Contents lists available at GrowingScience

International Journal of Industrial Engineering Computations

homepage: www.GrowingScience.com/ijiec

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[a†*], Stibel Alejandro Camayo Muñoz[a†], Diego Fernando Coral Coral[a] and Carlos Alberto Cobos Lozada[b]

[a]*Universidad del Cauca, Departamento de Física, Colombia*
[b]*Universidad del Cauca, Departamento de Sistemas, Colombia*

C H R O N I C L E

Article history:
Received June 12 2024
Received in Revised Format July 21 2024
Accepted September 10 2024
Available online September 10 2024

Keywords:
Small-angle scattering
Metaheuristics
Evaluation of alternative
Harmony search
Genetics algorithm

A B S T R A C T

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.

1. Introduction

Small Angle Scattering (SAS) techniques, widely recognized as standard analysis techniques, are instrumental in studying the structure of matter and its interactions. These techniques are particularly useful for the analysis of non-periodic structures of colloidal size, with scales ranging from about 10 Å to several thousand Å (Glatter & Kratky, 1982). SAS provides crucial information about the analyzed sample, including its morphology, dimensions, aggregation, and packing state. This information is of significant importance in various scientific and technological fields, such as condensed matter physics, molecular biology, biophysics, polymer science, and metallurgy (Coral-Coral & Mera-Córdoba, 2019; Heil et al., 2022; Jeffries et al., 2021). To extract this information, established mathematical models and methodologies supported by computational processing are used (Breßler et al., 2015; Petoukhov & Svergun, 2005; Ye et al., 2021).

Based on wave scattering principles, SAS techniques involve directing X-ray (SAXS) or neutron (SANS) beams onto samples to produce scattering patterns captured using detectors. These patterns are then averaged to create SAS profiles, depicting scattered intensity (I) against wave vector magnitude (Q), improving data quality and aiding result analysis in some cases (Jeffries et al., 2021; Schnablegger & Singh, 2023).

The Computational Reverse-Engineering Analysis for Scattering Experiments (CREASE) method is a recently developed tool for analyzing SAS profiles. It uses the model-fitting methodology to obtain key structural information from one or more SAXS or SANS profiles of a sample (Heil et al., 2022; Ye et al., 2021). The model used by CREASE is based on Rigid-Body

* Corresponding author †These authors have contributed equally to this work and share first authorship.
E-mail rcdiego@unicauca.edu.co (D. F. R. Chávez)
ISSN 1923-2934 (Online) - ISSN 1923-2926 (Print)
2024 Growing Science Ltd.
doi: 10.5267/j.ijiec.2024.9.001

modeling, which consists of structural descriptors making a 3D reconstruction of the sample or part of it, modeling it as a spatial conformation of rigid bodies, from which its SAS profile is calculated ($I_{comp}(Q)$) using Debye's model for dispersion, and compares it with the experimental SAS profile of the sample ($I_{exp}(Q)$). This approach has the particularity of having a considerable computational cost because the calculation of $I_{comp}(Q)$ requires considerable arithmetic operations, although in some cases, it has been successfully replaced with the use of Machine Learning (ML) models (Heil et al., 2022; Ye et al., 2021). The structural descriptors are changed, and with it, the 3D reconstruction in such a way as to $I_{comp}(Q)$ as close as possible to $I_{exp}(Q)$, and the final 3D reconstruction of the sample is useful for further analysis. In practice, the CREASE tool has proven to perform successfully for SAS profile analysis, achieving structural parameters like the ones obtained with analytical models. CREASE can analyze distinct types of samples, called shapes, such as low and high-concentration nanoparticle solutions and mixtures, and is easily adaptable to new sample types (Heil et al., 2022; Ye et al., 2021).

To search the structural descriptors, an optimization process in which the goal is minimizing the difference between $I_{comp}(Q)$ and $I_{exp}(Q)$ is performed using the Genetic Algorithm (GA) with dynamic adaptation (Vasconcelos et al., 2001). This metaheuristic is a population-based optimization algorithm whose evolution is based on Darwinian evolutionary theory. In this approach, a population of candidate solutions evolves over several generations to find the best possible solution (Vasconcelos et al., 2001). The GA works on a binary encoding of the structural descriptors of the shape, in which the number of bits used for each descriptor is the same and is determined by the user. The number of GA hyperparameters to be set by the user for analysis is eleven.

This internal optimization stage is a critical phase in the performance of CREASE, so improving this optimization would reduce the number of evaluated solutions necessary for the analysis, speeding it up and reducing its computational cost. In this regard, the CREASE team has implemented a version in which they support the GA optimization process with the use of an Artificial Neural Network (ANN), thus achieving an improvement in some cases in the convergence speed and others in the accuracy in the case study of amphiphilic polymer blocks in solution (Wessels & Jayaraman, 2021).

This study exposed a different approach, proposing the replacement of GA with an alternative metaheuristic based on Harmony Search (HS), specifically the Novel Global Harmony Search (NGHS) algorithm (Zou et al., 2010). HS is a type of population-based optimization algorithm inspired by the musical improvisation process, and which, like GA, has multiple variants (Dubey et al., 2011; Qin et al., 2022). It has been reported that HS shows superior performance to GA in different scenarios (Ghiduk & Alharbi, 2022; Peraza et al., 2014; Ranjbar et al., 2021, Ruano-Daza et al., 2018; Ruano-Daza et al., 2018).

In this way, it was decided to assess the performance of NGHS in CREASE for the analysis of SAS profiles, specifically in the shape of a low-concentrated solution of vesicles-assembled amphiphilic polymers, as described in (Ye et al., 2021). For this, four SAS profiles were chosen as benchmarks to compare and validate the results of including NGHS in CREASE concerning the same process but performed with GA.

Results showed that with the use of NGHS, CREASE presented a superior performance in speed and adjustment, achieving on average the same result as with the GA, with approximately only one-sixth, and in some cases one-tenth, part of solutions evaluated achieving $I_{comp}(Q)$ better adjustments to $I_{exp}(Q)$. On the other hand, it allows the user to individually configure the accuracy of each shape's structural descriptor, replacing the binary discretization used by GA with a more intuitive and versatile decimal one. Likewise, the NGHS achieves a notable improvement in the use of computational resources and the exploration capacity of CREASE by limiting the number of reevaluations of solutions, thus achieving a four-fold increase in the number of unique solutions evaluated in the search process and decreasing to a third the number of reevaluations that with GA was as high as an order of magnitude of $\sim 10^3$, with NGHS being an order of magnitude less. In addition, NGHS has only three hyperparameters, eight less than GA, facilitating its usability and adjustment for practical and effective use in different shapes.

The rest of the paper is structured as follows: Section 2 provides an overview of the CREASE methodology, details the GA step of CREASE, introduces the NGHS metaheuristic, and outlines the experimental methodology. Section 3 presents the results and discusses the performance of the CREASE method using the NGHS compared to the GA approach in analyzing the scattering intensities of low-concentration solutions of vesicles-assembled amphiphilic polymers. Finally, the conclusion in Section 4 summarizes the paper and suggests potential future directions.

2. Material and methods

This section outlines the methodologies employed in this study. First, the calculation of $I_{comp}(Q)$ is explained, detailing the calculations needed. Then, the operation of CREASE using Genetic Algorithms (GA) with dynamic adaptation is described. The introduction of the Novel Global Harmony Search (NGHS) follows, explaining its key components. Finally, the experimental methodology for the comparison is outlined.

2.1 Calculation of $I_{comp}(Q)$

As previously mentioned, CREASE describes the sample's morphology using structural descriptors. From these descriptors, CREASE currently has two ways of calculating the SAS profile $I_{comp}(Q)$: The Debye model and the use of ML models.

Like many other models used for SAS profile analysis, the Debye model is based on the wavelet dispersion model. It allows to calculate $I_{comp}(Q)$ of anisotropic samples from the spatial coordinates r_j of the sample scatterers. The Rigid-Body modeling uses the discrete version of the Debye model since the sample is modeled as a reconstruction of N_T rigid bodies. The computational cost of calculating $I_{comp}(Q)$ increases with the square of N_T since it uses the distance r_{jk} between each pair of the rigid bodies N_T rigid bodies, making it computationally demanding.

Depending on the shape, the expression for compute $I_{comp}(Q)$ takes different forms; for the shape low concentration of vesicles-assembled amphiphilic polymers, hereafter dilute vesicle solution, the amphiphilic macromolecules are constituted by two types of monomers, solvophilic A and solvophilic B, which are the rigid-bodies in this case. For this shape, $I_{comp}(Q)$ is calculated as:

$$I_{comp}(Q) = \sum_{\alpha\in[A,B]} \sum_{\beta\in[A,B]} b_\alpha b_\beta F_\alpha(Q) F_\beta(Q) \omega(Q) + I_{bg} \tag{1}$$

The monomers are modeled as spherical rigid bodies so that their form factor $F_\alpha\,(Q)$ is that of a sphere of diameter l_α, and I_{bg} is the background scattering intensity. In this case, the computational complexity is in the computation of $\omega(Q)$, known as the intravesicular structure factor.

$$\omega(Q) = \left\langle \frac{1}{N_A + N_B} \sum_{j=1}^{N_A+N_B} \sum_{k=1}^{N_A+N_B} \frac{\sin(Q r_{jk})}{Q r_{jk}} \right\rangle \tag{2}$$

It this case, $N_A + N_B = N_T$. Where N_A and N_B are the numbers of type A and type B rigid bodies, respectively, used in the 3D reconstruction of the vesicles.

To obtain the coordinates r_j of the rigid bodies, which are necessary for the calculation of $I_{comp}(Q)$, CREASE makes a 3D reconstruction of the sample from the structural descriptors, more details about this model can be found in (Ye et al., 2021). For the case of the dilute vesicle solution, descriptors are listed in Table 1.

Table 1
Structural parameters of the shape Solution of low-concentration vesicles assembled from amphiphilic polymers.

Structural parameter	Meaning
R_{core}	Radius of the vesicle core
t_{Ain}	Thickness of the internal solvophilic layer A
t_B	Thickness of the intermediate solvophobic B layer
t_{Aout}	Thickness of the external solvophilic layer A
s_{Ain}	The proportion of total solvophilic dispersants present in the inner layer
σ_R	Core radius dispersion R_{core}
$-\log(I_{bg})$	Negative of the logarithm of the intensity of the I_{bg}

As previously mentioned, the increase in N_T increases the computational cost of running CREASE; however, it also improves the accuracy of the structural parameters obtained from the analysis. The value of N_T value must be set by the user, considering these aspects. This value is set at the beginning of the run by the shape_params, which are the descriptors of the shape; for the dilute vesicle solution, the value of N_T is directly proportional to the ratio n_{sct}/N, being N and n_{sct} the shape_params corresponding to the number of monomers in a chain (polymer) and number of scatterers used to represent a chain, respectively.

Since the computation of $I_{comp}(Q)$ using the Debye method has a significant computational cost, the CREASE team has successfully implemented, for some shapes, ML models trained on thousands of computed scattering profiles calculated from the Debye method for various values of the structural descriptors, given structural descriptors by considerably speeding up their computation against the Debye method, after the initial time investment in gather the training dataset and the training the ML model training (Heil et al., 2022; Wessels & Jayaraman, 2021). Not all available shapes use this methodology; some, such as the dilute vesicle solution, still use the Debye model (Ye et al., 2021).

Once the $I_{comp}(Q)$ is computed, it is fitted to $I_{exp}(Q)$. For this purpose, the quadratic sum of errors (SSE) is used as a merit parameter. As can be seen in Equation 3, SSE metric decreases approaching zero when $I_{comp}(Q)$ fits better with $I_{exp}\,(Q)$.

$$SSE = \sum_i w_i \left[\log\left(I_{exp}(Q_i)\right) - \log\left(I_{comp}(Q_i)\right)\right]^2 \tag{3}$$

The weighing factor wi is calculated as $w_i = log(Q_{i+1}) - log(Q_i)$. For this case, the SSE constitutes the objective function to be minimized, depending on the structural parameters that constitute the decision variables of the problem and the search space.

Since when using the Debye model for the calculation of $I_{comp}(Q)$ in the location of the rigid bodies, there is a certain degree of randomness when $I_{comp}(Q)$ is calculated for the same combination of structural descriptors this can vary slightly and therefore also its fit to $I_{exp}(Q)$ which can be interpreted as noise in the SEE. Another factor to consider is the degeneracy phenomenon, where different combinations of structural descriptors result in similar SAS profiles. On the other hand, it is anticipated that the CREASE search landscape will have noise and be multimodal.

For the analysis of SAS profiles, it is essential to define the Q-range to be analyzed. The morphological information of the sample in real space is encoded in the reciprocal space of wave vectors $\boldsymbol{Q}$ so that a distance d in real space can be related to the value of Q through the expression $Q \approx \pi/d$ (Schnablegger & Singh, 2023); from here it can be seen which larger distances in real space will be related to lower values of Q values and vice versa. The above is usually considered for the choice of the fit metric and the analysis of the results.

2.2 CREASE internal optimization, Genetic Algorithm (GA) with dynamic adaptation

CREASE performs an internal optimization using the Genetic Algorithm with dynamic adaptation metaheuristic to search the combination of structural descriptor values that minimize the SSE. In this approach, an initially random population of number (pop) candidate solutions (individuals) evolve over a number ($gens$) of generations in search of the best possible solution. This version of GA differs from the original in that the hyperparameters mutation probability p_m and crossover probability p_c are dynamic throughout the run, unlike in the original GA where they are constant; these vary according to the genetic diversity measure (GDM) as shown in Equation 5, seeking to maintain an adequate genetic diversity in the population throughout the run.

$$GDM = \frac{minimum\ error\ of\ the\ population}{average\ error\ of\ the\ popualtion} = \frac{SSE_{min}}{SSE_{avg}} \tag{4}$$

The GDM can take values between zero (0) and one (1), being closer to 1 when the error values of the population (SSE in this case) are more homogeneous and to 0 when they are more diverse. The values of p_m and p_c have an initial value of $pm_{initial}$ and $pc_{initial}$ respectively, and are adjusted throughout the run so that the population has an adequate diversity, as follows:

if $GDM > GDM_{max}$ then: // seeking to increase the diversity.
    $pm = pm * k_{GDM}$
    $pc = pc/k_{GDM}$
else if $GDM < GDM_{min}$ then: // seeking to reduce the diversity.
    $pm = pm/k_{GDM}$
    $pc = pc * k_{GDM}$

The allowed values of pm are restricted to the interval (pm_{min}, pm_{max}), and allowed values of pc are restricted to the interval (pc_{min}, pc_{max}).

From now on, it represents a total of eleven hyperparameters for the Genetic Algorithm with Dynamic Adaptation Metaheuristic, GA. Due to the No Free Lunch (NFL) theorem (Wolpert & Macready, 1997), there is no universally optimal configuration of these hyperparameters to analyze all shapes with CREASE, so when adding a new shape, an adjustment of these hyperparameters would be the most appropriate, seeking to guarantee the performance of the tool that justifies its computational cost. More information about the rationale behind choosing these adaptation parameters can be found in (Beltran-Villegas et al., 2019).

To analyze low-concentration vesicle solutions, (Ye et al., 2021) suggest that multiple CREASE runs can provide useful information for the user to understand the degeneracy in vesicle dimensions corresponding to the I_{exp} (Q).

2.3 Novel Global Harmony Search (NGHS)

The NGHS, like other metaheuristics based on the HS, is a population-based optimization algorithm inspired by the process of musical improvisation, where each musician plays a note x_i within a possible range $[x_{iL}, x_{iU}]$ forming a harmonic vector x = ($x_1, x_2, \ldots\ x_k$). In the CREASE optimization problem, a note represents each structural parameter, and a harmony (a vector of notes) is a possible solution. If the set of notes played by the musicians is considered a good harmony, it is stored in the memory of each musician, increasing the possibility of making a good harmony. The HS-base metaheuristics are inspired by a simple concept that results in easy implementation, few hyperparameters, and easy integration to other metaheuristics (Dubey et al., 2021; Qin et al., 2022).

Like the GA, the HS starts from an initial random population of HMS (Harmony Memory Size) harmonies that comprise the initial harmonic memory (HM) $x^1, x^2, \ldots, x^{HMS}$. These harmonic vectors are evaluated, and in each iteration (t), a new harmony $x' = (x'_1, x'_2, \ldots, x'_k)$ is generated. The new harmony is evaluated, and if it is better than the worst existing harmony x^{worst} in the HM, x' replaces it. The process is repeated by NI (Number of Iterations) times.

The structure of NGHS differs from the original HS, mainly in the generation of new harmonies x' for which NGHS includes the adaptive step and the trust region x_R and, with these factors, a new improvisation scheme is designed so that the worst harmonic x^{worst} of the HM is moved to the best harmony x^{best} benefiting from swarm intelligence. Using this new improvisation approach can accelerate the convergence rate; however, it also accelerates premature convergence, soon stalling at local minima. To overcome this disadvantage, a genetic mutation is introduced, with a probability of pm (Zou et al., 2010). The scheme of improvisation of a new harmony x' used by the NGHS is:

for each $i \in [1, k]$ **do**:
  $x_R = 2 \times x_i^{best} - x_i^{worst}$
  if $x_R > x_{iU}$ **then:**
    $x_R = x_{iU}$
  elseif $x_R < x_{iL}$ **then**:
    $x_R = x_{iL}$
  end
  $x_i' = x_i^{worst} + rand() \times (x_R - x_i^{worst})$
  if $rand() > pm$ **then:** // $rand()$: random value ∈ [0,1]
    $x_i' = x_{iL} + rand() \times (x_{iU} - x_{iL})$
  end
end

In total, the NGHS has three hyperparameters. The choice and adjustment of the NGHS for this study was made based on preliminary tests in which it was compared with the following HS-bases metaheuristics: the original HS, Global-Best Harmony Search (GHS) (Omran & Mahdavi, 2008) and the Self-adaptative Global-best Harmony Search (SGHS) (Pan et al., 2010). These were chosen based on their reported performance in the literature and were assessed on the four benchmarks described below, among which the NGHS stood out for its convergence speed and simplicity. Premature convergence strategies based on the diversification of the HM were evaluated, seeking to improve the performance of the NGHS. However, applying these strategies showed slight improvement that did not justify their use. Therefore, it was finally determined that the best NGHS configuration evaluated was obtained with HMS = 40 y pm = 0.07.

2.4 Description of test cases (Benchmarks)

CREASE performance was assessed and compared on four benchmarks corresponding to four *in-silico* SAS profiles, $I_{exp}(Q)$ of diluted solutions of vesicles with known structural parameters. These SAS profiles were obtained, as in (Ye et al., 2021). Their structural parameters were chosen to cover a variety of combinations of relevant features, and they had a reasonable computational cost of analysis due to the considerable number of runs performed in this study. The structural parameters chosen for the benchmarks are listed in Table 2, and their corresponding SAS profiles can be seen in Fig. 1, depicting the dispersion intensity $I(Q)$ and Q-values in logarithmic scale, which is a usual representation.

Table 2
Structural Parameters of the benchmarks, 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

With a core radius dispersion (σ_R) of 20% and a proportion of the total solvophilic scatterers present in the inner layer (s_{Ain}) of 0.20, these values capture the type of scattering observed in experimental samples of vesicle assemblies (Ye et al., 2021).

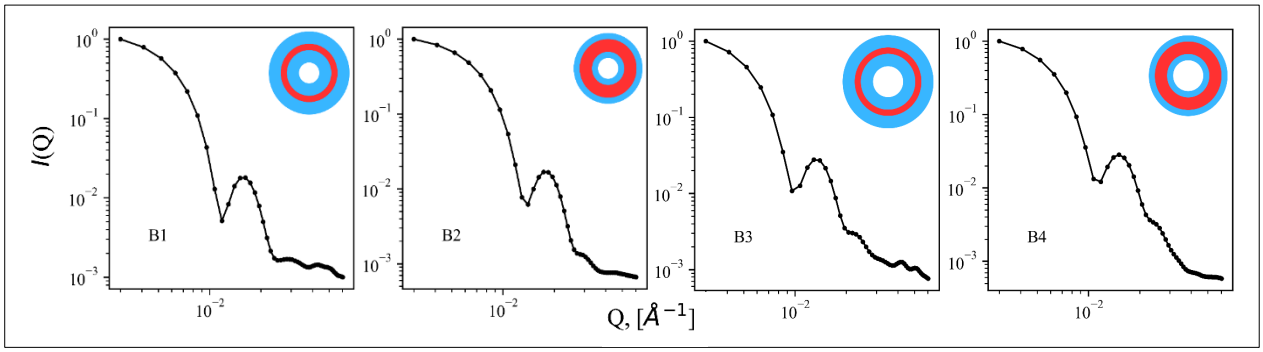

Fig. 1. SAS profiles, $I_{exp}(Q)$, of the four benchmarks (B1, B2, B3, and B4).

The range of Q of $I_{exp}(Q)$ considered for the analysis was from 0.003Å^{-1} to 0.060Å^{-1} with a total of fifty-three (53) Q-values, sufficient to resolve the key characteristics of the dilute vesicle solutions considered.

2.5 CREASE run configurations

All CREASE runs performed on the benchmarks in this research were done with the following configurations:

- The hyperparameter settings used in the GA were the recommended for the shape dilute vesicle solutions (Ye et al., 2021), listed below.
  $GDM_{max} = 0.85$, $GDM_{min} = 0.005$, $k_{GDM} = 1.1$, $pc_{initial} = 0.6$, $pc_{max} = 1$, $pc_{min} = 0.1$, $pm_{initial} = 0.001$, $pm_{max} = 0.25$, $pm_{min} = 0.006$, $pop = 80$ y $gens = 100$
- The following hyperparameter settings were used in the NGHS runs:
  $HMS = 40$, $pm = 0{,}07$ and $NI = 8000$. So that it computes the same number of solutions per run as the GA.
- A ratio $\frac{n_{sct}}{N} = 0.5$ was used, which allowed to obtain results like those reported by (Ye et al., 2021) in this *shape* with a moderate computational cost.
- The search ranges $[x_{iL}, x_{iu}]$ used for the search of the structural parameters in the CREASE executions were $[50\text{Å}, 250\text{Å}]$ for R_{core}, $[30\text{Å}, 200\text{Å}]$ for t_{Ain}, t_B and t_{Aout}, $[0.1, 0.45]$ for s_{Aint}, $[0.0, 0.45]$ for σ_R and [2.5,5.5] for $-log(I_{bg})$.
- For GA runs, it was used $nloci$ = 7, allowing each of the seven structural parameters to take 128 values. For the NGHS, since the user is allowed to choose the accuracy of each parameter, it was used for R_{core}, t_{Ain}, t_B y t_{Aout} zero decimal places and for the s_{Aint}, σ_R and for $-log(I_{bg})$ two decimal places. With this choice for both the GA and NGHS, the total number of combinations of structural parameters would be of the order of $\sim 10^{14}$. However, NGHS distributes the values more appropriately, improving the discretization of the search space for optimization. For example, the parameter $-log(I_{bg})$ parameter can now take 300 possible values instead of the 128 considered by the GA, which is desirable given the weight it has in the calculation of $I_{comp}(Q)$ as can be seen in Equation 1, and therefore of its fit to $I_{exp}(Q)$.
- The number of times a solution has been evaluated was added to the HM of the NGHS, retaining the least SSE obtained. The number of reevaluations was limited to a user-configurable value (mct), set to 10 for all tests. Once a solution has been evaluated mct times, it is added to a "tabu list" to avoid further reevaluations and ensure that new solutions are not equal to any in the list before being evaluated. The information of the lowest SSE and the number of evaluations is no longer considered once the solution leaves the HM, facilitating information management by the metaheuristic. All of this is because the GA does not adequately control the number of reevaluations of a solution during its execution, resulting in populations with numerous identical individuals, a considerable computational expense in reevaluations, and inefficient use of the information obtained.

2.6 Comparison of CREASE-GA and CREASE-NGHS

To make a fair and reliable comparison of the performance of CREASE with GA and the NGHS, thirty-one (31) complete independent runs of CREASE with each metaheuristic in each of the four benchmarks described before were done to obtain an accurate estimate of the average performance and assess the variability of the results. From the data collected from these runs, we compared the average convergence curves, the values of the best SSE obtained, the accuracy obtained from the structural parameters from their Root-Mean-Square Relative Error ($RMSRE$) and the exploration of the search space.

3. Results and Discussion

In this section, the results of the performance of CREASE using the NGHS and GA metaheuristics, CREASE-NGHS and CREASE-GA, respectively, are presented, comparing the average convergence curves, the best-achieved values SSE obtained, the obtained accuracy of the structural parameters for the analyzed benchmarks, as well as the exploration of the search landscape.

3.1 Inner Workings of the CREASE-GA Step and CREASE-NGHS Step

Below, the average convergence curves of the thirty-one runs on each benchmark for the CREASE-GA and CREASE-NGHS can be seen in Fig. 2. These convergence curves consist of the best SSE (SEE_{best}) found on a logarithmic scale as a function of the number of solutions evaluated throughout the run. The logarithmic scale in the SSE_{best} axis is a usual representation in this application (Wessels & Jayaraman, 2021; Ye et al., 2021) practical to improve visualization and facilitate the analysis of the results, but it should be noted that in this case, it amplifies the differences. The SSE_{best} corresponds to the smallest SEE the current population in the GA and the current MH in the NGHS.

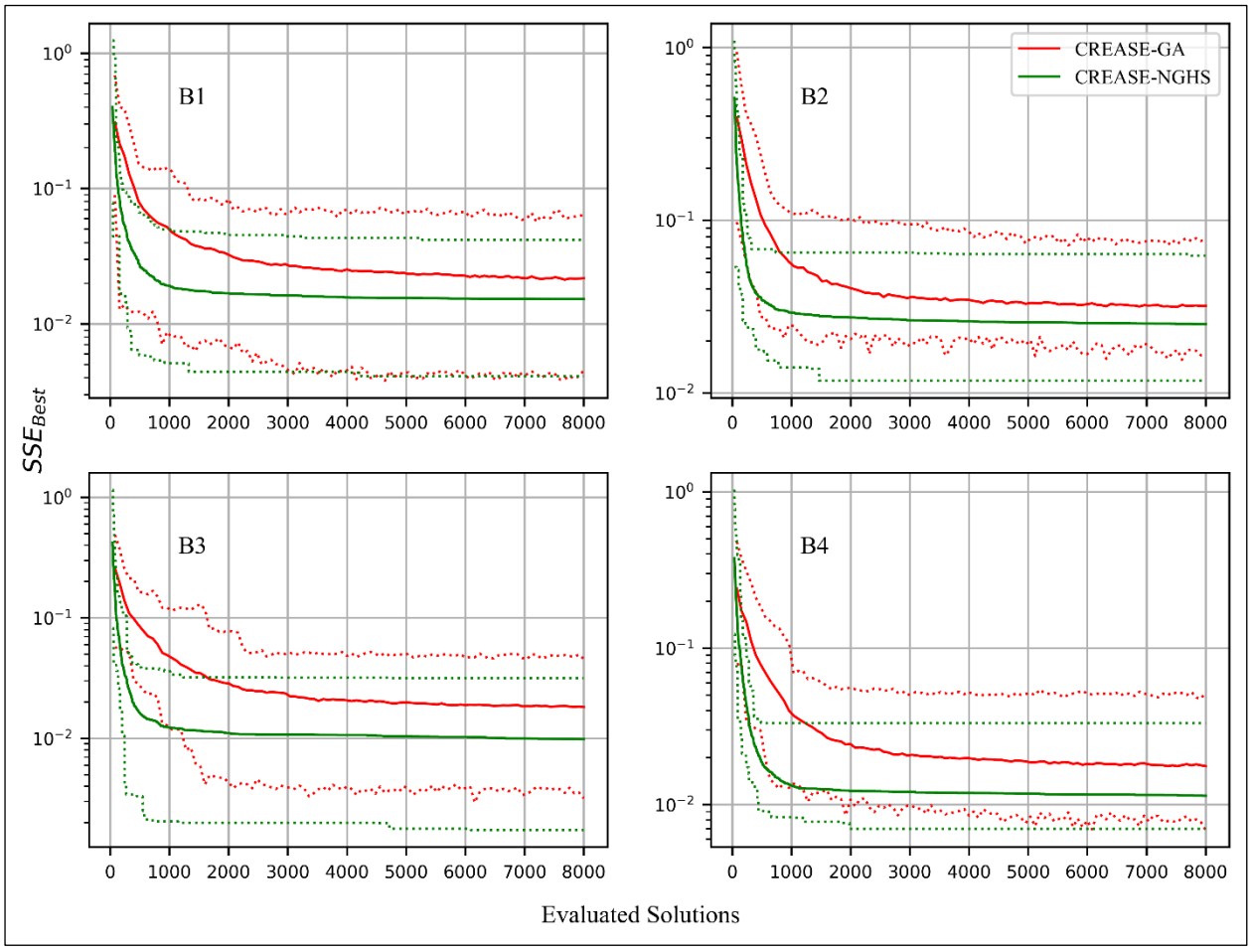

Fig. 2. The average (solid line) minimum (lower dashed line) and maximum (upper dashed line) convergence curves of CREASE-GA (red) and CREAS-NGHS (green) from their thirty-one runs on the four benchmarks.

The results reveal that, in B1, on average, CREASE NGHS (green continuous curve) achieves the same value of SSE_{best}, to which CREASE GA (red continuous curve) converges, but using a smaller number of evaluated solutions, approximately one-sixth. This ratio is reduced to about one-tenth for the other benchmarks (B2, B3, and B4). While, for all benchmarks, CREASE-GA needs to evaluate between 5000 and 7000 solutions to converge, CREASE NGHS achieves it between 2000 and 3000 solutions, in addition to obtaining lower values of SSE in all cases. These results consistently indicate an improvement in the convergence speed of CREASE-NGHS compared to CREASE-GA.

The results represented by the dashed lines correspond to the evolution of the worst (upper dashed line) and best (lower dashed line) SSE of the 31 CREASE-NGHS and CREASE-GA runs. It is observed that the worst CREASE-NGHS performance consistently outperforms its CREASE-GA counterpart, while the best CREASE-HS performance equals or improves the best CREASE-GA results.

The results in Fig. 2 for the average values indicate that in all benchmark cases, the CREASE-NGHS improves on the CREASE-GA both in terms of the SSE value and convergence speed; the average values and their standard deviation can be found in Table 2. From the behavior exhibited, the NI could be reduced for the CREASE-NGHS to 3500 to ensure its convergence in this *shape*, needing to evaluate less than half as many solutions as suggested for CREASE-GA in one run.

3.2 Comparison of CREASE-GA and CREASE-NGHS Outputs

In this section, the structural descriptors of the best structure determined by the 31 runs of CREASE-GA and CREASE-NGHS are compared, except the $-log(I_{bg}),$, which has information on the background scattering intensity and not of the sample, and from these, the precision of each metaheuristic is calculated. The average value and standard deviation of the SSE y $RMSRE$ of the best solution found for each run for each metaheuristic on each benchmark can be found in Table 2. Table 3 shows the average and standard deviation of the structural descriptors, in addition to the total vesicle radius ($R_T = R_{core} + t_{Aint} + t_B + t_{Aout}$), obtained in the 31 runs of both CREASE-GA and CREASE-NGHS, for each benchmark. It is observed that both metaheuristics obtained similar structural parameters in the benchmarks, which coincide within the margin of error in most cases with the expected values (Target), which is further evidenced by analyzing the values of $RMSRE$ values shown in Table 2, whose averages and standard deviations differ very little between the metaheuristics, being only in B3 where there

is an appreciable difference in the average value in favor of CREASE-NGHS. CREASE-NGHS generally achieves the same accuracy in the search for structural parameters as CREASE-GA.

Table 3
SSE y $RMSRE$ (mean and standard deviation) obtained for the thirty-one runs of CREASE GA and CREASE NGHS on each benchmark.

	Alg.	SSE	$RMSRE$
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

Table 4
Structural descriptors (mean and standard deviation) obtained from thirty-one runs of CREASE-GA and CREASE-NGHS on each benchmark (B1, B2, B3, and B4).

	Alg.	R_{core} [Å]	t_{Aint} [Å]	t_B [Å]	t_{Aout} [Å]	σ_{Rcore} [%]	s_{Ain} [%]	R_T [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

The structural parameters obtained show an arrangement and variability like that reported by the authors of this shape (Ye et al., 2021). Even though, as previously observed in Fig. 1, CREASE-NGHS obtained in general SSE lower compared to CREASE-GA, the coincidence between the results of both metaheuristics suggests that this improvement in the SSE was at values of Q which do not significantly influence the dimensions (Wessels & Jayaraman, 2021). Fig. 3 shows the B3 computed SAS profiles of the best solution found by the $I_{comp}(Q)$ of the best solution found by CREASE-GA and CREASE-NGHS, accompanied by the experimental profile $I_{exp}(Q)$.

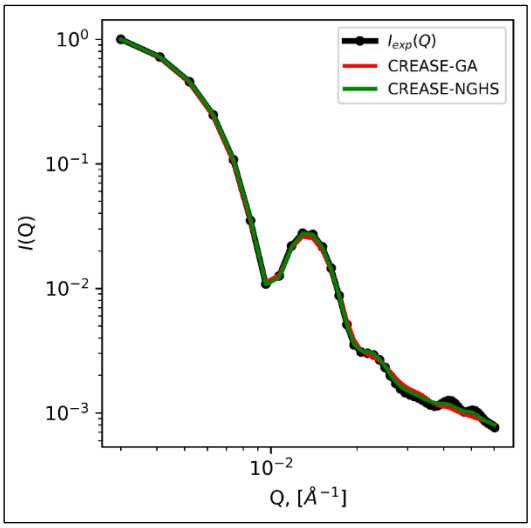

Fig. 3. The B3 computed SAS profile $I_{comp}(Q)$ of the best solution found by CREASE-GA and CREASE-NGHS accompanied by the experimental profile $I_{exp}(Q)$.

It is evident that the solution obtained, in this example, by the CREASE-NGHS achieves a SAS profile computed $I_{comp}(Q)$ slightly better adjusted in the values of Q values higher than $0.02Å^{-1}$ better capturing the characteristics of the profile $I_{exp}(Q)$ in that region.

The variability in the structural parameters identified in the CREASE runs can be attributed to the phenomenon of degeneration, previously mentioned, as well as to the value of n_{sct}/N value chosen for the analysis. As mentioned previously, this value is directly related to N_T; the use of higher values of N_T for the analysis tends to increase the tool's accuracy, although with an increase in its computational cost. Since CREASE-NGHS achieves results with a lower error (SSE) with fewer solutions evaluated, the computational savings in these calculations could be used for using higher values of N_T values for the analysis.

Something to highlight from the structural parameters obtained by CREASE-GA and CREASE-NGHS, listed in Table 1, is that when calculating the total radius of the vesicle, it is observed that it presents a low dispersion compared with the other structural parameters. R_T it is observed that it presents a relatively low dispersion compared to the other structural parameters, in addition to being consistently close to the expected value; this may be because the SAS dispersion is particularly sensitive to the larger dimensions of the structures present in the sample, in this case, the vesicles; this suggests that the total vesicle radius may be a more suitable structural R_T could be a more suitable structural parameter to perform the optimization and could even be used as heuristic information to improve the optimization process in the analysis of this shape.

3.3 Exploration of the Search Landscape

To evaluate the effectiveness of CREASE-GA and CREASE-NGHS in exploring the search space, the percentage of unique solutions (%SU) evaluated in each run was considered for the thirty-one runs of each benchmark. On average, overall benchmarks, in CREASE-GA, only 20.8% of the evaluated solutions were unique, while CREASE-NGHS was 70.9%, 3.5 times more; the remaining percentage corresponded to reevaluations of solutions; more details of these results can be found in Table 6. This high percentage of reevaluations by CREASE-GA is of concern since, although reevaluating can be useful to consider SSE noise, GA does not adequately control for solution repetition in generating new populations, resulting in future populations with multiple repeated solutions. Furthermore, the information obtained from reevaluating solutions is not exploited, as reflected in the noisy behavior of the CREASE-GA convergence curves in Fig. 1.

When analyzing the number of times the same solution is evaluated throughout a run, ES from now on, it is observed that for CREASE-NGHS concerning CREASE-GA, in all the benchmarks, the average of ES maximum (E_{max}) of all runs ($\overline{E_{max}}$) was decreased by one-sixth, the percentages of evaluations corresponding to solutions evaluated with more than one thousand ES (%[ES > 1000]) was reduced on average to 0%, with more than ten ES (%[ES > 10]) to 11.6%, with only 1.0% corresponding to solutions with more than one hundred ES (%[ES > 100]).

Table 5
Evaluation metrics concerning the number of times the same solutions were reevaluated in a CREASE-GA and CREASE-NGHS run for the thirty-one runs of each benchmark.

	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
Global Average	**GA**	**20.8**	**682.5**	**87.4**	**62.9**	**30.6**	**2.5**
	NGHS	**70.9**	**113.4**	**38.5**	**11.6**	**1.0**	**0.0**

Fig. 4 shows a frequency histogram of ES for each benchmark with the results of the 31 CREASE-GA and CREASE-NGHS runs. These histograms show that the NGHS obtains an increase by an order of magnitude for the solutions evaluated only once and a more pronounced decrease than in the CREASE-GA, in the frequency as one advance in values of ES in the graph, so much so that it is achieved to decrease the ES_{max} of all the benchmarks by order of magnitude concerning CREASE-GA, which reaches an order of magnitude of $\sim 10^3$.

Fig. 4 shows an excessive computational investment in the reevaluation of solutions by CREASE-GA, despite the GA's dynamic adaptation, which was significantly reduced in CREASE-NGHS. This is even though CREASE- NGHS, as shown in section 3.1, converges before CREASE-GA.

Additionally, for the CREASE-NGHS histograms of B2, B3, and B4, a peak for ES equal to 10 can be observed. This corresponds to one of the initial configurations, the objective of which is to establish a maximum value of ES (mct) to limit the reevaluation of solutions. Some solutions managed to evade the implemented measure, but better control of the reevaluations was achieved.

All of this implies a notable improvement in the use of computational resources and the exploration capacity of CREASE-NGHS compared to CREASE-GA, allowing the former to find new solutions without sacrificing the algorithm's ability to

deal with noise. Reevaluations of repeated solutions are mainly performed in the exploitation process, so it does not represent a considerable decrease in the convergence speed. Therefore, CREASE-NGHS not only significantly improves the exploration process but also significantly increases the convergence speed.

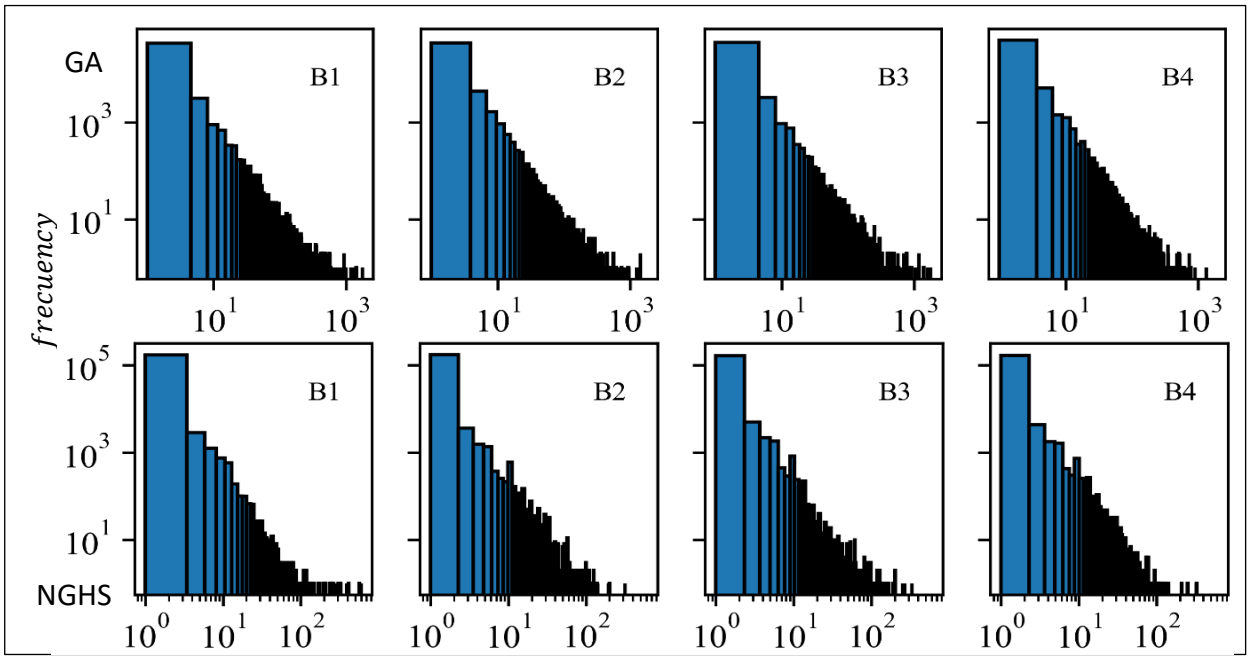

Fig. 4. Frequency histograms of the number of times the same solution is evaluated (ES) in a CREASE-GA (top) and CREASE-NGHS (bottom) run for the thirty-one runs of each benchmark.

4. Conclusions

The NGHS algorithm demonstrated in the CREASE execution a higher efficiency and similar accuracy compared to GA, achieving a significant reduction in the number of evaluated solutions required to reach a solution with the same fitness as the obtained by GA, with only one-sixth to one-tenth as many evaluations. It was observed that the NGHS achieved values of SSE values, which translates into computed profiles that were $I_{comp}(Q)$ more closely matched to the experimental profile $I_{exp}(Q)$, converging on average with the evaluation of between 2000 and 3000 solutions while the GA converges between 5000 and 7000.

The usability of the CREASE tool was also improved in two aspects: the first was the significant reduction in the number of hyperparameters that the user must configure to ensure the correct operation of the metaheuristic in CREASE, going from eleven hyperparameters for the GA to only three for the NGHS, which can facilitate its usability and adjustment for the analysis of new shapes with CREASE. Secondly, the binary discretization of the search space used by the GA was replaced by a version that allows the user to determine the precision with which he/she wishes to analyze each structural parameter separately and decimally, which is more intuitive and versatile.

A notable improvement in the use of computational resources and the exploration capacity of CREASE was achieved with the use of NGHS instead of GA, limiting the number of reevaluations of solutions, thus achieving a three-fold increase in the number of unique solutions evaluated in the search process and reducing to a third the number of reevaluations, which with GA was up to an order of magnitude $\sim 10^3$, the NGHS reduced this by an order of magnitude, without sacrificing the algorithm's ability to deal with the SSE noise in the search process.

5. Data and Tool Availability

The original version of the tool used in this study can be accessed at github.com/arthijayaraman-lab/crease_ga. The version proposed and implemented in this research, which includes the modifications discussed in the article, is available in the following repository: github.com/cha-do/crease_heuristic. Both repositories are freely accessible for reproducibility and further development.

Acknowledgements

This work was made possible thanks to the support of the Vicerrectoría de Investigaciones and the following research groups: Grupo de Óptica y Láser (GOL) and Ciencia y Tecnología de Materiales Cerámicos (CYTEMAC), affiliated with the Department of Physics, and the Grupo de Investigación y Desarrollo en Tecnologías de la Información (GTI), affiliated with the Department of Systems, all from the Universidad del Cauca.

References

Beltran-Villegas, D. J., Wessels, M. G., Lee, J. Y., Song, Y., Wooley, K. L., Pochan, D. J., & Jayaraman, A. (2019). Computational Reverse-Engineering Analysis for Scattering Experiments on Amphiphilic Block Polymer Solutions. *Journal of the American Chemical Society, 141*(37), 14916–14930. https://doi.org/10.1021/jacs.9b08028

Breßler, I., Kohlbrecher, J., & Thünemann, A. F. (2015). SASfit: A tool for small-angle scattering data analysis using a library of analytical expressions. *Journal of Applied Crystallography, 48*(5), 1587–1598. https://doi.org/10.1107/s1600576715016544

Coral-Coral, D. F., & Mera-Córdoba, J. A. (2019). Applying SAXS to study the structuring of Fe3O4 magnetic nanoparticles in colloidal suspensions. *DYNA, 86*(209), 135–140. https://doi.org/10.15446/dyna.v86n209.73450

Dubey, M., Kumar, V., Kaur, M., & Dao, T. P. (2021). A Systematic Review on Harmony Search Algorithm: Theory, Literature, and Applications. Mathematical Problems in Engineering, 2021(1), 5594267. https://doi.org/10.1155/2021/5594267

Ghiduk, A. S., & Alharbi, A. (2022). Generating of Test Data by Harmony Search Against Genetic Algorithms. *Intelligent Automation & Soft Computing, 36*(1), 647–665. https://doi.org/10.32604/IASC.2023.031865

Glatter, O., & Kratky, O. (Eds.). (1982). Small angle X-ray scattering. Academic Press.

Heil, C. M., Patil, A., Dhinojwala, A., & Jayaraman, A. (2022). Computational Reverse-Engineering Analysis for Scattering Experiments (CREASE) with Machine Learning Enhancement to Determine Structure of Nanoparticle Mixtures and Solutions. *ACS Central Science, 8*(7), 996–1007. https://doi.org/10.1021/acscentsci.2c00382

Jeffries, C. M., Ilavsky, J., Martel, A., Hinrichs, S., Meyer, A., Pedersen, J. S., Sokolova, A. V., & Svergun, D. I. (2021). Small-angle X-ray and neutron scattering. *Nature Reviews Methods Primers, 1*(1), 1–39. https://doi.org/10.1038/s43586-021-00064-9

Omran, M. G. H., & Mahdavi, M. (2008). Global-best harmony search. *Applied Mathematics and Computation, 198*(2), 643–656. https://doi.org/10.1016/j.amc.2007.09.004

Pan, Q. K., Suganthan, P. N., Tasgetiren, M. F., & Liang, J. J. (2010). A self-adaptive global best harmony search algorithm for continuous optimization problems. *Applied Mathematics and Computation, 216*(3), 830–848. https://doi.org/10.1016/J.AMC.2010.01.088

Peraza, C., Valdez, F., & Castillo, O. (2014). A harmony search algorithm comparison with genetic algorithms. In O. Castillo & P. Melin (Eds.), *Studies in Computational Intelligence* (Vol. 574, pp. 105–123). Springer Verlag. https://doi.org/10.1007/978-3-319-10960-2_7

Petoukhov, M. V., & Svergun, D. I. (2005). Global rigid body modeling of macromolecular complexes against small-angle scattering data. *Biophysical Journal, 89*(2), 1237–1250. https://doi.org/10.1529/biophysj.105.064154

Qin, F., Zain, A. M., & Zhou, K. Q. (2022). Harmony search algorithm and related variants: A systematic review. Swarm and Evolutionary Computation, 74, 101–126. https://doi.org/10.1016/j.swevo.2022.101126

Ranjbar, N., Anvari, S., & Delavar, M. (2021). The application of harmony search and genetic algorithms for the simultaneous optimization of integrated reservoir–FARM systems (IRFS)*. *Irrigation and Drainage, 70*(4), 743–756. https://doi.org/10.1002/IRD.2567

Ruano-Daza, E., Cobos, C., Torres-Jimenez, J., Mendoza, M., & Paz, A. (2018). A multiobjective bilevel approach based on global-best harmony search for defining optimal routes and frequencies for bus rapid transit systems. *Applied Soft Computing, 67*, 567–583. https://doi.org/10.1016/J.ASOC.2018.03.026

Schnablegger, H., & Singh, Y. (2023). *The SAXS Guide Getting acquainted with the princi*ples (5th ed.). Anton Paar GmbH. www.anton-paar.com

Vasconcelos, J. A., Ramírez, J. A., Takahashi, R. H. C., & Saldanha, R. R. (2001). Improvements in genetic algorithms. *IEEE Transactions on Magnetics, 37*(5 I), 3414–3417. https://doi.org/10.1109/20.952626

Wessels, M. G., & Jayaraman, A. (2021). Machine Learning Enhanced Computational Reverse Engineering Analysis for Scattering Experiments (CREASE) to Determine Structures in Amphiphilic Polymer Solutions. *ACS Polymers Au, 1*(3), 153–164. https://doi.org/10.1021/ACSPOLYMERSAU.1C00015

Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation, 1*(1), 67–82. https://doi.org/10.1109/4235.585893

Ye, Z., Wu, Z., & Jayaraman, A. (2021). Computational Reverse Engineering Analysis for Scattering Experiments (CREASE) on Vesicles Assembled from Amphiphilic Macromolecular Solutions. *JACS Au, 1*(11), 1925–1936. https://doi.org/10.1021/jacsau.1c00305

Zou, D., Gao, L., Wu, J., Li, S., & Li, Y. (2010). A novel global harmony search algorithm for reliability problems. *Computers & Industrial Engineering, 58*(2), 307–316. https://doi.org/10.1016/J.CIE.2009.11.003