Communication

Modeling Viscosity in Starch-Polymer Suspensions: A Comparative Analysis of Swarm Algorithm-Aided ANN Optimization

Mohamed Kouider Amar *, Faiza Omari, Achouak Madani and Mohamed Hentabli

Laboratory of Biomaterials and Transport Phenomena (LBMPT), Faculty of Technology, University Yahia Fares of Medea, Medea 26000, Algeria; omari.faiza@gmail.com (F.O.); achwakmadani08@gmail.com (A.M.); hentabli.mohamed@univ-medea.dz (M.H.)

* Corresponding author. E-mail: kouideramar.mohamed@univ-medea.dz (M.K.A.)

ABSTRACT: The analysis of rheological properties of suspensions requires the use of models such as Einstein's formulation for viscosity in dilute conditions, but its effectiveness diminishes in the context of concentrated suspensions. This study investigates the rheology of suspensions containing solid particles in aqueous media thickened with starch nanoparticles (SNP). The goal is to model the viscosity of these mixtures across a range of shear rates and varying amounts of SNP and SG hollow spheres (SGHP). Artificial neural networks (ANN) combined with swarm intelligence algorithms were used for viscosity modeling, utilizing 1104 data points. Key features include SNP proportion, SGHP content, log-transformed shear rate (LogSR), and log-transformed viscosity (LogViscosity) as an output. Three swarm algorithms—AntLion Optimizer (ALO), Particle Swarm Optimizer (PSO), and Dragonfly Algorithm (DA)—were evaluated for optimizing ANN hyperparameters. The ALO algorithm proved most effective, demonstrating strong convergence, exploration, and exploitation. Comparative analysis of ANN models revealed the superior performance of ANN-ALO, with an R^2 of 0.9861, mean absolute error (MAE) of 0.1013, root mean absolute error (RMSE) of 0.1356, and mean absolute percentage error (MAPE) of 3.198%. While all models showed high predictive accuracy, the ANN-PSO model had more limitations. These findings enhance understanding of starch suspension rheology, offering potential applications in materials science.

Keywords: Starch suspensions; Rheology modeling; Starch nanoparticles; Artificial Neural Networks; Swarm intelligence algorithms

1. Introduction

Colloid science has gained significance due to the commercial success of nanotechnologies, which generate, process, and employ nanomaterials. Suspensions are materials composed of liquid continuum and solid, submicrometre particles found in various products like ink, paints, and lotions. Their distinction from true solutions depends on physical phenomena like scattering, membrane penetration, and rheological studies [1]. Colloidal suspensions can be dispersed, coagulated or flocculated, depending on the particle-particle interaction energy and particle concentration. In weakly flocculated dispersions, particles form a volume-spanning network, which can be deformed to increase the total potential energy and produce an interparticle force. A colloidal gel is a special state of strongly flocculated systems where a continuous network of particles forms by aggregation, resulting in a high viscosity [2].

Starch is a natural, renewable, and biodegradable polymer found in plant roots, stalks, seeds, and staple crops like rice, corn, wheat, tapioca, and potato. Starch's composition consists of two glucosidic macromolecules: amylose and amylopectin. Some mutant types have high amylose content and low amylose content [3]. Starch suspensions play a significant role in stabilizing emulsions and enhancing the texture and mouthfeel of food products, resulting in a smooth and creamy consistency. These suspensions exhibit distinctive properties, including high viscosity, which can be modulated by altering the concentration of starch, and the capacity to form gels upon cooling. This versatility renders them valuable in a variety of culinary applications. Furthermore, starch suspensions can enhance the mechanical properties of composite materials, thereby contributing to their strength and durability while being environmentally sustainable. The microstructural properties of starch suspensions are defined by the presence of granules that exhibit

variability in size and shape, typically ranging from 1 to 100 μm. Upon hydration, these granules undergo swelling and may experience gelatinization, resulting in the disruption of their crystalline structure and the formation of a viscous gel-like matrix. The interactions among starch molecules, including hydrogen bonding and the presence of amylose and amylopectin, significantly influence the suspension's rheological behavior, stability, and texture. These properties are critical for the functionality of starch in both food and industrial applications [4].

The rheological property of starch is crucial as it is often used as a thickener in various applications. Viscosity measures a fluid's resistance to flow when a shear rate or strain is applied. Starch suspensions, which form from starch granules, determine the thickening power of starch for stability and various applications, including pharmaceutical formulations [5]. The analysis of rheological properties required using models specifically designed to distinguish discrete proportions or fractions of particle loadings. Among these models, Einstein's formulation for viscosity in dilute suspensions of spheres, as articulated in Equation (1) [6], has served as a foundational construct. Nevertheless, the efficacy of this model diminishes significantly in the context of concentrated suspensions [7]. The limitation of Einstein's model in addressing concentrated suspensions underscores the need for more versatile and comprehensive models capable of understanding the dynamics within suspension systems. The identification and application of more sophisticated models are indispensable for understanding rheological phenomena across a spectrum of particle loadings and types.

$$\eta(\phi) = \eta_0(1 + {}^5/_2\, \phi) \tag{1}$$

where: η and η_0 are the viscosities related to the volume fraction of particles and solvent, respectively.

Metaheuristic algorithms play a vital role in addressing complex optimization problems that may be inadequately solved by traditional methods, especially in high-dimensional or non-linear contexts. These algorithms utilize mechanisms such as population-based search, randomization, and local search to effectively navigate the solution space, thereby increasing the likelihood of identifying optimal or near-optimal solutions. Their applicability extends across diverse domains, including clustering, scheduling, and optimizing machine learning models, rendering them versatile instruments for confronting real-world challenges [8]. Machine learning models are optimized by applying metaheuristic algorithms, which automate the search for optimal hyperparameters, architectural designs, and feature representations. These algorithms emulate natural phenomena to navigate the solution space, utilizing intensification and diversification strategies to circumvent local minima and identify high-quality solutions. By assigning fitness values to potential solutions and generating new candidates via reproduction operators, metaheuristics effectively address complex optimization challenges within the field of machine learning [9].

The present study builds upon the recent research conducted by G. Ghanaatpishehsanaei and R. Pal, which focused on exploring the rheology of suspensions containing solid particles in aqueous matrix liquids thickened with starch nanoparticles (SNP) [10]. Their research aimed to comprehend the impact of SNP addition on the rheological behavior of these suspensions. Their findings revealed Newtonian and non-Newtonian shear-thinning behaviors in different concentration ranges of solid particles. In continuation of this work, this study aims to model the shear viscosity of these mixtures across a diverse range of shear rates and varying amounts of SNP and SGHP (solid particles—SG hollow spheres). To achieve this, Artificial Neural Networks (ANN) hybridized with three Swarm algorithms: AntLion Optimizer (ALO), Particle Swarm Optimizer (PSO), and Dragonfly Algorithm (DA) were employed. The objective is to compare the optimizing efficacy of these algorithms in determining the optimal hyperparameters of the ANN, thereby contributing to a deeper understanding of the rheological characteristics of the suspensions and enhancing the ability to predict and control their rheological behavior.

2. Methodology

2.1. Related Work

Suspensions' properties can vary significantly based on factors such as particle size, concentration, and interactions between particles and the medium (Figure 1a,b). The viscosity of suspensions, which describes their resistance to flow, is often influenced by the concentration of solids and their interactions with the surrounding fluid. Suspensions can exhibit Newtonian and non-Newtonian behaviors (Figure 1c), depending on their composition and conditions, such as shear rate. In Newtonian fluids, viscosity remains constant regardless of shear rate, while non-Newtonian fluids, like some suspensions, show changes in viscosity under stress, often exhibiting shear-thinning or thickening behavior.

In the context of the rheology of suspensions, the methodology employed in the referenced study [10] concentrated on measuring viscosity in SNP suspensions. The authors prepared a series of SNP dispersions within an aqueous matrix,

with SNP concentrations ranging from 9.89 to 34.60 wt%. Additionally, the suspensions incorporated solid particles at concentrations varying from 0 to 47 wt%. The viscosity behavior of the suspensions exhibited notable changes with the addition of solid particles: at lower concentrations, the suspensions demonstrated Newtonian behavior, whereas at higher solid concentrations, they displayed non-Newtonian shear-thinning behavior. Furthermore, as the concentration of SNP increased, the transition from Newtonian to non-Newtonian behavior occurred at lower solid concentrations, suggesting that the interactions between SNP and the solid particles significantly influenced the flow properties of the suspension.

The preparation of the suspensions (Figure 1d) entailed the incorporation of starch nanoparticles (SNPs) into an aqueous solution that contained a surfactant, specifically Triton X-100, and a biocide to inhibit bacterial proliferation. The dispersion was subjected to homogenization to achieve uniform mixing. Subsequently, solid particles, specifically SG hollow spheres, were gradually introduced while maintaining continuous homogenization. The experimental data obtained from this study exhibited a strong correlation with the predictions of the Pal relative viscosity model (Equation (2)) [11].

$$\eta_r = \left[1 - \left\{1 + \left(\frac{1 - \phi_m}{\phi_m}\right)\sqrt{1 - \left(\frac{\phi_m - \phi}{\phi_m}\right)^2}\right\}\phi\right]^{-2.5} \tag{2}$$

where ϕ is the volume fraction of particles, and ϕ_m is the maximum packing volume fraction of particles.
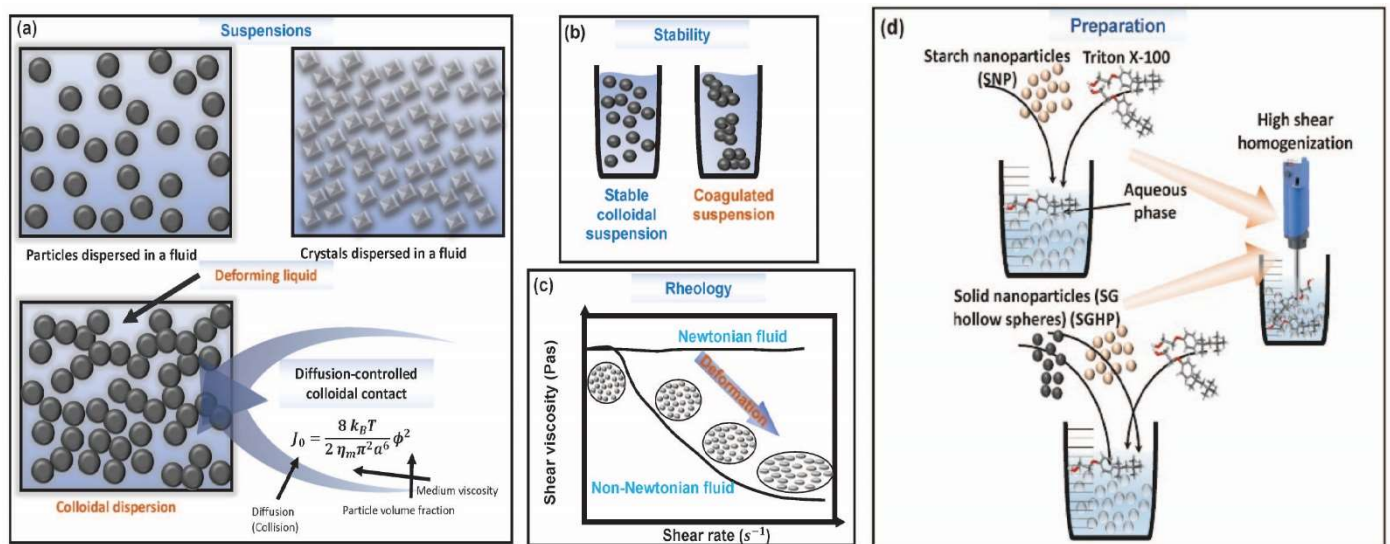

Figure 1. Panel (**a**) depicts examples of solid dispersions in fluids, such as silica particles, crystals, and colloidal suspensions. These exhibit complex structural collisions influenced by colloidal contacts, dependent on factors such as temperature, viscosity, particle size, and particle volume fraction. Panel (**b**) illustrates the coagulation of particles as an instability phenomenon. Panel (**c**) outlines different types of fluids, including Newtonian and complex non-Newtonian fluids. The structural deformation, resulting in decreased viscosity, occurs as the shear rate increases. Panel (**d**) summarizes the preparation steps of SNP and SNP/SGHP suspensions.

2.2. Data Pre-Processing and Modeling Using ANN

In order to model the viscosity of starch suspensions, a database was organized as a matrix of 1104 data points. The three input features that had the most significant impact on the viscosity of the suspensions were the proportions of starch nanoparticles (SNP), the amount of solid particles—SG hollow spheres (SGHP), and the log-transformed shear rate (LogSR). The output is the log-transformed viscosity, denoted as LogViscosity. It is important to note that this database covers a range of SNP content from 9.89% to 34.6%, while SGHP contents extend from 0% to 0.56%. The shear rate applied ranged from 0.017 s^{-1} to a maximum of 1214 s^{-1} and relative viscosities ranged from 6.53 mPas to 1,631,285.6 mPas.

The artificial neural network modeling was conducted without data scaling because inputs' scales did not impact convergence and training stability. However, the data was divided into a 70% training set and a 30% testing set, as illustrated in the flowchart in Figure 2. The flowchart also explains the steps and algorithms used to optimize the ANN models using three metaheuristic swarm algorithms: AntLion Optimizer (ALO), Particle Swarm Optimizer (PSO), and Dragonfly Algorithm (DA).
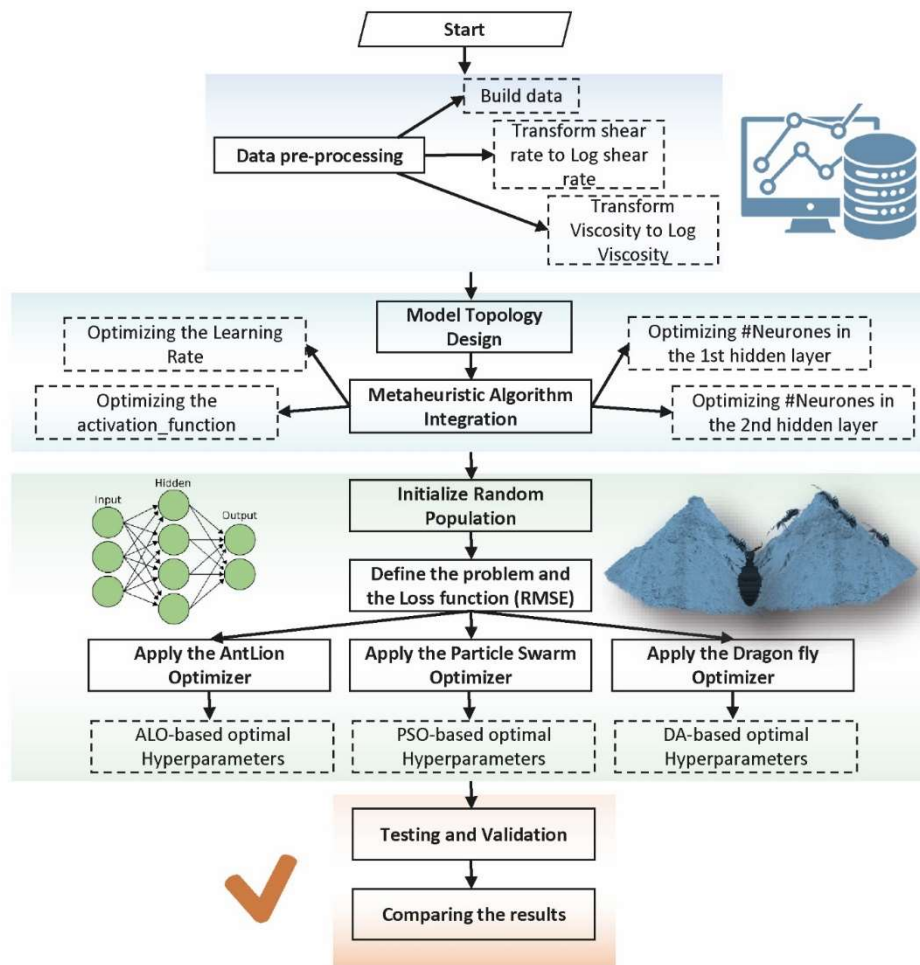
Figure 2. A general flowchart representing the viscosity modeling process, including data pre-processing, model building, hyperparameters optimization using swarm intelligence, and testing and validating the results. (Note that the symbol '#' denotes numerical values).

Artificial neural networks are structured into linear arrays called layers consisting of input, output, and hidden layers. The design of these networks involves determining the number of nodes at each layer, the number of layers in the network, and the algorithm hyperparameters. Learning is a crucial aspect of ANNs, with supervised learning aiming to adjust the weights between connections to minimize the error between the network output and the correct output. The ideal training set must represent the underlying model, as an unrepresentative set cannot produce a reliable and general model [12].

In this study, a Multi-layer Perceptron (MLP) network is employed, constituting one of the most prevalent and practical architectures within Artificial Neural Networks (ANN) [13]. The MLP structure involves interconnected neurons in each layer, as depicted in Figure 2. The learning process of the ANN relies on the weights assigned to these neurons, indicating their relative influence. An activation function is applied to transform the weighted summation of inputs. Subsequently, the output layer assimilates information from neurons in the hidden layers, essentially functioning as input layers. If multiple hidden layers are utilized, this sequential process is reiterated accordingly. The procedural representation is articulated by Equation (3):

$$y_{j,k} = F_k\left(\sum_{i=1}^{N_{k-1}} w_{ijk}\, y_{i(k-1)} + \beta_{jk}\right) \tag{3}$$

where: $y_{j,k}$ represents the output of the jth neuron in the kth layer of the MLP network. In other words, it is the activation value of a specific neuron in a specific layer. F_k is the activation function applied to the weighted summation of inputs. The subscript k indicates that different layers in the network may use different activation functions. β_{jk} is the bias term associated with the jth neuron in the kth layer. It is a constant added to the weighted sum before applying the activation function.

2.3. Metaheuristic Algorithms

2.3.1. AntLion Optimizer

The Ant Lion Optimizer (ALO) algorithm is inspired by the hunting behavior of natural antlions. Antlions are insects that trap ants in sand pits, and the ALO algorithm mimics this process to solve optimization problems. Seyedali Mirjalili proposed the algorithm in 2015 [14]. The ALO algorithm works by simulating the interactions between antlions and ants in their natural habitat. The algorithm consists of three main phases:

- Initialization: In this phase, a population of antlions and ants is randomly generated. Antlions represent the potential solutions to the optimization problem, while ants move around the antlion traps, representing the search for the optimal solution.
- Optimization: During this phase, the antlions update their positions based on the positions of the ants. If an ant finds a better position (higher fitness value), the antlion in that location will move towards the ant, mimicking the hunting behavior of antlions in nature. This process helps the algorithm explore the search space and exploit promising regions.
- Update: The best antlion obtained so far is saved as an elite solution, ensuring that the best solution found is preserved throughout the optimization process. This elitism feature helps the algorithm maintain the best solutions and avoid getting stuck in local optima.

The ALO algorithm leverages random walks, adaptive shrinking boundaries, and population-based strategies to effectively explore and exploit the search space.. By combining exploration and exploitation mechanisms inspired by the natural behavior of antlions, the ALO algorithm aims to approximate the global optimum of optimization problems.

2.3.2. Particle Swarm Optimizer

The Particle Swarm Optimization (PSO) algorithm is a nature-inspired optimization technique that was first introduced by Kennedy and Eberhart in 1995 [15]. PSO is based on the social behavior of bird flocking and fish schooling. Kennedy and Eberhart observed that birds in a flock and fish in a school exhibit collective behavior, moving towards a common goal while maintaining coordination with their neighbors. This observation led to the development of the PSO algorithm, where particles (representing solutions) in the search space adjust their positions based on their own experience and the experience of their neighbors [16]. The algorithm consists of six main phases:

- Initialization: The algorithm starts by initializing a population of particles randomly in the search space.
- Velocity and Position Update: Each particle adjusts its velocity and position based on two main components:
- Cognitive Component: The particle's memory of its best position (personal best).
- Social Component: The particle's neighborhood's best position (global best).
- Fitness Evaluation: The fitness of each particle is evaluated based on the objective function to be optimized.
- Update Personal Best and Global Best: Each particle updates its personal best position and shares information with its neighbors to update the global best position.
- Iteration: The process of velocity and position update, fitness evaluation, and best position update is iterated for a certain number of generations or until a stopping criterion is met.
- Optimal Solution: The algorithm aims to converge to an optimal solution by iteratively adjusting the positions of the particles in the search space based on their own experience and the experience of their neighbors.

2.3.3. Dragonfly Algorithm

The Dragonfly Optimizer is inspired by the swarming behavior of dragonflies. The algorithm is designed to simulate dragonflies' individual and social intelligence of, utilizing their unique characteristics for optimization purposes [17]. The DA algorithm is based on the Differential Evolution (DE) algorithm and incorporates a mechanism to manage an archive of solutions. It updates the position of search agents in a manner similar to DE, but food sources are selected from the archive. The algorithm aims to find a well-spread Pareto optimal front for multi-objective optimization problems, encouraging the artificial dragonflies to explore less populated regions of the solution space. It also includes a mechanism to manage the archive and select enemies from the archive based on their population. The Dragonfly Optimizer algorithm involves the following steps:

- Initialization: Initialize the population of dragonflies and set the algorithm parameters.
- Fitness Evaluation: Evaluate the fitness of each dragonfly in the population based on the objective function(s) of the optimization problem.
- Update Position: Update the position of each dragonfly based on its current position, the positions of neighboring dragonflies, and the best position found so far.
- Update Archive: Maintain an archive of the best solutions found (Pareto optimal set) during optimization.
- Selection of Food Sources: Choose food sources (solutions) for dragonflies from the least populated region of the obtained Pareto optimal front to improve the distribution of the solutions.
- Selection of Enemies: Select enemies (non-promising crowded areas) for the dragonflies from the most populated region of the Pareto optimal front to discourage exploration in those areas.
- Archive Management: Implement a mechanism to manage the archive, ensuring it is updated regularly and does not become full.
- Convergence and Coverage: Aim for convergence by determining accurate approximations of Pareto optimal solutions and ensure coverage by distributing the obtained solutions uniformly across all objectives.

These steps enable the Dragonfly Optimizer algorithm to efficiently explore the solution space and find high-quality solutions.

2.4. Optimizing Hyperparameters Using Swarm Intelligence

The first step in any machine learning task is loading and preparing the dataset. In this study, an excel file containing the required dataset is loaded into a Pandas DataFrame within a Python environment. The target variable, 'LogViscosity', and input features, 'SNP', 'SGHP', and 'LogSR' are defined, and the dataset is split into training and validation sets as previously detailed.

The fitness function is a crucial element in metaheuristic optimization. In this study, the fitness function evaluates the performance of an ANN with specified hyperparameters by utilizing root mean squared error (RMSE) on the validation dataset. The hyperparameters include the number of neurons in two hidden layers, the learning rate, and the activation function. The hyperparameter ranges are defined, specifying lower and upper bounds for the number of neurons in each layer from 5 to 50 neurons, learning rate from 10^{-5} to 10^{0}, batch size fixed at 100, and the choice of activation function between the most common activation functions in regression tasks, i.e., 'tanh' and 'relu' activation functions. Tuning these ranges can significantly impact the optimization process. After optimization, the best hyperparameters are extracted, and a new ANN model is trained using these parameters. The model's performance is then evaluated on the validation set using various metrics such as the coefficient of determination (R^2 score), mean absolute error (MAE), root mean squared error (RMSE), and mean absolute percentage error (MAPE). These metrics are mathematically described in Equations (4)–(7), respectively. Finally, the optimized model is evaluated on the entire dataset. The same set of metrics is computed to provide an overview of the model's performance across the entire data distribution.

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(\eta_i - \hat{\eta}_i)^2}{\sum_{i=1}^{N}(\eta_i - \bar{\eta}_i)^2} \tag{4}$$

$$MAE = \frac{\sum_{i=1}^{N}|\eta_i - \hat{\eta}_i|}{N} \tag{5}$$

$$RMSE = \sqrt{\sum_{i=1}^{N}\frac{(\eta_i - \hat{\eta}_i)^2}{N}} \tag{6}$$

$$MAPE = \left(\frac{1}{N}\sum_{i=1}^{N}\left|\frac{\eta_i - \hat{\eta}_i}{\eta_i}\right|\right) \times 100 \tag{7}$$

where η_i represent the actual viscosity in the dataset, and $\hat{\eta}_i$ represents the predicted from the model, while $\bar{\eta}_i$ is the mean of the observed viscosities in the dataset, and N denotes the number of data points.

3. Results and Discussion

3.1. Data Representation and Pre-Processing

A database of 1104 data points was used to model the viscosity of starch suspensions, incorporating proportions of starch nanoparticles, SGHP content, and log-transformed shear rate. The dataset covered a diverse range of viscosities.

Figure 3a shows the distribution of SNPs across six classes evenly dispersed throughout the dataset. Similarly, Figure 3b illustrates the distribution of SGHP across multiple ranges, indicating a well-distributed presence in the dataset. However, Figure 3c,e depict challenges in the distribution of shear rate and viscosity. Both exhibit high intensities at lower values, prompting the application of a log transformation to achieve a normal distribution, as demonstrated in Figure 3d,f. This transformation ensures a more suitable representation of their values. As a result, Figure 3g presents the final input and output features in a scaled violin representation. The dataset appears to be uniformly distributed, showing nearly equal ranges and means, which offers a comprehensive overview of the input-output relationships.
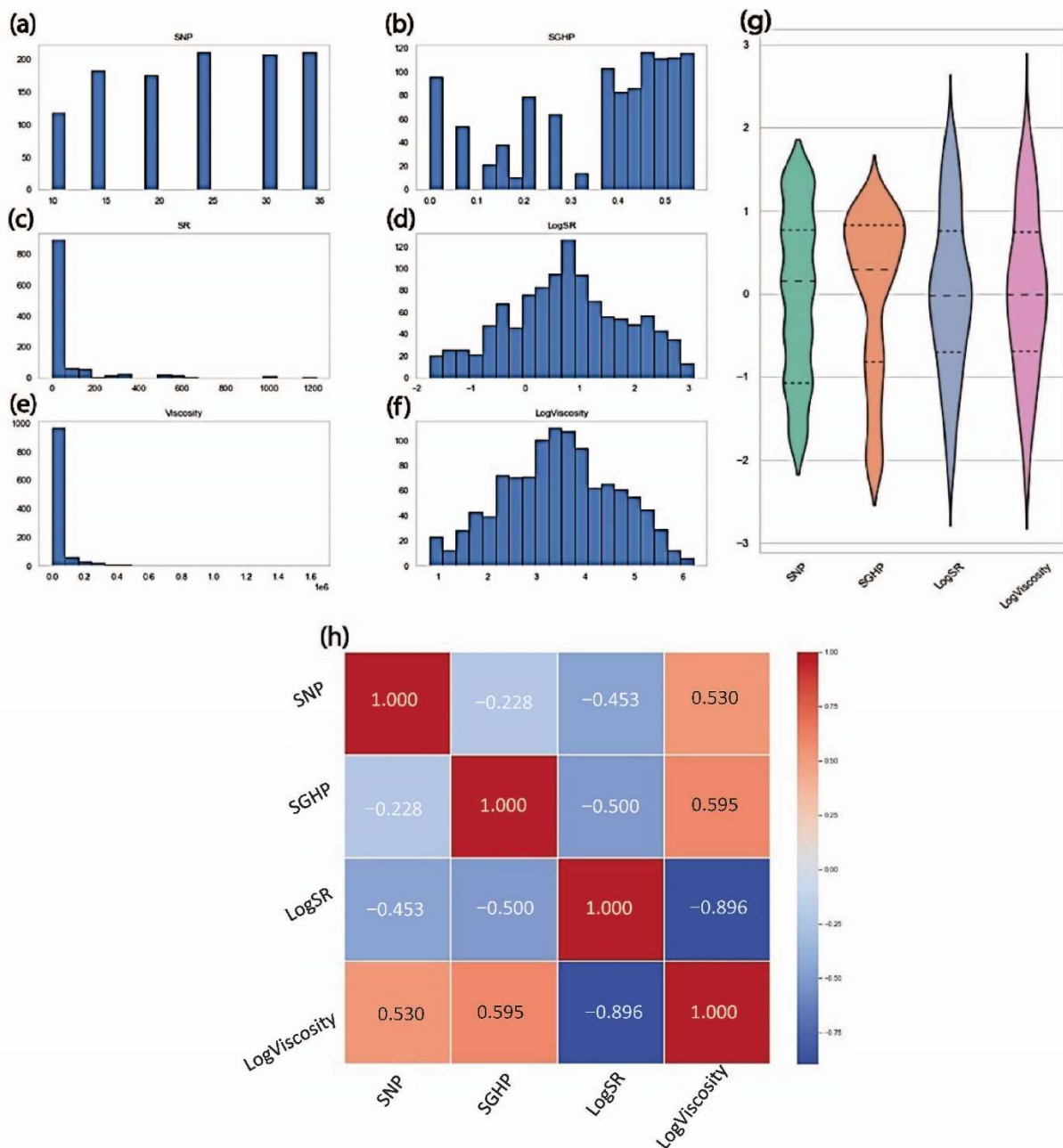
Figure 3. An overview of the dataset, showing distributions of starch nanoparticles (SNP) and solid particles of hollow spheres type (SGHP) in panels (**a**,**b**), shear rate and log-transformed shear rate in panels (**c**,**d**), viscosity and log-transformed viscosity in panels (**e**,**f**). Additional insights are provided by a violin plot of input feature distribution in panel (**g**) and a correlation heatmap in panel (**h**).

The collinearity among features was explored using a heatmap (Figure 3h). Notably, a strong negative collinearity of −0.896 was observed between LogSR and LogViscosity. This correlation plays a pivotal role in influencing the rheological behavior of starch suspensions, characterized by a non-Newtonian shear-thinning behavior. This collinearity suggests that viscosity decreases at a non-Newtonian rate as the shear rate increases, which is crucial for accurately predicting and understanding the flow behavior of these suspensions. Additionally, the suspensions exhibit Newtonian behavior, with viscosity having a relationship with SGHP of 0.595 and SNP of 0.530, indicating their distinct impact on the rheological properties.

3.2. Evaluation of Swarm Algorithms

In this experimental section, three swarm algorithms (ALO, PSO, and DA) are compared to determine their effectiveness in searching for optimal hyperparameters for developing the most robust ANN model capable of accurately modeling the viscosities of suspensions. Evaluation criteria were employed to assess the performance of these algorithms: monitoring their trajectories, exploration and exploitation of the search space, and achieving the lowest fitness function (RMSE).

Tracking the trajectory of particles over iterations helps assess whether the algorithm is converging to a solution. If particles converge toward a specific region in the search space, it indicates that the algorithm is homing in on a potential optimum. Figure 4 represents the trajectory of several agents (solutions) for the three algorithms searching for the optimal hyperparameters of four ANNs. As a result, the ALO algorithm exhibits a more uniform trajectory after nearly 20 iterations compared to the PSO or DA algorithms.

In addition, the spread of ALO agents shows better performance in covering the entire search space (hyperparameters lower and upper bounds), indicating better diversity and well-balanced solutions explored by the algorithm for finding a global optimum rather than getting stuck in local optima. The trajectory of ALO agents reveals that the algorithm is converging the search space faster. A faster convergence might indicate efficiency, but ensuring that the algorithm explores a sufficient portion of the search space is essential to avoid premature convergence.

The trajectories of the algorithms show the final hyperparameters achieved after 100 iterations [18]. For ALO, PSO, and DA, the number of neurons in the first and second hidden layers was determined as 23:7, 46:37, and 28:21, respectively. The learning rates were determined as 0.0079, 0.0055, and 0.0052 for the ALO, PSO, and DA algorithms. Additionally, the 'tanh' activation function was found to be optimal for all three swarm algorithms.
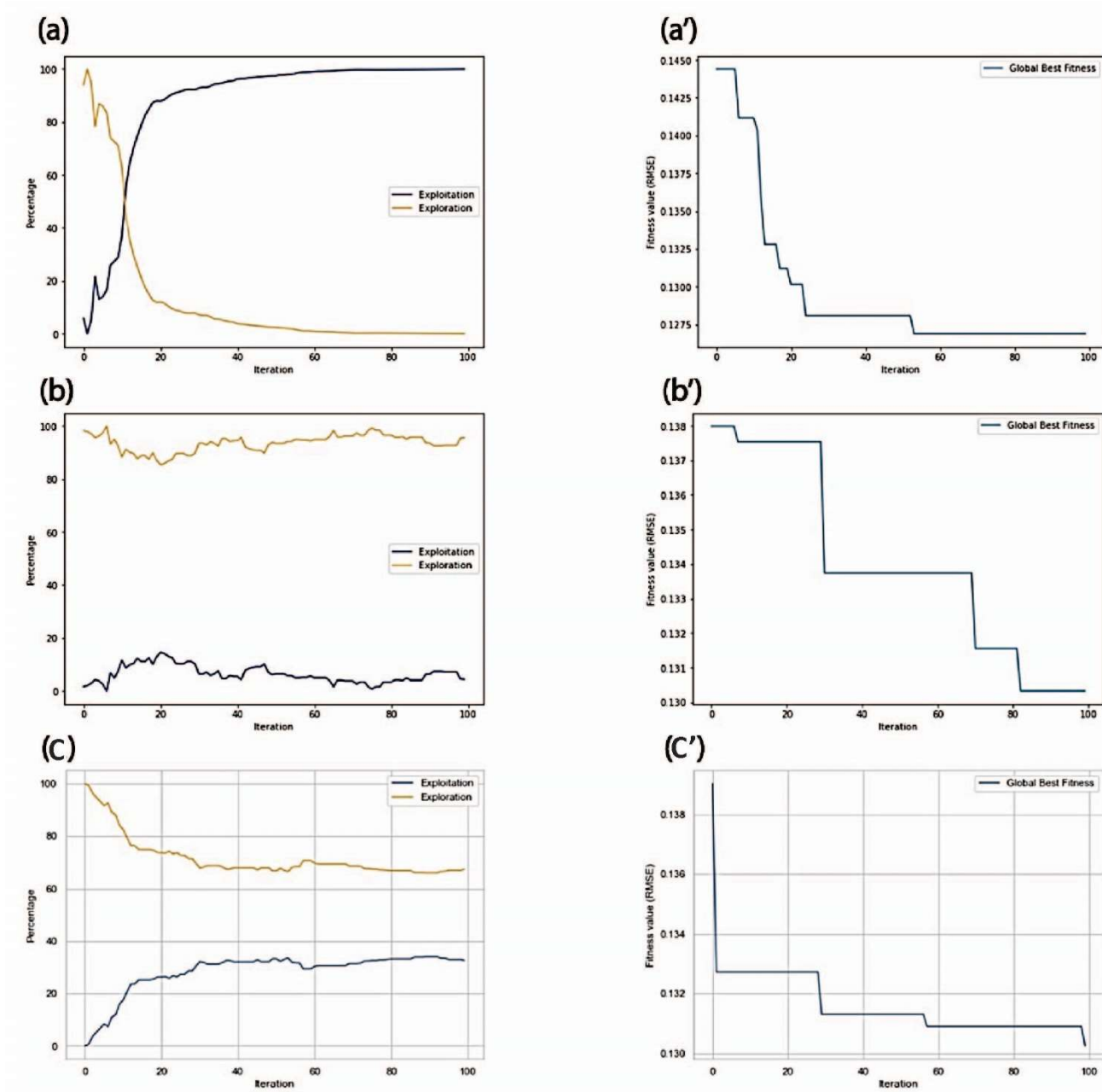
In addition, the Exploration vs. Exploitation graphs in Figure 5a–c represent how an optimization algorithm balances exploration (searching for new solutions) and exploitation (exploiting known solutions) over iterations. This graph provides insights into the algorithm's behavior and ability to navigate the search space efficiently. The graph helps visualize the trade-off between exploration and exploitation. The ALO algorithm appears to balance exploration and exploitation effectively, indicating its ability to avoid local optima while also exploiting promising regions. On the other hand, the PSO and DA algorithms appear to be trapped in local optima due to an imbalance between exploration and exploitation. This imbalance is particularly noticeable in the PSO algorithm, where exploration remains nearly linear at approximately 100%, while exploitation consistently remains low, fluctuating between 0% and 20%. Similarly, the utilization of solutions from the DA algorithm does not exceed 40%, while exploration does not surpass 60%, underscoring the algorithm's difficulties in traversing various regions of the search space.

Furthermore, Figure 5a'–c' represents the relative fitness value (RMSE) over iterations in the optimization algorithms. This can provide crucial information about the algorithms' progress and the quality of the solutions they are exploring when tuning hyperparameters. A consistently decreasing RMSE in all algorithms suggests that the optimization process effectively enhances the models' performance. However, it is important to consider whether the algorithm converges to a global minimum or gets stuck in a local minimum. The ALO algorithm, when compared to the PSO and DA algorithms, demonstrates greater convergence as the RMSE decreases and stabilizes after approximately 60 iterations. Additionally, the ALO algorithm reduces RMSE values more effectively than the other algorithms, indicating superior hyperparameter selection.
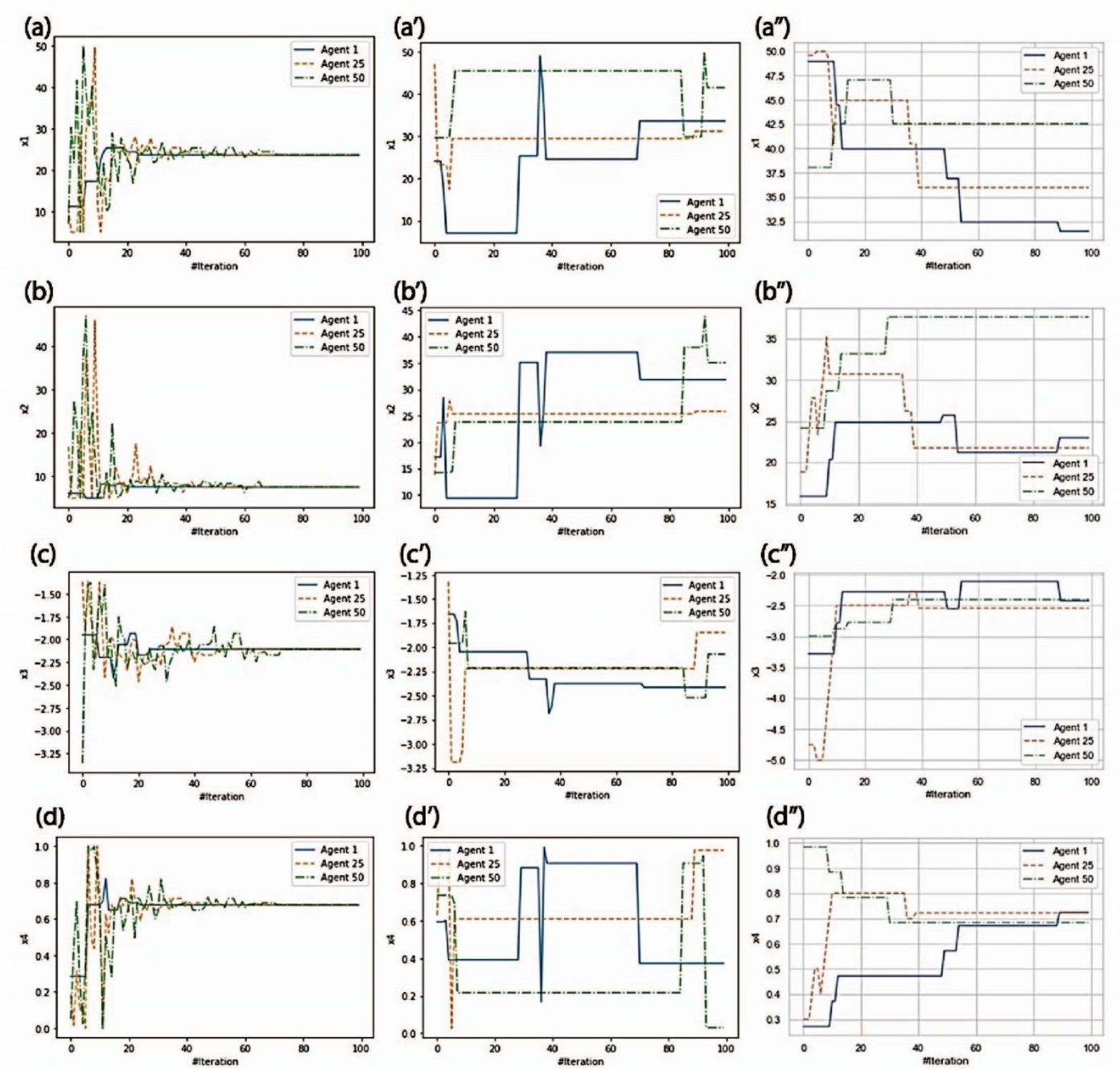
Figure 4. (**a–d**) The trajectory of three randomly selected agents (solutions) from the ALO algorithm across four crucial hyperparameters of the ANN model: number of neurons in the first hidden layer, number of neurons in the second hidden layer, learning rate, and activation function, respectively. Correspondingly, (**a'–d',a"–d"**) depict the trajectories of agents from PSO and DA, respectively, navigating the same set of hyperparameters. (Note that the symbol '#' denotes number of iterations).

Figure 5. Exploration vs. Exploitation graphs for (**a**,**b**,**c**) corresponding to the ALO, PSO, and DA algorithms, respectively. The convergence of algorithms is represented by the RMSE as a fitness function (**a'**,**b'**,**c'**) corresponding to the ALO, PSO, and DA algorithms, respectively.

3.3. Regression Results

Figure 5 provides an exploration of the model predictability through regression analysis. These plots visually convey the relationship between the predicted and actual Log viscosities, offering insights into the models' ability to capture underlying patterns and make accurate predictions. The left subplots demonstrate how well the models perform on the specific data used for testing. In contrast, the right subplots provide a broader perspective by showcasing their performance across the entire dataset. The regression plots reveal a good alignment, suggesting a robust fit of the developed models. This alignment indicates a strong correspondence between the models' predictions and the true outcomes, signifying a good fit to the underlying data. The proximity of the data points to the regression lines in both sets of plots suggests that the models capture the essential patterns within the datasets. Such uniformity across various datasets underscores the reliability and generalizability of the models, reinforcing their predictive efficacy.

In addition, in Figure 6, a comparative performance analysis of three prominent Artificial Neural Network variants—ANN-ALO, ANN-PSO, and ANN-DA—is presented. This study evaluated the performance of three algorithms—ANN-ALO, ANN-PSO, and ANN-DA—on a given dataset. The evaluation used four key metrics: R^2

Score, MAE, RMSE, and MAPE. The goal was to analyze the algorithms' effectiveness in predicting the transformed viscosity (Log viscosity) and determine the most robust model.
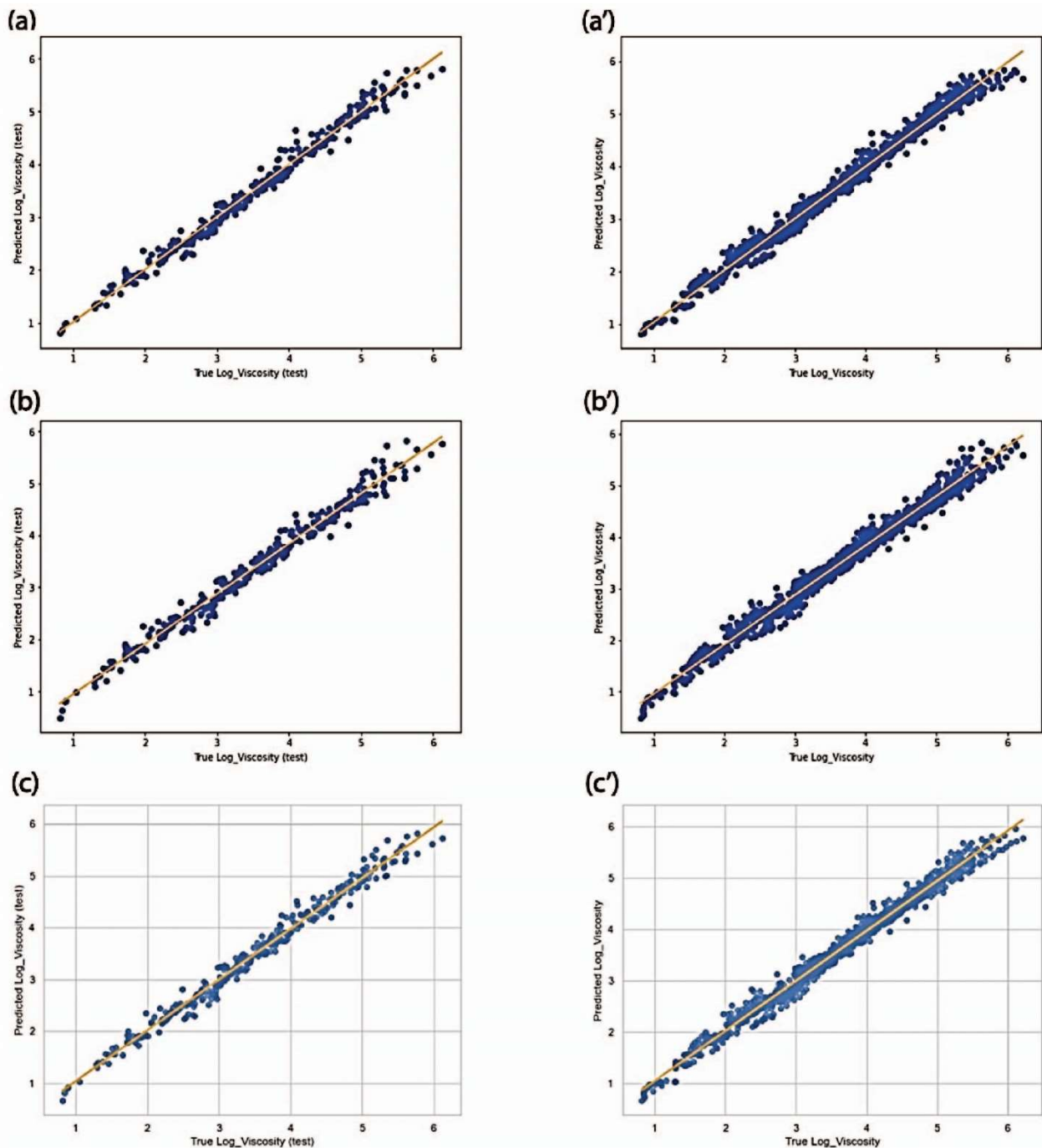

Figure 6. Exploring model predictability with regression analysis. Panels (**a–c**) present regression plots illustrating the predictive performance of three developed models—ANN-ALO, ANN-PSO, and ANN-DA. Each panel displays a pair of plots, with the left subplots (**a–c**) showcasing the regression for test datasets and the right subplots (**a'–c'**) representing the regression for entire datasets.

The R^2 Score measures the proportion of the variance in the dependent variable that is predictable from the independent variables. It ranges from 0 to 1, where a higher score indicates a better fit. On the other hand, the MAE is the average absolute difference between predicted and actual values. It provides a measure of the model's accuracy without considering the direction of the error. Lower MAE values indicate better prediction accuracy. Similarly, RMSE is the square root of the average squared differences between predicted and actual values. It penalizes larger errors more than MAE. Like MAE, lower RMSE values indicate better prediction accuracy. Furthermore, MAPE calculates the percentage difference between predicted and actual values, averaged over all observations. As an error indicator, lower MAPE values suggest a smaller percentage of prediction error.

The results reveal that ANN-ALO consistently outperforms the other models across all evaluated metrics, boasting a higher coefficient of determination of 0.9861, the lowest MAE at 0.1014, and the smallest RMSE of 0.1356. These values reflect its superior accuracy in predicting the target variable, showcasing robust predictive capabilities. The ANN-DA model also exhibits competitive performance, closely trailing ANN-ALO with an R^2 Score of 0.9856 and MAE of 0.1046. In contrast, ANN-PSO, while showing acceptable results, lags in terms of MAE (0.1713), RMSE (0.2090), and MAPE (5.2337%), indicating a comparatively higher margin of error in predictions. Based on these metrics' evaluation, ANN-ALO emerges as the most robust model. Its consistent and superior performance suggests its suitability for predicting the viscosity.

In Figure 7e, the comparison between the original and predicted viscosity, facilitated by Artificial Neural Network (ANN) models optimized through three distinct swarm algorithms (ALO, PSO, and DA), is presented. Notably, the predictions demonstrate a remarkable degree of overlap across all models, indicating the effectiveness of the models in capturing the underlying patterns of the viscosity data. However, a nuanced observation emerges when scrutinizing the predictions at very low shear rates, where the viscosity experiences a sharp increase. At these critical points, the models are susceptible to errors, which can lead to potential discrepancies, especially in predicting extremely high viscosities.
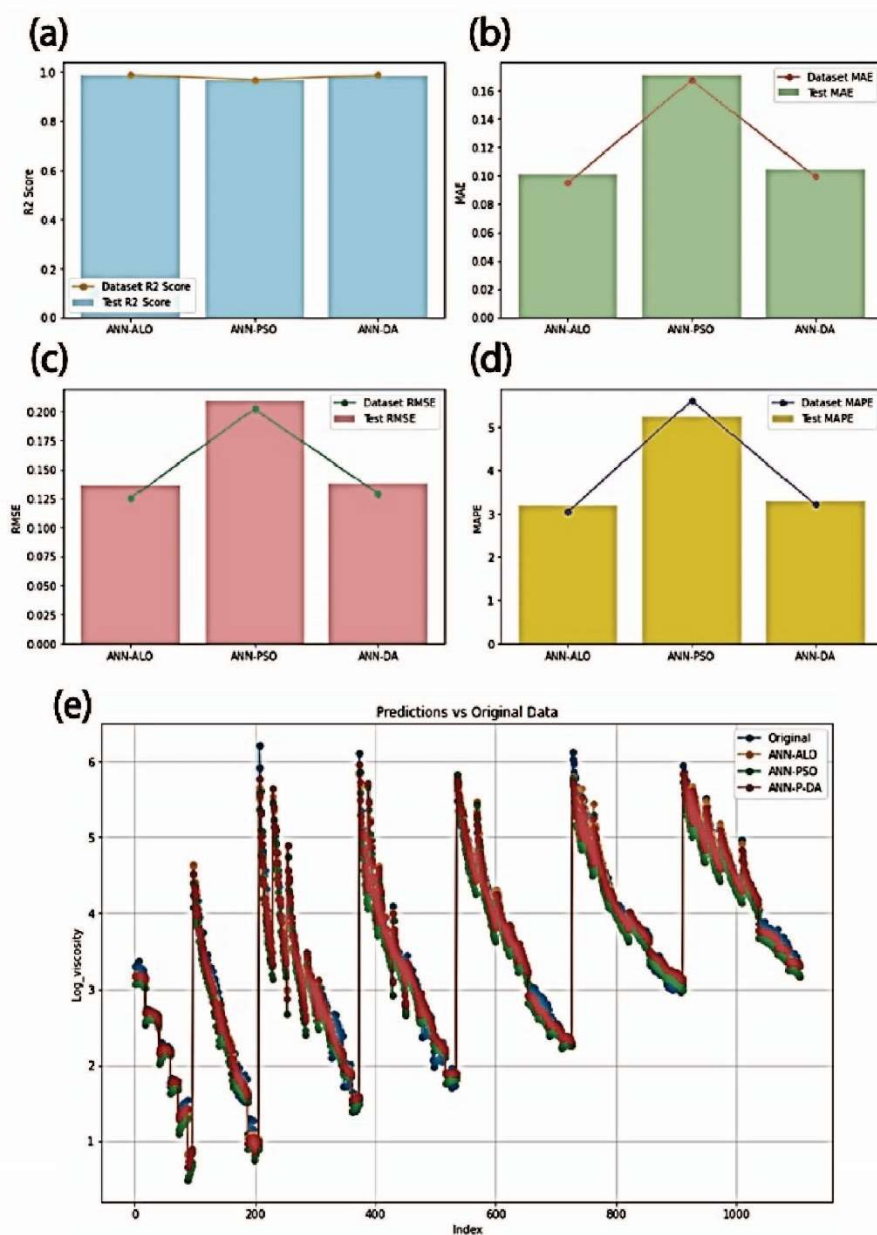
Figure 7. Comparative performance analysis of ANN variants. Panels (**a–d**) illustrate the evaluation of three prominent Artificial Neural Network variants—ANN-ALO, ANN-PSO, and ANN-DA—across key performance metrics. Each panel showcases the (**a**) R^2 scores, (**b**) MAE, (**c**) RMSE, and (**d**) MAPE. Panel (**e**) illustrates the superimposition of models' predictions against the original log viscosities.

It becomes evident that the ANN-PSO model stands out due to a more pronounced deviation from the original data. This discrepancy aligns with the model's comparatively lower coefficient of determination (R^2) and higher error metrics compared to the ANN-ALO or ANN-DA models. The divergence observed in the ANN-PSO model highlights the algorithm's limitations in accurately capturing the complexities of viscosity behavior, particularly under extreme shear conditions.

3.4. Discussion and Comparative Analysis

The application of optimization algorithms in machine learning (ML) modeling has gained significant traction, particularly in refining the predictive accuracy of artificial neural networks (ANNs) in complex datasets. In the context of viscosity modeling for suspensions, numerous works have explored the utility of different optimization algorithms to improve model performance. To place our study in context, we present a comparison between our results and relevant studies, emphasizing the role of optimization algorithms in achieving high predictive accuracy (Table 1).

Table 1. Comparison of machine learning models and optimization algorithms for viscosity prediction in various materials.

Material	ML Modeling	Optimization Algorithm	R^2	Ref.
Liquid paraffin-Fe_3O_4 nanofluids.	ANN for viscosity modeling.	Group Method of Data Handling (GMDH).	0.96	[19]
Suspensions containing microencapsulated phase change materials (MPCMs).	Gaussian process regression (GPR) for dynamic viscosity modeling.	Genetic algorithm (GA), particle swarm optimization (PSO), and marine predators algorithm (MPA).	0.998	[20]
Polydisperse SiO_2 nanoparticles in a glycerol and water mixture.	Gaussian Process Regression for viscosity modeling.	Bayesian optimization.	0.99	[21]
TiO2/SAE 50 nano-lubricant.	ANN for viscosity modeling.	Genetic Algorithm-Radial Basis Function.	>0.99	[22]
Honey	ANN for viscosity modeling.	Genetic Algorithm.	0.997	[23]
Starch-nanosuspensions (current study)	ANN for viscosity modeling.	Ant lion optimizer.	0.986	

In the context of liquid paraffin-Fe_3O_4 nanofluids, an ANN model was optimized utilizing the Group Method of Data Handling (GMDH), resulting in a high R^2 of 0.96. Similarly, for suspensions containing microencapsulated phase change materials (MPCMs), Gaussian process regression (GPR) models were optimized through various algorithms, including the Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and Marine Predators Algorithm (MPA), achieving an even higher R^2 of 0.998. Additionally, in the viscosity modeling of polydisperse SiO_2 nanoparticles, Bayesian optimization produced a model with an R^2 of 0.99.

In comparison, our study demonstrates that the Ant Lion Optimizer (ALO), when utilized in conjunction with artificial neural networks (ANN) for modeling starch-nanosuspensions, achieves a coefficient of determination (R^2) of 0.986, which indicates a strong predictive capability. This R^2 value is comparable to that of other machine learning models optimized through metaheuristic algorithms, such as the ANN-GA model, which attained an R^2 of 0.997 in the context of honey viscosity modeling. Although our ANN-ALO model does not exceed all previously reported results, it exhibits robust performance, specifically in the domain of starch-based suspensions, highlighting a distinct advantage in effectively balancing exploration and exploitation within the search space during hyperparameter optimization.

A significant finding from our study is the superior performance of ALO in comparison to PSO and Dragonfly Algorithm (DA), as evidenced by the lower RMSE and higher R^2 achieved by the ANN-ALO model. This outcome emphasizes the efficacy of ALO in identifying optimal hyperparameters, as it effectively balances exploration and exploitation, thereby minimizing the likelihood of convergence to local optima. Consistent with previous research, such as the optimization of TiO2/SAE 50 nano-lubricants using Genetic Algorithm-Radial Basis Function ($R^2 > 0.99$), our study underscores the importance of selecting an appropriate optimization algorithm to enhance model accuracy in non-linear systems, such as viscosity.

4. Conclusions

In conclusion, this study explored the complex domain of modeling the viscosity of starch suspensions by utilizing a combination of artificial neural networks (ANN) and swarm intelligence algorithms. The comprehensive examination of a dataset comprising 1104 data points enabled a nuanced understanding of the rheological behavior of these

suspensions. Key features such as starch nanoparticles (SNP) proportions, solid particles of hollow spheres (SGHP) content, and log-transformed shear rate (LogSR) were identified as crucial determinants in predicting viscosity.

The investigation involved exploring three swarm algorithms—AntLion Optimizer (ALO), Particle Swarm Optimizer (PSO), and Dragonfly Algorithm (DA)—for hyperparameter optimization in developing ANN models. Through trajectory analysis, exploration versus exploitation graphs, and relative fitness value (RMSE) comparisons, the ALO algorithm emerged as the most efficient in balancing exploration and exploitation, showcasing faster convergence and covering the entire search space.

Regression analysis and a comparative performance evaluation of ANN variants—ANN-ALO, ANN-PSO, and ANN-DA—underscored the superior predictive capabilities of ANN-ALO. The model consistently outperformed its counterparts in terms of R^2 score, MAE, RMSE, and MAPE, reflecting its robustness and accuracy in predicting transformed viscosity (Log viscosity). While all models demonstrated a high degree of alignment between predicted and actual viscosities, the ANN-PSO model exhibited limitations, especially in predicting extremely high viscosities under extreme shear conditions.

The findings contribute to the understanding of starch suspension rheology but also highlight the efficacy of the integrated approach employing ANN and ALO in predicting viscosity. The identified correlations between input features and viscosity, and the strengths and limitations of the swarm algorithms, offer valuable insights for future studies in this domain. In essence, the combination of artificial intelligence and swarm intelligence offers a promising approach to understanding the complexities of colloidal suspensions, thereby contributing to the wider field of materials science and industrial applications.

Author Contributions

Conceptualization, M.K.A. and F.O.; Methodology, M.K.A. and F.O. and A.M.; Software, M.K.A.; Validation, M.K.A. and M.H.; Formal Analysis, M.K.A.; Investigation, M.K.A. and F.O. and A.M. and M.H.; Writing—Original Draft Preparation, M.K.A.; Writing—Review & Editing, M.K.A.

Ethics Statement

Not applicable.

Informed Consent Statement

Not applicable.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this article.

References

1. Babick F. *Suspensions of Colloidal Particles and Aggregates*, 20th ed.; Springer: Berlin, Germany, 2016.
2. Genovese DB, Lozano JE, Rao MA. The Rheology of Colloidal and Noncolloidal Food Dispersions. *J. Food Sci.* **2007**, *72*, R11–R20.
3. Le Corre D, Bras J, Dufresne A. Starch Nanoparticles: A Review. *Biomacromolecules* **2010**, *11*, 1139–1153.
4. Fazeli M. Development of hydrophobic thermoplastic starch composites. Unpublished work, 2018.
5. Ai Y, Jane JL. Gelatinization and rheological properties of starch. *Starch—Stärke* **2015**, *67*, 213–224.
6. Einstein A. *Investigations on the Theory of Brownian Movement*; Dover: New York, NY, USA, 1975; p. 591.
7. Mendoza CI, Santamaría-Holek I. The rheology of hard sphere suspensions at arbitrary volume fractions: An improved differential viscosity model. *J. Chem. Phys.* **2009**, *130*, 44904.

8. Nanda SJ, Panda G. A survey on nature inspired metaheuristic algorithms for partitional clustering. *Swarm Evol. Comput.* **2014**, *16*, 1–18.

9. Akay B, Karaboga D, Akay R. A comprehensive survey on optimizing deep learning models by metaheuristics. *Artif. Intell. Rev.* **2022**, *55*, 829–894.

10. Ghanaatpishehsanaei G, Pal R. Rheology of Suspensions of Solid Particles in Liquids Thickened by Starch Nanoparticles. *Colloids Interfaces* **2023**, *7*, 52.

11. Pal R. A new model for the viscosity of asphaltene solutions. *Can. J. Chem. Eng.* **2015**, *93*, 747–755.

12. Zou J, Han Y, So SS. Overview of Artificial Neural Networks. In *Artificial Neural Networks: Methods and Applications*, 1st ed.; Livingstone DJ, Ed.; Humana Press: Totowa, NJ, USA, 2009; pp. 1–30.

13. Heidari E, Sobati MA, Movahedirad S. Accurate prediction of nanofluid viscosity using a multilayer perceptron artificial neural network (MLP-ANN). *Chemom. Intell. Lab. Syst.* **2016**, *155*, 73–85.

14. Mirjalili S. The Ant Lion Optimizer. *Adv. Eng. Softw.* **2015**, *83*, 80–98.

15. Eberhart R, Kennedy J. A new optimizer using particle swarm theory. In Proceedings of the MHS'95 Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, 4–6 October 1995; pp. 39–43.

16. Wang D, Tan D, Liu L. Particle swarm optimization algorithm: An overview. *Soft Comput.* **2018**, *22*, 387–408.

17. Mirjalili S. Dragonfly algorithm: A new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. *Neural Comput. Appl.* **2016**, *27*, 1053–1073.

18. Arbia W, Kouider AM, Adour L, Amrane A. Maximizing chitin and chitosan recovery yields from fusarium verticillioides using a many-factors-at-a-time approach. *Int. J. Biol. Macromol.* **2024**, 136708. doi:10.1016/j.ijbiomac.2024.136708.

19. Shahsavar A, Khanmohammadi S, Karimipour A, Goodarzi M. A novel comprehensive experimental study concerned synthesizes and prepare liquid paraffin-Fe_3O_4 mixture to develop models for both thermal conductivity & viscosity: A new approach of GMDH type of neural network. *Int. J. Heat Mass Transf.* **2019**, *131*, 432–441.

20. Hai T, Basem A, Alizadeh A, Sharma K, Jasim DJ, Rajab H, et al. Optimizing Gaussian process regression (GPR) hyperparameters with three metaheuristic algorithms for viscosity prediction of suspensions containing microencapsulated PCMs. *Sci. Rep.* **2024**, *14*, 20271.

21. Sharma KV, Talpa Sai PHVS, Sharma P, Kanti PK, Bhramara P, Akilu S. Prognostic modeling of polydisperse SiO_2/Aqueous glycerol nanofluids' thermophysical profile using an explainable artificial intelligence (XAI) approach. *Eng. Appl. Artif. Intell.* **2023**, *126*, 106967.

22. Hemmat Esfe M, Tatar A, Ahangar MRH, Rostamian H. A comparison of performance of several artificial intelligence methods for predicting the dynamic viscosity of TiO_2/SAE 50 nano-lubricant. *Phys. E Low-Dimens. Syst. Nanostruct.* **2018**, *96*, 85–93.

23. Ramzi M, Kashaninejad M, Salehi F, Sadeghi Mahoonak AR, Ali Razavi SM. Modeling of rheological behavior of honey using genetic algorithm–artificial neural network and adaptive neuro-fuzzy inference system. *Food Biosci.* **2015**, *9*, 60–67.