Engineering & Innovation Conference" en Manizales, Colombia. Adicionalmente, se generó un artículo de investigación basado en este trabajo, el cual está actualmente sometido en la revista "International Journal of Industrial Engineering Computations" de ISSN 1923-2934, indexada en categoria A1 en Publindex.

El documento está estructurado de la siguiente manera: la Sección III contextualiza la investigación y proporciona conceptos clave, incluyendo las metaheurísticas, la técnica SAS, una descripción general de la metodología CREASE, con detalle en el modelo físico y el GA que usa, y presenta las variantes HS elegidas para el estudio. La Sección IV describe la metodología experimental y los recursos computacionales empleados. La Sección V presenta los resultados y analiza el rendimiento del método CREASE utilizando la metaheurística basada en HS en comparación con el enfoque GA en el análisis de perfiles SAS in silico de soluciones de baja concentración de vesículas ensambladas a partir de polímeros anfifílicos. Finalmente, la Sección VI ofrece las conclusiones de la investigación y sugiere posibles direcciones futuras.

III. Marco teórico

En este capítulo se verán algunos de los conceptos clave para el desarrollo y contextualización de la presente investigación.

III.1. Metaheurísticas

Las metaheurísticas son una clase poderosa y flexible de algoritmos de optimización útiles para abordar problemas complejos y variados donde otros métodos no son prácticos. Una metaheurística es una estrategia algorítmica general que guía el proceso de búsqueda para encontrar soluciones (cercanas a óptimas) a problemas de optimización complejos. A diferencia de las heurísticas, que son métodos específicos para localizar buenas soluciones sin garantizar su optimalidad, las metaheurísticas combinan múltiples procedimientos heurísticos utilizando una estrategia de alto nivel lo que les da una aplicabilidad general. Lo que facilita su uso en gran variedad de problemas de optimización con pocas modificaciones específicas para cada uno; pudiendo incluso utilizar información heurística específica del problema para mejorar la eficiencia y eficacia de la búsqueda [28], [29].

Las metaheurísticas buscan explorar eficientemente el espacio de búsqueda para encontrar soluciones que sean buenas o cercanas a óptimas sin la necesidad de probar todas las posibles soluciones (lo cual sería computacionalmente inviable en muchos casos) y evitar quedarse atrapadas en áreas confinadas. Las metaheurísticas avanzadas utilizan la experiencia de búsqueda, que se manifiesta en alguna forma de memoria, para guiar la búsqueda [28], [29].

En su uso es importante tener en cuenta que son algoritmos aproximados, lo que significa que no garantizan encontrar la solución óptima, sino una solución lo suficientemente buena. Además, son no deterministas, por lo que pueden producir diferentes soluciones en diferentes ejecuciones debido al uso de aleatoriedad [28], [29].

III.1.A. Conceptos clave

Algunos conceptos importantes en el campo de las metaheurísticas son [19], [27], [29].:

Función Objetivo: Representa la medida de calidad o valor que se busca optimizar (maximizar o minimizar). Por lo general, la función objetivo toma como entrada una solución candidata al problema y devuelve un valor que indica qué tan buena es esa solución en términos del objetivo deseado. En términos de su expresión matemática, la función objetivo puede ser clasificada como analítica o no analítica: una función objetivo no analítica es aquella cuya expresión matemática no es conocida o no puede ser evaluada de forma directa utilizando operaciones matemáticas convencionales, caso contrario a cuando es analítica.

Variable de Decisión: Se refiere a una cantidad o parámetro que puede ajustarse o controlarse para influir en la solución de un problema de optimización. Estas variables representan las opciones que el algoritmo puede modificar para encontrar soluciones óptimas o aceptables para el problema en cuestión.

Dimensionalidad del Problema: se refiere al número de variables de decisión o características que deben ser consideradas para describir una solución al problema de optimización, la complejidad del problema suele aumentar con su dimensionalidad.

Espacio de Búsqueda: También conocido como espacio de soluciones, es el conjunto de todas las posibles soluciones factibles para un problema dado. En el dominio en el que opera una metaheurística, cada solución en este espacio representa una configuración posible del problema, entre las que se busca las soluciones óptimas o cercanas a óptimas. este espació de búsqueda está limitado por el valor máximo x_{lj} y mínimo x_{uj} de cada variable de decisión j

Paisaje de búsqueda: El paisaje de búsqueda conocido también como LandScape es la representación visual o conceptual del valor de la función objetivo a lo largo del espacio de búsqueda. Algunas características usuales de los paisajes de búsqueda son los valles (mínimos locales), picos (máximos locales), mesetas y pendientes, y su estructura puede influir significativamente en la dificultad de encontrar el óptimo global.

Exploración: La exploración es el proceso de investigar nuevas áreas del espacio de búsqueda para descubrir soluciones potencialmente mejores y evitar quedarse atrapado en óptimos locales.

Explotación: La explotación es el proceso de refinar y mejorar las soluciones actuales mediante la búsqueda local intensiva alrededor de las soluciones de alta calidad encontradas.

Velocidad de Convergencia: La velocidad de convergencia es la rapidez, medida en tiempo o número de evaluaciones de la función objetivo (EFOs), con la que una metaheurística se acerca a un punto en el que no podrá mejorar más el valor de la función objetivo, esto idealmente porque se acerca a la solución óptima o aceptable a medida que progresa en su ejecución, pero en algunos casos porque el proceso de optimización se quedó atrapado en un óptimo local. Por lo anterior, aunque una velocidad de convergencia alta es deseable no siempre es adecuada.

Hiperparámetros: son los parámetros externos al algoritmo de optimización que deben ser configurados antes de la ejecución y que afectan su comportamiento y el rendimiento. A diferencia de los parámetros internos del algoritmo, que se ajustan durante el proceso de optimización, los hiperparámetros se mantienen constantes durante la ejecución y pueden influir en aspectos como la exploración y explotación del espacio de búsqueda, la intensificación de la búsqueda, la diversificación de las soluciones y la convergencia del algoritmo.

Benchmarks: son problemas específicos que se utilizan como estándar para medir el rendimiento de metaheurísticas; los benchmarks poseen características extrapolables a los espacios de búsqueda de problemas de optimización a los que se enfrentan los metaheurísticas. Un benchmark consiste en problemas de optimización bien definidos y estandarizados, junto con sus soluciones óptimas o estimadas. Los benchmarks pueden tener diversas naturalezas, como teórica, empírica (basados en datos reales o problemas del mundo real), sintética o específica de la aplicación; su elección depende de los objetivos de evaluación y la relevancia para el problema o dominio de aplicación en cuestión.

Precisión: Se refiere a la capacidad de un algoritmo para encontrar soluciones x que se aproximan a la solución óptima x_o del problema que se está resolviendo, está relacionada de la fiabilidad de las soluciones encontradas por el algoritmo. Existen diversas métricas para cuantificar la precisión la empleada una de ella es la Raíz Cuadrada de la Media Cuadrática de los Errores Relativos (Root Mean Square Relative Error, $RMSRE$), cuya expresión puede verse en la Ecuación 1, este representa la raíz cuadrada de la media de los errores relativos al cuadrado de cada variable de decisión respecto a su valor objetivo definido por el valor optimo del bechmark usado, proporcionando una medida que penaliza más los errores grandes.

$$RMSRE = \sqrt{\frac{1}{k}\sum_{j=1}^{k}\left(\frac{x_j - x_{jo}}{x_{jo}}\right)^2} \qquad (1)$$

En esta expresión, x_j representa el valor de la variable de decisión j de la solución encontrada (x) por el algoritmo, x_{jo} representa su valor correspondiente en la solución óptima (x_o) del problema (benchmark) y k representa la cantidad de variables de decisión del problema. Entre menor sea el valor de esta métrica mayor es la precisión del algoritmo.

Eficiencia: La eficiencia de una metaheurística se relación con la cantidad de recursos computacionales (tiempo y memoria) necesarios para encontrar una solución aceptable, muy usualmente relacionada con la velocidad de convergencia y la precisión.

Teoremas No Free Lunch (NFL): establecen que, en términos globales, no existe un algoritmo de optimización que sea superior a todos los demás para resolver todos los tipos de problemas. Es decir, cuando se considera el rendimiento promedio sobre todos los posibles problemas, todos los algoritmos de optimización en general tienen el mismo desempeño. Por lo tanto, la efectividad de un algoritmo de optimización está altamente influenciada por las características específicas del problema en cuestión. Esto aplica a las metaheurísticas y sus hiperparámetros [25].

III.1.B. Clasificación de metaheurísticas

Hay varias maneras de clasificar metaheurísticas, a continuación, algunas de las más usuales son [28], [29]:

Basadas en poblaciones vs. basadas en trayectorias: Las metaheurísticas basadas en poblaciones trabajan con un conjunto de soluciones, llamadas población, que evolucionan gradualmente, mientras que las basadas en trayectorias trabajan con una única solución que se modifica gradualmente. Ejemplos de metaheurísticas basadas en poblaciones

incluyen Algoritmos Genéticos (GA) y Optimización por Enjambre de Partículas (PSO), mientras que ejemplos de metaheurísticas basadas en trayectorias incluyen Recocido Simulado y Búsqueda Tabú.

Basadas en memoria vs. sin memoria: Las metaheurísticas basadas en memoria utilizan información de iteraciones previas, mientras que las metaheurísticas sin memoria no lo hacen. Ejemplos de metaheurísticas basadas en memoria incluyen Búsqueda Tabú y Algoritmos de Colonia de Hormigas, mientras que ejemplos de metaheurísticas sin memoria incluyen GA y Recocido Simulado.

Basadas en poblaciones continuas vs. discretas: Dependiendo de si las soluciones se representan con variables continuas o discretas. Los algoritmos genéticos fueron inicialmente pensados para desenvolverse en un espacio de búsqueda discreto, mientras que HS se basa en poblaciones continuas.

Basadas en inspiración biológica vs. no biológica: Algunas metaheurísticas se inspiran en procesos naturales, mientras que otras no. Algoritmos Genéticos y Optimización por Enjambre de Partículas (PSO) son ejemplos de metaheurísticas basadas en inspiración biológica, mientras que Búsqueda Tabú, Recocido Simulado y Harmony Search son ejemplos de metaheurísticas no biológicas.

III.1.C. Comparación de Metaheurísticas

En la comparación de metaheurísticas, es común evaluar su desempeño utilizando benchmarks. A partir de los resultados obtenidos al ejecutar las metaheurísticas sobre estos benchmarks, se pueden realizar comparaciones mediante diversas técnicas, como pruebas no paramétricas, comparación de promedios de medidas de desempeño, análisis de curvas de convergencia y visualización de los resultados en el paisaje de búsqueda. Estas técnicas se describen a continuación [19], [28], [29].

C.i. Pruebas no paramétricas

Las pruebas no paramétricas son métodos estadísticos útiles para comparar grupos de datos y determinar si hay diferencias significativas entre ellos en términos de una variable de interés. Estas pruebas no requieren asumir una distribución específica para los datos, como la normalidad, a diferencia de las pruebas paramétrica como el t-student o la prueba z. Por lo que se utilizan cuando los datos no cumplen con los supuestos de los métodos paramétricos o cuando se trabaja con escalas ordinales o con datos categóricos, teniendo la ventaja de ser menos sensibles a valores atípicos, siendo más robustas frente a resultados extremos y de no requerir grandes tamaños de muestra para obtener resultados significativos [29].

Por lo anterior las pruebas no paramétricas son útiles para comparar el desempeño de metaheurísticas en diferentes benchmarks, ya que los resultados de las ejecuciones sobre los benchmarks no tienen una distribución conocida a priori y pueden ser no normales, asimétricos o multimodales. Además, al ser más apropiadas para muestras pequeñas puede ser beneficioso cuando se dispone de un número limitado de benchmarks, debido a restricciones de tiempo o recursos computacionales. Por lo anterior son ampliamente usadas para comparar múltiples algoritmos o configuraciones [29], [30]. En este estudio se usará la prueba no paramétrica de Wilcoxon [31] disponible en la herramienta Keel 3.0 [32].

a. Prueba de Wilcoxon de rangos con signo

La prueba de Wilcoxon de rangos con signo es una prueba no paramétrica diseñada para comparar dos conjuntos de datos de observaciones en parejas, cuando las muestras son independientes. Específicamente, la prueba de Wilcoxon asume que las observaciones emparejadas se obtienen de forma aleatoria e independiente. Esta prueba evalúa si hay diferencias significativas entre las medianas de dos conjuntos de datos emparejados [29], [31]. Por lo anterior se ha usado con éxito como criterio en la comparación del desempeño de metaheurísticas [30].

Esta prueba recibe como entrada las medidas de desempeño (como los valores de la función objetivo, tiempos de ejecución, etc.) de ambas metaheurísticas en cada uno de los benchmarks. Estos datos deben estar emparejados, es decir, para cada benchmark se debe tener un par de valores correspondiente a las dos metaheurísticas. La prueba calcula el estadístico de Wilcoxon (W), que se obtiene sumando los rangos de las diferencias entre los pares de datos, considerando los signos (positivos y negativos). Además, produce un valor p ($p\text{-}value$), que indica la probabilidad de obtener un resultado tan extremo como el observado bajo la hipótesis nula. Un valor p menor que el nivel de

significancia preestablecido α (por ejemplo, 0.05) sugiere que existe una diferencia significativa entre las dos metaheurísticas [29], [30], [31].

C.ii. Comparación del promedio de medidas de desempeño y curvas de convergencia

Otra forma habitual de comparar el desempeño de metaheurísticas es analizar los valores promedio de distintas medidas de desempeño obtenidas de su ejecución sobre cada benchmark, como el valor de la función objetivo al que convergen (relacionado con la precisión) y velocidad de convergencia (relacionado con la eficiencia). Estos promedios se obtienen a partir de un número considerable de ejecuciones, garantizando así que reflejen un comportamiento representativo de la metaheurística sobre cada bechmark y no se vean influenciados por valores atípicos.

Este análisis puede realizarse de forma visual utilizando las curvas de convergencia de las ejecuciones de las metaheurísticas sobre los benchmarks. Una curva de convergencia muestra el mejor valor obtenido de la función objetivo contra la cantidad de EFOs a lo largo de una ejecución. Las curvas obtenidas de varias ejecuciones de una metaheurística sobre un benchmark se pueden promediar, permitiendo observar visualmente la velocidad de convergencia y el valor promedio de la función objetivo al que convergen.

Estos promedios se comparan para cada benchmark, buscando observar una tendencia. Esta metodología es una de las más ampliamente utilizadas para la comparación de metaheurísticas debido a su sencillez [16], [20], [21], [23], [30], [33], [34], [35].

C.iii. Visualización en el paisaje de búsqueda

La visualización del paisaje de búsqueda puede ser útil para la comparación de metaheurísticas permitiendo de forma visual determinar las soluciones obtenidas por cada metaheurística en que región se encuentra, permitiendo determinar posibles estancamientos en mínimos locales. En el caso problemas con una dimensionalidad mayor de 3 la visualización del paisaje de búsqueda no se puede hacer de forma directa, se debe evaluar métodos de visualización que permitan bajar la dimensionalidad del espacio de búsqueda, algunos de los más comunes son los gráficos en 2D o 3D, que consisten de representaciones visuales que muestran cómo la función objetivo varía con dos o tres variables de decisión, y los mapas de contorno que consisten de representaciones bidimensionales que usan líneas de contorno para mostrar niveles de la función objetivo. Estas representaciones pueden ser no muy útiles para problemas en el que la dimensionalidad es muy alta y también cuando su función objetivo es no analítica, ya que el análisis de solo un par de esto parámetros a la vez puede ser complejo y no concluyente. Por lo anterior pueden ser mayor utilidad técnicas que permitan la reducción de dimensionalidad sin requerir la eliminación directa de dimensiones, como los mapas autoorganizativos (Self-Organising Maps, SOM) [36], [37].

C.iv. Mapa Auto-organizativo (Self-Organizing Maps, SOM)

Los SOM son una clase específica de redes neuronales artificiales no supervisadas en las que las neuronas se sitúan en un espacio que suele ser unidimensional o bidimensional, en el que se adaptan selectivamente a los patrones recibidos como entrada durante un proceso de aprendizaje competitivo. El SOM permite proyectar datos de alta dimensionalidad en un espacio de menor dimensión, generalmente un mapa bidimensional, pudiendo funcionar como una herramienta de Minería Visual de Datos, para la visualización y la detección de patrones en los datos [37]. Y ha sido usada con éxito para el análisis de procesos de optimización con metaheurísticas [36].

Un SOM bidimensional se representa generalmente como una cuadrícula donde cada celda, o nodo, tiene un vector de pesos asociado que representa una posición en el espacio de características de los datos originales. Las celdas adyacentes en el mapa están relacionadas entre sí, y los datos similares tienden a agruparse en celdas vecinas [37].

III.2. Small-angle Scatering (SAS)

Las técnicas de análisis de dispersión a bajos ángulos, conocidas en inglés como Small Angles Scattering (SAS) se basan en el principio físico de dispersión de ondas por la materia que es uno de los fenómenos más comúnmente usados para investigar la estructura espacial de la materia, en la Figura 1 se presenta un esquema de este fenómeno. En los experimentos de dispersión, un haz primario de radiación incide sobre el objeto de estudio dispersando de forma elástica la radiación y produciendo un patrón de dispersión, que puede medirse y analizarse para obtener información sobre la estructura de la muestra. En la Figura 2a se muestra la intensidad de la radiación dispersada I en cada dirección

representada por el vector de onda $\boldsymbol{Q} = (Q_x, Q_y)$ [4]. La resolución espacial de las técnicas basadas en este fenómeno está determinada por la longitud de onda λ de la radiación y los ángulos de dispersión 2θ analizados, permitiendo estudiar la materia en todas las escalas, desde las partículas elementales hasta los macroobjetos [1], [38].

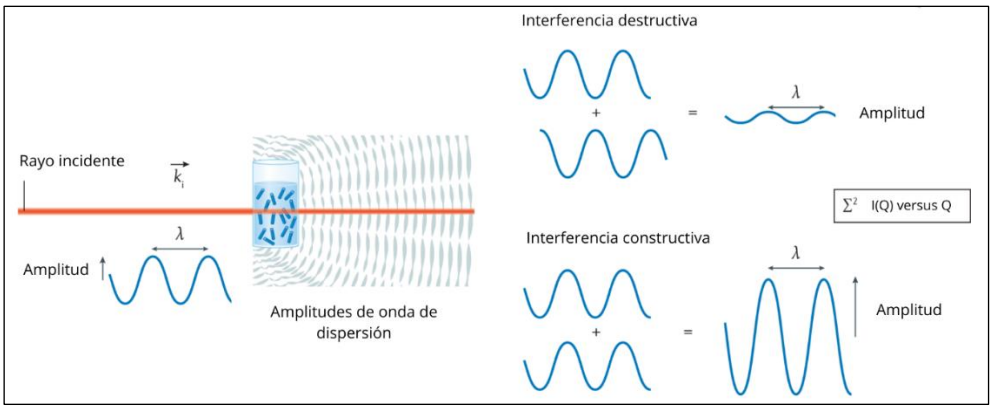

Figura 1. Representación ondulatoria del fenómeno de dispersión elástica. La magnitud de las amplitudes de dispersión coherente en función del ángulo está relacionada con las correlaciones espaciales entre los centros de dispersión. Tomada de [4].

El uso de un haz de radiación con longitud de onda de 1Å a 10Å permite estudiar el estado condensado de la materia, sólidos y líquidos, hasta una resolución atómica, Figura 3. La determinación de la estructura atómica y molecular con una resolución espacial de hasta 0.01Å para muestras con orden de corto alcance o no periódicas se puede obtener con las técnicas Wide Angle Scattering (WAS), y para muestras con estructuras cristalinas con periodicidad de largo alcance con las técnicas de difracción, por ejemplo Difracción de Rayos X (DRX); para ambos casos esta información estructural está codificada en el patrón de dispersión y difracción, respectivamente, en ángulos del orden de grados a decenas de grados. Las técnicas SAS, como se mencionó previamente, se usan en el estudio de la estructura de la materia a nivel superatómico, con una resolución espacial desde diez hasta miles e incluso varias decenas de miles de angstroms, la información de estas dimensiones está codificada en el patrón de difracción en la región de ángulos pequeños del orden de decimas de grados, Figura 4, de ahí su nombre. Para analizar estructuras con dimensiones aún mayores se hace uso de las técnicas Ultra Small Angle Scattering (USAS) y Very Small Angle Scattering (VSAS), en las que se analiza el patrón de difracción en la región del orden de décimas a centésimas de grados [1], [5].

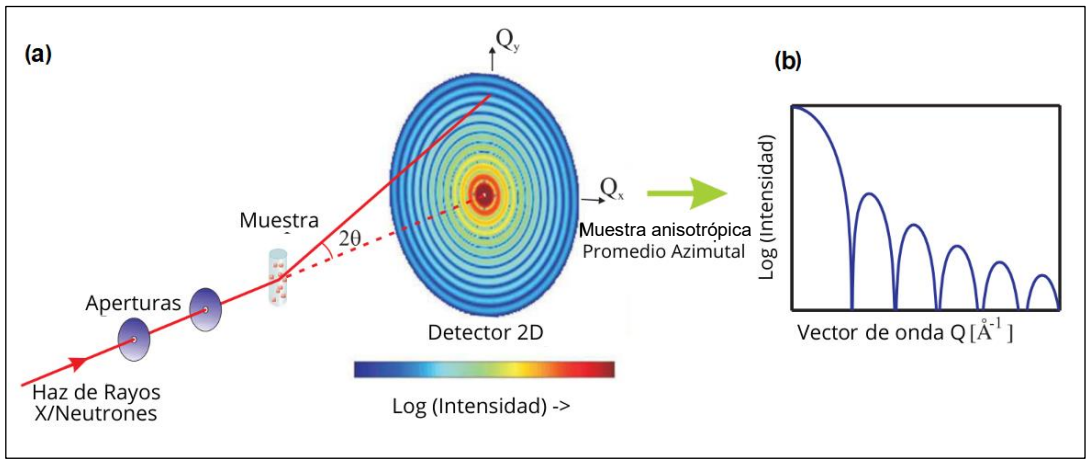

Figura 2. (a) Representación esquemática de un experimento de dispersión de ángulo pequeño (SAS). **(b)** Obtención de un patrón SAS 2D de muestra isotrópica, obteniendo mediante promedio azimutal un perfil SAS 1D. Adaptada de [39].

La cantidad de información estructural que se obtiene mediante SAS depende en gran medida del grado de orden dentro de la muestra. Por ejemplo, en una suspensión diluida de partículas se puede determinar la forma, el radio de giro y la

polidispersidad, mientras que una muestra semicristalina muy ordenada puede proporcionar un modelo estructural de resolución molecular o de ordenamiento estructural. Por lo general, los experimentos SAS proporcionan información estructural promediada por conjuntos, a diferencia de las características más selectivas observadas mediante microscopías electrónicas o de fuerza atómica. En consecuencia, los métodos SAS y las técnicas de microscopía electrónica y otros tipos de técnicas obtienen información complementaria para la elucidación estructural de materiales a nanoescala (ver Figura 3) [5].

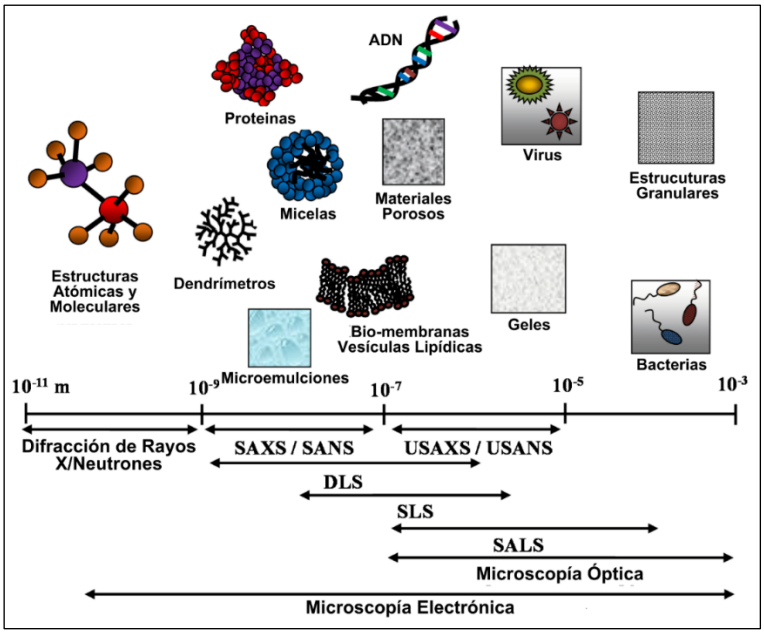

Figura 3. Resolución espacial de las técnicas de difracción de rayos X y neutrones, y de las principales técnicas de dispersión SAS: dispersión de rayos X y neutrones de ángulo pequeño (SAXS y SANS), ultra-SAXS y ultra-SANS (USAXS y USANS) y dispersión de luz dinámica, estática y de ángulo pequeño (DLS, SLS y SALS). Comparación con técnicas complementarias de microscopía óptica y electrónica. Adaptada de [5].

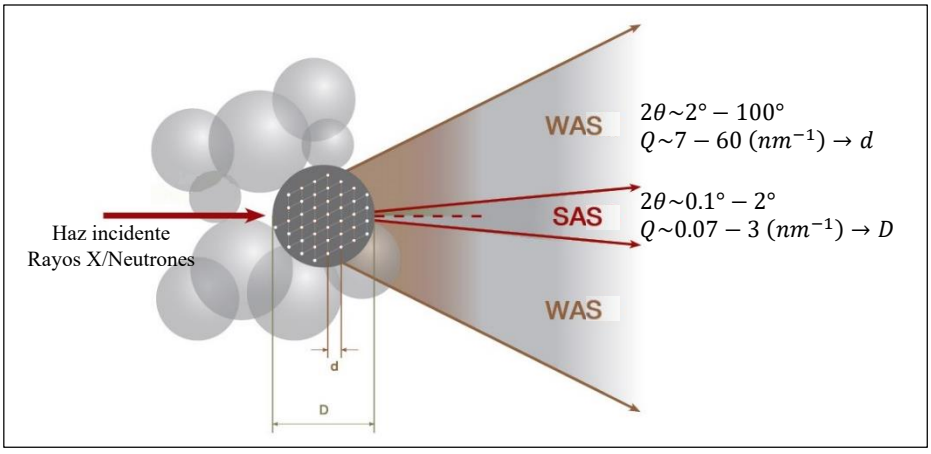

Figura 4. SAXS y WAXS alcanzan resoluciones diferentes: WAXS mide ángulos más amplios y alcanza una resolución atómica, mientras que SAXS mide ángulos muy pequeños y alcanza una resolución a nanoescala. Adaptada de [40].

Existen dos tipos principales de técnicas SAS que se diferencian por el tipo de radiación usada para la dispersión: la dispersión de rayos X (Small Angle X-ray Scattering SAXS) y la dispersión de neutrones térmicos (Small Angle Neutron Scattering SANS) [1], [2], [4], [5], [12], [38], [40]. Los mecanismos físicos con los que su radiación interactúa con la materia difieren entre las técnicas SAXS y SANS; la dispersión de los rayos X por la materia viene determinada

casi en su totalidad por la interacción de la radiación incidente con los electrones; el contraste en el patrón de dispersión se produce debido a las fluctuaciones espaciales de la densidad de electrones en la muestra que varían en función del número atómico de los elementos presentes [1]. Por su parte los neutrones no son dispersados apreciablemente por electrones y su contraste de dispersión proviene principalmente de las fluctuaciones espaciales de la densidad y tipo de isótopos presentes en la muestra [4], [7].

Conocer estos mecanismos permite determinar las limitaciones y campos de acción de cada técnica; así como la preparación de las muestras más adecuada para mejorar el análisis, por ejemplo con el uso de las técnicas de contraste que consisten en la modificación de la densidad de dispersores del blanco o la matriz de soporte para maximizar o alterar de forma controlada la dispersión de alguno o algunos de los componentes de interés de la muestra, permitiendo obtener más información de estos [4]. Aun así, las técnicas SAXS y SANS suelen usarse en conjunto permitiendo la obtención de información complementaria para el estudio estructural de sistemas multicomponentes [1], [4], [38].

III.2.A. Obtención de un patrón SAS y un perfil SAS

En la Figura 2a se puede ver una representación esquemática de un experimento de SAS, en este se hace incidir un haz primario de radiación, rayos X o neutrones térmicos, sobre el objeto de estudio que dispersa de forma elástica la radiación en todas las direcciones, esta radiación dispersada puede ser medida usando dispositivos de carga acoplada o detectores de matriz de píxeles en el caso de SAXS y detectores basados en helio o boro para SANS [4], produciendo un patrón de dispersión 2D [4], [12]. Este patrón 2D SAS, de ahora en adelante patrón SAS, puede presentar determinadas simetrías dependiendo de dos factores principalmente, el sistema de colimación del haz incidente y si la muestra es isotrópica o anisotrópica; en el caso de una muestra isotrópica su patrón SAS medido usando un sistema de colimación puntual presenta una simetría radial (o azimutal), este caso se puede ver en la Figura 2a.

Una vez medido el patrón SAS este es sometido a distintos tratamientos antes de ser analizados, entre ellos la eliminación de background que es la parte de la intensidad de dispersión $I(\boldsymbol{Q})$ que no es generada por la muestra, sino por otros objetos como el porta-muestra u otras fuentes externas de radiación; así mismo se le aplican correcciones debido a la colimación y a los efecto de la longitud de onda ya que el haz de radiación tiene una dimensión finita y un perfil dado de longitud de onda, diferente al caso ideal en el que el haz es puntual y posee una única longitud de onda [10]. Adicionalmente también los patrones SAS suelen promediarse mejorando la calidad estadística de los datos[4], [12].

Los patrones SAS pueden promediarse en 1 o más perfiles SAS 1D, perfiles SAS de ahora en adelante, en la mayoría de las ocasiones este promedio tiene la virtud de facilitar el análisis. Para el caso de patrones SAS de muestras isotrópicas, estos puede representarse en un único perfil SAS promediando de forma lineal para el caso de colimación lineal o acimutal para el caso de colimación puntual, en la Figura 2b se puede ver una ejemplo de este último caso; el perfil SAS resultante sólo depende del ángulo de dispersión 2θ con respecto al haz directo, representándose como la intensidad de dispersión $I(Q)$ en función de la magnitud del vector de onda $Q = |\boldsymbol{Q}| = 4\pi sin\theta/\lambda$, siendo λ la longitud de onda del haz dispersado [2], [4], [12].

III.2.B. Análisis de un perfil SAS ($\boldsymbol{I_{exp}(Q)}$)

Ahora bien, para poder obtener información de la muestra a partir de su patrón de dispersión SAS se modela dicho patrón a partir de modelos fisicomatemáticos que en su mayoría se desprenden del modelo físico ondulatorio de la dispersión, como el presentado en la Figura 1. La intensidad de la radiación dispersada ($I(\boldsymbol{Q})$) depende del valor del vector de onda $\boldsymbol{Q}$ dispersado y de la densidad de longitudes de dispersión $\rho(\boldsymbol{r})$ de los elementos que dispersan la onda dentro del material. En la Ecuación 2, la función $F(\boldsymbol{Q})$, representa la transformada de Fourier de $\rho(\boldsymbol{r})$. Mientras que $\rho(\boldsymbol{r})$ representa la distribución espacial de los elementos dispersantes en el espacio, $F(\boldsymbol{Q})$ representa esta misma distribución, pero en el espacio recíproco (por eso depende de $\boldsymbol{Q}$) [1], [2], [39].

$$F(\boldsymbol{Q}) = \iiint \rho(\boldsymbol{r}) e^{-i\boldsymbol{Q}\cdot\boldsymbol{r}} dV$$
$$I(\boldsymbol{Q}) = F(\boldsymbol{Q})F^*(\boldsymbol{Q}) \qquad (2)$$

Para el caso de patrones SAS de muestras isotrópicas, la expresión anterior puede reescribirse de forma más sencilla usando la formulación de Debye; en dichos patrones su intensidad solo depende de la magnitud Q del vector de onda

y no de su dirección, $I(Q)$ [1], [2]. Si consideramos dos puntos de dispersión ubicados en las posiciones $\boldsymbol{r}_1$ y $\boldsymbol{r}_2$ el patrón de dispersión será un patrón promedio producido por las distribuciones $\rho(\boldsymbol{r_1})$ y $\rho(\boldsymbol{r_2})$:

$$\tilde{\rho}(r) = \iiint \rho(\boldsymbol{r_1})\,\rho(\boldsymbol{r_2})dV \quad r = \|\boldsymbol{r_1} - \boldsymbol{r_2}\|$$

$$Q = \|\boldsymbol{Q}\| = \frac{4\pi\, sen\theta}{\lambda} \quad dV = 4\pi r^2 dr$$

$$I(Q) = \int 4\pi r^2 \tilde{\rho}(r) \frac{\sin(Qr)}{Qr} dr \; = 4\pi \int p(r) \frac{\sin(Qr)}{Qr} dr \qquad (3)$$

Donde la expresión:

$$\frac{\sin(Qr)}{Qr} = \left\langle e^{-i\boldsymbol{Q}\cdot\boldsymbol{r}} \right\rangle$$

Se conoce como *factor de Debye* y corresponde al promedio del factor de fase $e^{-i\boldsymbol{Q}\cdot\boldsymbol{r}}$. La Ecuación 3 es conocida como la formulación de Debye; de esta formulación del perfil SAS se desprenden la mayoría de las metodologías de análisis.

Una de estas metodologías consiste en obtener $p(r)$ a partir de $I(Q)$, esto mediante la Ecuación 3 que puede invertirse, usando métodos de inversión indirecta regularizada [4]. El término $p(r)$ es conocido como la función de distribución par-distancia o frecuencia probable de distancias en el espacio real, que consiste en un histograma ponderado de la distancia entre dispersores de la muestra y contiene información estructural del espacio real de la muestra como la dimensión máxima de las partículas D_{max} para el caso de partículas en suspensión [2].

Otra metodología habitual de análisis es la de mínimos cuadrados ponderados que consiste en ajustar modelos a uno o varios perfiles SAS experimentales $I_{\text{exp}}(Q)$ de la muestra, donde se calculan perfiles SAS $I_{comp}(Q)$ a partir del modelo escogido; a continuación se hablará del análisis a partir de un solo perfil experimental $I_{\text{exp}}(Q)$, cuando se usan varios la metodología es muy parecida con algunas diferencias. Estos modelos dependen de algunos parámetros estructurales que se optimizan buscando mejorar el ajuste del perfil calculado $I_{comp}(Q)$ a los datos experimentales $I_{\text{exp}}(Q)$ [4], por lo que el objetivo de esta forma de análisis es, a partir de $I_{\text{exp}}(Q)$, entregar información estructural de la muestra representada en los parámetros estructurales del modelo o en algunos casos una representación de la estructura [10], [17]. En esta metodología hay tres partes fundamentales, que son: el modelo usado para calcular los perfiles $I_{comp}(Q)$, la métrica usada para evaluar el ajuste de los perfiles $I_{comp}(Q)$ generados durante el proceso de análisis respecto a $I_{\text{exp}}(Q)$, y el proceso de optimización que permite obtener el $I_{comp}(Q)$ más ajustado a $I_{\text{exp}}(Q)$ y con este sus parámetros estructurales correspondientes.

Existen diferentes métricas que se pueden usar para evaluar el ajuste de cada $I_{comp}(Q)$ respecto a $I_{exp}(Q)$, unas de las más comunes son la prueba χ^2, Ecuación 4, y la suma de errores cuadráticos SSE, así como sus variantes como la presentada en la Ecuación 5; estas se van acercando a uno y cero respectivamente, cuando el ajuste mejora; cada métrica puede darle distinto ponderado $w(Q)$ en el ajuste a los valores de $I_{exp}(Q)$ o Q, priorizando el ajuste en distintas regiones del perfil SAS experimental $I_{exp}(Q)$ [10], [11]. La expresión matemática de estas métricas es distinta cuando se considera más de un perfil experimental $I_{exp}(Q)$ en el análisis. Para el análisis de perfiles SAS es importante determinar el rango de valores de Q del perfil a analizar, la información morfológica de la muestra en el espacio real se codifica en el espacio reciproco de vectores de onda $\boldsymbol{Q}$, de modo que una distancia d en el espacio real se puede relación con el valor de Q mediante la expresión $Q \approx \pi/d$ [12]; de aquí se puede ver qué distancias más grandes en el espacio real estarán relacionadas con valores de Q bajos y viceversa. Lo anterior se suele tener en cuenta para la elección de la métrica de ajuste, así como para el análisis de los resultados. Esta métrica constituye la función objetivo del problema de optimización.

$$\chi^2 = \sum_Q \left(\frac{I_{\text{exp}}(Q) - I_{comp}(Q)}{w(Q)} \right)^2 \qquad (4)$$

$$SSE = \sum_i w(i)\left(\log\left(I_{\exp}(Q_i)\right) - \log\left(I_{comp}(Q_i)\right)\right)^2 \qquad (5)$$
$$w(i) = \log(Q_{i+1}) - \log(Q_i)$$

El proceso de optimización busca los parámetros del modelo, que constituyen las variables de decisión del problema de optimización, generando el $I_{comp}(Q)$ con mejor ajuste posible, este proceso puede hacerse mediante distintas metodologías y algoritmos, como los algoritmos de descenso del gradiente, la búsqueda de fuerza bruta, el método Monte Carlo, los algoritmos metaheurísticos, entre otros [4], [9], [10], [11]; su elección está sujeta al modelo sobre el que se esté trabajando, la métrica escogida para evaluar el ajuste, la robustes deseada en el análisis, los recursos computacionales, entre otros aspectos; así mismo debe tenerse en cuenta que cada procedimiento de optimización también proporciona errores estándar sobre los parámetros de ajuste [4]. Un factor para tener en cuenta es el fenómeno de la degeneración, en el que diferentes combinaciones de parámetros estructurales dan como resultado perfiles SAS muy parecidos [10], lo que se refleja en un paisaje de búsqueda multimodal.

Finalmente, el modelo usado en el análisis. Esta es la parte más importante ya que el éxito de esta metodología de análisis radica en la capacidad de elegir el modelo adecuado que permita obtener información física sobre el sistema analizado [8], el modelo define la cantidad y calidad de la información que se puede extraer del perfil SAS experimental. Existen varios modelos que pueden usarse dependiendo especialmente del tipo de muestra que se esté analizando en la técnica SAS, para cada tipo existen diferentes modelos que se han venido formulando, cada modelo considera o no diferentes parámetros estructurales de la muestra [4], esto define su nivel de complejidad. Aunque, idealmente, se desearía que el modelo tuviera en cuenta todos los parámetros estructurales posibles de la muestra, esto no es viable en la práctica, ya que esto aumentaría su complejidad, así como la dimensionalidad del problema de optimización y por lo tanto los recursos computacionales necesarios para su uso en el análisis. Muchos de estos modelos se desprenden del modelo fisicomatemático de la dispersión como el deducido en las Ecuaciones 2 y 3. A continuación se presentan algunos de los modelos más comúnmente usados.

Para el caso de muestras que se puedan modelar estructuralmente como un conjunto de unidades con una estructura uniforme en una distribución espacial dada, por ejemplo, soluciones de nanopartículas, una de las formas más usuales de modelar su perfil SAS se puede expresar como en la Ecuación 6, [39]. Donde $F(Q)$ es el factor de forma, $S(Q)$ es el factor de estructura, y K es una constante de proporcionalidad que está relacionada especialmente con el volumen y cantidad de unidades que participan de la dispersión. $F(Q)$ representa la intensidad de dispersión a partir de la distribución de distancias entre centros de dispersión espacialmente correlacionados dentro de una unidad y no tiene en cuenta la distribución/interacciones entre las unidades, que se describen mediante el factor de estructura $S(Q)$ [4]. Las expresiones matemáticas para $F(Q)$ y $S(Q)$ se pueden obtener solucionando la formulación de Debye, Ecuación 3, para diferentes geometrías y disposiciones.

$$I_{comp}(Q) = K \cdot S(Q) \cdot F^2(Q) \qquad (6)$$

Los parámetros morfológicos y estructurales asociados a la Ecuación 6 son los que se optimizan en el análisis de mínimos cuadrados ponderados. Para $F(Q)$ se asume para las unidades una geometría y longitud de dispersión dada, que se remplaza en la Ecuación 3 y al resolverla se obtiene una expresión en función de las características geométricas de las unidades [4]. Uno los modelos analíticos de factor de forma $F(Q)$ más comúnmente usado es el de unidades esféricas de radio R y longitud de dispersión uniforme, que está dado por la Ecuación 7 [39].

$$F^2(Q) = 9\left(\frac{\sin(QR) - QR\cos(QR)}{Q^3R^3}\right)^2 \qquad (7)$$

Para $S(Q)$ existen deferentes aproximaciones para analizar la distribución de subunidades, como los potenciales centro-simétricos, tales como los potenciales hard-sphere, de Coulomb apantallado y el sticky hard-sphere [4], estos potenciales se relacionan matemáticamente como $p(r)$, permitiendo resolver la Ecuación 3. Cuando la concentración de unidades es baja la interacción de dispersión entre estas es despreciable y el factor de estructura es aproximadamente 1, $S(Q)\sim 1$, por lo que la intensidad de dispersión I_{comp} es función solamente a el factor de forma $F(Q)$ [10].

Los modelos analíticos son unos de los modelos más comúnmente usados en la metodología de mínimos cuadrados ponderados por su versatilidad y sencillez para describir los perfiles de dispersión $I_{comp}(Q)$ y fácil acople con procesos de optimización, siendo el modelo empleado por diferentes paquetes de software de análisis de perfiles SAS, como SASfit y SASview [11]. Los modelos analíticos existentes funcionan bien para perfiles de dispersión de estructuras ensambladas canónicas y químicas de polímeros convencionales, pero pueden no funcionar para obtener dimensiones importantes de la estructura para químicas y arquitecturas de polímeros novedosas y/o estructuras ensambladas no vistas previamente [10] así como para fracciones de empaquetamiento de las unidades elevadas, superiores a 0.4 [3]. Otras modelos tienen mayor desempeño en estos casos, pero a cambio de un costo computacional mayor, la técnica de modelado de Rigid-Body es una de estas.

El modelado de Rigid-Body consiste en modelar la muestra o parte de esta como una conformación espacial de subunidades, los Rigid-Bodies, como se puede ver en la Figura 5a, con estructura y longitud de dispersión conocida para la cual se puede calcular su perfil de dispersión SAS $I_{comp}(Q)$ a partir de la Ecuación 8 que se deriva de formulación de Debye, Ecuación 3 [2], ya que la posición de cada subunidad $\boldsymbol{r_j}$ es conocida y por lo tanto la de los dispersores.

$$I_{comp}(Q) = \left\langle \sum_{j,k}^{\square\square} b_j F_j(\boldsymbol{Q}) b_k F_k^*(\boldsymbol{Q}) e^{-i\boldsymbol{Q}\cdot\boldsymbol{r}_{jk}} \right\rangle \quad \boldsymbol{r}_{jk} = \boldsymbol{r}_j - \boldsymbol{r}_k$$

$$r_{jk} = \|\boldsymbol{r}_{jk}\| \quad F_j(Q) = \langle F_j(\boldsymbol{Q}) \rangle = \langle F_j^*(\boldsymbol{Q}) \rangle$$

$$I_{comp}(Q) = \sum_{j}^{N_T} b_j^2 F_j(Q)^2 + 2 \sum_{j \neq k}^{\square\square} b_j b_k F_j(Q) F_k(Q) \frac{\sin(Q r_{jk})}{Q r_{jk}} \qquad (8)$$

En esta expresión N_T corresponde a la cantidad total de subunidades usadas para modelar la muestra. $F_j(\boldsymbol{Q})$ corresponde al factor de forma de la subunidad j cuya posición es $\boldsymbol{r}_j$ y su densidad de longitud de dispersión es b_j. $F_j(\boldsymbol{Q})$ se puede calcular a partir de la Ecuación 2 o expresiones derivadas de esta. Para el caso en el que se usan subunidades esféricas cuya longitud de dispersión uniforme es común usar $F_j(Q)$ como la raíz cuadrada del modelo analítico de la Ecuación 7, un ejemplo de este caso es el modelado Ab Inito [9].

En el modelado Rigid-Body los parámetros estructurales que se buscan optimizar son las coordenadas $\boldsymbol{r}_j$ y orientación de cada una de estas subunidades, estas se trasladan y giran como cuerpos rígidos, lo que modifica su amplitud de dispersión $I_{comp}(Q)$ [9]. La elección de la estructura y longitud de dispersión adecuada de las subunidades, que pueden ser fijos o variables, así como la cantidad a usar en el modelado y sus grados de libertad, son cruciales en esta técnica ya que establecen la dimensionalidad del problema de optimización, y son decisivos a la hora de escoger el algoritmo de optimización a usar y con este el coste computacional. Con este modelo los métodos de optimización usados comúnmente son la búsqueda exhaustiva, el método Monte Carlo y los algoritmos metaheurísticos [9], [10]; para su elección se busca que el algoritmo logre conseguir la configuración espacial de subunidades óptima con la generación y evaluación de la menor cantidad posible de estas configuraciones, ya que el cálculo de la dispersión es computacionalmente exigente, Ecuación 8; se puede observar que la cantidad de operaciones aritméticas necesarias para calcular $I_{comp}(Q)$ aumenta con el cuadrado del número de subunidades N_T usadas en el modelado.

De este modelado se derivan otros como el ad Initio [9], y el usado en la herramienta 'Computational Reverse-Engineering Analysis for Scattering Experiments' (CREASE) [13].

III.3. Computational Reverse-Engineering Analysis for Scattering Experiments (CREASE)

CREASE es una herramienta recientemente desarrollada en el laboratorio de investigación de la Profesora Arthi Jayaraman en la Universidad de Delaware, que usa la metodología de mínimos cuadrados ponderados para obtener información estructural clave del factor de forma y factor de estructura de uno o varios perfiles SAXS y/o SANS de

una muestra, e incluso hacer una estructura 3D representativa de la misma, útil para análisis adicionales[14]; en la Figura 5 se puede ver su flujo de trabajo.

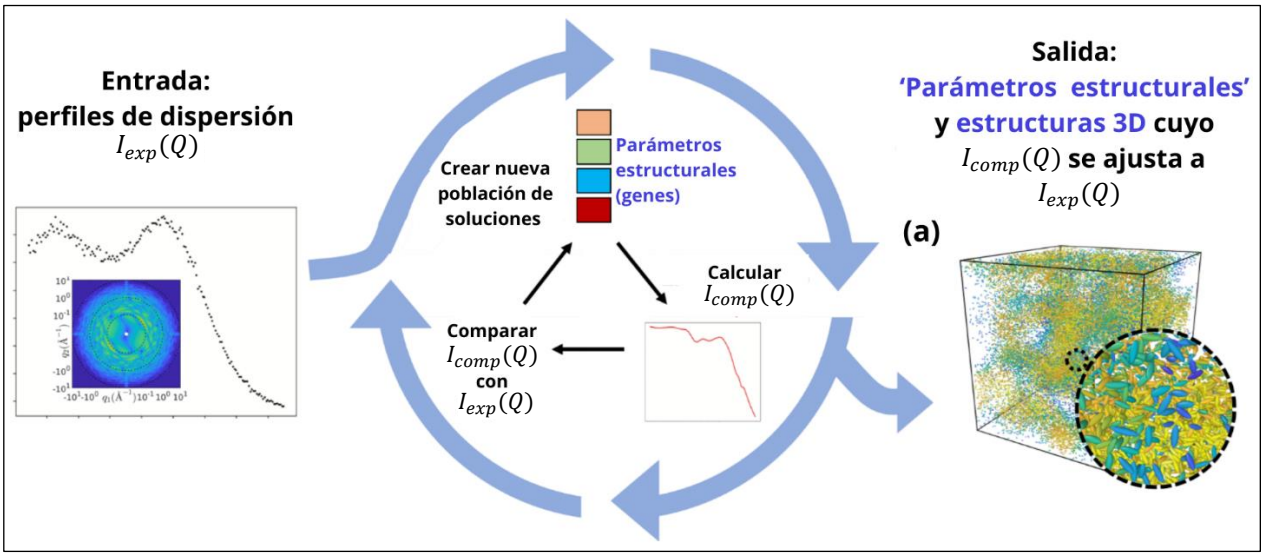

Figura 5. Flujo de trabajo del CREASE. CREASE toma como entrada perfiles SAS experimentales $I_{exp}(Q)$ y mediante una optimización interna utilizando GA identifica como salida las características estructurales clave, así como estructuras representativas del espacio real 3D cuyos $I_{comp}(Q)$ coinciden con la entrada de dispersión experimental. **(a)** Estructura 3D de muestra de solución nanopartículas construida con el modelado Rigid-Body usando subunidades con forma de elipsoide. Adaptado de [13].

El CREASE tiene capacidad para analizar distintos tipos de muestras, llamadas *shapes*, como soluciones de micelas y vesículas, y fibrillas y mezclas de nanopartículas, de baja y alta concentración, y es fácilmente adaptable para nuevos tipos de muestras [3], [10], [13], ya que es una herramienta de código abierto codificada en Python disponible en GitHub (github.com/arthijayaraman-lab/crease_ga). En la práctica la herramienta CREASE ha demostrado desempeñarse muy bien para el análisis de perfiles SAS, logrando conseguir parámetros estructurales más cercanos a los experimentales en comparación con los modelos analíticos disponibles [3], [10].

III.3.A. Cálculo de $I_{comp}(Q)$

Cómo se mencionó previamente el CREASE, describe la morfología de la muestra a partir de parámetros estructurales. CREASE tiene dos maneras de calcular el perfil SAS $I_{comp}(Q)$: usando la formulación de Debye y con modelos de aprendizaje automático (Machine Learning, ML).

Ya que el cálculo de $I_{comp}(Q)$ mediante el método de Debye tiene un coste computacional importante, el equipo del CREASE ha implementado exitosamente, para algunas *shapes,* modelos de ML entrenados en miles de perfiles de dispersión calculados a partir del método Debye para diversos valores de los parámetros estructurales, para calcular $I_{comp}(Q)$ dados los parámetros estructurales, acelerando considerablemente su cálculo respecto al método de Debye, después de la inversión de tiempo inicial en el entrenamiento del modelo de ML [3], [16] y en la obtención del dataset de entrenamiento. Por lo anterior el uso de ML no se ha aplicado a todas las *shapes* disponibles en CREASE. *Shapes* como las de soluciones de baja concentración de vesículas ensambladas a partir de polímeros anfifílicos aún emplean la formulación de Debye [10].

Con la formulación Debye, dependiendo de la *shape* la Ecuación 8 tiene algunas modificaciones, a continuación, se verá como es el modelo para la *shape* usada en esta investigación.

A.i. shape "Soluciones de baja concentración de vesículas ensambladas a partir de polímeros anfifílicos"

Las vesículas son nanoestructuras esféricas con una arquitectura característica de núcleo hueco encerrado por una membrana de pared, estas constituyen una importante clase de nanoestructuras autoensambladas capaces de encapsular

materiales que permiten su uso en una amplia gama de aplicaciones, como las nanorreacciones y la administración de fármacos. Las vesículas suelen encontrarse al ensamblar copolímeros anfifílicos sintéticos, biomiméticos o bioderivados en solución. Estudiar el autoensamblaje de vesículas a partir de polímeros anfifílicos como tensioactivos, copolímeros sintéticos en bloque y macromoléculas bioinspiradas es necesario para diseñar nanotransportadores personalizables e inteligentes con funciones controladas [10]. En ese camino el equipo CREASE desarrolló una *shape* para el análisis de ese tipo de muestras, de ahora en adelante solución diluida de vesículas.

El análisis de esta *shape* se hace a partir un solo perfil experimental $I_{exp}(Q)$, las macromoléculas anfifilicas están constituidas por dos tipos de monómeros, solvofílicos A y solvofóvicos B, que son representados como las subunidades en este caso. Para esta *shape* $I_{comp}(Q)$ se calcula como:

$$I_{comp}(Q) = \sum_{\alpha\in[A,B]}^{\square} \sum_{\beta\in[A,B]}^{\square} b_\alpha b_\beta F_\alpha(Q) F_\beta(Q) \omega(Q) + I_{bg} \quad (9)$$

$$\omega(Q) = \left\langle \frac{1}{N_A + N_B} \sum_{j=1}^{N_A+N_B} \sum_{k=1}^{N_A+N_B} \frac{\sin(Qr_{jk})}{Qr_{jk}} \right\rangle \quad N_A + N_B = N_T \quad (10)$$

Los monómeros son modelados como subunidades esféricas por lo que su factor de forma $F_\alpha(Q)$, habiendo un total N_A subunidades de tipo A y N_B de tipo B, y I_{bg} es la intensidad de dispersión de fondo. Se puede ver que la complejidad computacional en este caso está en el cálculo de $\omega(Q)$, Ecuación 10, conocido como el factor de estructura intravesicular, en este al igual que en el factor de Debye, los paréntesis angulares denotan un promedio. Para cada individuo, $\omega(Q)$ es el resultado de promediar siete configuraciones diferentes de colocación aleatoria de subunidades, esto con el fin de mitigar el riesgo de que una sola configuración dentro de las dimensiones de la vesícula sesgue la intensidad de dispersión calculada. Se podría optar por usar una sola configuración con un número significativamente mayor de subunidades N_T, lo cual podría reducir los sesgos introducidos por pocas colocaciones de dispersores o densidades más pequeñas. Sin embargo, esto aumentaría el coste computacional. Pero, otra razón para considera múltiples configuraciones diferentes es que esto se aprovecha para capturar la dispersión del radio de la vesícula σ_R, que, como se ver en la Tabla 1, es uno de los parámetros estructurales que considera esta *shape* [10].

Tabla 1. Parámetros estructurales de la *shape* Solución de baja concentración de vesículas ensambladas a partir de polímeros anfifílicos [10].

Parámetros Estructurales	**Significado Físico**
R_{core}	Radio del núcleo de la vesícula
t_{Ain}	Espesor de la capa solvofílica A interna
t_B	Espesor de la capa solvofóbica B intermedia
t_{Aout}	Espesor de la capa solvofílica A externa
s_{Ain}	Proporción de los dispersores solvofílicos totales presentes en la capa interna
σ_R	Dispersión del radio del núcleo R_{core}
$-\log(I_{bg})$	Negativo del logaritmo la intensidad del I_{bg}

Para obtener las coordenadas $\boldsymbol{r}_j$ de las subunidades, necesarias para el cálculo de $I_{comp}(Q)$, se hace una reconstrucción 3D de la muestra a partir de los parámetros estructurales, en la Figura 6 se pude ver para el caso de la *shape* de solución diluida de vesículas. Esta *shape* tiene un total de siete parámetros estructurales, que se pueden ver listados en la Tabla 1, siendo estas las variables de decisión del problema de optimización. Más detalles sobre este modelo se pueden encontrar en [10].

Otras *shapes* como la de mezcla binaria de nanopartículas, requieren de un paso adicional de uso de dinámica molecular para la obtención de la ubicación $\boldsymbol{r}_j$ de las subunidades para el cálculo de $I_{comp}(Q)$ [3] lo que aumenta aún más su coste computacional.

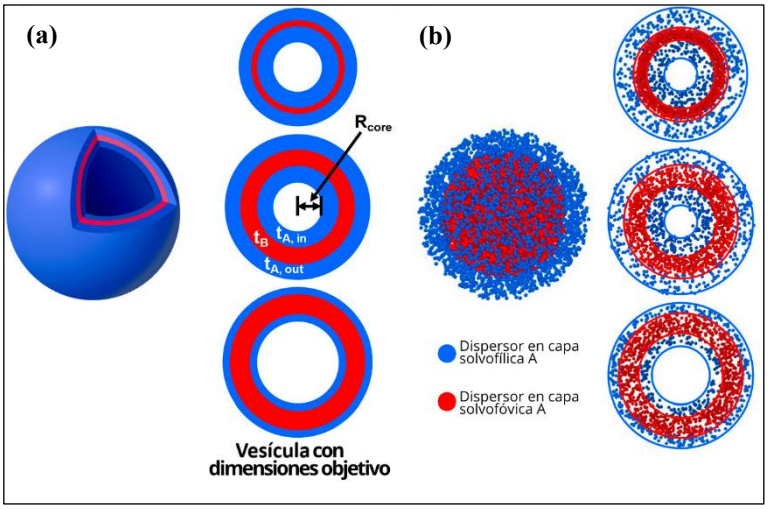

Figura 6. Ubicación de las subunidades para la *shape* de solución de bajas concentración de vesículas ensambladas a partir de polímeros anfifílicos, dados sus parámetros estructurales objetivo. **(a)** Dimensiones relevantes de la vesícula compuesta por las capas solvofílicas A (azul) interior (t_{Ain}) y exterior (t_{Aout}) y una capa solvofóbica B (rojo) intermedia (t_B). **(b)** Colocación de las subunidades (esferas) dentro de las dimensiones objetivo (líneas continuas) de las capas solvofílicas A y solvofóbica B visualizadas para la vesícula completa (izquierda) y un corte del centro de la vesícula (derecha). Adaptada de [10].

Como se mencionó previamente, el aumento de N_T incrementa el costo computacional de la ejecución del CREASE; sin embargo, también mejora la precisión de los parámetros estructurales obtenidos del análisis. El valor de N_T debe ser configurado por el usuario teniendo en cuenta estos aspectos. En el CREASE este valor se configura mediante los *shape_params* que se ajustan al inicio de la ejecución, estos son descriptores de la *shape* que en algunos casos incluye información estructural conocida a priori de la muestra; para la solución diluida de vesículas, el valor de N_T es directamente proporcional a la razón n_{sct}/N, siendo N y n_{sct} los *shape_params* correspondiente al número de monómero en una cadena y el número de dispersores usados para representar una cadena, respectivamente.

Para esta *shape* cuyo análisis se hace sobre un único perfil $I_{exp}(Q)$, para evaluar el ajuste de $I_{comp}(Q)$, CREASE, permite escoger entre la métrica SSE del logaritmo de $I_{exp}(Q)$ e $I_{comp}(Q)$ ponderada sobre el rango de Q (ver Ecuación 5); o la métrica χ^2, usando el error experimental de $I_{exp}(Q)$, $\sigma_{exp}(Q)$, como ponderado $w(Q)$, si dicha información es aportada de lo contario se iguala a 1 (ver Ecuación 4) [3], [10], [16]. Ya que al usar el modelo de Debye para el cálculo de $I_{comp}(Q)$ en la ubicación de las subunidades hay cierto grado de aleatoriedad, al calcular $I_{comp}(Q)$ para la misma combinación de parámetros estructurales, este puede variar ligeramente y por lo tanto también su ajuste a $I_{exp}(Q)$, lo que puede interpretarse como ruido. Para este estudio se usó el SSE como función objetivo que es el usado en la referencia [10] para el análisis de la *shape* solución diluida de vesículas.

De lo anterior se puede observar que la función objetivo del problema de optimización del CREASE es una función no analítica ya que para unos parámetros estructurales dados el cálculo de su SSE involucra el uso de operaciones matemáticas no convencionales. Por otro lado, se anticipa que el paisaje de búsqueda del CREASE tendrá presencia de ruido y de múltiples mínimos locales. El ruido debido a que se mencionó previamente en la subsección III.3.A, al usar Debye para el cálculo de $I_{comp}(Q)$ hay cierto nivel de aleatoriedad en su cálculo, por lo que el valor de SSE correspondiente no es siempre el mismo, sino que presenta una variación alrededor de un valor medio, para la misma combinación de parámetros estructurales, esto se puede interpretar en el paisaje de búsqueda como ruido. Y la presencia de múltiples mínimo locales es debido al fenómeno de la degeneración, en el que diferentes combinaciones de parámetros estructurales dan como resultado perfiles SAS muy parecidos [10].

III.3.B. Optimización interna del CREASE, Algoritmo Genético de adaptación dinámica (GA)

Para la búsqueda de la combinación de valores de los parámetros estructurales que minimicen el error (SSE), el CREASE hace una optimización interna usando la metaheurística algoritmo genético (Genetic Algorithm) de adaptación

dinámica descrito en [18], GA de ahora en adelante, Figura 5, un tipo de algoritmo de optimización basado en población; en este enfoque, una población de pop soluciones candidatas (individuos), inicialmente aleatoria, evoluciona a lo largo de $gens$ generaciones en busca de la mejor solución posible, este proceso se basa en la teoría de la evolución de Darwin, donde los individuos más aptos de una generación continúan y son capaces de heredar sus "genes" a individuos de generaciones futuras [19].

Los genes son los k parámetros estructurales de la *shape*, codificados binariamente por el GA con $nloci$ bits por parámetro. Cada gen i puede tomar por lo tanto 2^{nloci} valores entre su mínimo x_{iL} y máximo x_{iU}, representando un individuo como una serie de $k*nloci$ bits, conocida como cromosoma.

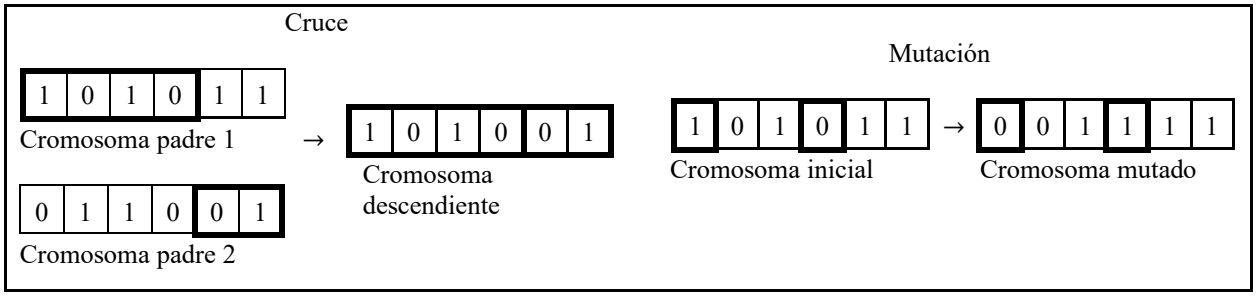

Figura 7. Ejemplo de la generación de nuevos individuos mediante la aplicación de las operaciones genéticas cruce y mutación a sus cromosomas. Fuente propia.

El proceso evolutivo del GA implica la aplicación sucesiva de tres operadores genéticos: selección, cruce y mutación [19]. En la selección, los individuos con mejor aptitud (*fitness*) tienen más posibilidades de ser elegidos para la reproducción en la próxima generación, el *fitness* de un individuo se calcula según la Ecuación 11, en función su SSE para este caso. En el cruce, dos individuos seleccionados (padres) tienen una probabilidad pc de generar un descendiente cuyo cromosoma será una combinación de sus cromosomas. La mutación introduce variabilidad genética cambiando uno o más bits del cromosoma de un individuo con una probabilidad pm. Tras aplicar estos operadores, los nuevos individuos reemplazan a los menos aptos en la población actual, conservando únicamente al mejor individuo (elitismo). Así se forma la nueva población, que se evalúa y luego se somete a los mismos procesos para dar lugar a la siguiente generación, este proceso se repite iterativamente $gens$ veces. En la Figura 7 se puede observar un ejemplo de la aplicación del cruce y la mutación sobre el cromosoma de unos individuos dados para generar nuevos individuos.

$$fitness = X(SSE_{max} - SSE) + Y$$

$$X = (cs - 1)\frac{SSE_{max} - SSE_{min}}{(SSE_{max} - SSE_{min}) - P} \qquad Y = (1 - X)P \quad P = \frac{1}{pop}\sum_{i}^{pop} SSE_{max} - SSE_i \tag{11}$$

En esta expresión el SSE_{max} y SSE_{min} son el valor máximo mínimo del SSE en la población, respectivamente, X y Y son factores de escala que buscan evitar que soluciones de baja aptitud (alto SSE) sean eliminadas prematuramente [10], y $cs = 10$ que es otro factor de escala [10]. Cuando el $I_{comp}(Q)$ de un individuo se acerca más a $I_{exp}(Q)$ su SSE disminuye y su $fitness$ aumenta. Se puede ver que para un individuo el valor su $fitness$ depende no solo de su SSE si no del de los demás individuos en la población actual (SSE_i).

En el GA los hiperparámetros probabilidad de mutación pm y probabilidad de cruce pc son dinámicos a lo largo de su ejecución, a diferencia de su implementación original en el que son constantes [19]; estos varían en función del GDM (Genetic Diversity Measure, medida de diversidad genética), Ecuación 12, buscando mantener una diversidad genética adecuada en la población a lo largo de la ejecución.

$$GDM = \frac{error\ mínimo\ de\ la\ población}{error\ promedio\ de\ la\ población} = \frac{SSE_{min}}{SSE_{avg}} \tag{12}$$

El GDM puede tomar valores entre 0 y 1, acercándose mas a 1 cuando los valores de error de la población, SSE, son más homogéneos, y a 0 cuando son más diversos. Los valores de pm y pc tienen un valor inicial de $pc_{initial}$ y $pc_{initial}$ respectivamente, y se actualizan de acuerdo con lo siguiente; cuando el valor del GDM alcanza un valor umbral superior (GDM_{max}), indicando que la población carece de suficiente diversidad, se aumenta pm y se reduce pc, ambos en un factor k_{GDM}, buscando aumentar la diversidad genética de la población en las siguientes generaciones. Cuando el GDM alcanza un umbral inferior (GDM_{min}), indicando que la población se ha vuelto demasiado diversa, se reduce pm y aumenta pc, en el mismo factor k_{GDM}, para reducir la diversidad genética de las poblaciones futuras. Los valores que puede tomar pm están restringidos al intervalo $[pm_{min}, pm_{max}]$, y los de pc al intervalo $[pc_{min}, pc_{max}]$ [10], [18].

Con lo anterior, son en total 11 hiperparámetros del GA; los valores optimizados de estos hiperparámetros usados por los autores para el análisis de soluciones diluidas de vesículas se pueden encontrar en [10]. Pero, debido a que el campo de la optimización obedece los teoremas NFL [25], no existe una configuración universalmente óptima en estos hiperparámetros para analizar todas la $shapes$ con el CREASE, por lo que al agregar una nueva *shape*, un ajuste de estos hiperparámetros sería lo más adecuado, buscando garantizar el desempeño de la herramienta que justifique su coste computacional.

Para el análisis de soluciones diluidas de vesículas, los autores sugieren que múltiples ejecuciones del CREASE pueden brindar información útil para que el usuario comprenda la degeneración en las dimensiones de las vesículas correspondientes al $I_{exp}(Q)$ [10] y sobre los resultados de estas ejecuciones hacer un promedio.

El equipo del CREASE en el artículo [16] reportó una modificación en el proceso de optimización interna implementando una versión en la que apoyan el proceso de optimización del GA con el uso de una red neuronal artificial, logrando una mejora en algunos casos en la velocidad de convergencia y en otros de la precisión, en el caso de estudio de bloques de polímeros anfifílicos en solución. El presente estudio se realizó sobre la versión del GA descrita antes, ya que es la más ampliamente documentada y es la que se encuentra implementada en su repositorio de GitHub.

III.3.C. Paquete *crease_ga*

La herramienta CREASE está programada en Python en una Arquitectura Modular, permitiendo con este enfoque dividir el paquete en módulos distintos y autónomos, cada uno responsable de funcionalidades específicas como funcionalidades del GA, cálculo de $I_{comp}(Q)$ de cada *shape*, manejo de excepciones, entre otras. Esta modularidad facilita el mantenimiento, escalabilidad y comprensión del código. En la Figura 8 se puede observar la arquitectura de la herramienta CREASE dispuesta en el repositorio del equipo del CREASE (github.com/arthijayaraman-lab/crease_ga), *crease_ga* de ahora en adelante.

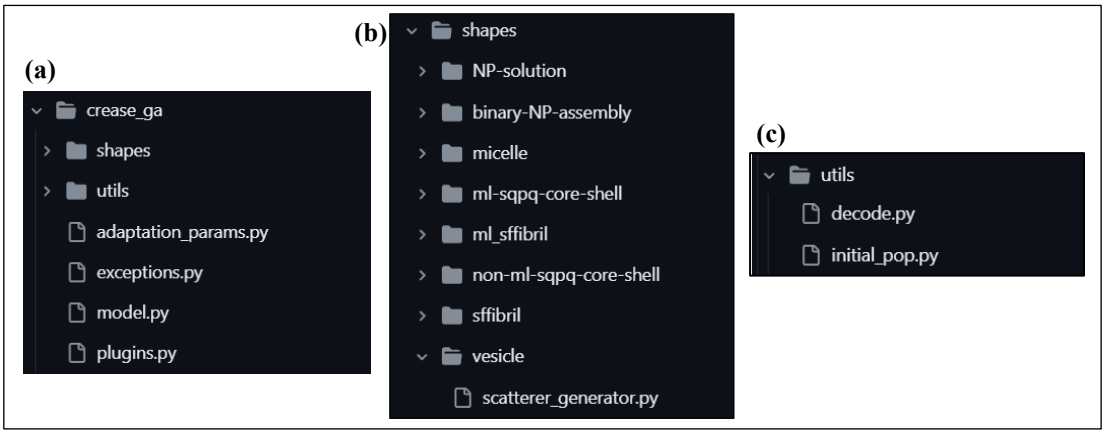

Figura 8. (a) Arquitectura modular del paquete *crease_ga*. Módulos contenidos en el directorio **(b)** *shapes* y **(c)** *utils*. Fuente propia.

El paquete *crease_ga* comprende una serie de módulos esenciales que desempeñan roles específicos dentro de su funcionamiento:

1. *model.py*: Este módulo es central en el paquete, ya que contiene la clase principal que define el modelo utilizado para analizar perfiles de dispersión $I_{exp}(Q)$. Se encarga de coordinar el proceso de optimización del GA, además de invocar las funcionalidades de otros módulos del paquete.
2. *adaptation_params.py*: Contiene la clase que gestiona los parámetros para la adaptación del GA, entregando a *model.py* en cada generación los hiperparámetros pm y pc actualizados según lo descrito en la subsección III.3.B.
3. *exceptions.py*: Define las excepciones personalizadas utilizadas en el paquete.
4. *plugins.py*: Administra el acceso a los *plugins* externos. En particular *crease_ga* permite a los usuarios desarrollar y contribuir con sus propias *shapes* a través de estos plugins externos, permitiendo que sean descubribles a través de los metadatos del paquete.
5. *shapes* (directorio): Contiene módulos con las funcionalidades de las *shapes* desarrolladas por el equipo CREASE. Cada *shape* tiene una clase contenedora en su respectivo módulo *scatterer_generator.py*, cuya función principal es calcular y devolver $I_{comp}(Q)$ a *model.py,* a partir de los parámetros estructures que este último le entrega.
6. *utils* (directorio): Contiene módulo de utilidad para el GA.
    a. *decode.py*: Entrega los parámetros estructurales de una población a partir de sus cromosomas binarios.
    b. *initial_pop.py*: Genera los cromosomas de la población inicial del GA.

Este paquete entre sus muchas funciones permite almacenar información del proceso de análisis de $I_{exp}(Q)$ a lo largo de la ejecución en archivos .txt, útiles para análisis posteriores, así como para retomar el proceso de análisis en caso de interrupción. Para su funcionamiento requiere de una versión de Python superior a la 3.8, para soportar las dependencias necesarias en su funcionamiento: *numpy*, *matplotlib* y *numexpr*. Mas detalle al respecto puede encontrarse en su repositorio y en la documentación del paquete *crease_ga* disponible en [13].

III.4. Harmony Search (HS)

La búsqueda armónica (Harmony Search, HS) es un tipo de algoritmo de optimización basado en población, inspirado en el proceso de improvisación musical, donde cada músico toca una nota dentro de un posible rango, de tal manera que forman un vector armónico, en el problema de optimización cada parámetro es representado por una nota y un vector armónico es una posible solución. Si el conjunto de notas tocadas por los músicos es considerado una buena armonía, esta es guardada en la memoria de cada músico, incrementando la posibilidad de hacer una buena armonía y este proceso se logra mediante aplicaciones repetidas de tres operadores: uso de la memoria armónica, ajuste de tono y aleatoriedad. Cabe resaltar que, así como el GA, el HS cuenta con múltiples variantes, entre las más conocidas, se encuentran: Improve Harmony Search (IHS), Global-Best Harmony Search (GHS), Novel Global-best Harmony Search (NGHS), Self-adaptative Global-best Harmony Search (SGHS) [27].

El HS es, como el GA, un algoritmo de optimización basado en población, que se basa en el proceso de improvisación musical, donde cada músico toca una nota x_i dentro de un posible rango $[x_{iL}, x_{iU}]$, formando un vector armónico x, en el problema de optimización del CREASE cada parámetro estructural es representado por una nota y un vector armónico es una posible solución. El HS está inspirado en un concepto simple lo que resulta en fácil implementación, pocos hiperparámetros, y fácil integración a otras metaheurísticas [26], [27].

Al igual que el GA el HS parte de una población inicial aleatoria de HMS (Harmony Memory Size) vectores armónicos que conforman la denominada memoria armónica (HM) inicial $x^1, x^2, \ldots, x^{HMS}$. Estos vectores harmónicos se evalúan, y a partir de estos, en cada iteración t se genera una nueva armonía $x' = (x'_1, x'_2, \ldots x'_k)$, que se evalúa y en caso de ser mejor que la peor armonía existente en la HM entra a reemplazarla, este proceso se repite iterativamente NI (Number of Iterations) veces; este proceso se puede ver representado de forma esquemática en la Figura 9.

El proceso de generar una nueva solución o armonía en el contexto del HS es llamado improvisación. En la Figura 10 se muestra el esquema de improvisación de la variable de decisión x'_i que conforma la nueva armonía x' usado por el HS.

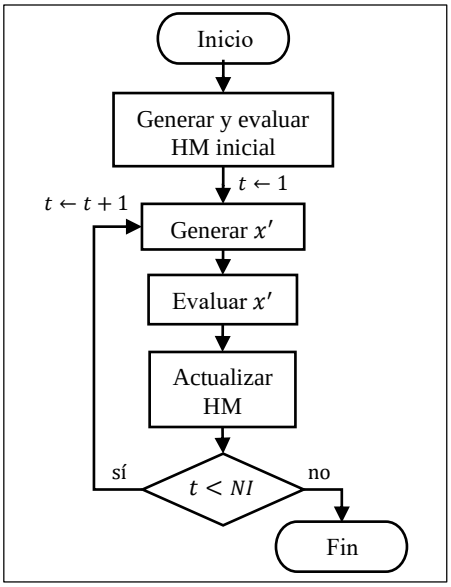

Figura 9. Diagrama de flujo esquemático del algoritmo HS. Fuente propia.

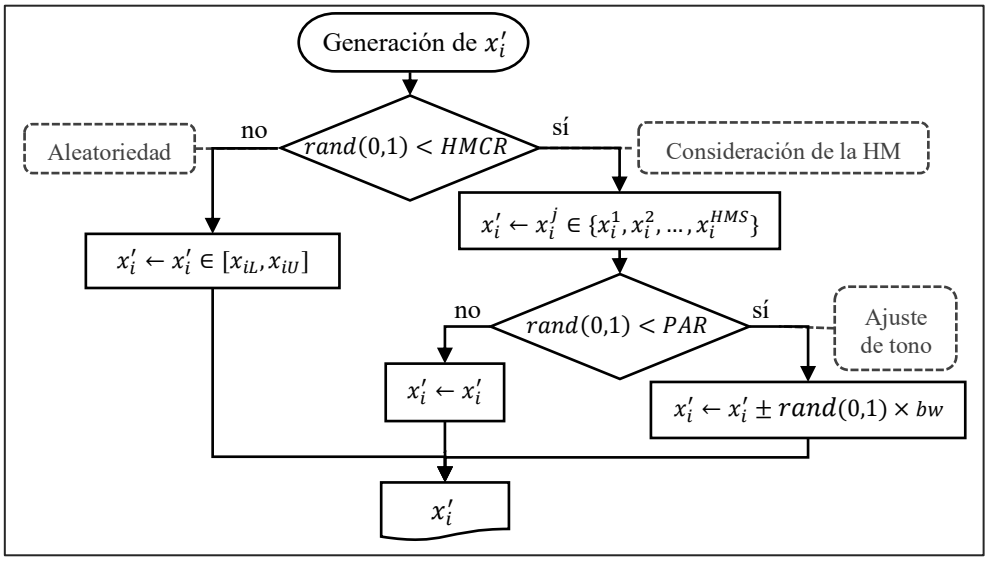

Figura 10. Esquema de improvisación de la variable de decisión x_i' de una nueva armonía x' usada por el HS. Fuente propia.

Donde $HMCR$ (Harmony Memory Consideration Rate) es la probabilidad de considerar la HM para la creación del x_i' y PAR (Pitch Adjusting Rate) es la probabilidad de aplicar un ajuste de todo entre $\pm bw$ (bandwidth) al x_i', buscando evitar el estancamiento en mínimos locales.

HS suele tener deficiencias como convergencia prematura y baja velocidad de convergencia. Por lo tanto, diversas variantes han sido planteadas para superar estas deficiencias. Estas variantes se ven reflejadas principalmente en el proceso de improvisación (Generar x') [27]. Para este estudio se escogieron las siguientes variantes en base a su desempeño reportado en la literatura.

III.4.A. Global-best Harmony Search (GHS)

En GHS la nueva armonía x' es generada a partir del historial de búsqueda, directamente de la mejor armonía actual x^{best} dentro del HM; es decir la nueva armonía está influenciada por la mejor posición hasta el momento encontrada, simplificando el procedimiento de ajuste de tono; este procedimiento está inspirado en la inteligencia de enjambre del

Particle Swarm Optimization (PSO), donde los individuos tienden a agruparse alrededor del mejor individuo [35]. En la Figura 11 se muestra el esquema de improvisación de la variable de decisión x_i' que conforma una nueva armonía x', usado por el GHS.

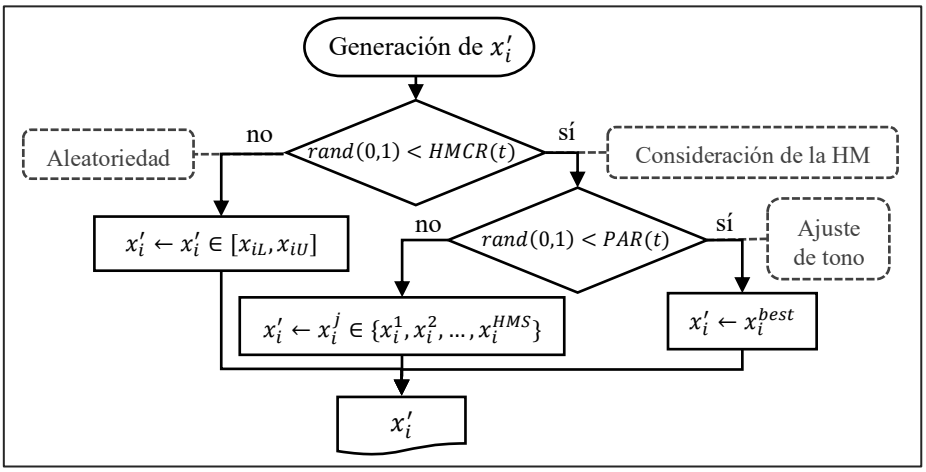

Figura 11. Esquema de improvisación de la variable de decisión x_i' de una nueva armonía x' usada por el GHS. Fuente propia.

III.4.B. Self-Adaptive GHS (SGHS):

Para la improvisación de una nueva armonía, como se ve en la Figura 12, el SGHS aprovecha la inteligencia de enjambre al usar la mejor armonía existente x^{best} en la HM, al igual que el GHS, y utiliza el bw para evitar estancamientos locales, como el HS. A diferencia de estos, el SGHS ajusta dinámicamente sus hiperparámetros; el bw cambia según la Ecuación 13, y para el $HMCR$ y PAR se asume una distribución normal con valores medios iniciales $HMCRm_o$ y $PARm_o$, y desviación estándar σ_{HMCR} y σ_{PAR}. A partir de esta distribución en cada iteración t se genera un nuevo valor de $HMCR(t)$ y $PAR(t)$, que se guardan cuando la armonía generada logra reemplazar la peor de la HM; después de un número especificado de iteraciones (LP), los valores medios de las distribuciones, $HMCRm$ y $PARm$, se actualizan como la media de los valores guardados, y se continua, actualizando dichos valores medios cada LP iteraciones. Los valores que pueden tomar el $HMCR$ y el PAR, están restringido a los intervalos $[HMCR_{min}, HMCR_{max}]$ y $[PAR_{min}, PAR_{max}]$, respectivamente. Con todo lo anterior el SGHS busca ajustarse gradualmente al problema específico y a las fases particulares del proceso de búsqueda [34].

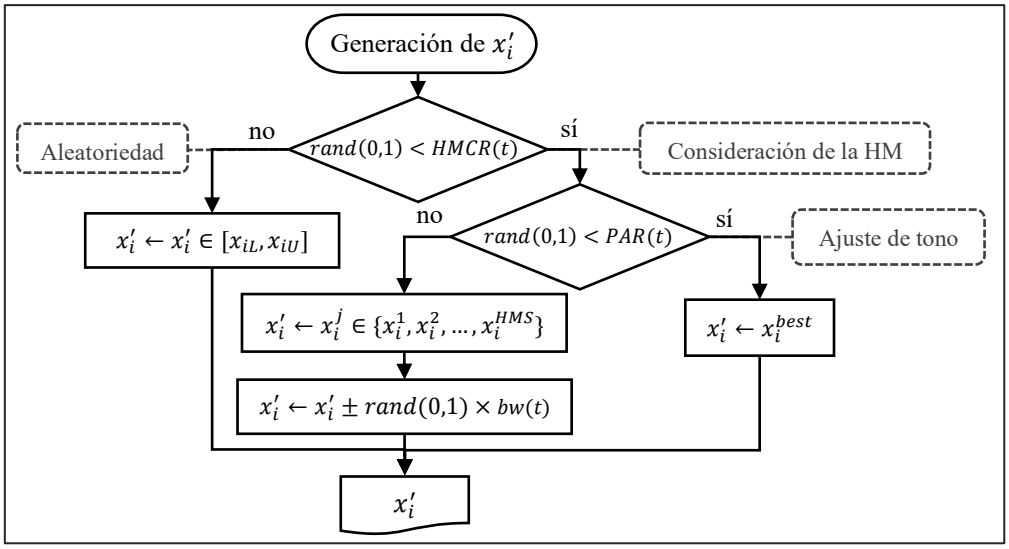

Figura 12. Esquema de improvisación de la variable de decisión x_i' de una nueva armonía x' usada por el SGHS. Fuente propia.

$$bw = \begin{cases} bw_{max} - \dfrac{bw_{max} - bw_{min}}{NI} \times 2t & ; t < NI/2 \\ bw_{min} & ; t \geq NI/2 \end{cases} \quad (13)$$

III.4.C. Novel Global-best Harmony Search (NGHS)

La estructura de NGHS es muy diferente del original HS en cuanto a que parámetros como $HMCR$, PAR y bw son eliminados, en su lugar NGHS incluye el paso adaptativo y la región de confianza x_R, y con estos factores se diseña un nuevo esquema de improvisación para que la peor armonía x^{worst} de la HM se mueva hacia la mejor armonía x^{best} de la HM en cada improvisación, beneficiándose de la inteligencia de enjambre al igual que el GHS. El uso de este nuevo enfoque de improvisación puede acelerar la tasa de convergencia; sin embargo, también acelera la convergencia prematura, estancándose pronto en mínimos locales; para superar esta desventaja, se introduce una mutación genética pm [33]. En la Figura 13 se muestra el esquema de improvisación de una variable de decisión x_i' que conforma una nueva armonía x' usado por el NGHS.

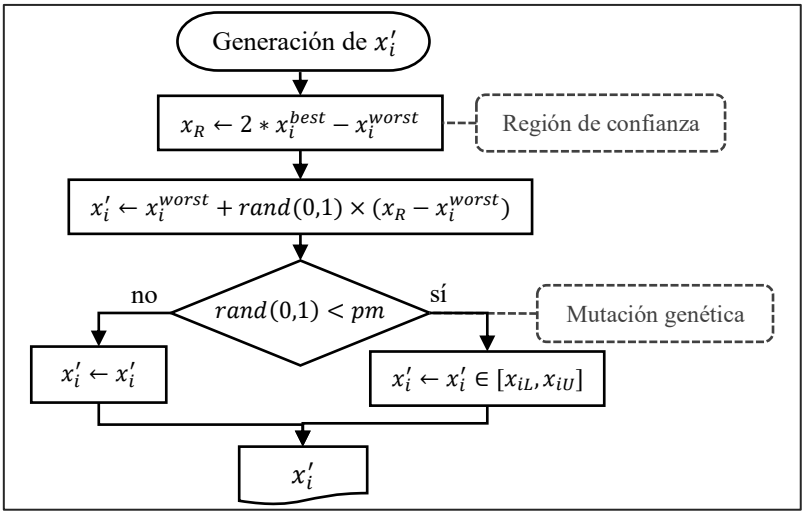

Figura 13. Esquema de improvisación de la variable de decisión x_i' de una nueva armonía x' usada por el NGHS. Fuente propia.

El valor de pm es sugerido que se escoja en función del número de variables de decisión (n_x) del problema, de la región de $\left[0.2 \times \frac{(1-50\%)}{n_x}, 0.2\frac{(1+50\%)}{n_x}\right]$ cuando $n_x \leq 4$ o, para $n_x > 4$, de la región $\left[\frac{(1-50\%)}{n_x}, \frac{(1+50\%)}{n_x}\right]$ [33].

III.4.D. SGHS2

De las metaheurísticas basadas en HS vistas el SGHS es el que más cantidad de hiperparámetros tiene, ascendiendo a once. Por lo anterior se decidió implementar las siguientes modificaciones a esta metaheurística: en lugar de permitir que el $HMCR$ y el PAR cambien de forma estadística, se mantendrán constantes, lo que resulta en una reducción de seis hiperparámetros. Obteniendo una metaheurística también muy parecida al GHS con el agregado del bw, que varía a lo largo de la ejecución según la Ecuación 13.

IV. Metodología

En este capítulo se describen los pasos metodológicos llevados a cabo para esta investigación. Se comienza con la modificación del paquete *crease_ga* para implementar las versiones de HS en la ejecución de CREASE. A continuación, se detallan los benchmarks utilizados para evaluar el desempeño de las metaheurísticas en CREASE, las configuraciones de ejecución, así como los recursos computacionales empleados en todas las ejecuciones. Se realiza un diagnóstico de la herramienta CREASE en su versión original para, posteriormente, llevar a cabo pruebas preliminares que permitan seleccionar la metaheurística basada en HS adecuada para comparar su desempeño con el GA en el uso del CREASE para el análisis de perfiles SAS de soluciones diluidas de vesículas.

IV.1. Implementación de las versiones de HS en el CREASE, modificaciones al paquete *crease_ga*: *crease_he*

Se realizaron modificaciones significativas en la arquitectura del paquete *crease_ga* para permitir la ejecución de la herramienta CREASE con algoritmos de optimización distintos al GA. Estas modificaciones implicaron la creación de un módulo contenedor para cada algoritmo de optimización, encargado de gestionar el proceso de optimización y realizar el proceso de almacenamiento de información necesaria para retomar el proceso en caso de interrupción, así como para análisis posterior. Esto otorgó a la herramienta CREASE una mayor flexibilidad y adaptabilidad al facilitar la implementación no solo de metaheurísticas basadas en HS, sino también de otros algoritmos de optimización, adicionales al GA.

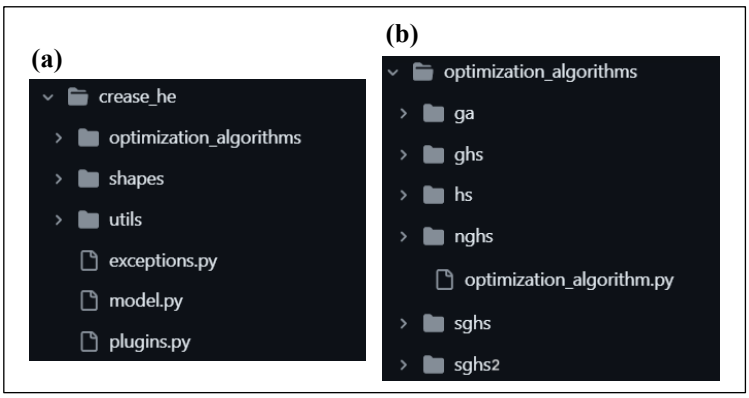

Figura 14. (a) Arquitectura del paquete *crease_he*. **(b)** Módulos de los algoritmos de optimización alojados en el directorio *optimization_algorithms.* Fuente propia.

El paquete resultante de estas modificaciones fue renombrado como *crease_he* y se encuentra en el repositorio de GitHub disponible en el siguiente enlace: github.com/cha-do/crease_heuristic/. En la Figura 14a se presenta la arquitectura del paquete *crease_he*. Se puede observar que, respecto al *crease_ga* (ver Figura 8), se eliminó el directorio *utils* y el módulo *adaptation_params.py*, ya que contenían funciones relacionadas con el funcionamiento específico del GA. Estas funciones se encuentran ahora alojadas en la clase contenedora del GA ubicada en el directorio añadido *optimization_algorithms*. Este directorio, como se muestra en la Figura 14b, contiene los módulos con las funcionalidades de cada algoritmo de optimización que *crease_he* puede utilizar. Cada algoritmo tiene una clase contenedora en su respectivo módulo *optimization_algorithm.py*, que gestiona su proceso de optimización, entregando a *model.py* las nuevas configuraciones de parámetros estructurales a evaluar, y recibiendo su evaluación correspondiente (SSE) para continuar el proceso de optimización. Con estas modificaciones, se implementó exitosamente el uso de todas las versiones de HS necesarias para el desarrollo de la investigación en el CREASE, utilizando las mismas dependencias que el *crease_ga*: *numpy, matplotlib* y *numexpr.*

IV.2. Descripción de los casos de estudio (Benchmarks)

Para las pruebas de desempeño del CREASE se utilizaron cuatro benchmarks correspondientes a cuatro perfiles SAS, $I_{exp}(Q)$, in silico de soluciones diluidas de vesículas con parámetros estructurales conocidos. Estos se obtuvieron, al igual que en [10], con el proceso descrito en la subsección III.3.A, usando de las Ecuaciones 9 y 10. Sus parámetros estructurales se escogieron buscando cubrir una variedad de combinaciones en las características relevantes como en

[10], así como procurando que tuvieran un coste computacional de análisis razonable debido al número considerable de ejecuciones que se realizaron en este estudio. Los parámetros estructurales escogidos para los benchmarks se listan en la Tabla 2, y sus respectivos perfiles SAS, $I_{exp}(Q)$, se pueden ver en la Figura 15.

Tabla 2. Parámetros estructurales de los benchmarks (B1, B2, B3 y B4), s_{Ain}=0.20, σ_R=20%.

	R_{core} [Å]	t_{Ain} [Å]	t_B [Å]	t_{Aout} [Å]
B1	100	120	60	120
B2	100	60	120	60
B3	150	120	60	120
B4	150	60	120	60

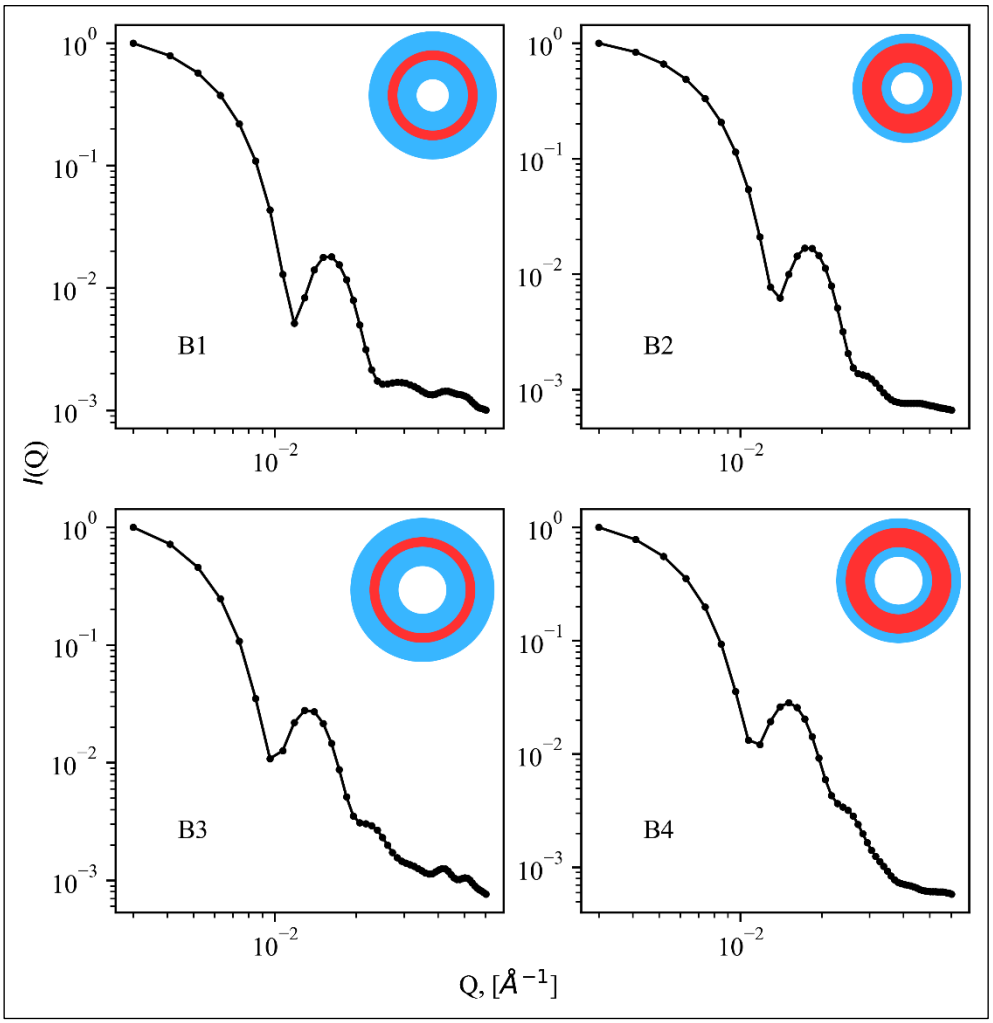

Figura 15. Perfil SAS, $I_{exp}(Q)$, de los cuatro benchmarks (B1, B2, B3 y B4) con una representación esquemática de las dimensiones de las vesículas en azul sus capas solvofílicas y en rojo su capa solvofóbica. Fuente propia.

La dispersión del radio del núcleo (σ_R) del 20% y la proporción de los dispersores solvofílicos totales presentes en la capa interna (s_{Ain}) del 0.20, captan el tipo de dispersión observada en muestras experimentales de ensamblajes de vesículas [10]. El rango de Q de $I_{exp}(Q)$ considerado para el análisis de los cuatro bechmarcks fue de 0.003Å^{-1} a 0.060Å^{-1} con un total de 53 valores de Q, suficientes para resolver las características clave de las soluciones diluidas de vesículas consideradas.

IV.3. Configuraciones de ejecución del CREASE

Todas las ejecuciones del CREASE realizadas sobre los bechmarks en esta investigación se hicieron con las siguientes configuraciones:

- En las ejecuciones con GA se usó la configuración de hiperparámetros recomendada para esa *shape* en [10], listada a continuación.
  $GDM_{max} = 0.85,\ \ GDM_{min} = 0.005,\ \ k_{GDM} = 1.1,\ \ pc_{initial} = 0.6,\ \ pc_{max} = 1,\ \ pc_{min} = 0.1,$ $pm_{initial} = 0.001, pm_{max} = 0.25, pm_{min} = 0.006, pop = 80$ y $gens = 100$
- Se usó una relación n_{sct}/N de 0.5, lo que permitió obtener resultados similares a los reportados por los autores de esta *shape* [10] con un costo computacional moderado.
- Los rangos de búsqueda $[x_{iL}, x_{iu}]$ usados para la búsqueda de los parámetros estructurales en las ejecuciones del CREASE fueron $[50\text{Å}, 250\text{Å}]$ para el R_{core}, $[30\text{Å}, 200\text{Å}]$ para el t_{Ain}, t_B y t_{Aout}, $[0.1, 0.45]$ para el s_{Aint}, $[0.0, 0.45]$ para el σ_R y [2.5,5.5] para el $-log_{10}(bg)$.
- Para las ejecuciones con GA se utilizó un $nloci$ de 7, permitiendo a cada uno de los 7 parámetros estructurales tomar 128 posibles valores. Para las versiones del HS, ya que se permite al usuario escoger la precisión de cada parámetro, se usó para R_{core}, t_{Ain}, t_B y t_{Aout} cero decimales y para el s_{Aint}, σ_R y $-log_{10}(bg)$ 2 decimales. Con esta elección, tanto para el GA como para las versiones de HS, la cantidad total de posibles combinaciones de parámetros estructurales sería de un orden de $\sim 10^{14}$.

IV.4. Recursos computacionales

Para llevar a cabo las pruebas necesarias para el desarrollo de esta investigación, se contó con los recursos computacionales detallados en la Tabla 3. Estos recursos estaban ubicados en la Sala de Sistemas y el Laboratorio de Física Moderna del Departamento de Física de la Universidad del Cauca, así como en el Servidor del Departamento de Telemática de la misma universidad (Servidor Telemática) y en los servidores del servicio Google Cloud Platform (GCP, cloud.google.com).

Tabla 3. Recursos computacionales usados.

No.	Ubicación	Tipo	CPU	Núcleos		RAM	Cantidad
				Físicos	Lógicos		
1	Sala	Ordenador	Intel i7 5th gen	6	12	16Gb	11
2	Sala	Ordenador	Intel i5 5th gen	6	6	16Gb	1
3	Sala	Ordenador	AMD 10	2	4	16Gb	11
4	Laboratorio	Ordenador	Intel i9 7th gen	8	16	32Gb	1
5	GCP	Máquina virtual	AMD EPYC™ 7B13	4	8	16Gb	4
6	Servidor Telemática	Máquina virtual	Intel Xenon	8	8	16Gb	5

Para la realización de las ejecuciones se tuvo en cuenta que la herramienta CREASE esta codificada para realizar el procesamiento necesario en la CPU, y puede configurarse fácilmente para aprovechar sus múltiples núcleos con multiprocesamiento.

Su administración se hizo mediante la herramienta a AnyDesk, a excepción de las máquinas virtuales del servicio de GCP, que se administraron mediante su plataforma. Y se usaron según su disponibilidad.

Para garantizar la correcta ejecución del *crease_he*, en cada uno de los recursos mencionados, se clonó el repositorio del paquete y se instaló Python 3.10 junto con las dependencias *numpy*, *matplotlib* y *numexpr* en un entorno virtual gestionado por Miniconda.

IV.5. Diagnóstico del CREASE original (CREASE-GA):

Aunque el estudio sobre el uso del CREASE utilizando la metaheurística GA, CREASE-GA de ahora en adelante, para el análisis de perfiles SAS de soluciones diluidas de vesículas [10] proporciona información sobre el rendimiento, los alcances y las limitaciones de la herramienta en el análisis de este tipo de muestras, fue necesario realizar una revisión

más exhaustiva para cumplir los objetivos de este trabajo. Por lo tanto, se replicaron las pruebas reportadas utilizando la herramienta CREASE-GA en el análisis de los cuatro benchmarks, descritos en la subsección IV.2, cada uno se analizó 31 veces para obtener una muestra estadística significativa que permitiera realizar un análisis detallado. Utilizando esta información se identificaron las fortalezas y los puntos críticos que debían abordarse en la herramienta, prestando especial atención a aquellos relacionados con la metaheurística GA.

Los resultados obtenidos de este diagnóstico permitirán establecer un comportamiento base, a partir del que se podrá comparar los resultados de las pruebas posteriores sobre el CREASE.

IV.6. Determinación y ajuste de la metaheurística basada en HS adecuada para la comparación

Debido a que el campo de la optimización obedece los teoremas NFL [25], es imposible establecer de antemano qué versión de HS y con que configuración de hiperparámetros es la más adecuada para abordar el problema de optimización del CREASE, por lo que para determinar cuál de las metaheurística basada en HS consideras (HS, GHS, SGHS, NGHS y SGHS2) sería la elegida para la comparación final y con qué hiperparámetros, se decidió realizar pruebas preliminares.

Como ya se mencionó, estas ejecuciones tienen un costo computacional considerable por lo que para estas pruebas preliminares, para balancear el costo computacional y la fiabilidad de los resultados, se determinó realizar doce ejecuciones independientes de cada algoritmo sobre cada uno de los benchmarks, hasta alcanzar 3600 cálculos de $I_{comp}(Q)$ (correspondientes a 45 generaciones del CREASE con GA); debido a que todas las pruebas ejecutadas mostraron convergencia por debajo de este número de evaluaciones. Para las pruebas, cada una de las doce ejecuciones independientes tiene una semilla de aleatoriedad que permitiría su repetibilidad. La metodología utilizada en estas pruebas preliminares se basó en Iterative Field Research [41] que implica ciclos iterativos de investigación, diseño y evaluación de resultados para guiar el siguiente paso.

Estas pruebas preliminares consistieron en 3 ciclos. En el ciclo 1 se realizó comparación del desempeño entre las metaheurísticas HS, GHS, SGHS, NGHS y SGHS2 en el CREASE sobre los cuatro benchmarks. Esta comparación se realizó principalmente a través de las curvas de convergencia promedio para seleccionar la metaheurística más adecuada. Sobre la metaheurística seleccionada en el primer ciclo, se realizó un afinamiento de sus hiperparámetros en el ciclo 2 y se evaluaron estrategias para evitar la convergencia prematura en el ciclo 3. Finalmente, se determinó cuál de las configuraciones evaluadas era la más adecuada para continuar al siguiente paso: la comparación con el GA en el desempeño del CREASE.

IV.7. Comparación del GA y la metaheurística basada en HS en el desempeño del CREASE

Para hacer una comparación con validez estadística, justa y fiable del desempeño del CREASE con GA y la metaheurística basada en HS, se determinó hacer sobre cada uno de los 4 bechmarcks un total 31 ejecuciones independientes completas, para obtener una estimación precisa del rendimiento medio y evaluar la variabilidad de los resultados. Estas ejecuciones se realizaron de forma que ambas metaheurísticas calculen la misma cantidad de $I_{comp}(Q)$, 8000.

Los resultados obtenidos de estas ejecuciones se utilizaron para comparar el desempeño de las dos metaheurísticas. Se emplearon pruebas no paramétricas de Wilcoxon de rangos con signo, se compararon directamente los promedios de los resultados finales de las ejecuciones y sus curvas de convergencia, y se visualizó mediante un SOM entrenado en el paisaje de búsqueda de cada benchmark las soluciones obtenidas de las ejecuciones por las dos metaheurística.

V. Resultados

En esta sección se presentan y analizan los resultados de las pruebas realizadas en esta investigación. En primer lugar, se muestran los resultados del diagnóstico del CREASE-GA, posteriormente se muestra el proceso de selección de la metaheurística basada en HS (NGHS) y el ajuste de sus hiperparámetros, para la ejecución del CREASE. Y, finalmente, se ven los resultados de la comparación entre el NGHS y el GA en el desempeño del CREASE.

V.1. Diagnóstico del CREASE-GA

A continuación, se verá el diagnóstico realizado al CREASE-GA a partir de las 31 ejecuciones realizada a cada uno de los benchmarks, para esto se tuvo en cuenta las curvas de convergencia, los parámetros estructurales obtenidos, así como la cantidad de configuraciones de parámetros estructurales (soluciones) y el número de veces que se evaluaron a lo largo de las ejecuciones. Los tiempos de ejecución no se pudieron analizar cuantitativamente debido a la variedad de los recursos computacionales usados en las ejecuciones, sí se observó que los tiempos para analizar el mismo benchmark era consistente entre ejecuciones y que era del orden de horas, la gran mayoría de este tiempo invertido en los cálculos de $I_{comp}(Q)$. Los benchmarks organizados de mayor a menor tiempo de ejecución son: B3, B4, B1 y B2; esto concuerda con lo esperado, ya que en se orden disminuyen sus dimensiones respectivas, por lo que la cantidad de subunidades necesarias también.

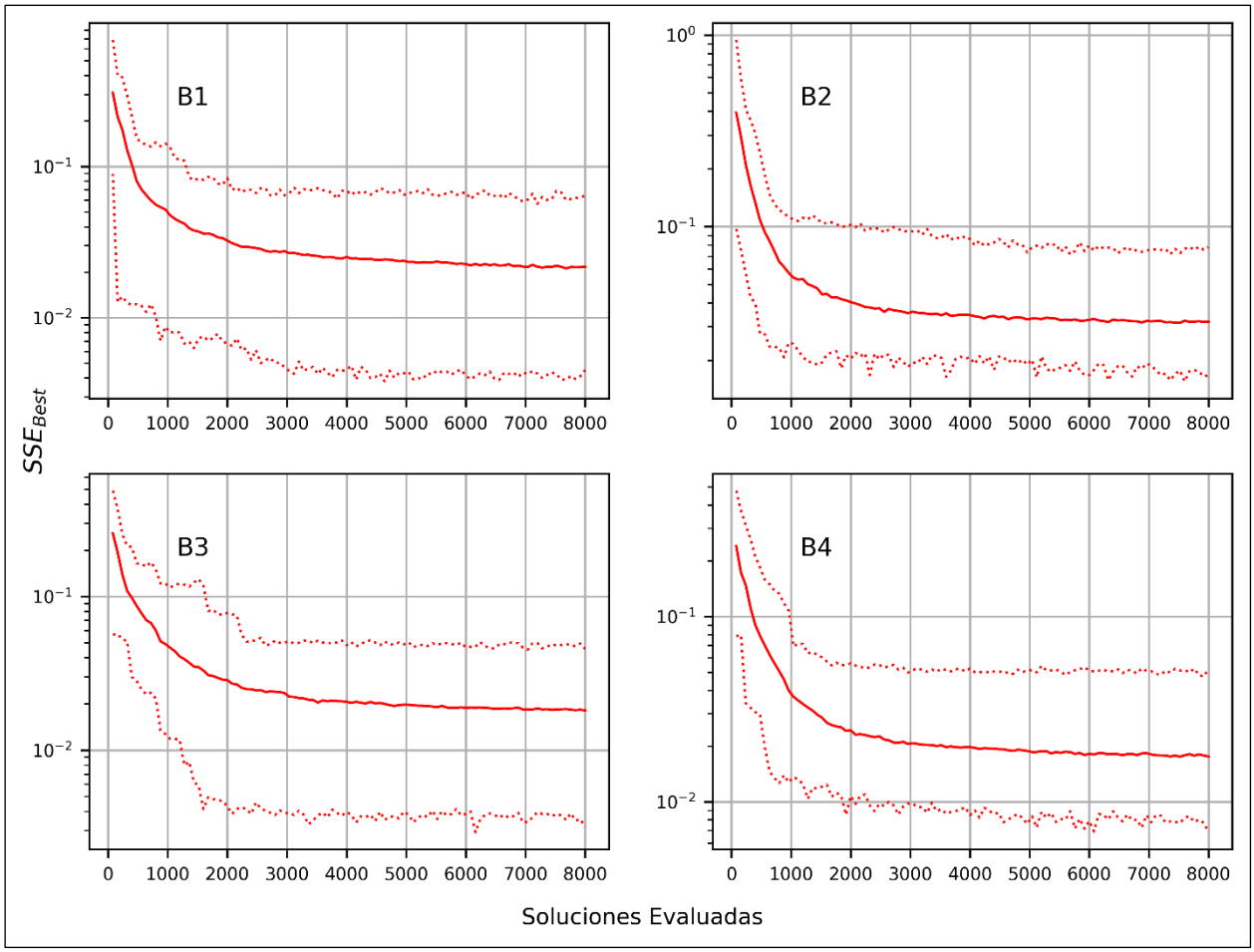

Figura 16. Curvas de convergencia promedio, mínima y máxima de las 31 ejecuciones del CREASE-GA sobre todos los bechmarks (B1, B2, B3 y B4). La línea continua es el valor promedio y las punteadas corresponden a la mínima (línea inferior) y máxima (línea superior) de todas las ejecuciones. Fuente propia.

V.1.A. Curvas de convergencia

Las curvas de convergencia del CREASE-GA representan el mejor (menor) SSE obtenido en función de la cantidad de soluciones evaluadas durante la ejecución de la herramienta. Para facilitar su análisis, se decidió promediar estas curvas. En la Figura 16 se presentan las curvas de convergencia promedio, así como las curvas mínima y máxima, de las 31 ejecuciones del CREASE-GA para cada benchmark. Las curvas de convergencia presentadas a continuación tendrán el eje vertical (mejor SSE, SSE_{Best}) representado en escala logarítmica, que es una representación usual [10], [16], útil para mejorar la visualización y facilitar el análisis de los resultados, se debe tener en cuenta que la escala logarítmica amplifica las diferencias.

De la Figura 16, se puede observar que en promedio el GA converge después de evaluar aproximadamente entre 5000 y 7000 soluciones, equivalente a 62 y 87 generaciones completas, respectivamente, siendo B1 en el que más tarda. Por otro lado, se pude notar que el valor del SSE_{Best} tiene un comportamiento ruidoso, aumentando en algunos momentos y disminuyendo en otros; esto puede parecer contra intuitivo pero es la consecuencia de que, como se mencionó antes, el SSE de una misma solución no siempre da el mismo valor, al evaluarse en una generación la mejor solución de la anterior, que se conservó debido al elitismo, el valor de su SSE puede dar mayor que el previamente obtenido que será el nuevo mejor si en el resto de la población no consigue menor SSE.

En la Tabla 4 se listan los valores promedios con sus respectivas desviaciones estándar del SSE_{Best} obtenido al final de las ejecuciones para cada benchmark, lo primero que salta a la vista es el valor considerable de la desviación respecto al valor medio, esto es un comportamiento usual en esta métrica [10], [16]. En B4 fue en el que menores valores de SSE se obtuvieron y mientras que en B2 en el que mayores.

Tabla 4. Mejor SSE (promedio y desviación estándar) obtenido en las ejecuciones del CREASE-GA para los benchmarks (B1, B2, B3 y B4).

B1	B2	B3	B4
0.0217 ± 0.0144	0.0318 ± 0.0147	0.0181 ± 0.0129	0.0176 ± 0.0116

V.1.B. Parámetros estructurales

Los valores promedio y deviación estándar de los parámetros estructurales obtenidos para cada benchmark en las ejecuciones del CREASE-GA se listan en la Tabla 5, exceptuando el $-log(I_{bg})$, cuyo valor no aporta información estructural de la muestra. Estos parámetros muestran una desviación estándar considerable, y aunque los valores promedio en algunos casos se acercan bastante a los esperados (Target), en otros se alejan notablemente. Este comportamiento coincide con lo reportado por los autores de la *shape* [10], y se debe en parte a la relación n_{sc}/N usada en el análisis y al fenómeno de la degeneración.

Tabla 5. Parámetros estructurales reales (Target) y obtenidos (promedio y desviación estándar) en las ejecuciones del CREASE-GA para los benchmarks (B1, B2, B3 y B4).

		R_{core}[Å]	t_{Aint} [Å]	t_B [Å]	t_{Aout} [Å]	σ_{Rcore} [%]	s_{Ain} [%]
B1	Target	100	120	60	120	20	20
	GA	106.5 ± 32.0	107.0 ± 33.3	81.8 ± 17.3	97.4 ± 18.8	18.3 ± 5.3	25.2 ± 10.6
B2	Target	100	60	120	60	20	20
	GA	98.6 ± 21.6	62.1 ± 20.5	123.6 ± 8.4	56.6 ± 9.5	21.5 ± 5.2	24.9 ± 10.5
B3	Target	150	120	60	120	20	20
	GA	143.0 ± 38.3	122.4 ± 39.9	78.7 ± 19.4	96.2 ± 21.4	22.4 ± 7.4	25.7 ± 11.2
B4	Target	150	60	120	60	20	20
	GA	127.3 ± 32.5	81.8 ± 32.4	124.6 ± 7.6	41.8 ± 10.7	25.1 ± 6.9	23.99 ± 12.0

Otro factor que puede influir es la discretización binaria que el GA aplica a los parámetros estructurales, lo cual podría impedir la evaluación de configuraciones más cercanas al objetivo. En particular, aunque $-log(I_{bg})$ no aporta información estructural, tiene un impacto considerable en el valor del SSE de una configuración, ya que, como se muestra en la Ecuación 9, I_{bg} se suma al valor de $I_{comp}(Q)$ en todo el rango de Q, afectando significativamente su ajuste a $I_{exp}(Q)$. Por lo tanto, una mayor precisión en la búsqueda de este parámetro es deseable; esto podría

solucionarse en el GA aumentando el valor de $nloci$, lo que aumentaría la precisión con la que se evalúan todos los parámetros. Sin embargo, esto también incrementaría exponencialmente la cantidad de posibles soluciones entre las que el GA tendría que buscar, comprometiendo potencialmente su velocidad de convergencia. Por este motivo, permitir al usuario escoger la precisión con la que se buscará cada parámetro estructural parece ser la opción más adecuada. Además, hacerlo de forma decimal en lugar de binaria facilitaría su comprensión.

V.1.C. Exploración del espacio de búsqueda del CREASE-GA

Con el fin de evaluar la eficacia del CREASE-GA en la exploración del espacio de búsqueda, se analizó el porcentaje de soluciones únicas ($\%SU$) evaluadas en cada ejecución, para las 31 ejecuciones de cada benchmark. En promedio sobre todos los benchmarks, solo el 20.8% de las soluciones evaluadas fueron únicas, mientras que el 79.2% restante correspondió a reevaluaciones de soluciones, de las cuales solo el 1.25% se debió al elitismo. Este alto porcentaje de reevaluaciones es preocupante, ya que, aunque reevaluar puede ser útil para considerar el ruido del SSE, el GA no controla adecuadamente la repetición de soluciones en la generación de nuevas poblaciones, resultando en poblaciones con múltiples soluciones repetidas. Además, no se aprovecha la información obtenida de la reevaluación de soluciones, lo que se refleja en el comportamiento ruidoso de las curvas de convergencia del CREASE-GA en la Figura 16.

Para analizar la cantidad de veces que se evalúa una misma solución a lo largo de una ejecución, ES de ahora en adelante, se creó un histograma de frecuencias de ES para cada benchmark con los resultados de las 31 ejecuciones del CREASE-GA. Los resultados, mostrados en la Figura 17 con ejes en escala logarítmica para facilitar la visualización, revelan que para todos los benchmarks el máximo de ES que se alcanzó en las 31 ejecuciones fue del orden de $\sim 10^3$, siendo en promedio de 681. Esto indica que, en promedio, el CREASE-GA invierte aproximadamente una doceava parte de las evaluaciones totales en reevaluar una misma solución, llegando en algunos casos hasta una quinta parte. La Tabla 6 presenta para cada benchmark: el $\%SU$, el promedio del máximo de ES ($\overline{ES_{max}}$), y los porcentajes de evaluaciones correspondientes a soluciones evaluadas más de una ($\%[ES > 1]$), diez ($\%[ES > 10]$), cien ($\%[ES > 100]$) y mil ($\%[ES > 1000]$) veces.

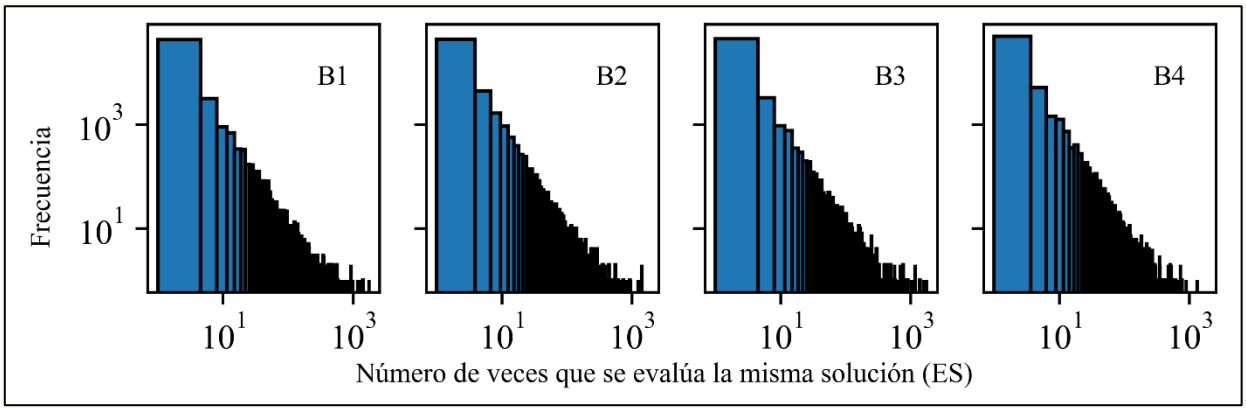

Figura 17. Histograma de frecuencia del número de veces que se evalúa la misma solución (ES) en una ejecución del CREASE-GA, para las 31 ejecuciones de cada benchmark (B1, B2, B3 y B4). Fuente propia.

Tabla 6. Métricas de evaluación de la cantidad de veces que se evalúan las mismas soluciones por parte del CREASE-GA en una ejecución, para las 31 ejecuciones de cada benchmark (B1, B2, B3 y B4).

	$\%SU$	$\overline{ES_{max}}$	$\%[ES > 1]$	$\%[ES > 10]$	$\%[ES > 100]$	$\%[ES > 1000]$
B1	19.8	772.5	87.8	65.2	34.2	3.3
B2	21.2	660.4	87.3	61.7	28.2	2.1
B3	20.0	727.5	87.6	64.8	34.3	4.1
B4	22.1	569.7	86.8	59.9	25.8	0.4

Lo anterior pone en evidencia una inversión excesiva de cómputo en la reevaluación de soluciones por parte del CREASE-GA, esto a pesar de la adaptación dinámica del GA, descrita en la sección III.3.B. Reducir las evaluaciones

repetidas podría mejorar la eficiencia del algoritmo y su capacidad para explorar el espacio de búsqueda de manera más efectiva.

V.1.D. Consideraciones en las versiones de HS a partir del diagnóstico del CREASE-GA

A partir del diagnóstico del CREASE-GA, se identificaron dos factores principales que se pueden mejorar en el proceso de optimización del CREASE, que se buscaron abordar en las versiones de HS consideradas:

a. Discretización de los parámetros estructurales:

La discretización binaria usada por el GA es poco versátil e intuitiva. Por lo anterior, en las versiones de HS se habilitó la opción de permitir al usuario elegir la precisión de cada parámetro estructural de forma independiente y en formato decimal en lugar de binario. Para las pruebas realizadas en esta investigación se les dio una precisión a los parámetros R_{core}, t_{Ain}, t_B y t_{Aout} de cero decimales y para el s_{Aint}, σ_R y $-log_{10}(bg)$ de 2 decimales. Esto mantiene un total de posibles soluciones del mismo orden ($\sim 10^{14}$) que el usado por el GA, pero distribuye los valores de manera más adecuada, mejorando la discretización del espacio de búsqueda para la optimización. Por ejemplo, el parámetro $-log(I_{bg})$ ahora puede tomar 300 posibles valores en lugar de los 128 considerados por el HS; lo que es deseable en vista del peso que tiene en el cálculo de $I_{comp}(Q)$, como se puede ver en la Ecuación 9, y por lo tanto de su ajuste a $I_{exp}(Q)$.

b. Control de reevaluaciones de la misma solución:

El GA no hace un control adecuado del número de reevaluaciones de una solución en su ejecución, conllevando un gasto considerable de cómputo, además de que no hace uso provechoso de la información obtenida de dichas reevaluaciones. Para abordar esto, en las versiones de HS se añadió a la HM la información sobre la cantidad de veces que una solución (armonía) ha sido evaluada en la ejecución, conservando el menor SSE obtenido de estas evaluaciones.

Se limitó la cantidad de reevaluaciones de una misma solución de la HM a un valor configurable por el usuario (*mct*), que se estableció en 10 para todas las pruebas realizadas. Una vez una solución de la HM se ha evaluado *mct* veces, se agrega a una "lista tabú" para evitar más reevaluaciones, validando que las nuevas soluciones candidatas no sean iguales a ninguna solución en la lista. La información del menor SSE y la cantidad de evaluaciones de una solución deja de ser considerada una vez esta deja de ser parte de la HM, buscando facilitar el manejo de la información por parte de la metaheurística.

V.2. Determinación y ajuste de la versión de HS

Una vez realizado el diagnóstico del CREASE-GA e implementadas las configuraciones en las versiones de HS consideradas, buscando solucionar las falencias del proceso de optimización encontradas en el diagnóstico, se procedió a la determinación y ajuste de la metaheurística basada en HS adecuada para el CREASE. A continuación, se describen los resultados y análisis de los ciclos de investigación que se llevaron a cabo para la elección de la versión de HS adecuada para el CREASE, su posterior afinamiento de hiperparámetros y evaluación de estrategias de convergencia prematura; para su análisis se tuvo en cuenta principalmente las curvas de convergencia. Cabe resaltar que estas pruebas se hicieron sobre doce ejecuciones independientes buscando tener un equilibrio entre la fiabilidad de los resultados y el coste computacional.

V.2.A. Ciclo I: Comparación del desempeño de las versiones de HS

Para este primer ciclo se buscó determinar cuál versión de HS sería la más adecuada para su uso en el CREASE. Se consideraron las metaheurísticas HS, GHS, SGHS, NGHS y SGHS2, descritas en la sección III.4, con un HMS de 20 y 3600 iteraciones (NI), y sus otros hiperparámetros basados en valores reportados en la literatura:

- HS: $PAR = 0.33, HMRC = 0.85,\ bw = 5\%$
- GHS: $PAR = 0.33, HMRC = 0.85$
- SGHS: $PARm_o = 0.9, \sigma_{PAR} = 0.05, HMCRm_o = 0.98\ , \sigma_{HMCR} = 0.01, bw_{max} = 5\%, bw_{min} = 1\%, LP = 100, PAR_{min} = 0.0, PAR_{max} = 1.0, HMCR_{min} = 0.9, HMCR_{max} = 1.0$
- NGHS: $pm = 0.14$
- SGHS2: $PAR = 0.33, HMRC = 0.85,\ bw_{max} = 5\%, bw_{min} = 1\%,$

En la Figura 18 se muestran las curvas de convergencia promedio del mejor SSE (SSE_{Best}) en escala logarítmica en función del número de soluciones evaluadas durante las ejecuciones del CREASE con cada versión del HS sobre cada benchmark, acompañadas de la curva de convergencia promedio de doce ejecuciones del CREASE-GA, para su comparación. Se presentan el promedio de estas curvas para facilitar su análisis.

Las curvas de convergencia promedio evidenciaron un buen comportamiento en todos benchmarks en los algoritmos HS, NGHS y SGHS2, mejoran al GA en velocidad de convergencia y en SSE_{Best} alcanzado, exceptuando el NGHS en B1 que mostró signos de convergencia prematura, quedándose estancado en un SSE_{Best} más alto que las demás metaheurísticas consideradas. En cuanto a la velocidad de convergencia, NGHS destacó sobre los otros algoritmos en todos los benchmarks.

Por lo tanto, se decidió seleccionar NGHS como la mejor opción, apoyado también en que es la metaheurística que tiene menor cantidad de hiperparámetros, lo que facilitaría su ajuste y uso por parte de los usuarios de CREASE, dando espacio para futuras modificaciones. Para abordar su problema de convergencia prematura, se evaluó el afinamiento de sus hiperparámetros (Ciclo II), y la diversificación de la HM (Ciclo III).

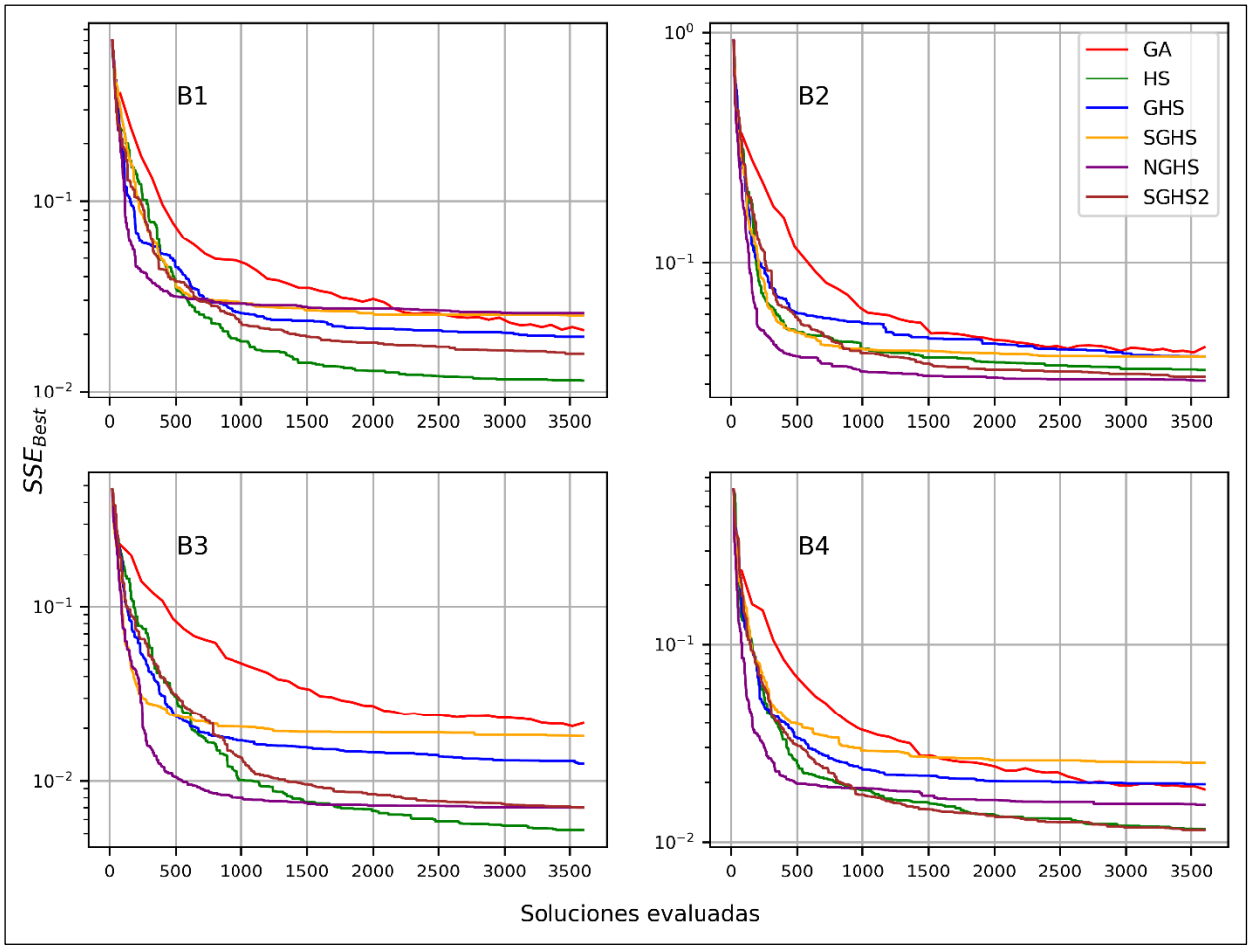

Figura 18. Curvas de convergencia promedio de doce ejecuciones del CREASE sobre todos los bechmarks (B1, B2, B3 y B4) con las versiones de HS consideradas, y con el GA. Fuente propia.

V.2.B. Ciclo II: Ajuste de hiperparámetos del NGHS

En este ciclo, se buscó la combinación de hiperparámetros de NGHS que brinde el mejor desempeño del CREASE en los benchmarks, tanto en velocidad como en SSE_{Best}. El ajuste de los hiperparámetros de NGHS es sencillo, ya que solo tiene dos: HMS y pm. Los valores probados de HMS fueron 5, 10, 20 y 40, y para pm siguiendo las recomendaciones de los autores [33], ya que el problema de optimización tiene siete variables de decisión, el intervalo

recomendado para su escogencia es $[0.07, 0.21]$, por lo que se escogió 0.07, 0.14 y 0.21. Las doce combinaciones de HMS y pm resultantes se probaron con doce ejecuciones independientes sobre B1, ya que fue en el que el NGHS mostró el desempeño más deficiente en el Ciclo I, las curvas de convergencia resultante pueden observarse en la Figura 19.

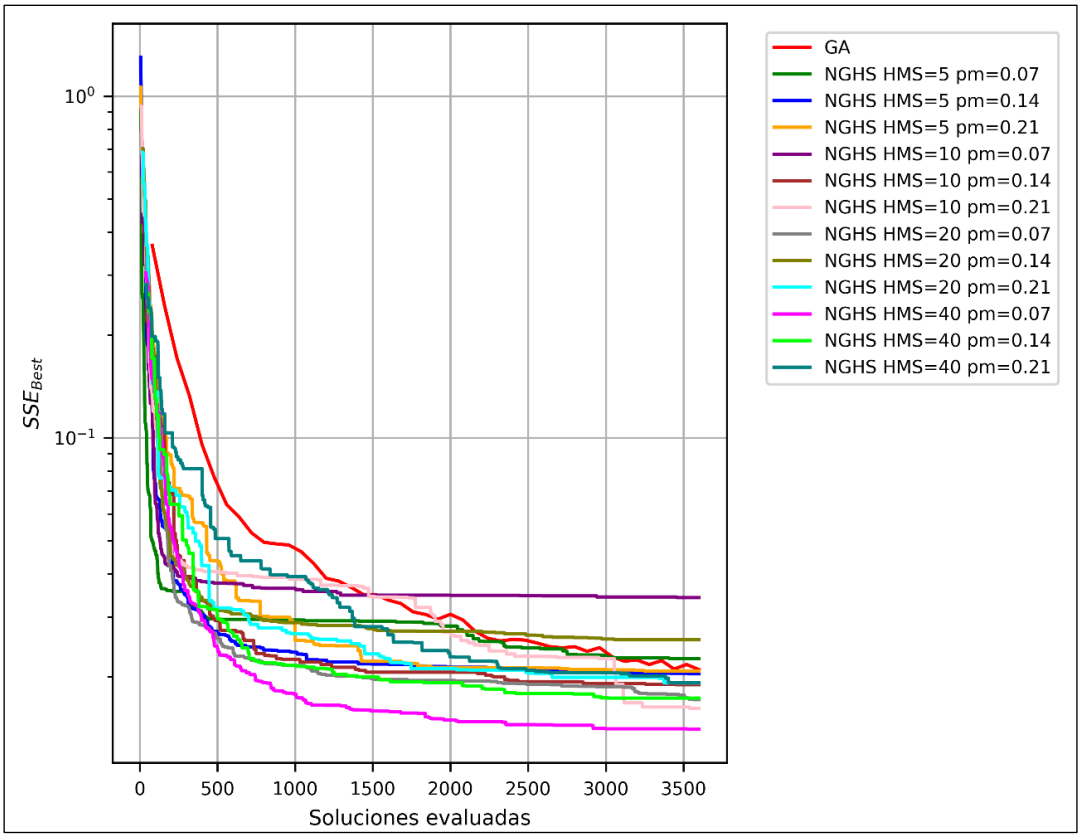

Figura 19. Curvas de convergencia promedio de doce ejecuciones del CREASE sobre B1 con GA y con NGHS, para las doce combinaciones de HMS y pm consideradas. Fuente propia.

Al analizar los resultados, considerando para cada HMS evaluado el pm con el que obtuvo un mejor desempeño se observa que las combinaciones resultantes son para HMS pequeños y pm intermedio, y con HMS grandes y pm bajos, siendo (5, 0.14), (10, 0.14), (20, 0.07), y (40, 0.07). Estas combinaciones se evaluaron en los benchmarks restantes. Los resultados se observan en la Figura 20, acompañadas de los resultados de la configuración consideradas en el Ciclo I: (20, 0.14).

Al analizar las curvas de convergencia promedio, se puede observar que la configuración (HMS, pm) (40, 0.07) presenta una velocidad de convergencia levemente menor que las demás en todos los benchmarks, pero es la que mejor SSE_{Best} alcanza para B1, B2 y B4, quedándose en B3 por detrás de la configuración (20, 0.14). Las otras dos configuraciones que mostraron un buen desempeño fueron (20, 0.07) y (20, 0.14), presentando una velocidad de convergencia muy parecida entre sí. Por su parte, la configuración (20, 0.14) obtuvo en promedio el mejor SSE_{Best} en B3 mientras que, en B1, B2 y B4, fue la 5ta, 2da y 3ra configuración con mejor SSE_{Best}, respectivamente. Mientras que (20, 0.07) fue, para B1, B2, B3 y B4, la 2da, 5da, 4ta y 2da configuración con mejor SSE_{Best}, respectivamente. Las demás configuraciones evaluadas mostraron un menor desempeño en general.

A partir del análisis de curvas de convergencia se determinó que la configuración (40, 0.07) fue la que presentó un mejor desempeño de forma consistente, seguida de (20, 0.14); siendo estas las configuraciones escogidas para la evaluación de estrategia de convergencia prematura en el Ciclo III, en busca de posibles mejoras en su desempeño.

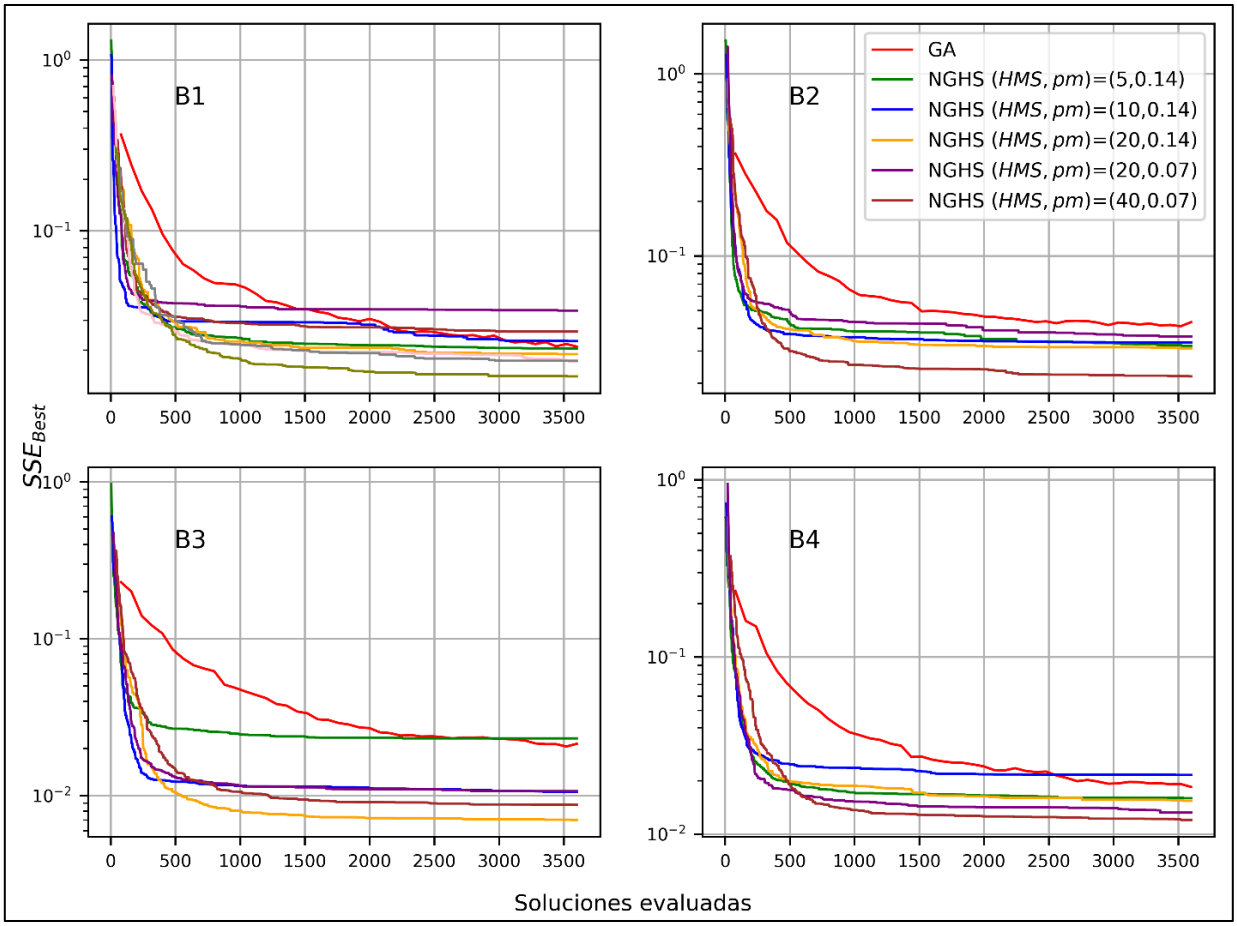

Figura 20. Curvas de convergencia promedio de doce ejecuciones del CREASE sobre todos los benchmarks (B1, B2, B3 y B4) con GA y NGHS, para las combinaciones de HMS y pm escogidas. Fuente propia.

V.2.C. Ciclo III: Estrategias de convergencia prematura sobre el NGHS

En este ciclo, se evaluaron estrategias de convergencia prematura basadas en la diversificación de la HM, buscando aumentar la exploración del paisaje de búsqueda para prevenir la convergencia prematura. Las diversificaciones implementadas consistieron de, en un momento dado (en el que se cumpla una *condición de diversificación*), reemplazar una porción de la HM correspondiente a las peores armonías (con mayor SSE) por armonías aleatorias (*Diversificar HM*), conservando las mejores armonías restantes. Esto debido a que, en el NGHS, las nuevas armonías se generan a partir de la mejor y peor armonía de la HM; conservar las mejores armonías ayuda a mantener el progreso de la optimización hasta ese punto, mientras que la inclusión de armonías aleatorias diversifica la HM, previniendo la explotación temprana y fomentando la exploración para evitar la convergencia prematura. Para su análisis se tuvo en cuenta principalmente sus curvas de convergencia promedio, al igual que el Ciclo II. En la Figura 21 se puede ver un diagrama el flujo esquemático de cómo está estrategia convergencia prematura se añade al algoritmo NGHS, y general a cualquier de las versiones de HS vistas antes, para su comparación se puede observar el diagrama de flujo del HS Figura 9.

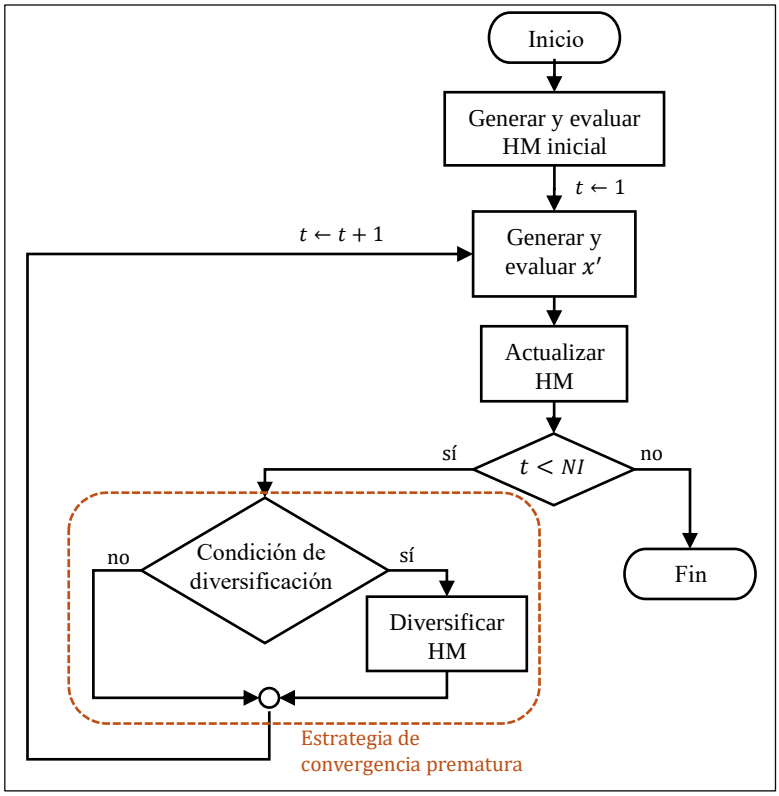

Figura 21. Diagrama de flujo esquemático del algoritmo NGHS con la estrategia de convergencia prematura implementada. Fuente propia.

C.i. HMS=20 *y* pm=0.14

Inicialmente, se probó la diversificación para HMS 20 y pm 0.14, ya que fue la que presentó una mayor convergencia prematura de las dos mejores configuraciones obtenidas en el Ciclo II. Primero, para *diversificar HM* se reemplazó solo la mitad de las armonías presentes en la HM y se realizó una evaluación de la diversificación durante toda la ejecución en distintos puntos de la ejecución que obedecen a la *condición de diversificación*, y estas son: en divHM1 cada que una nueva armonía candidata x' fuera igual a una de las armonías presentes en la HM ($x' \in HM$), y en divHM2 cada cierto número fijo de iteraciones $inter$ ($t \ mod \ inter = 0$) en particular 100, 150, 200 y 300. Estas versiones de diversificación se probaron en B1, sus curvas de convergencia resultantes pueden verse en la Figura 22a, con la versión sin diversificación para su comparación.

En la Figura 22a se observa que la versión de diversificación divHM1 presentó un desempeño menor que la versión sin diversificación obteniendo un SSE_{Best} más alto, por lo que se descarta; por su lado la divHM2 mostró una mejora significativa, siendo la diversificación con $inter$ de 200 y 300 las que destacan; el desempeño de la diversificación de $inter$ 100 y 150 parecen sugerir ser muy prematuras, mientras que para divHM1 muy tardía.

Se decidió entonces implementar divHM3 que hace una *diversificación de HM* más fuerte reemplazando toda la HM, exceptuando la mejor armonía, por armonías aleatorias, con la misma *condición de diversificación* de divHM2 usando $inter$ de 200 y 300 que fueron las que mejor resultados presentaron para divHM2. Esta versión se probó también en B1; sus curvas de convergencia resultantes pueden verse en la Figura 22b, con los resultados de divHM2 correspondientes, y la versión sin diversificación para su comparación. Las curvas de convergencia promedio muestran que la diversificación divHM3 logra SSE_{Best} mejores que divHM2 para los dos valores de $inter$ probados, mostrando ser más beneficiosa la diversificación más fuerte, y permitiendo con esto eliminar la escogencia de qué proporción de la HM a cambiar.

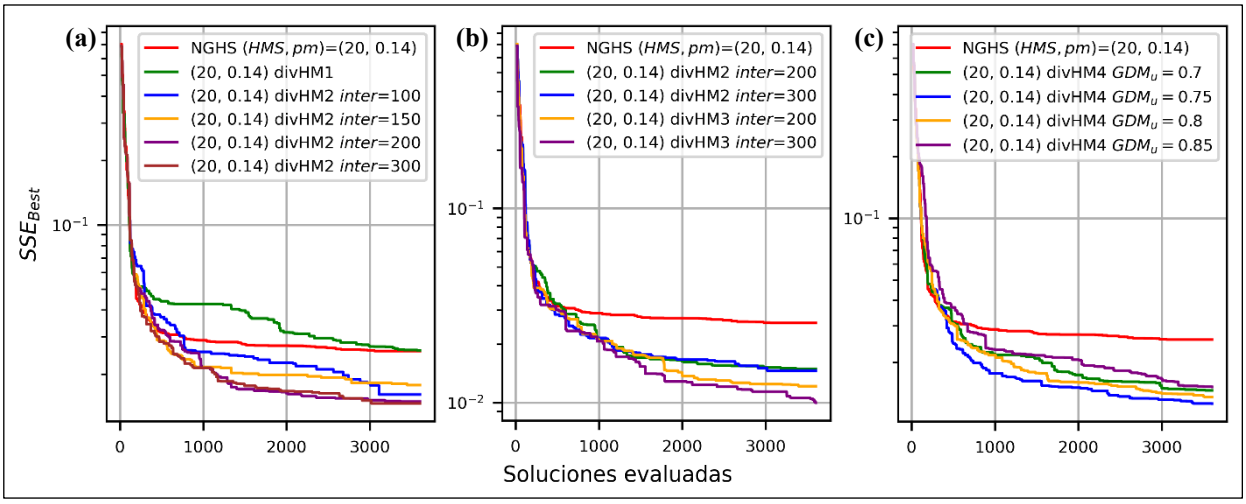

Figura 22. Curvas de convergencia promedio de doce ejecuciones del CREASE con NGHS (HMS 20, pm 0.14) sobre B1, en **(a)** con diversificación divHM1 y divHM2 ($inter$: 100, 150, 200 y 300), en **(b)** con diversificación divHM2 ($inter$ = 200 y 300) y divHM3 ($inter$: 200 y 300) y en **(c)** con diversificación divHM4 ($GDMu$ = 0.7, 0.75, 0.8 y 0.85); acompañadas de la versión sin diversificación (NGHS). Fuente propia.

A pesar de esto, una desventaja que se vislumbró en la versión de divHM3 es que el valor de este intervalo de iteraciones $inter$ entre las diversificaciones podría estar muy ligado al HMS usado, por lo que al cambiar el HMS el valor adecuado de dicho intervalo probablemente también y, ya que los posibles valores que puede tomar este intervalo no tienen un valor máximo claro, hace complicado su ajuste. Por lo anterior se implementó divHM4 que usa una nueva *condición de diversificación* basada en la forma que el GA, analizado en este trabajo, actualiza sus hiperparámetros, determinando los momentos de diversificación con el uso de la métrica GDM (Ecuación 12) sobre los SSE de la HM, haciendo la diversificación cuando el GDM alcance un valor umbral $GDMu$ ($GDM(SSE_{HM}) \geq GDMu$) y *diversificando HM* al igual que en divHM3. El $GDMu$, a diferencia de los $inter$, está acotado en [0,1], por lo que su ajuste puede ser más sencillo. Así mismo, el valor del $GDMu$ se puede extrapolar más fácilmente entre distintos HM*S*, debido a que tiene una interpretación directamente relacionada con la diversidad de la HM (entre más cercano a 1, menos diversidad hay en la HM) [10], [18].

A partir de lo recomendado en las referencias [10], [18], se decidió probar divHM4 con los siguientes valores de GDM_u: 0.7, 0.75, 0.8 y 0.85, estos escogidos a partir del GDM_{max} (0.85) sugerido para el GA en la *shape* de estudio [10]. Las pruebas se hicieron sobre B1, sus curvas de convergencia resultantes pueden verse en la Figura 22c, con los resultados de la versión sin diversificación para su comparación. Las curvas de convergencia promedio muestran que la diversificación divHM4 presenta una mejora significativa respecto a la versión sin diversificación, siendo la diversificación con $GDMu$ de 0.75 y 0.8 las que destacan; los otros valores de $GDMu$ parecen sugerir ser muy prematuras o tardías.

Por lo anterior, se determinó que las mejores versiones de diversificación probadas en B1 fueron divHM3 con $inter$ de 200 y 300 y divHM4 con $GDMu$ de 0.75 y 0.8. Se procedió a probar estas versiones en el todos los benchmarks. En la Figura 23 se pueden ver las curvas de convergencia resultantes, con los resultados de la versión sin diversificación para su comparación. Los resultados muestran que, en general, las dos versiones de diversificación resuelven satisfactoriamente la convergencia prematura del NGHS en B1, y sobre los demás benchmarks, obtuvieron $SSEs$ similares a los obtenidos sin diversificación con una leve desmejora en la velocidad de convergencia. Se determinó que la mejor versión de diversificación probada fue la de divHM3 con un inter de 300, que exhibe un leve mejor SSE, en B1 y B2 que las demás versiones de diversificación.

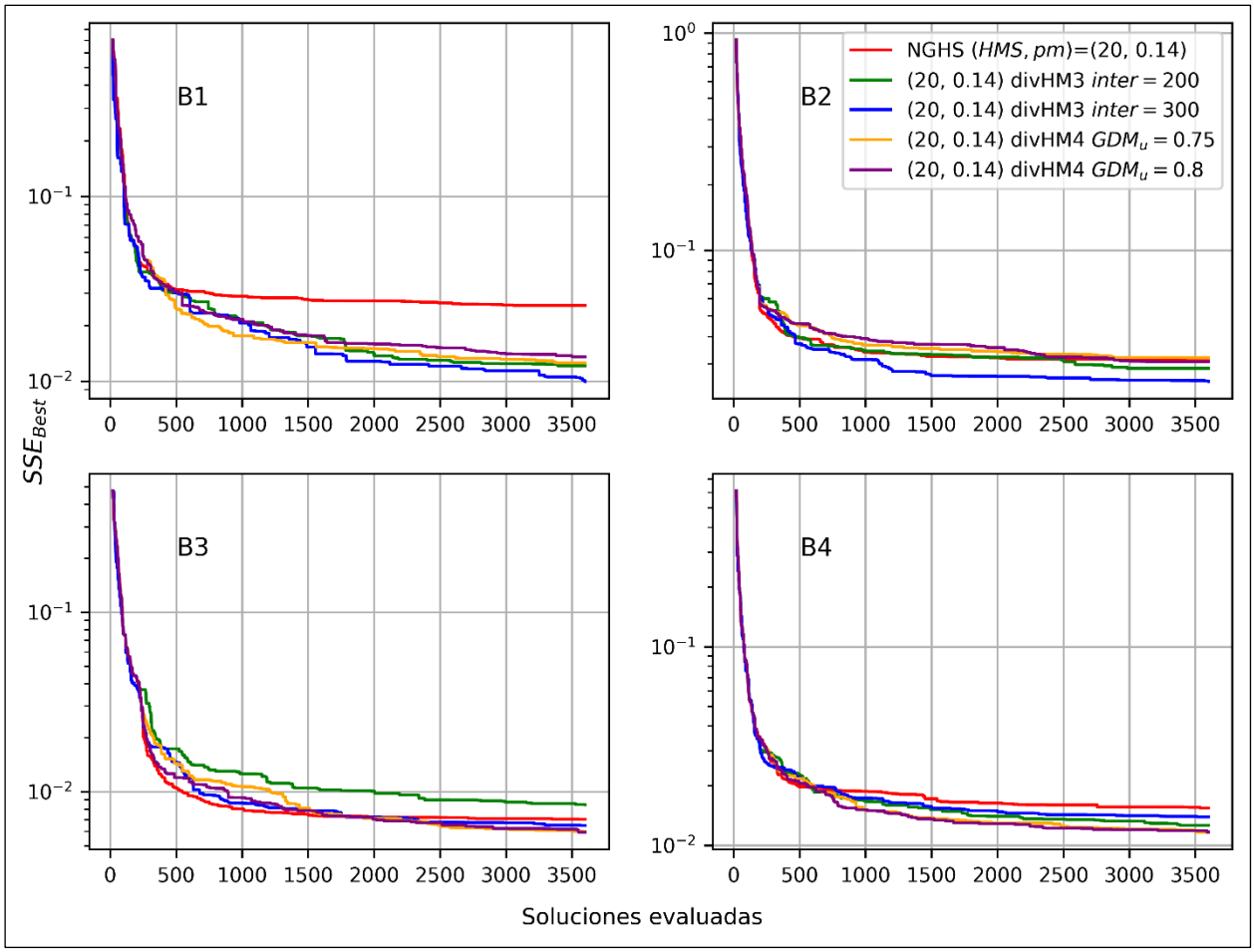

Figura 23. Curvas de convergencia promedio de doce ejecuciones del CREASE sobre todos los benchmarks (B1, B2, B3 y B4), de NGHS (HMS=20, pm=0.14), con diversificación divHM3 (*inter*: 200 y 300) y divHM4 ($GDMu$ = 0.75, 0.8) y sin diversificación. Fuente propia.

C.i. HMS=40 y pm=0.07

Para esta configuración de hiperparámetros se decidió probar directamente la estrategia de diversificación divHM4 dados los resultados obtenidos y analizados de las estrategias de diversificación probadas para HMS 20 y pm 0.07. Ya que en este caso el HMS es el doble del antes considerado, no es claro cuales valores de $inter$ probar cuando se usa divHM3. Se probó entonces divHM4 con GDM_u de 0.75 y 0.8, que fueron los que se desempeñaron mejor para la configuración anterior. Sus curvas de convergencia resultantes pueden verse en la Figura 24, con los resultados de la versión sin diversificación y con la mejor configuración de diversificación para HMS 20 y pm 0.07: divHM3 con *inter*=300, para su comparación.

Las curvas de convergencia promedio muestran que, para HMS 40 y Pm 0.07, la diversificación divHM4 muestra una mejora respecto a la versión sin diversificación en B1 y B3, obteniendo valores de SSE más bajos, siendo la diversificación con $GDMu$ de 0.8 la que destaca. En B2 y B4 los valores de SSE son similares a los obtenidos sin diversificación. Respecto a la mejor versión de diversificación para NGHS HMS 20 y pm 0.14, se obtienen resultados muy similares.

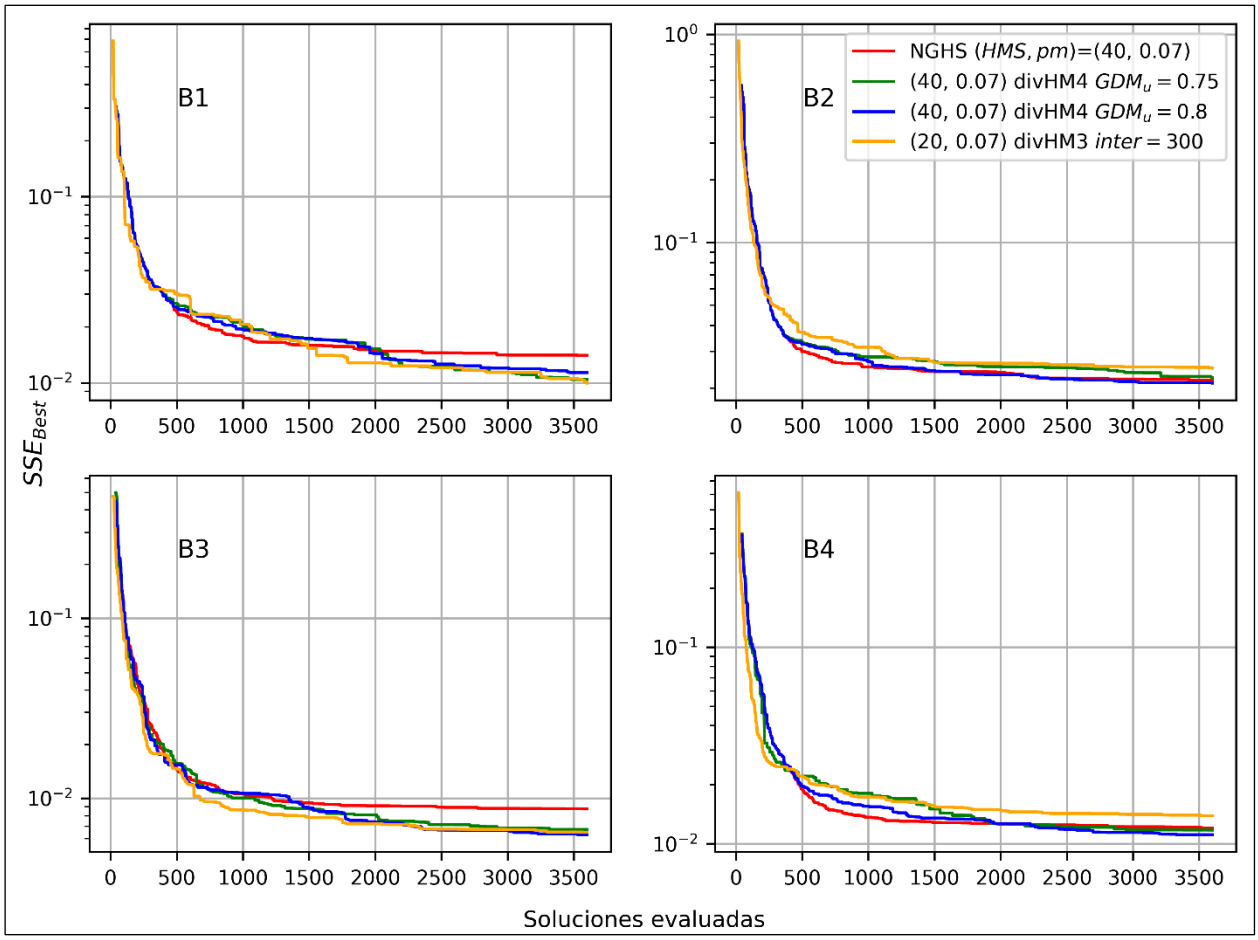

Figura 24. Curvas de convergencia promedio de doce ejecuciones del CREASE sobre todos los benchmarks (B1, B2, B3 y B4) de NGHS (HMS=40, pm=0.07) con diversificación divHM4 ($GDMu$: 0.75, 0.8), y sin diversificación; acompañador los resultados del NGHS (HMS=20, pm=0.14) con divHM3 ($inter$=300). Fuente propia.

Finalmente, aunque con el NGHS con diversificación se logra obtener en promedio valores de SSE_{Best} similares o mejores que sin diversificación, esta mejora viene acompañada con un aumento considerable del tiempo de ejecución de herramienta, de alrededor del 30%, esto debido a que con la diversificación de la HM, a pesar de evaluar la misma cantidad de soluciones, las soluciones aleatorias presentan usualmente dimensiones demandantes en cuanto al cálculo de $I_{comp}(Q)$; adicionalmente hay una leve pérdida en la velocidad de convergencia, además de aumentar un hiperparámetros a la metaheurística. Por lo anterior no se ve favorable usar ninguna de las versiones de diversificación evaluadas.

La determinación de la cantidad de armonías a conservar en la HM, el momento y frecuencia la diversificación, adecuados para mejorar el rendimiento del NGHS constituyen áreas de investigación potenciales.

V.2.A. Conclusión: Versión de HS final

En conclusión, tras estas pruebas preliminares, el NGHS con HMS=40 y pm=0.07 se identifica como la metaheurística basada en HS evaluada más adecuada para la comparación final con el GA en el desempeño del CREASE para el análisis de perfiles SAS de soluciones de baja concentración vesículas ensambladas a partir de polímeros anfifílicos. A continuación, se presenta el resultado y análisis de dicha comparación.

V.3. Comparación CREASE-GA vs CREASE-NGHS

En esta sección, se evalúa el desempeño de CREASE usando las metaheurísticas NGHS y GA, CREASE-NGHS y CREASE-GA respectivamente de ahora en adelante; comparando sus curvas de convergencia, los parámetros

estructurales obtenidos y la velocidad de convergencia, para los bechmarks analizados. Los tiempos de ejecución no se pudieron comparar debido a la variedad de los recursos computacionales usados; pero sí se observó que con ambas metaheurísticas el CREASE presentó tiempos de ejecución parecidos, indicando que el tiempo de ejecución se invirtió principalmente evaluación de las soluciones.

V.3.A. Curvas de convergencia del CREASE-GA y el CREASE-NGHS

A continuación, se pueden ver las sus curvas de convergencia promedio de las 31 ejecuciones sobre cada benchmark para el CREASE-GA y el CREASE-NGHS, estas curvas de convergencia consisten en el mejor SSE (SEE_{best}) encontrado en escala logarítmica en función de la cantidad de soluciones evaluadas hasta ese momento a lo largo de la ejecución, Figura 25. El SSE_{best} corresponde al menor SEE la población actual en el GA y de la HM actual en el NGHS.

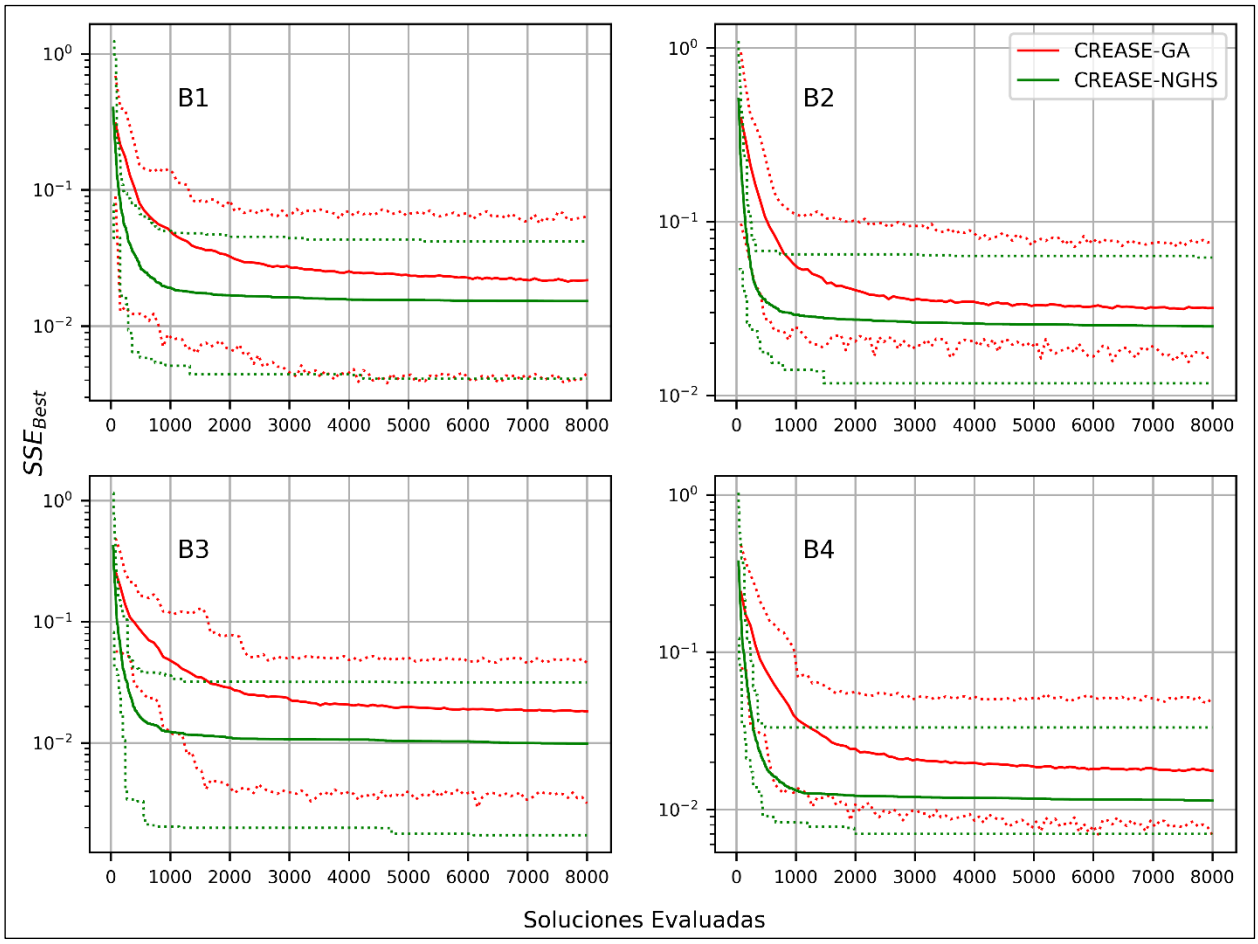

Figura 25. Curvas de convergencia promedio, mínima y máxima de las 31 ejecuciones en los cuatro benchmarks (B1, B2, B3 y B4). En rojo los resultados para CREASE-GA y en verde para el CREASE-NGHS, la línea continua es el valor promedio y las punteadas corresponden la peor (línea superior) y mejor (línea inferior) de todas las ejecuciones. Fuente propia.

Los resultados revelan que, en B1, el CREASE-NGHS (curva continua verde) logra el mismo valor de SSE_{Best} al que converge CREASE-GA (curva continua roja) con la sexta parte de cálculos de $I_{comp}(Q)$, aproximadamente. Mientras que, en los demás perfiles, esta relación se reduce a aproximadamente a la décima parte. Entretanto, para todos los perfiles CREASE-GA necesita evaluar entre 5000 y 7000 soluciones para converger, mientras el CREASE-NGHS lo logra entre 2000 y 3000 soluciones, además de obtener en todos los casos valores de SSE más bajos. Estos resultados indican consistentemente una mejora en la velocidad de convergencia del CREASE-NGHS en comparación con CREASE GA.

Los resultados representados por las líneas discontinuas corresponden a la evolución del peor (línea discontinua superior) y mejor (línea discontinua inferior) SSE_{Best} a lo largo de las 31 ejecución del CREASE-NGHS y CREASE-GA. Se observa que la peor ejecución del CREASE-NGHS siempre supera a su contraparte en CREASE-GA, mientras que la mejor ejecución del CREASE-NGHS iguala o mejora los mejores resultados del CREASE-GA.

En todos los casos benchmarks el CREASE-NGHS mejora al CREASE GA tanto en el valor de SSE_{Best} obtenido como en velocidad de convergencia; los valores de SSE_{Best} promedio y su desviación estándar se pueden encontrar en la Tabla 7. A partir del comportamiento exhibido se podría reducir para el CREASE-NGHS el NI a 3500, de forma que asegure su convergencia en esta *shape*.

V.3.B. Comparación de las salidas del CREASE-GA y el CREASE-NGHS

En esta sección se comparan el SSE_{Best}, $RMSRE$ y los parámetros estructurales de la mejor estructura determinada por las 31 ejecuciones del CREASE-GA y CREASE-NGHS, a excepción del $-log_{10}(I_{bg})$ que tiene información de la intensidad de dispersión de fondo y no de la muestra, y a partir de estos se evalúa la precisión de cada versión.

Para analizar los valores de SSE y $RMSRE$, calculado a partir de la Ecuación 1, de la mejor solución encontrada de cada ejecución para cada metaheurística sobre cada benchmark, se calculó su promedio y desviación estándar que se pueden encontrar listados en la Tabla 7, y se visualizó sus distribuciones mediante diagramas de violín y de caja, que se pueden ver en la Figura 26, para su análisis se debe tener en cuenta que: el diagrama de caja muestra la distribución de datos mediante una caja que representa el rango intercuartílico, una línea para la mediana y bigotes que indican el mínimo y máximo, excluyendo los valores atípicos; por su parte, el diagrama de violín muestra una distribución aproximada más detallada de los datos a través de su anchura, que indica la densidad de los valores en diferentes rangos.

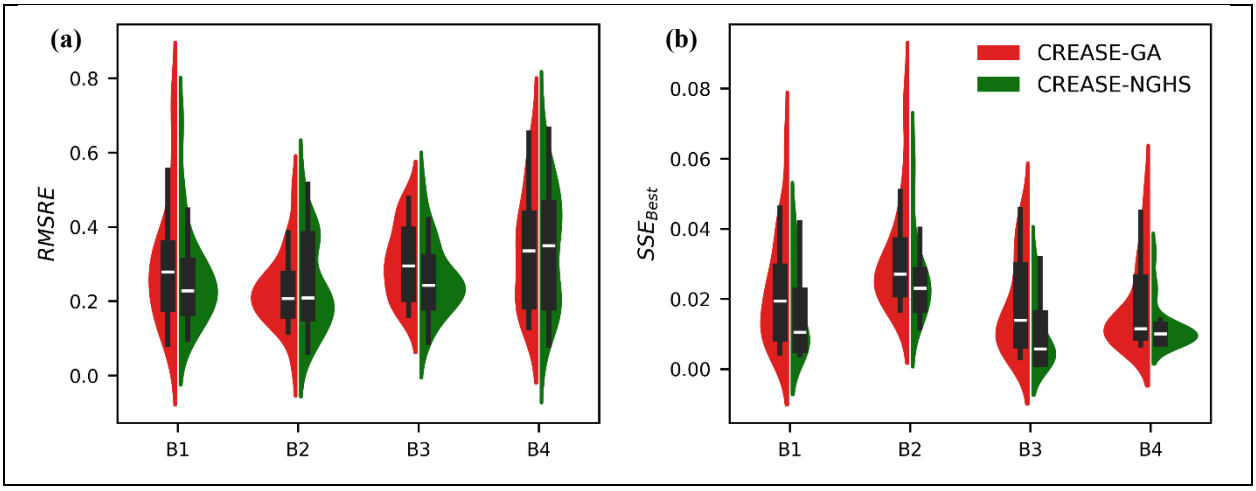

Figura 26. Diagramas de violín y de caja de los **(a)** $RMSRE$ y **(b)** SSE_{Best} obtenidos en las 31 ejecuciones del CREASE-GA (rojo) y CREASE-NGHS (verde). Fuente propia.

Tabla 7. SSE_{Best} y $RMSRE$ (promedio y desviación estándar) obtenidos en las 31 ejecuciones del CREASE-GA y CREASE-NGHS para cada benchmark (B1, B2, B3, y B4).

	Alg.	$\boldsymbol{SSE_{Best}}$	$\boldsymbol{RMSRE}$
B1	GA	0.0217 ± 0.0144	0.295 ± 0.158
	NGHS	0.0152 ± 0.0111	0.256 ± 0.119
B2	GA	0.0318 ± 0.0147	0.228 ± 0.090
	NGHS	0.0249 ± 0.0108	0.243 ± 0.117
B3	GA	0.0181 ± 0.0128	0.307 ± 0.097
	NGHS	0.0098 ± 0.0089	0.257 ± 0.093
B4	GA	0.0175 ± 0.0116	0.338 ± 0.148
	NGHS	0.0113 ± 0.0054	0.345 ± 0.152

Al analizar los diagramas de violín y de caja del SSE_{Best} (Figura 26b) para el CREASE-GA y CREASE-NGHS, se observa que ambos muestran en general una distribución normal sesgada positivamente, indicando una concentración de valores de SSE_{Best} alrededor de la media, con una tendencia hacia valores más bajos, pero con presencia de algunos de valores altos, sugiriendo un posible comportamiento multimodal en ciertos casos donde se puede identificar un pico secundario de amplitud considerablemente menor que el pico principal. Se destaca que el CREASE-NGHS exhibe en todos los benchmarks, distribuciones más compactas y desplazadas hacia valores más bajos de SSE_{Best}, teniendo valores máximos y medianas menores, en comparación con las distribuciones del CREASE-GA. Respecto a sus valores mínimos los diagramas de violín parecen sugerir que el CREASE-GA alcanza valores menores de SSE_{Best}, pero esto último es resultado de las aproximaciones hechas por esta técnica para suavizar las distribuciones, resultando también en la sugerencia de que el SSE_{Best} alcanzó valores negativos para B1, B3 y B4. Los valores mínimos de SSE_{Best} para ambas metaheurísticas se pueden corroborar al ver los bigotes de los gráficos de caja, así como con las curvas de convergencia mínimas en la Figura 25, confirmando que el CREASE-NGHS alcanzó SSE_{Best} menores. Todo lo anterior coincide con los resultados presentados en la Tabla 7 y en la subsección anterior, donde se evidencia que los valores promedio de SSE_{Best} obtenidos por el CREASE-NGHS fueron consistentemente menores, al igual que sus desviaciones estándar. Esto se refleja en perfiles SAS $I_{comp}(Q)$ mejor ajustados a $I_{exp}(Q)$.

Los diagramas de violín del el $RMSRE$ (Figura 26a) de B1, B2 y B3 muestran distribuciones similares a las observadas para el SSE_{Best}, pero con la presencia en algunos casos de un pico secundario más pronunciado que los observados en el SSE_{Best}, habiendo en el caso de B4 la presencia de dos picos principales para ambas metaheurísticas; esta multimodalidad pone en evidencia la naturaleza multimodal del paisaje de búsqueda de los benchmarks y en general del problema de análisis de perfiles SAS debido a la degeneración. Se destaca que el CREASE-NGHS tiene medianas y dispersiones menores que el CREASE-GA en B1 y B3, y medianas muy similares con dispersiones mayores en B2 y B4. Estas observaciones se corresponden con los valores de promedio y desviaciones estándar del $RMSRE$ expuestos en la Tabla 7, que difieren muy poco entre las metaheurísticas, siendo en B1 y B3 donde se presenta una diferencia más apreciable en el valor promedio a favor del CREASE-NGHS. De los diagramas de cajas se puede observar que CREASE-NGHS obtuvo los valores de RMSRE más altos para B2 y B4, y los más bajos para B2, B3 y B4.

Tabla 8. Parámetros estructurales obtenidos por las 31 (media y desviación estándar) ejecuciones del CREASE-GA y CREASE-NGHS para benchmark (B1, B2, B3, y B4).

	Alg.	$\boldsymbol{R_{core}}$ [Å]	$\boldsymbol{t_{Aint}}$ [Å]	$\boldsymbol{t_B}$ [Å]	$\boldsymbol{t_{Aout}}$ [Å]	$\boldsymbol{\sigma_{Rcore}}$ [%]	$\boldsymbol{s_{Ain}}$ [%]	$\boldsymbol{R_{Total}}$ [Å]
B1	Target	100	120	60	120	20	20	400
	GA	106.5 ± 32.0	107.0 ± 33.3	81.8 ± 17.3	97.4 ± 18.8	18.3 ± 5.3	25.2 ± 10.6	391.0 ± 11.9
	NGHS	109.0 ± 26.6	108.8 ± 31.6	73.0 ± 15.0	104.0 ± 12.0	17.2 ± 4.9	26.3 ± 9.8	396.1 ± 10.1
B2	Target	100	60	120	60	20	20	340
	GA	98.6 ± 21.6	62.1 ± 20.5	123.6 ± 8.4	56.6 ± 9.5	21.5 ± 5.2	24.9 ± 10.5	340.8 ± 9.3
	NGHS	99.0 ± 26.4	64.0 ± 22.7	120.8 ± 5.8	60.1 ± 7.3	21.2 ± 7.1	25.7 ± 11.0	343.8 ± 7.4
B3	Target	150	120	60	120	20	20	450
	GA	143.0 ± 38.3	122.4 ± 39.9	78.7 ± 19.4	96.2 ± 21.4	22.4 ± 7.4	25.7 ± 11.2	440.4 ± 18.1
	NGHS	155.9 ± 30.2	110.8 ± 34.1	77.0 ± 15.3	97.3 ± 15.2	19.51 ± 3.9	26.2 ± 10.0	441.1 ± 12.7
B4	Target	150	60	120	60	20	20	390
	GA	127.3 ± 32.5	81.8 ± 32.4	124.6 ± 7.6	41.8 ± 10.7	25.1 ± 6.9	24.0 ± 12.0	375.5 ± 10.6
	NGHS	118.1 ± 33.5	87.5 ± 31.8	126.8 ± 4.4	40.6 ± 5.6	27.4 ± 8.0	17.7 ± 8.8	373.0 ± 5.3

Para profundizar en el análisis de la precisión del CREASE se debe recordar que los autores de la *shape* de solución diluida de vesículas recomiendan para el análisis de perfiles SAS [10] múltiples ejecuciones de la herramienta CREASE permitiendo un mejor entendimiento de la degeneración, usando el promedio y desviación estándar de los resultados para sacar conclusiones del perfil SAS analizado. Por lo anterior, se analizará el promedio y desviación estándar de los parámetros estructurales obtenidos en las 31 ejecuciones sobre cada benchmark con el CREASE-GA y el CREASE-NGHS, estos se listan en la Tabla 8, se añade también los resultados obtenidos para el radio total de la vesícula ($R_T = R_{core} + t_{Aint} + t_B + t_{Aout}$). Se observa que ambas metaheurísticas obtuvieron parámetros estructurales similares en los benchmarks, los cuales coinciden dentro del margen de error en la mayoría de los casos

con los valores esperados (Target). Estos resultados los vistos para el RMSRE permiten ver que en general, tanto el CREASE-NGHS como el CREASE-GA logran una precisión similar en la búsqueda de parámetros estructurales.

Los parámetros estructurales obtenidos muestran una disposición y variabilidad similar a la reportada por los autores de esta *shape* [10]. A pesar de que, como se observó previamente en las Figuras 25 y 26 , el CREASE-NGHS obtuvo en general SSE_{Best} menores en comparación con el CREASE-GA, la coincidencia entre los resultados de ambas metaheurísticas sugiere que esta mejora en el SSE fue en valores de Q que posiblemente no influyen de manera significativa en las dimensiones [16]. En la Figura 27 se muestra el perfil SAS computado $I_{comp}(Q)$ de la mejor solución encontrada por el CREASE-GA y el CREASE-NGHS para B3, acompañado del perfil experimental $I_{exp}(Q)$. Se obversa que la solución obtenida, en este ejemplo, por el CREASE-NGHS logra un perfil SAS computado $I_{comp}(Q)$ levemente mejor ajustado en los valores de Q superiores a 0.01Å^{-1} capturando mejor las características del perfil $I_{exp}(Q)$ en esa región.

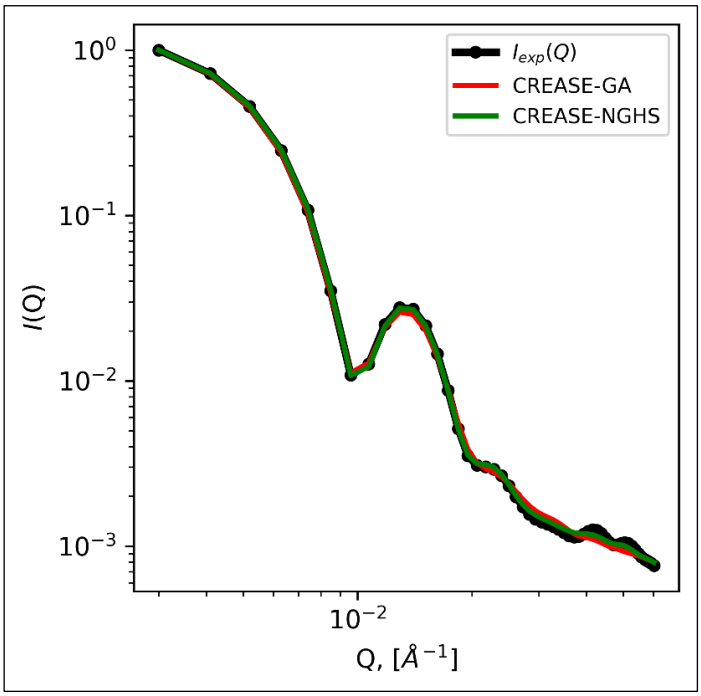

Figura 27. Perfil SAS computado $I_{comp}(Q)$ de la mejor solución encontrada por el CREASE-GA y el CREASE-NGHS para B3, acompañado del perfil experimental $I_{exp}(Q)$. Fuente propia.

La variabilidad en los parámetros estructurales identificados en las ejecuciones del CREASE puede atribuirse al fenómeno de la degeneración, previamente mencionado, así como al valor de n_{sct}/N elegido para el análisis. Como se mencionó anteriormente, este valor está directamente relacionado con N_T, y el uso de valores más altos de N_T para el análisis tiende a aumentar la precisión de la herramienta, aunque con un incremento en su costo computacional. Dado que el CREASE-NGHS logra resultados con un menor SSE con menos soluciones evaluadas, el ahorro computacional en estos cálculos podría ser destinado al uso de valores más altos de N_T para el análisis.

Algo a resaltar de los parámetros estructurales obtenidos por el CREASE-GA y CREASE-NGHS, listados en la Tabla 8, es que al calcular el radio total de la vesícula R_T se observa que presenta una dispersión relativamente baja en comparación con los otros parámetros estructurales, además de ser consistentemente cercano al valor esperado, esto puede deberse a que la dispersión SAS es particularmente es sensible a las dimensiones más grandes de las estructuras presentes en la muestra, en este caso las vesículas. Lo anterior sugiere que el radio total de la vesícula R_T podría ser un parámetro estructural más adecuado para hacer la optimización e incluso que se podría usar como información heurística para mejorar el proceso de optimización en el análisis de esta *shape*.

V.3.C. Prueba no paramétrica de Wilcoxon de Rangos con Signo

Los resultados listados en las Tabla 7 se usaron para realizar la comparación del desempeño de CREASE-GA y CREASE-NGHS usando la prueba no paramétrica de Wilcoxon de rangos con signo con la herramienta Keel 3.0 [32]. Se le entregó el promedio del SSE_{Best} y $RMSE$ para cada benchmark y metaheurística. Los resultados entregados se pueden ver resumidos en la Tabla 9.

Tabla 9. Resumen del resultado de la prueba Wilcoxon para los resultados obtenidos de SSE_{Best} y $RMSRE$ en las 31 ejecuciones, arrojado por la herramienta Keel 3.0 [32].

	SSE_{Best}		$RMSRE$	
	(1)	(2)	(1)	(2)
CREASE-GA (1)	-		-	
CREASE-NGHS (2)		-		-

Para interpretar los resultados obtenidos en la Tabla 9 se debe tener en cuenta que Keel usa la siguiente notación para los resultados obtenidos: aparecerá un • si el método en la fila mejora al método de la columna, y un ◦ si el método en la columna mejora al método de la fila, si no aparece ninguno de los dos indican que no hay mejora estadísticamente significativa, esto con un nivel de significancia de α = 0.9. Con esto, y a pesar de que CREASE-NGHS provee en promedio resultados más bajos de SSE con menor desviación estándar que el CREASE-GA consistentemente para todos los benchmarks, como se puede ver en la Tabla 7 y en sus curvas de convergencia en la Figura 25, esta mejora no es estadísticamente significativa según los resultados de las pruebas de Wilcoxon de rangos con signo. Sin embargo, valores más bajos en la desviación estándar del SSE_{Best} muestra la solidez de los resultados obtenidos por CREASE-NGHS en todos los benchmarks probados.

Por su parte los resultados para el $RMSRE$ muestran que tampoco hay una diferencia estadísticamente significativa entre los valores obtenidos por el CREASE-GA y el CREASE-NGHS sobre los benchmark. Con esto se corrobora que ambas metaheurísticas logran una precisión estadísticamente igual.

V.3.D. Exploración del paisaje de búsqueda

Para comparar la eficacia en términos de exploración del paisaje de búsqueda, se calcularon para los resultados del CREASE-NGHS las mismas métricas del diagnóstico del CREASE-GA expuestas en la Tabla 6 (subsección V.1.C); dichos resultados, junto con los del CREASE-NGHS, se lista en la Tabla 10. A partir de estos, de observa que el CREASE-NGHS obtuvo, en promedio sobre todos los benchmarks, un porcentaje de soluciones únicas (%SU) evaluadas en cada ejecución del 70.9%, 3.5 veces más que el CREASE-GA, dejando solo el 29.1% restante a reevaluaciones de soluciones. Adicionalmente, se observa que el CREASE-NGHS, en comparación con el CREASE-GA, logró disminuir la cantidad de veces que se evalúa una misma solución en una ejecución (ES) para todos los benchmarks. En particular, se aprecia que el ES máximo (E_{max}) de cada una de las 31 ejecuciones promedio ($\overline{ES_{max}}$) para cada Benchmark disminuyó aproximadamente a una sexta parte. Asimismo, para todas las ejecuciones, se redujeron considerablemente los porcentajes de evaluaciones correspondientes a soluciones con más de mil ES (%$[ES > 1000]$), cien ES (%$[ES > 100]$) y diez ES (%$[ES > 10]$).

Tabla 10. Métricas de evaluación de la cantidad de veces que se evalúan las mismas soluciones por parte del CREASE-NGHS y el CREASE-GA en una ejecución, para las 31 ejecuciones de cada benchmark (B1, B2, B3, y B4).

	Alg.	%SU	$\overline{ES_{max}}$	%$[ES > 1]$	%$[ES > 10]$	%$[ES > 100]$	%$[ES > 1000]$
B1	GA	19.8	772.5	87.8	65.2	34.2	3.3
	NGHS	70.5	143.7	39.0	12.1	0.9	0.0
B2	GA	21.2	660.4	87.3	61.7	28.2	2.1
	NGHS	73.1	93.6	35.0	12.1	0.9	0.0
B3	GA	20.0	727.5	87.6	64.8	34.3	4.1
	NGHS	69.7	113.6	40.8	10.3	1.4	0.0
B4	GA	22.1	569.7	86.8	59.9	25.8	0.4
	NGHS	70.4	102.8	39.2	12.0	0.9	0.0

Adicionalmente, al igual que como se hizo para el diagnóstico del CREASE-GA se creó un histograma de frecuencias de ES para cada benchmark con los resultados de las 31 ejecuciones del CREASE-NGHS, estos se pueden observar en la Figura 28. Estos histogramas muestran que, para el CREASE-NGHS, hay un aumento en un orden de magnitud en el número de soluciones evaluadas una sola vez, y una disminución más pronunciada en la frecuencia a medida que aumentan los valores de ES en el gráfico, en comparación con los resultados obtenidos para el CREASE-GA (Figura 17), logrando reducir el ES máximo de cada benchmark en un orden de magnitud.

En los histogramas de B2, B3 y B4 se observar un pico para ES igual a diez para, esto se corresponde con una de las configuraciones iniciales hechas a todas las versiones de HS a partir del diagnóstico CREASE-GA, que se pueden ver en la subsección V.1.D, cuyo objetivo es establecer un valor máximo de ES (mct) para limitar la reevaluación de soluciones, usándose para estas pruebas $mct = 10$. Claramente algunas soluciones lograron evadir la medida implementada, pero aun así se logró hacer un mejor control de las reevaluaciones.

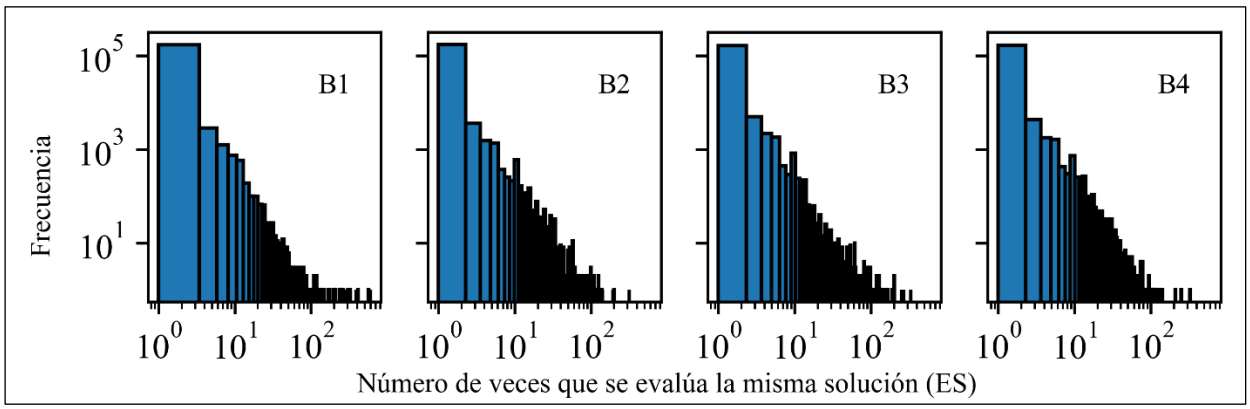

Figura 28. Histograma de frecuencia del número de veces que se evalúa la misma solución (ES) en una ejecución del CREASE-NGHS, para las 31 ejecuciones de cada benchmark. Fuente propia.

Lo anterior implica una mejora notable en el uso de los recursos computacionales y en la capacidad de exploración del CREASE-NGHS respecto al CREASE-GA, permitiendo al primero encontrar nuevas soluciones sin sacrificar la capacidad del algoritmo para tratar el ruido. Las reevaluaciones de soluciones repetidas se realizan principalmente en el proceso de explotación, por lo que no representa una disminución considerable en la velocidad de convergencia. De hecho, el CREASE-NGHS no solo mejora significativamente el proceso de exploración, sino que también aumenta de manera considerable la velocidad de convergencia.

D.i. Visualización del paisaje de búsqueda usando un SOM

En un esfuerzo por visualizar el paisaje de búsqueda de los benchmarks usados en esta investigación, se hizo un entrenamiento no supervisado de un Mapa Auto-organizativo (Self-Organizing Map, SOM) bidimensional rectangular de 50×50 neuronas, entrenada sobre un conjunto de cinco millones de datos correspondientes a posibles soluciones aleatorias en el espació de búsqueda considerado para los benchmarks, estas soluciones se normalizaron en función del límite superior x_{ui} e inferior x_{li} de cada parámetro i. Una vez entrenado el SOM, se clasificaron todas las soluciones evaluadas en las ejecuciones del CREASE-GA y CREASE-NGHS que sumaban un total de aproximadamente quinientos mil datos, de forma que cada solución quedara asignada a la neurona cuyo vector de pesos se acerca más. Finalmente, con todas las soluciones clasificadas, se graficó en escala de color el resultado de tomar el menor valor de SSE encontrado entre los individuos asignados a cada neurona. El resultado se puede observar en la Figura 29.

Para la interpretación de esta representación del paisaje de búsqueda debe recordarse que las celdas adyacentes en el SOM están relacionadas entre sí de forma que los datos similares tienden a agruparse en celdas vecinas. Adicionalmente, es importante tener en cuenta que esta representación bidimensional de datos de siete dimensiones, en este caso, inevitablemente conlleva una pérdida de información, ya que no toda la estructura de los datos puede ser capturada; entre las implicaciones más importantes es que, por ejemplo, dos celdas que estén en extremos opuestos del SOM podrían representar datos cercanos entre sí en el espacio real, por lo que es más acertado sacar conclusiones sobre datos de celdas aledañas [37]. Por otro lado, las celdas moradas observadas en los paisajes de búsqueda (Figura

29), corresponden a neuronas a las que ninguna solución fue asignada, estas se ubican en regiones de SSE altos, mostrando que estas zonas fueron menos exploradas tanto por el GA como el NGHS, como era de esperarse.

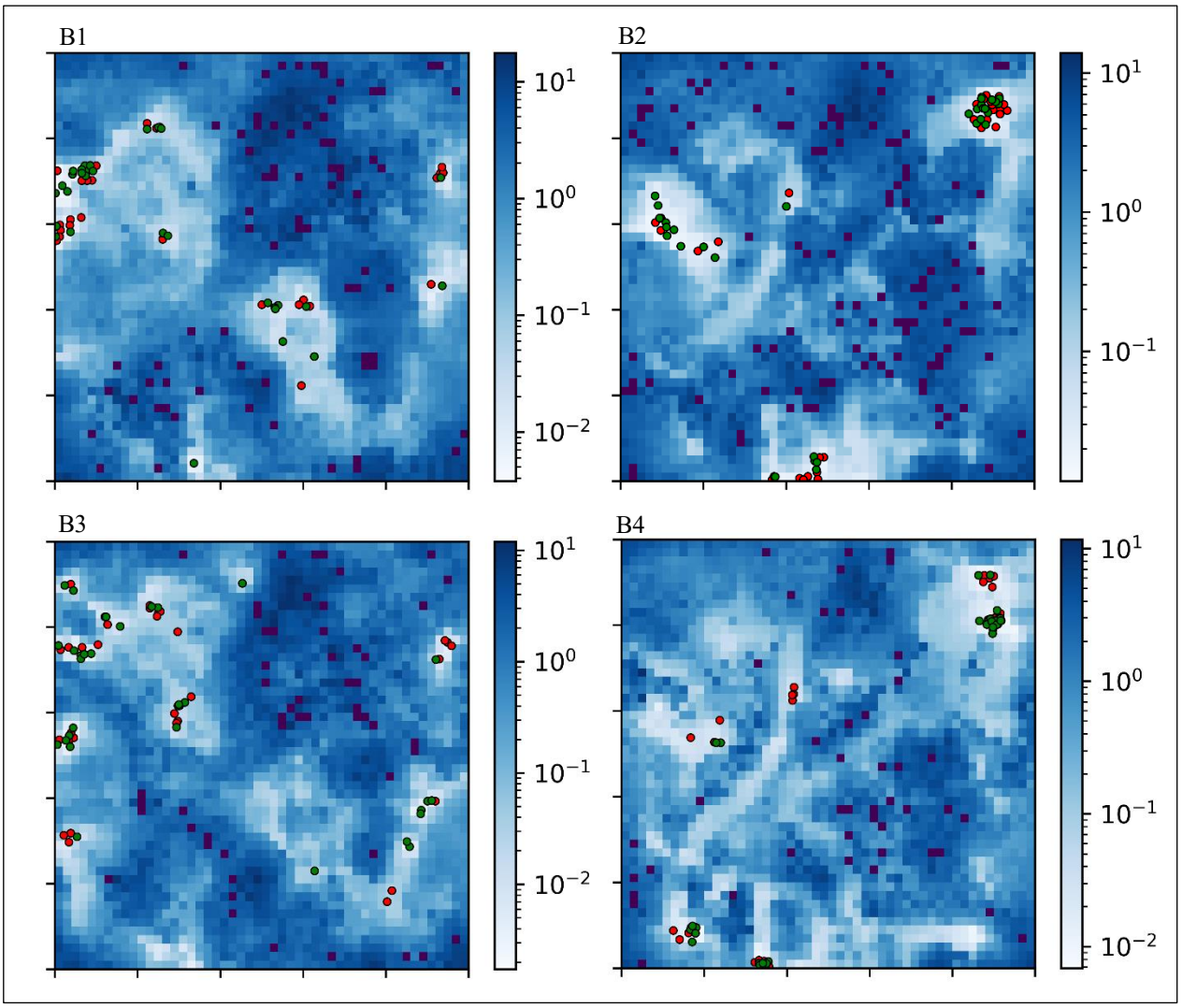

Figura 29. Visualización del paisaje de búsqueda de los benchmarks (B1, B2, B3 y B4) en el SOM, con el valor de la función objetivo SSE en escala logarítmica en tonos azules y ubicación de las mejores soluciones obtenidas por el CREASE-GA (puntos rojos) y el CREASE-NGHS (puntos verdes) en sus 31 ejecuciones. Fuente propia.

Se puede visualizar en la Figura 29 la presencia de los múltiplos mínimos para todos los benchmarks, que la degeneración anticipaba; en varios de estos mínimos se agrupan los resultados de tanto las ejecuciones del CREASE-NGHS (puntos verdes) como del CREASE-GA (puntos naranjas). Esto se traduce en $SSEs$ y parámetros similares como ya se vio en las subsecciones previas. Sin embargo, en B4 se observa una clara diferencia en este agrupamiento, precisamente en este fue donde hubo una diferencia más notable en los parámetros promedio obtenidos por ambas metaheurísticas respecto al objetivo; se pude observar que cuatro ejecuciones del CREASE-GA se agrupan solas, puede indicar un estancamiento del CREASE-GA, que en este caso le favoreció ya que, en este benchmark, el CREASE-GA obtuvo en promedio parámetros más cercanos al objetivo. Lo anterior se debe a los efectos ya mencionados de la escogencia de la razón n_{sct}/N para el análisis y la degeneración.

VI. Conclusiones

Se planteó y evaluó la metaheurística basada en HS, NGHS, como sustituta del GA de adaptación dinámica en la herramienta CREASE para el análisis de perfiles SAS de soluciones de baja concentración de vesículas ensamblas a partir de polímeros anfifílicos, con el potencial de ser usada en otros tipos de muestras.

El algoritmo NGHS demostró, en la ejecución del CREASE, reducir significativamente la cantidad de soluciones evaluadas requeridas para alcanzar una solución que concuerde con las dimensiones determinadas por el algoritmo GA, con solo la sexta y hasta la doceava parte de evaluación, y logrando soluciones con la misma precisión. En particular se observó que el NGHS logró en promedio converger con la evaluación de entre 2000 y 3000 soluciones, mientras que el GA lo hace entre 5000 y 7000, con valores de SSE menores, lo que se traduce en perfiles computados $I_{comp}(Q)$ más ajustados al perfil experimental $I_{exp}(Q)$; esta mejora se traduce en una mayor eficiencia de la herramienta CREASE, tanto si el cálculo de $I_{comp}(Q)$ se hace a partir del modelo de Debye o con un modelo de ML.

Así mismo, se aportó en la usabilidad de la herramienta CREASE en dos aspectos: el primero fue la reducción significativa de la cantidad de hiperparámetros que el usuario debe configurar para asegurar el correcto funcionamiento de la metaheurística en el CREASE, pasando de once hiperámetros para el GA a solo tres para el NGHS, lo que puede facilitar su usabilidad y ajuste para el análisis de nuevas *shapes* con el CREASE. En segundo lugar, se reemplazó la discretización binaria del espacio de búsqueda usada por el GA por una versión que permite al usuario determinar la precisión con la que desea analizar cada parámetro estructural por separado y de forma decimal, siendo esto más intuitivo y versátil.

Se contribuyó en una mejora notable del uso de los recursos computacionales y en la capacidad de exploración del CREASE con el uso del NGHS en lugar del GA, limitando la cantidad de reevaluaciones de soluciones, logrando un aumento de cuatro veces en la cantidad de soluciones únicas evaluadas en el proceso de búsqueda. Además, se disminuyó a una tercera parte la cantidad de reevaluaciones, que con el GA llegaba a ser de hasta un orden de $\sim 10^3$, y con el NGHS se redujo en un orden de magnitud, sin sacrificar la capacidad del algoritmo para tratar el ruido en el SSE.

VII. Trabajos Futuros

A partir de la realización de este trabajo se vislumbran los siguientes posibles trabajos futuros:

- Evaluar el desempeño del CREASE-NGHS en otras *shapes*.
- Implementar y evaluar en el CREASE metaheurísticas orientadas a resolver problemas multimodales, para abordar el problema de la degeneración que genera múltiples mínimos en el paisaje de búsqueda, como el Algoritmo de Búsqueda de Ranas Saltarinas (Shuffled Frog Leaping Algorithm, SFLA) y Optimización por Búsqueda de Campo de Fútbol (Football Field Optimization, FFO).
- Determinar los parámetros de las diversificaciones planteadas como estrategia de convergencia prematura que permitan mejorar el rendimiento del NGHS en CREASE y otras aplicaciones.
- Implementar y evaluar el uso de información heurística en el proceso de optimización de la *shape* de soluciones de baja concentración de vesículas ensamblas a partir de macromoléculas anfifílica.

VIII. Bibliografía

[1] L. A. Feigin and D. I. Svergun, *Structure Analysis by Small-Angle X-Ray and Neutron Scattering*, 1st ed. Springer US, 1987. doi: 10.1007/978-1-4757-6624-0.

[2] O. Glatter and O. Kratky, Eds., *Small angle X-ray scattering*. London: Academic Press, 1982.

[3] C. M. Heil, A. Patil, A. Dhinojwala, and A. Jayaraman, "Computational Reverse-Engineering Analysis for Scattering Experiments (CREASE) with Machine Learning Enhancement to Determine Structure of Nanoparticle Mixtures and Solutions," *ACS Central Science*, vol. 8, no. 7, pp. 996–1007, Jul. 2022, doi: 10.1021/acscentsci.2c00382.

[4] C. M. Jeffries *et al.*, "Small-angle X-ray and neutron scattering," *Nature Reviews Methods Primers*, vol. 1, no. 1, pp. 1–39, Oct. 2021, doi: 10.1038/s43586-021-00064-9.

[5] D. Lombardo, P. Calandra, and M. A. Kiselev, "Structural Characterization of Biomaterials by Means of Small Angle X-rays and Neutron Scattering (SAXS and SANS), and Light Scattering Experiments," *Molecules*, vol. 25, no. 23, p. 5624, Nov. 2020, doi: 10.3390/molecules25235624.

[6] Y. Wei and M. J. A. Hore, "Characterizing polymer structure with small-angle neutron scattering: A Tutorial," *Journal of Applied Physics*, vol. 129, no. 17, p. 171101, May 2021, doi: 10.1063/5.0045841.

[7] X. Li *et al.*, "Contrast Variation Application in Small-Angle Neutron Scattering Experiments," *Journal of Physics: Conference Series*, vol. 351, no. 1, p. 012002, Mar. 2012, doi: 10.1088/1742-6596/351/1/012002.

[8] D. F. Coral-Coral and J. A. Mera-Córdoba, "Applying SAXS to study the structuring of Fe3O4 magnetic nanoparticles in colloidal suspensions," *DYNA*, vol. 86, no. 209, pp. 135–140, Apr. 2019, doi: 10.15446/dyna.v86n209.73450.

[9] M. V. Petoukhov and D. I. Svergun, "Global rigid body modeling of macromolecular complexes against small-angle scattering data," *Biophysical Journal*, vol. 89, no. 2, pp. 1237–1250, Aug. 2005, doi: 10.1529/biophysj.105.064154.

[10] Z. Ye, Z. Wu, and A. Jayaraman, "Computational Reverse Engineering Analysis for Scattering Experiments (CREASE) on Vesicles Assembled from Amphiphilic Macromolecular Solutions," *JACS Au*, vol. 1, no. 11, pp. 1925–1936, Nov. 2021, doi: 10.1021/jacsau.1c00305.

[11] I. Breßler, J. Kohlbrecher, and A. F. Thünemann, "SASfit: A tool for small-angle scattering data analysis using a library of analytical expressions," *Journal of Applied Crystallography*, vol. 48, no. 5, pp. 1587–1598, Oct. 2015, doi: 10.1107/s1600576715016544.

[12] H. Schnablegger and Y. Singh, *The SAXS Guide Getting acquainted with the principles*, 5th ed. Graz - Austria: Anton Paar GmbH, 2023. Accessed: Dec. 23, 2023. [Online]. Available: www.anton-paar.com

[13] Jayaraman Research Lab, "CREASE Documentation." Accessed: Sep. 22, 2023. [Online]. Available: https://crease-ga.readthedocs.io/

[14] A. Patil *et al.*, "Modeling Structural Colors from Disordered One-Component Colloidal Nanoparticle-Based Supraballs Using Combined Experimental and Simulation Techniques," *ACS Materials Letters*, vol. 4, no. 9, pp. 1848–1854, Sep. 2022, doi: 10.1021/acsmaterialslett.2c00524.

[15] S. V. R. Akepati, N. Gupta, and A. Jayaraman, "Computational Reverse Engineering Analysis of Scattering Experiments Method for Interpretation of 2D Small-Angle Scattering Profiles (CREASE-2D)," *JACS Au*, Jan. 2024, doi: 10.1021/jacsau.4c00068.

[16] M. G. Wessels and A. Jayaraman, "Machine Learning Enhanced Computational Reverse Engineering Analysis for Scattering Experiments (CREASE) to Determine Structures in Amphiphilic Polymer

Solutions," *ACS Polymers Au*, vol. 1, no. 3, pp. 153–164, Dec. 2021, doi: 10.1021/acspolymersau.1c00015.

[17] R. L. McGreevy, "Reverse Monte Carlo modelling," *Journal of Physics: Condensed Matter*, vol. 13, no. 46, p. R877, Nov. 2001, doi: 10.1088/0953-8984/13/46/201.

[18] J. A. Vasconcelos, J. A. Ramírez, R. H. C. Takahashi, and R. R. Saldanha, "Improvements in genetic algorithms," *IEEE Transactions on Magnetics*, vol. 37, no. 5, pp. 3414–3417, 2001, doi: 10.1109/20.952626.

[19] R. Hassan, B. Cohanim, O. De Weck, and G. Venter, "A comparison of particle swarm optimization and the genetic algorithm," *AIAA Journal*, pp. 1138–1150, 2005, doi: 10.2514/6.2005-1897.

[20] A. S. Ghiduk and A. Alharbi, "Generating of Test Data by Harmony Search Against Genetic Algorithms," *Intelligent Automation & Soft Computing*, vol. 36, no. 1, pp. 647–665, Sep. 2022, doi: 10.32604/iasc.2023.031865.

[21] N. Ranjbar, S. Anvari, and M. Delavar, "The application of harmony search and genetic algorithms for the simultaneous optimization of integrated reservoir–FARM systems (IRFS)," *Irrigation and Drainage*, vol. 70, no. 4, pp. 743–756, Oct. 2021, doi: 10.1002/ird.2567.

[22] Y. H. Kim, Y. Yoon, and Z. W. Geem, "A comparison study of harmony search and genetic algorithm for the max-cut problem," *Swarm and Evolutionary Computation*, vol. 44, pp. 130–135, Feb. 2019, doi: 10.1016/j.swevo.2018.01.004.

[23] M. Ghozi and A. Budiati, "Comparison of Genetic Algorithm and Harmony Search Method for 2D Geometry Optimization.," *MATEC Web of Conferences*, vol. 159, p. 01009, Mar. 2018, doi: 10.1051/matecconf/201815901009.

[24] C. Peraza, F. Valdez, and O. Castillo, "A harmony search algorithm comparison with genetic algorithms," in *Studies in Computational Intelligence*, 1st ed., vol. 574, O. Castillo and P. Melin, Eds., Springer Verlag, 2014, pp. 105–123. doi: 10.1007/978-3-319-10960-2_7.

[25] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67–82, 1997, doi: 10.1109/4235.585893.

[26] C. Cobos., J. Pérez., and D. Estupiñan., "Una revisión de la búsqueda armónica," 2011, Accessed: Sep. 26, 2023. [Online]. Available: https://repositorio.unal.edu.co/handle/unal/38743

[27] F. Qin, A. M. Zain, and K. Q. Zhou, "Harmony search algorithm and related variants: A systematic review," *Swarm and Evolutionary Computation*, vol. 74, pp. 101–126, Oct. 2022, doi: 10.1016/j.swevo.2022.101126.

[28] J. Brownlee, *Clever Algorithms: Nature-inspired Programming Recipes*, 2nd ed. Lulu.com, 2011. Accessed: May 20, 2024. [Online]. Available: http://www.cleveralgorithms.com

[29] S. Luke, *Essentials of Metaheuristics*, 2nd ed. Lulu.com, 2013. Accessed: May 20, 2024. [Online]. Available: https://cs.gmu.edu/~sean/book/metaheuristics/

[30] S. Gupta, "Enhanced harmony search algorithm with non-linear control parameters for global optimization and engineering design problems," *Engineering with Computers*, vol. 38, no. 4, pp. 3539–3562, Oct. 2022, doi: 10.1007/s00366-021-01467-8.

[31] F. Wilcoxon, "Individual Comparisons by Ranking Methods," *Biometrics Bulletin*, vol. 1, no. 6, pp. 80–83, Dec. 1945, doi: 10.2307/3001968.

[32] J. Alcalá-Fdez *et al.*, “KEEL: A software tool to assess evolutionary algorithms for data mining problems,” *Soft Computing*, vol. 13, no. 3, pp. 307–318, May 2009, doi: 10.1007/s00500-008-0323-y.

[33] D. Zou, L. Gao, J. Wu, S. Li, and Y. Li, “A novel global harmony search algorithm for reliability problems,” *Computers & Industrial Engineering*, vol. 58, no. 2, pp. 307–316, Mar. 2010, doi: 10.1016/j.cie.2009.11.003.

[34] Q. K. Pan, P. N. Suganthan, M. F. Tasgetiren, and J. J. Liang, “A self-adaptive global best harmony search algorithm for continuous optimization problems,” *Applied Mathematics and Computation*, vol. 216, no. 3, pp. 830–848, Apr. 2010, doi: 10.1016/j.amc.2010.01.088.

[35] M. G. H. Omran and M. Mahdavi, “Global-best harmony search,” *Applied Mathematics and Computation*, vol. 198, no. 2, pp. 643–656, May 2008, doi: 10.1016/j.amc.2007.09.004.

[36] M. Lotif, “Visualizing the population of meta-heuristics during the optimization process using self-organizing maps,” *Proceedings of the 2014 IEEE Congress on Evolutionary Computation, CEC 2014*, pp. 313–319, Sep. 2014, doi: 10.1109/cec.2014.6900265.

[37] A. Flexer, “On the use of self-organizing maps for clustering and visualization,” *Intelligent Data Analysis*, vol. 5, no. 5, pp. 373–384, Jan. 2001, doi: 10.3233/ida-2001-5502.

[38] T. Narayanan, “Small-Angle Scattering,” in *Structure from Diffraction Methods: Inorganic Materials Series*, D. W. Bruce, D. O’Hare, and R. I. Walton, Eds., John Wiley & Sons, Ltd, 2014, pp. 259–324. doi: 10.1002/9781118695708.ch5.

[39] J. Als-Nielsen and D. McMorrow, “Kinematical scattering I: non-crystalline materials,” in *Elements of Modern X-ray Physics*, John Wiley & Sons, Ltd, 2011, pp. 113–146. doi: 10.1002/9781119998365.ch4.

[40] Anton Paar GmbH, “SAXS nanostructure analysis.” Accessed: Jun. 07, 2024. [Online]. Available: https://wiki.anton-paar.com/mx-es/analisis-de-la-nanoestructura-de-saxs/

[41] K. S. Pratt, “Design Patterns for Research Methods: Iterative Field Research,” *AAAI Spring Symposium*, 2009, Accessed: Apr. 06, 2024. [Online]. Available: www.aaai.org

IX. Anexos

IX.1. Anexo I: Enlaces de interés

Enlaces para acceder a:

1. Documentación de la herramienta CREASE:
    - https://crease-ga.readthedocs.io/en/latest/index.html
2. Repositorio en GitHub del CREASE original (*crease_ga*):
    - https://github.com/arthijayaraman-lab/crease_ga
3. Repositorio en GitHub del CREASE adaptado para la realización de este trabajo (*crease_he*):
    - https://github.com/cha-do/crease_heuristic/

IX.2. Anexo II: Participaciones en congresos y contribuciones científicas

Como logros del presente trabajo de investigación, desarrollado en el grupo de investigación de Ciencia y Tecnología de Materiales Cerámicos CYTEMAC y en el grupo de investigación y desarrollo en Tecnologías de la Información GTI de la Universidad del Cauca, se tienen cuatro contribuciones científicas tituladas:

1. Using Data Science Techniques to Analyze SAXS patterns
2. Using metaheuristic algorithms to analyze Small-angle Scattering patterns
3. Uso de algoritmos metaheurísticos para el análisis de patrones de Small-angle Scattering
4. Performance evaluation of the NGHS metaheuristic as an alternative to the dynamic adaptive GA in the CREASE tool in SAS profile analysis of nanoparticulate systems

El primer trabajo (Anexo II-1) se presentó en la modalidad de poster en el *41st International Conference on Vacuum Ultraviolet and X-ray Physics (VUVX 2023),* celebrado desde el 3 al 7 de junio de 2023 en el *Brazilian Synchrotron Light Laboratory (LNLS) of the Brazilian Center for Research in Energy and Materials (CNPEM) and the Institute of Physics of University of Campinas (UNICAMP),* en Campinas-SP, Brasil (modalidad presencial).

El segundo trabajo (Anexo II-2) se presentó en la modalidad de poster en el *31st international materials research congress (IMRC2023),* celebrado desde el 13 al 18 de agosto de 2023 organizado por la *sociedad Mexicana de Materiales (SMMater) y la Materials Research Society® (MRS),* en Cancún, México (modalidad virtual).

El tercer trabajo (Anexo II-3) se presentó en modalidad oral en el *VII Congreso Nacional de Ingeniería Física - II Applied Physics, Engineering and Innovation conference,* celebrado desde el 22 al 27 de octubre de 2023 en la *Universidad Nacional de Colombia sede Manizales - Sociedad Colombiana de Ingeniería Física SCIF,* en Manizales, Colombia (modalidad presencial).

El cuarto trabajo (Anexo II-4) es un artículo científico que está sometido en proceso de revisión para su publicación en la “International Journal of Industrial Engineering Computations” (ISSN 1923-2934) indexada en Publindex en categoría A1 para el 2023.

A continuación, se presenta la imágenes y resúmenes de las contribuciones.

USING DATA SCIENCE TECHNIQUES TO ANALYZE SAXS PATTERNS

Stibel Alejando Camayo Muñoz [1†*], Diego Felipe Ramírez Chávez [1†], Diego Fernando Coral [1], Carlos Alberto Cobos Lozada [2],

[1] Departamento de Física, Universidad del Cauca, Popayán-Colombia
[2] Departamento de Sistemas, Universidad del Cauca, Popayán-Colombia
[†] These authors have contributed equally to this work and share first authorship
*e-mail: stibelal@unicauca.edu.co

ABSTRACT: Small-angle X-ray scattering (SAXS) is a useful technique to analyze the physical structure of colloids formed by proteins, polymers, and nanomaterials, among others [1]. The success of SAXS lies in the ability to choose the appropriate mathematical model that allows obtaining physical information about the analyzed system [2]. The "Computational Reverse Engineering Analysis for Small-Angle Scattering Experiments" (CREASE) is a method recently developed to analyze results from SAXS experiments [3]. Results show that it is possible to reconstruct the morphology of the scattering object by analyzing a SAXS curve, using the appropriate mathematical model (size distribution, form factor and structure factor) and to validate results using data science techniques. The aim of this work is the evaluation of the metaheuristic algorithms: Particle Swarm Optimization (PSO) and Global-best Harmony Search (GHS), as an alternative to the adaptive genetic algorithm (AGA) of discrete variables used in the CREASE.

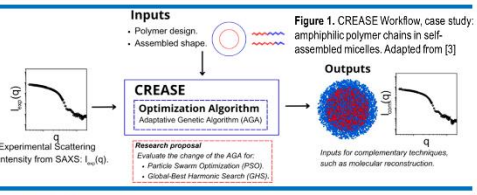

Figure 1. CREASE Workflow, case study: amphiphilic polymer chains in self-assembled micelles. Adapted from [3]

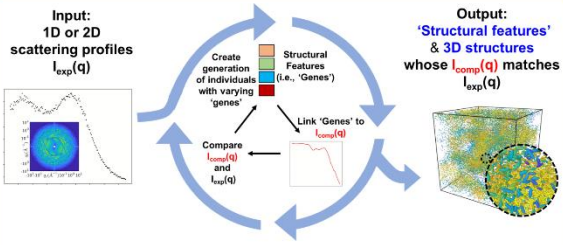

Figure 2. General workflow in CREASE where experimentally measured 1D scattering profiles are taken as input and CREASE, through an internal optimization, generates as output the key structural features as well as representative 3D real space structures whose computed scattering profiles match the experimental scattering input. Taken from [4].

INTRODUCTION

The *CREASE method* is based on the Rigid-body modeling technique which consists of modeling the sample or part of it as a conformation of subunits (rigid-bodies), with known structure and by means of displacements and rotations of these to find the spatial arrangement whose dispersion intensity $I_{comp}(q)$ best fits the experimental $I_{exp}(q)$ [5], *Figure 2*. The CREASE method uses parameters from which the subunits will be located, *Figure 3a*, these parameters depend on the type of sample to be analyzed. With this modification it is possible to lower the dimensionality of the optimization problem and therefore reduce its computational cost.

Calculation of $I_{comp}(q)$:

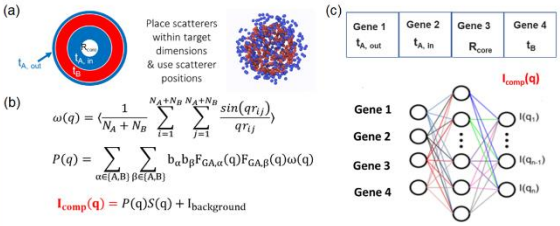

Figure 3. (a) Methodology for rigid-body location from parameters used in CREASE for vesicles. **(b)** Calculation of the scattering intensity $I_{comp}(q)$ by Debye's formulation. **(c)** Calculation of the scattering intensity using NNs. Adapted from [6].

The function used to evaluate the fit of $I_{comp}(q)$ with $I_{exp}(q)$ is the Sum of Squared estimate of Errors (SSE) metric.

$$SSE = \sum_i \left[\log\left(\frac{I_{exp}(q_i)}{I_{comp}(q_i)} \right) \right]^2 \qquad (1)$$

The objective of this research is to evaluate the effectiveness (efficiency and efficacy) of CREASE using the metaheuristic algorithms PSO and GHS as an alternative to AGA.

METHODOLOGY

To perform this evaluation, the CREASE method will be run with the AGA, PSO and GHS on 4 benchmark functions of vesicle solution samples, for each combination of optimization algorithm and benchmark, 30 repetitions of the experiment will be performed. This will be performed on a cluster of 20 Intel® Core™ i7 -core processors.

From these runs, a record will be taken of the parameters and the SSE of the best solution obtained, and the total execution time; on these data, the Friedman non-parametric tests will be performed, which will allow evaluating if there is a significant difference in the results between the algorithms and ordering them from best to worst.

PRELIMINARY RESULTS

CREASE was run with the GHS on computers with the characteristics described before 5 times, on the following benchmark.

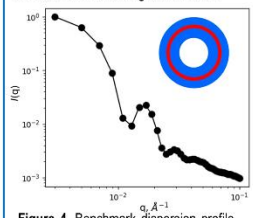

Figure 4. Benchmark dispersion profile, $I_{exp}(q)$. Taken from [3].

Benchmark parameters: Small core ($R_{core} = 10\ nm$), thick A layers ($t_{A_{in}} = t_{A_{out}} = 12\ nm$), thin B layer ($t_B = 6\ nm$).

AGA hyperparameters used: Population size: 80, generations: 100.

GHS hyperparameters used: chosen in such a way that the same number of structures (8000) were evaluated as with AGA.
Harmony memory size (HMS): 80
Harmony memory considering rate ($HMCR$): 0.9
Pitch adjusting rate (PAR): 0.35
Number of improvisations (NI): 7920

The results obtained were as follows:

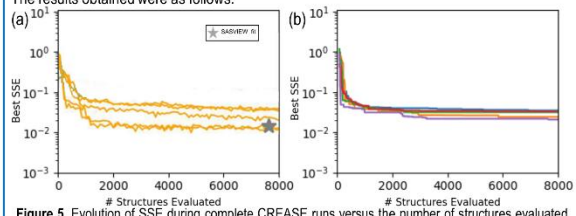

Figure 5. Evolution of SSE during complete CREASE runs versus the number of structures evaluated, using **(a)** AGA, adapted from [3], and **(b)** GHS.

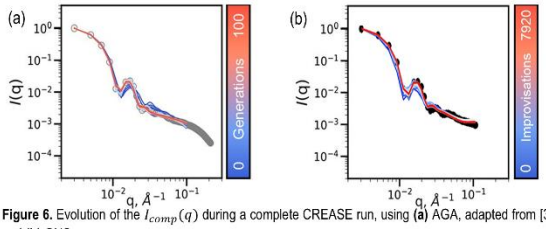

Figure 6. Evolution of the $I_{comp}(q)$ during a complete CREASE run, using **(a)** AGA, adapted from [3], and **(b)** GHS.

Method	R_{core} (nm)	$t_{A_{in}}$ (nm)	t_B (nm)	$t_{A_{out}}$ (nm)	$D_{vesicle}$ (nm)
Target	10.0±2.0	12.0	6.0	12.0	80.0±4.0
AGA*	9.2±3.0	11.3±3.8	7.4±1.6	9.0±1.7	73.9±10.8
GHS	10.8±**2.7**	**12.3±3.0**	**6.7±1.3**	**9.6**±2.0	**78.8±9.4**

Table 1. Structural parameters of vesicles expected and obtained. *Taken from [3].

ANALYSIS AND CONCLUSIONS

The GHS appears to exhibit good performance on the benchmark used, matching and surpassing the results achieved by the AGA, *Figure 5,6* and *Table 1*. However, further testing is needed on other benchmarks to verify the performance of the CREASE using GHS. It can be observed that the GHS shows faster evolution of the SSE at the beginning of the CREASE executions compared to the AGA, *Figure 5*. This convergence speed can be adjusted to avoid getting stuck in local minima by adjusting the PAR and HMCR parameters.

Acknowledgments: The authors want to thank the Universidad del Cauca. Also, Prof. Arthi Jayaraman's research lab for the support in using its tool CREASE.

References:

[1] Glatter-Kratky. Small angle x-ray scattering. Academic Press, 1982.
[2] D. F. Coral-Coral et al., Dyna, 2019, doi: 10.15446/dyna.v86n209.73450.
[3] Z. Ye et al., JACS Au, 2021, doi: 10.1021/jacsau.1c00305.
[4] M. V. Petoukhov et al., Biophys J. Aug. 2005, doi: 10.1529/biophysj.105.064154.
[5] C. M. Heil et al., ACS Cent Sci, 2022, doi: 10.1021/acscentsci.2c00382.
[6] M. G. Wessels et al., ACS Polymers Au, 2021, doi: 10.1021/acspolymersau.1c00015.

IX.2.A. Anexo II-2

3ST INTERNATIONAL MATERIALS RESEARCH CONGRESS 2023 Cancún, México
13-18 August, 2023
Sociedad Mexicana de Materiales A.C.
MRS MATERIALS RESEARCH SOCIETY

USING METAHEURISTIC ALGORITHMS TO ANALYZE SMALL-ANGLE SCATTERING PATTERNS

Diego Felipe Ramírez Chávez[1†*], Stibel Alejando Camayo Muñoz[1†], Diego Fernando Coral[1], Carlos Alberto Cobos Lozada[2],
[1] Departamento de Física, Universidad del Cauca, Popayán-Colombia
[2] Departamento de Sistemas, Universidad del Cauca, Popayán-Colombia
[†] These authors have contributed equally to this work and share first authorship
*e-mail: rcdiego@unicauca.edu.co

Universidad del Cauca
GRUPO CYTEMAC
GRUPO I+D EN TECNOLOGÍAS DE LA INFORMACIÓN

ABSTRACT: Small-angle scattering (SAS) is a useful technique to analyze the physical structure of colloids formed by proteins, polymers, and nanomaterials, among others [1]. The success of SAS lies in the ability to choose the appropriate mathematical model that allows obtaining physical information about the analyzed system [2]. The "Computational Reverse Engineering Analysis for Small-Angle Scattering Experiments" (CREASE) is a method recently developed to analyze results from SAS experiments [3]. Results show that it is possible to reconstruct the morphology of the scattering object by analyzing a SAS curve, using the appropriate mathematical model (size distribution, form factor and structure factor) and to validate results using data science techniques. The aim of this work is the evaluation of the metaheuristic algorithm Global-best Harmony Search (GHS), as an alternative to the adaptive genetic algorithm (AGA) of discrete variables used in the CREASE.

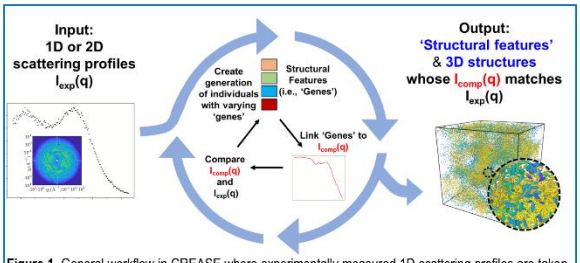

Figure 1. General workflow in CREASE where experimentally measured 1D scattering profiles are taken as input and CREASE, through an internal optimization, generates as output the key structural features as well as representative 3D real space structures whose computed scattering profiles match the experimental scattering input. Taken from [4].

INTRODUCTION

The *CREASE method* is based on the Rigid-body modeling technique which consists of modeling the sample or part of it as a conformation of subunits (rigid-bodies), with known structure and by means of displacements and rotations of these to find the spatial arrangement whose dispersion intensity $I_{comp}(q)$ best fits the experimental $I_{\exp}(q)$ [5], *Figure 1*. The CREASE method uses parameters from which the subunits will be located, *Figure 2a*, these parameters depend on the type of sample to be analyzed. With this modification it is possible to lower the dimensionality of the optimization problem and therefore reduce its computational cost.

Calculation of $I_{comp}(q)$:

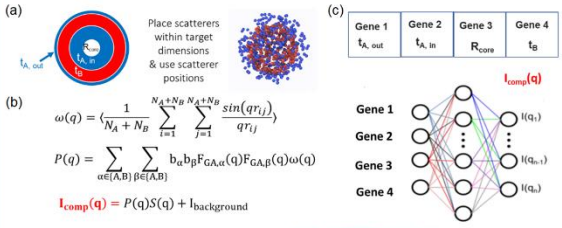

Figure 2. **(a)** Methodology for rigid-body location from parameters used in CREASE for vesicles. **(b)** Calculation of the scattering intensity $I_{comp}(q)$ by Debye's formulation. **(c)** Calculation of the scattering intensity using NNs. Adapted from [6].

The function used to evaluate the fit of $I_{comp}(q)$ with $I_{exp}(q)$ is the Sum of Squared estimate of Errors (SSE) metric.

$$SSE = \sum_i \left[\log\left(\frac{I_{exp}(q_i)}{I_{comp}(q_i)} \right) \right]^2 \quad (1)$$

The objective of this research is to evaluate the effectiveness (efficiency and efficacy) of CREASE using the metaheuristic algorithm GHS as an alternative to AGA.

METHODOLOGY

To perform this evaluation, the CREASE method will be run with the AGA and GHS on 4 benchmark functions of vesicle solution samples, for each combination of optimization algorithm and benchmark, 30 repetitions of the experiment will be performed. This will be performed on a cluster of 20 Intel® Core™ i7 -core processors.

From these runs, a record will be taken of the parameters and the SSE of the best solution obtained, and the total execution time; on these data, the Friedman non-parametric tests will be performed, which will allow evaluating if there is a significant difference in the results between the algorithms and ordering them from best to worst.

PRELIMINARY RESULTS

Benchmark parameters: Small core ($R_{core} = 10\ nm$), thick A layers ($t_{A_{in}} = t_{A_{out}} = 12\ nm$), thin B layer ($t_B = 6\ nm$).

AGA hyperparameters used: Population size: 80 Generations: 100

GHS hyperparameters used: Harmony memory size (HMS): 80
Harmony memory considering rate ($HMCR$): 0.9
Pitch adjusting rate (PAR): 0.35
Number of improvisations (NI): 7920

The results obtained were as follows:

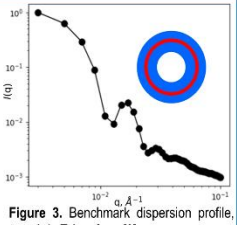

Figure 3. Benchmark dispersion profile, $I_{exp}(q)$. Taken from [3].

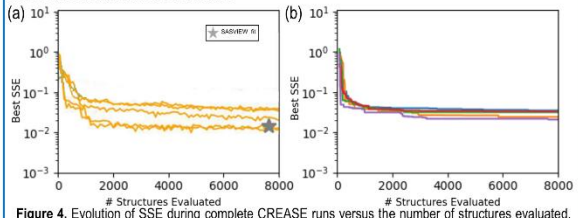

Figure 4. Evolution of SSE during complete CREASE runs versus the number of structures evaluated, using **(a)** AGA, adapted from [3], and **(b)** GHS.

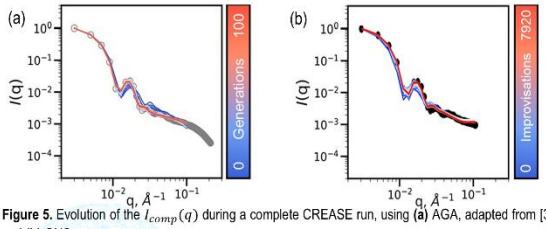

Figure 5. Evolution of the $I_{comp}(q)$ during a complete CREASE run, using **(a)** AGA, adapted from [3], and **(b)** GHS.

Method	R_{core} (nm)	$t_{A_{in}}$ (nm)	t_B (nm)	$t_{A_{out}}$ (nm)	$D_{vesicle}$ (nm)
Target	10.0±2.0	12.0	6.0	12.0	80.0±4.0
AGA*	9.2±3.0	11.3±3.8	7.4±1.6	9.0±1.7	73.9±10.8
GHS	10.8±**2.7**	**12.3±3.0**	**6.7±1.3**	**9.6**±2.0	**78.8±9.4**

Table 1. Structural parameters of vesicles expected and obtained. *Taken from [3].

ANALYSIS AND CONCLUSIONS

The GHS appears to exhibit good performance on the benchmark used, matching and surpassing the results achieved by the AGA, *Figure 4,5* and *Table 1*. However, further testing is needed on other benchmarks to verify the performance of the CREASE using GHS. It can be observed that the GHS shows faster evolution of the SSE at the beginning of the CREASE executions compared to the AGA, *Figure 4*. This convergence speed can be adjusted to avoid getting stuck in local minima by adjusting the PAR and HMCR parameters.

Acknowledgments: The authors want to thank the Universidad del Cauca. Also, Prof. Arthi Jayaraman's research lab for the support in using its tool CREASE.

References:

[1] Glatter-Kratky. Small angle x-ray scattering. Academic Press, 1982.
[2] D. F. Coral-Coral et al., Dyna, 2019, doi: 10.15446/dyna.v86n209.73450.
[3] Z. Ye et al., JACS Au, 2021, doi: 10.1021/jacsau.1c00305.
[4] M. V. Petoukhov et al., Biophys J. Aug. 2005, doi: 10.1529/biophysj.105.064154.
[5] C. M. Heil et al., ACS Cent Sci, 2022, doi: 10.1021/acscentsci.2c00382.
[6] M. G. Wessels et al., ACS Polymers Au, 2021, doi: 10.1021/acspolymersau.1c00015.

IX.2.A. Anexo II-3

USO DE ALGORITMOS METAHEURÍSTICOS PARA EL ANÁLISIS DE PATRONES DE SMALL-ANGLE SCATTERING

Diego Felipe Ramírez Chávez[1], Stibel Alejandro Camayo Muñoz[1], Carlos Alberto Cobos Lozada[2], Diego Fernando Coral Coral[1]
[1]Universidad del Cauca, Departamento de Física, Colombia.
[2]Universidad del Cauca, Departamento de Sistemas, Colombia.
rcdiego@unicauca.edu.co

Las técnicas de caracterización Small-Angle Scattering (SAS) son útiles para analizar la estructura física en escalas coloidales de muestras formados por proteínas, polímeros y nanomateriales, entre otros, [1]. El éxito de las técnicas SAS radica en la capacidad de elegir el modelo matemático adecuado que permita obtener información física sobre el sistema analizado [2]. El "Computational Reverse-Engineering Analysis for Scattering Experiments" (CREASE) es un método desarrollado recientemente para analizar resultados de experimentos SAS [3]. Los resultados muestran que es posible reconstruir la morfología del objeto de dispersión mediante el análisis de una curva SAS, utilizando el modelo matemático apropiado (distribución de tamaño, factor de forma y factor de estructura), y validar los resultados utilizando algoritmos metaheurísticos y técnicas de ciencia de datos. El objetivo de este trabajo es la evaluación del algoritmo metaheurístico Global-best Harmony Search (GHS), como alternativa al algoritmo genético adaptativo (AGA) de variables discretas utilizado en el CREASE.

Referencias:

[1] Glatter-Kratky, Small angle x-ray scattering. Academic Press, 1982.
[2] D. F. Coral-Coral et al., Dyna, 2019, doi: 10.15446/dyna.v86n209.73450.
[3] Z. Ye et al., JACS Au, 2021, doi: 10.1021/jacsau.1c00305.

IX.2.A. Anexo II-4

Performance evaluation of the NGHS metaheuristic as an alternative to the dynamic adaptive GA in the CREASE tool in SAS profile analysis of nanoparticulate systems

Diego Felipe Ramírez Chávez[1], Stibel Alejandro Camayo Muñoz[1], Diego Fernando Coral Coral[1] Carlos Alberto Cobos Lozada[2],
[1]Universidad del Cauca, Departamento de Física, Colombia.
[2]Universidad del Cauca, Departamento de Sistemas, Colombia.
rcdiego@unicauca.edu.co

Abstract:

This research focused on intervening in the optimization algorithm used by the Computational Reverse-Engineering Analysis for Scattering Experiments (CREASE) tool to analyze small-angle scattering (SAS) profiles using the Rigid-Body model. CREASE uses the genetic algorithm (GA) with dynamic adaptation as its optimization algorithm. The aim is to evaluate the performance of CREASE by replacing the GA with a Harmony Search (HS)-based metaheuristic, specifically the Nobel Global Harmony Search (NGHS), in the analysis of SAS profiles of low-concentration solutions vesicles-assembled amphiphilic macromolecules. Results showed that NGHS achieved similar accuracy to GA but with higher efficiency, achieving similar quality solutions with only one-sixth, and in some cases one-tenth, the number of fitness function evaluations used by GA. Besides, CREASE-NGHS achieved SAS profile analysis convergence with less than half the number of fitness function evaluations, saving computational resources and facilitating a more complete analysis. In addition, NGHS addressed some shortcomings of the GA optimization process and facilitated its use and adaptation to distinct types of samples for users with little experience in optimization.

Keywords: small-angle scattering, metaheuristics, evaluation of alternative, harmony search, genetics algorithm.

Submission ID: 3243