POLITECNICO DI TORINO
Repository ISTITUZIONALE

Interpretable machine-learning enhanced parametrization methodology for Pluronics–water mixtures in DPD simulations

Original
Interpretable machine-learning enhanced parametrization methodology for Pluronics–water mixtures in DPD simulations / Lauriello, N., Ponnana, D.N., Ma, Z., Šindelka, K., Buffo, A., Boccardo, G., Marchisio, D., Pan, W.. - In: SOFT MATTER. - ISSN 1744-683X. - 21:29(2025), pp. 5833-5851. [10.1039/d5sm00291e]

Availability:
This version is available at: 11583/3005671 since: 2025-12-05T15:19:09Z

Publisher:
Royal Society of Chemistry

Published
DOI:10.1039/d5sm00291e

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

19 November 2026

Soft Matter

ARTICLE TYPE

Cite this: DOI: 00.0000/xxxxxxxxxx

Interpretable machine-learning enhanced parametrization methodology for Pluronics-Water Mixtures in DPD simulations†

Nunzia Lauriello,[a] Deekshith Naidu Ponnana,[b] Zhan Ma,[b] Karel Šindelka,[c] Antonio Buffo,[a] Gianluca Boccardo,[a] Daniele Marchisio[a] and Wenxiao Pan*[b]

Received Date
Accepted Date

DOI: 00.0000/xxxxxxxxxx

Dissipative Particle Dynamics (DPD) is an incredibly powerful tool for simulating the behavior of structured fluids. However, identifying the appropriate model parameters to accurately replicate physical properties remains a challenge. This study showcases the benefits of integrating machine learning techniques into the top-down parameterization of Pluronic systems. The proposed workflow outlines a data-driven approach to accurately determine model parameters tailored to various Pluronic systems. Gaussian Process Regression (GPR)-based surrogate models effectively replicate the results of DPD simulations, delivering faster responses that streamline parameter optimization and enable the calibration of Pluronic systems against experimental data. Although DPD simulations provide valuable insight, their high computational cost, due to extensive simulations and post-processing, presents a challenge. The GPR-based surrogate model addresses this by modeling the relationships between input parameters and output properties. SHAP (SHapley Additive exPlanations) analysis enhances model interpretability, providing deeper insights into the relationships and causal mechanisms between the input parameters and the predicted properties. The combination of GPR and SHAP analysis provides an interpretable machine learning approach, enabling a more efficient optimization process and reducing the need for exhaustive simulations. This work lays a foundation for generalizing the parameterization process across Pluronic systems and conditions, such as varying temperatures, by incorporating additional DPD model input parameters.

1 Introduction

Pluronics, or poloxamers, are synthetic amphiphilic copolymers with a triblock structure: a central hydrophobic poly(propylene oxide) (PPO) block flanked by two hydrophilic poly(ethylene oxide) (PEO) chains[1]. Available in various types, they are highly flexible systems capable of meeting the specific requirements of a wide range of applications. Their versatility, biocompatibility, low toxicity, temperature sensitivity, and self-assembly capabilities make them ideal in numerous fields. These include drug delivery, tissue engineering, bioprinting, and nanomedicine, along with uses in dispersion stabilization, emulsification, lubrication, formulation, and surface modification to improve biocompatibility in medical applications[2,3]. They are also commonly applied in the cosmetics industry. The wide range of commercially available Pluronics is due to the flexibility in adjusting the lengths of the PPO and PEO chains. This allows for the customization of their properties and overall molecular weight. They exhibit different behavior and phase diagrams in solution. They can exist in the form of liquids, solids, and pastes. Their molecular weight ranges from 2000 to 20000 g / mol, with a PEO content of 10 to 80 wt. %. These materials are named using an alphanumeric code that combines a letter and two or three digits[4]. The letter represents the physical state at room temperature: L for liquids, P for pastes, and F for flakes. The digits denote the molecular weight and composition. Specifically, the molecular weight of the hydrophobic PO block is calculated by multiplying the first one or two digits by 300, while the hydrophilic EO content (in weight percentage) is determined by multiplying the final digit by 10. Different Pluronics are cataloged according to the molecular weight of the PPO and the PEO contents in the Pluronic grid, reported in Fig. 1. Pluronics are often referred to as $EOxPOyEO_x$, where x and y represent the number of EO and PO units as shown in Fig. 2, respectively. The degree of polymerization of the Pluronic hydrophilic

[a] DISAT – Institute of Chemical Engineering, Politecnico di Torino, C.so Duca degli Abruzzi 24, Turin, Italy.
[b] Department of Mechanical Engineering, University of Wisconsin-Madison, Madison, WI 53706, USA. E-mail: wpan9@wisc.edu
[c] Department of Molecular and Mesoscopic Modelling, The Czech Academy of Sciences, Institute of Chemical Process Fundamentals, Rozvojová 135/1, Prague, Czech Republic.

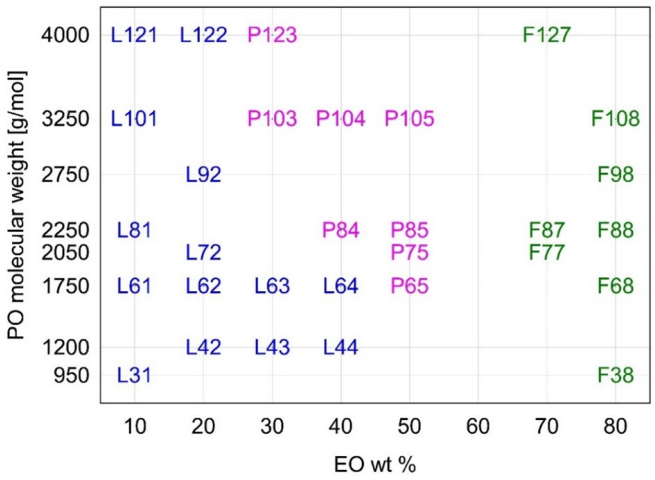


Fig. 1 Pluronic grid. This figure is reproduced from Ref. [3].

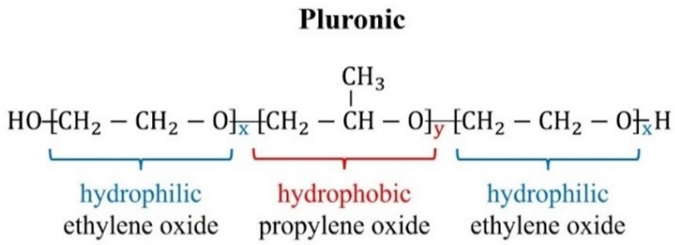


Fig. 2 Molecular structure of Pluronics. The variables x and y denote the lengths of the PEO and PPO chains, respectively. This figure is reproduced from Ref. [3].

and hydrophobic segments determines its hydrophilic-lipophilic properties. The ratio between the lengths of the EO and PO chains defines the hydrophilic-lipophilic balance (HLB), which quantifies the Pluronic degree of hydrophilicity or lipophilicity[5]. The HLB of Pluronic molecules can be calculated using the following equation[6]:

$$\mathrm{HLB} = -36.0\frac{\mathrm{N_{PO}}}{\mathrm{N_{PO}}+\mathrm{N_{EO}}} + 33.2, \quad (1)$$

where $\mathrm{N_{PO}}$ and $\mathrm{N_{EO}}$ refer to the PO and EO block repeating units, respectively.

Depending on factors such as HLB, concentration, temperature, and shear stresses, these materials can exist in aqueous solutions either as individual macromolecules, unimers, or self-assemble into micelles once they surpass the Critical Micellar Concentration (CMC) and the Critical Micellar Temperature (CMT). They can evolve in various microstructures[7,8], which in turn govern the ultimate material properties they exhibit. The CMC represents the concentration threshold at which self-assembly in micelles occurs, which impacts various functionalities, such as drug delivery. Micelles can exhibit a variety of shapes and sizes, typically described by the radius of gyration R_g and the aggregation number A_s.

Pluronic micelles, widely researched for medical applications, are among the most studied systems for drug delivery[9,10]. These micelles can encapsulate hydrophobic drugs within the PPO core, allowing them to be transported throughout the body and to target specific areas, as has recently been done in cancer therapy[11]. For these reasons, the micellar phase has been investigated in depth using numerous experimental techniques. For example, the CMC has been studied by surface tension measurements. Micelle characterization is performed by static light scattering or small-angle X-ray and/or neutron scattering[2,12,13]. Key properties such as CMC, R_g, and A_s, which significantly influence functionalities like drug delivery, are determined by factors such as concentrations, temperatures, and the types of Pluronics. Although predicting and controlling these properties is vital for optimizing performance, the experimental campaigns required to explore different conditions, such as composition, concentration, and temperature, are both time-intensive and costly.

Computational tools offer valuable support that enables the efficient and cost-effective exploration of various scenarios. Atomistic simulations, however, face limitations due to the large time and length scales involved, making them unsuitable for studying self-assembly phenomena directly. Coarse-grained (CG) methods, which represent groups of atoms or molecules as single entities called *beads*, bridge the gap between micro- and macro-scale models[14]. These methods operate at scales inaccessible to molecular-level approaches. Among these, Dissipative Particle Dynamics (DPD) is one of the most widely used techniques for such systems. It offers greater computational efficiency and can simulate time and length scales that accurately reproduce the morphologies of diverse systems[15]. However, its primary drawback is the challenge of capturing chemical specificity and linking the model to atomic-level structures. These challenges are reflected in the parameterization problem: identifying and transferring the parameters of a CG model, such as DPD, is not straightforward but crucial for accurately representing physical properties. CG model parameters are often neither easily identifiable nor directly transferable across different thermodynamic states. These challenges arise regardless of whether the model is developed by reference to experimental data, top-down coarse-graining (CGing), or by reference to an underlying atomistic model, bottom-up CGing[16,17]. In a top-down approach, interaction parameters are typically related to thermodynamic properties or fit to experimental observables. In contrast, bottom-up parameterization derives conservative interactions through structure matching, force matching, or relative entropy methods.

A notable example of bottom-up coarse-graining is multi-scale CGing (force matching)[16,17], where the Mori-Zwanzig projection operator method[18,19] provides a rigorous framework to determine the CG equations of motion (EoM). These equations, derived from microscopic dynamics, take the form of generalized Langevin equation, which under Markovian approximation reduces to DPD EoM with conservative, dissipative force and random forces whose expressions can be obtained directly from atomistic simulations. Despite its theoretical rigor, this bottom-up approach can be computationally prohibitive, particularly for high molecular weight polymers, where extensive molecular dynamics trajectories are required to inform the DPD model. In such cases, a top-down approach, where parameters are empirically fitted to system-specific properties, remains more practical. In

fact, for large macromolecules the identification of DPD interaction parameters is still largely performed relying on this empirical approach[20–22]. Furthermore, the simple form of DPD interactions limits the portability of model parameters. As a means of addressing these deficiencies, a new class of DPD variants is emerging with more expressions for forces to capture interactions more realistically, representing an open and active field of research[23]. For many practical purposes involving such systems, it may be useful and computationally advantageous to retain the original simple form of linear repulsive forces and fit the parameters to the specific properties of the system under investigation, following a top-down procedure. In this work, we focus on different types of Pluronic systems.

The state-of-the-art approach to treat Pluronic systems, or in general other polymer systems, with DPD is the top-down, following the Flory-Huggins theory. Groot and Warren first presented this approach in their pioneering work [14]. In polymer science, established theories connect the χ-parameter to the solubility and the mixing energies of polymeric components[24]. These parameters can be obtained through either atomistic simulations or experimental data. Since Flory-Huggins (FH) parameters are experimentally derived, their calculation does not require additional simulations. However, the Flory-Huggins-based mapping procedure has intrinsic limitations, which hinder the accurate quantitative prediction of critical properties, such as the critical micelle concentration (CMC), the micelle radius of gyration (R_g), and the aggregation number (A_s), especially under varying conditions such as concentration or temperature. Given these limitations, the parameters of a CG model are not universal, hindering dependencies on temperature, composition, and Pluronic types. This dependency cannot be known a priori and need to be investigated for every CG model. The disagreement between experimental results and simulations partly arises from using fixed parameters across different concentrations, temperatures, and Pluronic types. A more flexible model, allowing parameters to vary accordingly, could improve quantitative predictions.

Preliminary work is required to tune and validate these parameters for each system to closely match the experimental results. The adjustment process must account for various $itype - jtype$ interactions (e.g., 1-1, 1-2, 2-2 for a binary system), requiring numerous trial-and-error iterations to find suitable parameters. Indeed, by employing a top-down approach, these can be acquired through multiple iterations of DPD simulations and a systematic comparison with experimental data. However, adjustments are needed to satisfactorily reproduce the physical properties of the system under study. In a top-down parameterization approach, the parameter-tuning task is typically performed through a manual trial-and-error process. This involves using a penalty function that compares the calculated values of selected properties, A_i, with their corresponding target values, A_i^T, from experiments:

$$\mathscr{P}(\boldsymbol{\theta}) = \sqrt{\sum_i \left(1 - \frac{A_i(\boldsymbol{\theta})}{A_i^T}\right)^2}, \tag{2}$$

where $\boldsymbol{\theta}$ represents the model parameters. Each evaluation of $\mathscr{P}(\boldsymbol{\theta})$ requires running a full DPD simulation, followed by a post-processing step to extract the relevant properties. In addition, for each time variable like temperature or the type of Pluronic change, a new simulation is required, adding complexity to the process. DPD simulations are known to be computationally expensive, making them time-consuming and impractical for high-dimensional optimization tasks. This extensive search through the parameter space is not only labor-intensive and costly but also demands substantial expertise from researchers. Machine learning techniques could potentially streamline this parameter optimization process by reducing the effort and enhancing accuracy.

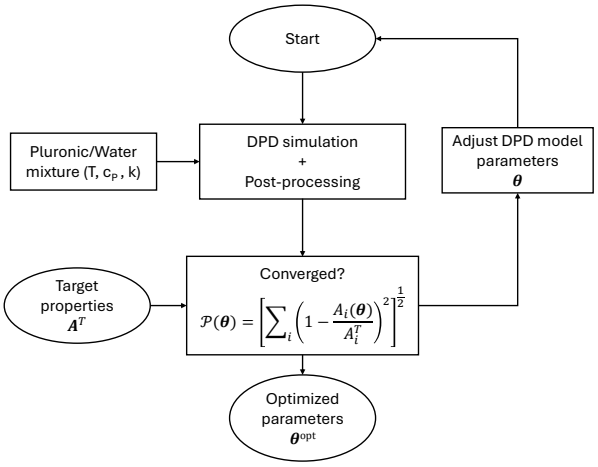


Fig. 3 Typical trial-and-error parameters identification workflow.

This work proposes a data-driven approach for parameterizing Pluronic systems. In particular, a reliable data-driven model is developed to accurately surrogate DPD simulations. This model's ability to deliver faster responses facilitates more efficient parameter optimization and enables the calibration of various Pluronic systems against experimental ground truth.

At the heart of this workflow lies the development of an accurate data-driven surrogate model to replace traditional physics-based DPD simulations. The key parameters of the DPD model are identified as input features, and a series of simulations are performed with these features varied within physically meaningful ranges. To ensure physical consistency, the conservative force parameters were varied were varied within ranges that are physically meaningful for DPD modeling of polymer–water systems: a_{AW} from 25 to 30 (with a grid spacing of 1) for the hydrophilic PEO, and a_{BW} from 30 to 44 (with a grid spacing of 2) for the more hydrophobic PPO. These values are consistent with the expected relative solubility behavior of the PEO and PPO in water and prevent excessively strong repulsions that could lead to non-physical behaviors such as system freezing. For each simulation, physical properties are calculated, creating a dataset in which each sample represents input features paired with their corresponding outcomes. The dataset created by DPD simulations is divided into two subsets: training dataset and validation dataset. The Gaussian Process Regression (GPR) model is trained on the training dataset, and its performance is evaluated on the valida- **R2C4**

tion dataset. The GPR accurately models the relationships between input features and outputs, and once trained, it can make predictions at a much faster rate. However, as a "black-box" machine learning model, it lacks sufficient interpretability. Beyond predictive accuracy and efficiency, understanding why a surrogate model makes a given prediction is crucial. Therefore, we employ SHAP (SHapley Additive exPlanations) analysis[25] to interpret the outputs of the GPR models designed to predict each physical property. SHAP analysis quantifies each input feature's contribution to a specific prediction, comparing their relative importance and revealing any interactive effects. This combination of a fast and accurate GPR surrogate model and SHAP analysis provides an interpretable machine learning approach to facilitate efficient grid-search optimization of the parameters.

DPD simulations are conducted using the open-source software LAMMPS. Physical observables are extracted through a post-processing workflow involving C++ code and a Python routine. The GPR model is developed, trained and tested using the Python library *Scikit learn*. SHAP analysis is conducted using the KernelSHAP method from the SHAP library in Python. The grid-search optimization is performed with a dedicated Python script. The outline of the article is as follows. Section 2 provides an overview of the DPD methodology, including the CG model for Pluronic macromolecules and the procedures used to extract physical observables from equilibrium trajectories. Section 3 outlines the data-driven workflow, comprising the creation of datasets from DPD simulations, post-processing routines, the construction of a GPR-based surrogate model, SHAP analysis, and the methodology employed to solve the parameterization problem. Section 4 presents and discusses the results, emphasizing the surrogate model's accuracy, and efficiency, and interpretability in supporting parameter optimization, as well as its potential for generalization across different Pluronic systems and conditions. Finally, the conclusions are summarized in Section 5.

2 Physics-based simulations

2.1 Dissipative particle dynamics

DPD is a particle-based mesoscopic simulation technique, where groups of atoms are projected into a statistically equivalent ensemble of structureless CG particles, referred to as *beads*[26]. In the standard formulation, these beads are considered to have the same size. The *beads* interact via a mesoscopic force field. The DPD force field consists of variables in reduced units, the so-called DPD units. Typically in DPD, the mass of a single DPD bead, conservative cutoff radius, and thermal energy are taken as, respectively, mass, time, and energy units. The time evolution of each bead can be calculated by the Newton second law as follows:

$$\frac{d\boldsymbol{r}_i}{dt} = \boldsymbol{v}_i, \quad \frac{d\boldsymbol{v}_i}{dt} = \frac{\boldsymbol{f}_i}{m_i}, \tag{3}$$

with $i = 1,..,N$; $\boldsymbol{r}_i$ and $\boldsymbol{v}_i$ are the position and velocity of the bead i with mass m_i, respectively, and N is the number of DPD *beads* in the system. In the case of a DPD fluid, the force $\boldsymbol{f}_i$ acting on the $i-$th bead is the sum of three pairwise contributions:

$$\boldsymbol{f}_i = \sum_{j \neq i} (\boldsymbol{F}_{ij}^C + \boldsymbol{F}_{ij}^D + \boldsymbol{F}_{ij}^R). \tag{4}$$

In 4, the sum runs over the indices of *beads* contained in the closest vicinity of the bead i within a certain cutoff radius r_c. The conservative contribution, $\boldsymbol{F}_{ij}^C$, is a soft-repulsive force acting between a pair of *beads* i and j and having the following functional form

$$\boldsymbol{F}_{ij}^C = \begin{cases} a_{ij}\left(1 - \frac{r_{ij}}{r_c^C}\right)\hat{\boldsymbol{r}}_{ij}, & r_{ij} < r_c^C \\ 0, & r_{ij} > r_c^C \end{cases} \tag{5}$$

where a_{ij} denotes a maximum repulsion between *beads* i and j, $r_{ij} = |\boldsymbol{r}_{ij}| = |\boldsymbol{r}_i - \boldsymbol{r}_j|$ is the separation distance between a pair of *beads*, and $\hat{\boldsymbol{r}}_{ij} = \boldsymbol{r}_{ij}/r_{ij}$ is the unit vector of the bead-bead separation distance and r_c^C is the cutoff radius for the conservative interactions.

To model chain macromolecules, such as Pluronics, an extra spring force is defined to bind the neighboring *beads*. The chemical bonds between the monomers are represented using the *bead-and-spring* model, in which adjacent *beads* in the polymer chain interact via a harmonic potential, described by the equation:

$$E_{harm} = K_b(r_{ij} - R_0)^2, \tag{6}$$

where R_0 is the nominal equilibrium distance and K_b the harmonic constant. CGing, which involves eliminating degrees of freedom from the system, reduces friction and smooths the free-energy landscape, leading to faster dynamics at the mesoscale level. In DPD, these removed degrees of freedom are effectively reintroduced through pairwise dissipative and random forces. Dissipative and random forces, $\boldsymbol{F}_{ij}^D$ and $\boldsymbol{F}_{ij}^R$, respectively, represent the effect of viscosity slowing down the particles motion with respect to each other and of thermal/vibrational energy of the system. They act together as a thermostat. The random and dissipative forces are coupled via the fluctuation–dissipation theorem (FDT)[27], ensuring sampling from the appropriate probability distribution. In addition, DPD conserves the total momentum. The weight functions for the dissipative force and the stochastic force provide the interaction range for their respective forces and are linked by a relationship necessary to ensure that the system, in the limit of an infinitesimal time step, reaches Gibbs equilibrium, derived from Español and Warren[27]. Obtaining from the simulations an equilibrium distribution corresponding to the Boltzmann distribution of a canonical or Gibbs ensemble is fundamental to relating to the classical relations of thermodynamics in the system under consideration. One of the two weight functions that appear can thus be chosen arbitrarily, and this choice determines the other:

$$w_D(r) = [w_R(r)]^2, \tag{7}$$

$$\sigma^2 = \frac{2\gamma k_B T}{m}, \tag{8}$$

where k_B is the Boltzmann constant and T is the equilibrium temperature. These conditions ensure that the DPD equations act as a thermostat, and since the algorithm depends on relative velocities and the interactions between particles are symmetric, it is

a Galilean-invariant thermostat that preserves hydrodynamics[28]. The conventional functional form for the dissipative and stochastic weight functions is as follows:

$$w^D(r_{ij}) = \left[w^R(r_{ij})\right]^2 = \begin{cases} \left(1-\frac{r_{ij}}{r_c}\right)^2 \hat{\mathbf{r}}_{ij}, & r_{ij} < r_c \\ 0. & r_{ij} \geq r_c \end{cases} \quad (9)$$

2.2 Coarse-grained model and parameters identification

In this work, adhering to the standard approach, the CG factors are selected such that all *beads* represent almost the same mass and volume[14,26,27,29] as shown in Tab. 1. The *mapping* scheme adopted is the same as described by van Vlimmeren et al.[30], shown in Fig. 4 for Pluronic L64. The number of monomers clustered into one bead is 4.3 for the ethylene oxide (EO) repeating units and 3.3 for the propylene oxide (PO) repeating units. Based on this scheme, the CG topology of the Pluronic L64 macro-

R2C1

Table 1 *Coarse-grained* (CG) *beads*, molecular volumes of H_2O, EO, and PO calculated according to the approach by Durchschlag and Zipper[29], and the corresponding *bead* volumes expressed in Å^3.

CG *bead*	Molecular volume (Å^3)	Volume *bead* (Å^3)
$[H_2O]_{10}$	30	300
$[CH_2CH_2O]_{4.3}$	64.6	278
$[CH_3CHCH_2O]_{3.3}$	96.5	318

molecule results in a chain of 15 *beads* and is shown in Fig. 4. It can be schematically represented as $A_3B_9A_3$, where A stands for the CG bead for EO and B for PO. In the Tab. 2 are reported the coarse-grained topologies of the analyzed Pluronic systems. In the simulations, a third type of *beads* (W) is considered that represents water molecules, with a corresponding degree of CGing equal to 10.

R2C1

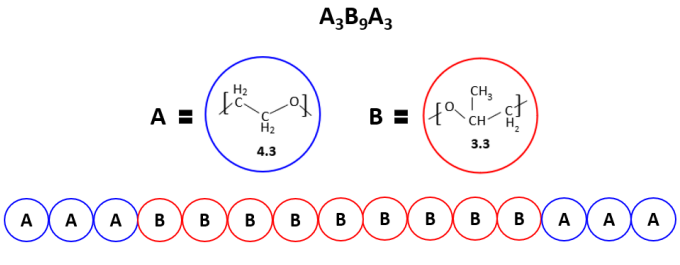


Fig. 4 Coarse-grained model of Pluronic L64.

Table 2 Coarse-Grained topologies of the analyzed Pluronic systems

Pluronic Name	Composition	DPD model composition
L64	$EO_{13}PO_{30}EO_{13}$	$A_3B_9A_3$
P65	$EO_{20}PO_{30}EO_{20}$	$A_5B_9A_5$
P104	$EO_{27}PO_{63}EO_{27}$	$A_4B_{18}A_4$
F68	$EO_{82}PO_{31}EO_{82}$	$A_{18}B_9A_{18}$

The repulsive coefficients for *beads* of the same type are calculated according to the following relationship[31]:

$$a_{ii} = \frac{\kappa^{-1} - k_B T}{2\alpha\rho}, \quad (10)$$

where α is a constant, κ^{-1} is the dimensionless bulk modulus at the considered temperature (the inverse of dimensionless compressibility), ρ is the number density, k_B denotes the Boltzmann constant and T is the system temperature. This relationship (Eq. 10) ensures that the compressibility in the simulation model matches the compressibility of the liquid to be studied, correctly describing the density fluctuations as they appear in a molecular liquid. From this condition, the repulsion parameters between equal *beads* can be fixed. Specifically, this relationship uses the generalized approach[32] extensively tested in our previous work[31], which incorporates both the simulation temperature T and the temperature-dependent compressibility κ in evaluating the repulsive interaction parameter. The repulsion coefficients for *beads* of different type, a_{ij}, is approximately given by[14]:

$$a_{ij} \approx a_{ii} + \frac{\chi_{ij}}{(0.231 \pm 0.001)}. \quad (11)$$

This expression (Eq. (11)), suggests that the repulsion parameters between *beads* representing different species contain an extra-repulsion term, which should be chosen to ensure that different solubilities of the involved species are accurately captured within the DPD model. However, direct transfer of a_{ij} values computed from χ_{ij} between different systems, in this case different Pluronics, is subject to limitations, dictated by different reasons. This problem is commonly known as the *parameters portability issue*. First, Eq. (11) is numerically derived for a specific system[14]. Furthermore, the experimental values of χ_{ij} available in the literature are obtained under specific experimental conditions. A key point is that comparisons with experiments indicate that χ_{ij} depends on both entropic and energetic contributions. In other words, the behavior of such systems arises from a subtle interplay between entropy and enthalpy[33]. However, predicting how entropic effects vary across different systems and incorporating them into a parameter remains a significant challenge. To conclude, while χ_{ij} and Eq. (11) provide a useful starting point for identifying DPD interaction parameters, fine-tuning of a_{ij} for a certain system, such as a Pluronic type, remains largely empirical when aiming to fit quantitative experimental targets. For consecutive *beads* in the chain, two additional parameters are involved: the nominal equilibrium distance $R_{0,ij}$ and the harmonic constant $K_{b,ij}$. The choice of the nominal equilibrium distance $R_{0,ij}$ and the harmonic constant $K_{b,ij}$ for each pair of beads must ensure that the topology of the polymer chains is preserved.

R2C3

2.3 System observables extraction

In this work, three different physical observables are selected as targets: the micelle aggregation or association number A_s, the CMC, and the micelle radius R_g. DPD simulations produce trajectories which are analyzed using our in-house software. The trajectory file stores information on all particles, such as their positions or velocities, in regular time intervals. Statistical analysis of

R2C7

these trajectory files allows us to gain valuable insights into the aggregation behavior during simulations, by detecting phenomena such as micelles, aggregates, and other structures. The core of this kind of analysis is a clustering algorithm able to determine if there is any aggregation occurring within the DPD simulation and to calculate various properties of aggregates. The following paragraphs describe how the in-house software we use extracts from DPD simulations each of the three selected target properties.

2.3.1 Aggregation number

The aggregation or association number, denoted as A_s, represents the number of chains present within a single aggregate. It is essential to first establish a criterion to establish the belonging of chains to the same aggregate in the simulation. A straightforward and commonly used criterion is that two molecules are considered to be part of the same aggregate if they have at least a specified number of contact pairs. A contact pair is defined as a pair of *beads* from different molecules that are closer than a given distance. The weight-average aggregation number is defined as

$$\langle A_s \rangle_w = \frac{\sum_i m_{A_s,i}^2}{\sum_i m_{A_s,i}}, \tag{12}$$

where $m_{A,i}$ is the weight of an aggregate i. The weight distribution function of aggregation numbers, F_w, is defined as:

$$F_w(A_s) = \frac{m_{A_s} N_{A_s}}{\sum_i m_{A_s,i} N_{A_s,i}}, \tag{13}$$

where $m_{A,i}$ represents the weight of an aggregate with an association number $A_s = i$ and $N_{A,i}$ denotes the number of aggregates with that association number. Similarly, the number distribution function of aggregation numbers, F_n, is expressed as:

$$F_n(A_s) = \frac{N_{A_s}}{\sum_i N_{A_s,i}}. \tag{14}$$

Both of these functions are normalized, so the value of F_w and F_n indicates the fraction of aggregates with a specific mass or aggregation number present in the system.
In this work, $\langle A_s \rangle_w$ is considered to be compared with experimental data. For simplicity of notation, in the following paragraphs $A_s \equiv \langle A_s \rangle_w$.

2.3.2 Determining the Critical Micellar Concentration

The CMC is the concentration above which Pluronics macromolecules self-assemble into micelles. Determination of the CMC is not straightforward and depend on the criteria used. Different methods have been presented in the literature. In this work, the CMC is defined as the concentration of sub-micellar aggregates. A cut-off aggregation number is defined to distinguish sub-micellar portion from micellar one. For a system with a concentration, c, above the CMC, the CMC is equal to the concentration of aggregates c_{sm} with A_s below a cut-off association number, $A_{s,m}$. The latter is identified from the number distribution function as done in Ref.[34].

2.3.3 Micelle radius from Gyration tensor

The micelle radius, denoted as R_g, is defined as the square root of the mean squared distance of *beads* in a chain or aggregate from its center of mass, r_{CM}, assuming uniform mass distribution among the particles:

$$R_g^2 = \frac{1}{N} \sum_{i=1}^{N} (\boldsymbol{r}_i - \boldsymbol{r}_{CM})^2, \tag{15}$$

where N is the number of particles forming the chain or aggregate. The radius of gyration is calculated using the gyration tensor, $\boldsymbol{S}$, which describes the second moments of the position vectors of the *beads*:

$$S_{mn} = \frac{1}{N} \sum_{i=1}^{N} (r_i^m - r_{\mathrm{CM}}^m)(r_i^n - r_{\mathrm{CM}}^n), \tag{16}$$

where r_i^m stands for the m-th Cartesian coordinate of $\boldsymbol{r}_i$ and r_{CM} is the m-th Cartesian coordinate of $\boldsymbol{r}_{\mathrm{CM}}$. The gyration tensor is a symmetric matrix which can be diagonalized:

$$\boldsymbol{S} = \begin{pmatrix} \lambda_x^2 & 0 & 0 \\ 0 & \lambda_y^2 & 0 \\ 0 & 0 & \lambda_z^2 \end{pmatrix}, \tag{17}$$

where λ_x^2,λ_y^2 and λ_z^2 are eigenvalues chosen so that $\lambda_x^2 < \lambda_y^2 < \lambda_z^2$. The eigenvectors indicate the principal axes of the aggregate, while the eigenvalues correspond to the gyration lengths along each principal direction. These eigenvalues can be interpreted as the semi-axes of an equivalent ellipsoid. They are used to calculate various properties, including shape descriptors of the aggregates, such as the radius of gyration:

$$R_g^2 = \lambda_x^2 + \lambda_y^2 + \lambda_z^2. \tag{18}$$

Analogously to the aggregation number, the weight-averaged radius of gyration is computed for comparison against experimental data. For simplicity of notation, in the remaining part of the work $R_g \equiv \langle R_g \rangle_w$.

3 Data-driven approach: methods and computational details

3.1 Dataset creation

The first step involves performing DPD simulations to generate the dataset necessary to train the GPR algorithm. These simulations provide the "ground truth" data for the data-driven model. A portion of data created by simulation is used to train the GPR, and another portion is subsequently used to assess the accuracy of GPR predictions. The workflow comprises two main stages. The first stage is the simulation setup, which involves identifying the key DPD input parameters that govern the accurate reproduction of the target physical properties: micelle gyration radius (R_g), aggregation number (A_s) and CMC. The second stage is the extraction of these physical properties from simulation trajectories through a structured post-processing routine. These two parts of the workflow are detailed in the subsequent paragraphs. This study investigates four different Pluronics, summarized in Tab. 3,

along with their physiochemical properties.

For each Pluronic, a dataset consisting of 40 data points is created based on DPD simulations, which serve as the "ground truth" for the surrogate model. The number of data points is determined for one Pluronic type and progressively enriched to reduce the uncertainty level. The input features, a_{BW} and a_{AB}, are selected due to their relevance to the system behavior, as will be discussed in Sect. 3.1.1. These parameters are systematically varied within physically meaningful ranges, whereas other simulation parameters are kept constant, as will be outlined in Sect. 3.1.1. For each simulation, physical properties are calculated, as will be explained in 3.1.2, resulting in a dataset where each sample consists of input features paired with their corresponding outcomes.

Table 3 Physiochemical properties of the considered Pluronics. [3]

Pluronic Name	Composition	Avg. Mol. Wt. (g/mol)	PO Mol. Wt. (g/mol)	EO Content (wt%)	HLB Value
L64	$EO_{13}PO_{30}EO_{13}$	2900	1750	40	15
P65	$EO_{20}PO_{30}EO_{20}$	3400	1750	50	17
P104	$EO_{27}PO_{63}EO_{27}$	5900	3250	40	13
F68	$EO_{82}PO_{31}EO_{82}$	8350	1750	80	28

3.1.1 Simulations setup and details

The simulation setup requires the identification of key model parameters that exert the greatest influence on the target properties. This section explains the considerations underlying parameter selection and provides an overview of the simulation details.

All conservative model parameters are summarized in Tab. 4. The repulsive parameters are denoted as a_{ij} and the cutoff distances are represented as $r_{C,ij}$. The chemical bonds between the monomers that form the polymer macromolecule are represented using the *beads and spring* model, in which adjacent *beads* in the polymer chain interact via a harmonic potential. Thus, for consecutive *beads* in the chain, two additional parameters are involved: the nominal equilibrium distance $R_{0,ij}$ and the harmonic constant $K_{b,ij}$. Accounting for all parameters creates a high-dimensional parameter space, making the problem computationally demanding. To address this, we initially restrict the number of parameters by leveraging physical insights. Future work may explore additional parameters to further refine the model. First, since ij-interactions are identical to ji-interactions, redundant parameters below (or above) the diagonal in Table 4 are excluded. Second, as all *beads* represent almost the same volume, all cutoff distances are assumed equal and set to 1. Third, the repulsive parameter a_{WW} is set to match the isothermal compressibility of water, as is commonly done in DPD simulations. a_{AA} and a_{BB} are assumed equal to a_{WW} as usually done in DPD models. As a result, the remaining adjustable parameters in the table are a_{AW}, a_{BW}, a_{AB}, $R_{0,AA}$, $R_{0,BB}$, $R_{0,AB}$, $K_{b,AA}$, $K_{b,BB}$, and $K_{b,AB}$. All bonded interaction parameters are assumed to be the same (R_0 and K_b for all bonds). The assumption is justified by the CG level. In addition, these parameters have a marginal impact on CMC and A_s. They can slightly influence the shape and compactness of the micelles, and then the R_g. In this work, preliminary simulations are carried out to establish reasonable values of bonded parameters: $K_b = 50$ and $R_0 = 0.7$ are found to give the desired behavior of the bond-bond distribution function.

Table 4 Conservative interaction parameters for the three species: PEO, PPO, and water.

	A (PEO)	**B (PPO)**	**W (H_2O)**
A (PEO)	$a_{AA}; r^C_{c,AA}; K_{b,AA}; R_{0,AA}$	$a_{AB}; r^C_{c,AB}; K_{b,AB}; R_{0,AB}$	$a_{AW}; r^C_{c,AW}$
B (PPO)	$a_{BA}; r^C_{c,BA}; K_{b,BA}; R_{0,BA}$	$a_{BB}; r^C_{c,BB}; K_{b,BB}; R_{0,BB}$	$a_{BW}; r^C_{C,BW}$
W (H_2O)	$a_{WA}; r^C_{C,WA}$	$a_{WB}; r^C_{C,WB}$	$a_{WW}; r^C_{C,WW}$

Furthermore, a_{AB} is assumed to be equal to a_{BW}, reducing the free parameters to a_{AW} and a_{BW}. These simplifications are reasonable because the self-assembly process is primarily driven by the solubility differences between the hydrophobic and hydrophilic blocks and water.

Thus, a_{AW} and a_{BW} are the key parameters of the DPD model to capture the chosen physical targets. As discussed in Introduction and in Sect. 2.2, these parameters are expected to be different between different Pluronic types.

All other simulation parameters remain constant. The simulations are conducted in a cubic box with a side length of $40r_c^C$, employing periodic boundary conditions in all directions. The number density is fixed at $\rho = 3$, and the temperature is set at $T = 30^{\circ}$C. The friction coefficient γ is maintained at a standard value of 4.5.

3.1.2 Post-processing routine

The post-processing is handled through an automated routine schematized in Alg. 1, which orchestrates the calculations necessary to derive from the equilibrium trajectories, produced by the DPD simulations, the system observables: CMC, R_g, and A_s. After the simulation data are converted into a suitable format, the post-processing phase begins by removing water beads from the coordinate file, isolating macromolecules. The post-processing of saved trajectories starts with the identification of Pluronic aggregates. Aggregates are identified on the basis of a predefined minimum distance between beads of different macromolecules and a required number of contacts to classify two macromolecules within the same aggregate. This step is followed by the calculation of the aggregation number distribution. The aggregation or association number, denoted as A_s, represents the number of chains within a single aggregate. The average aggregation number is then computed over the entire simulation to track its evolution and estimate the equilibration time. Once the equilibration time is obtained, the aggregation number distribution and average aggregation number are recomputed and discarded from the equilibration steps. Specifically, the weight-average aggregation number is computed for comparison with the experimental data. The CMC is then computed as detailed in Sect. 2.3. Eventually, the weight average R_g is calculated from the gyration tensor, as described in Sec. Section 2.3.

R2C1

3.2 Surrogate modeling via GPR

Once the dataset for each Pluronic system (e.g., L64 at 30°C) is generated from the simulations, it is used to construct separate surrogate models for each output property (e.g., R_g, A_s, CMC).

Algorithm 1 Automated routine for dataset creation

Input: Interaction parameters a_{AW}, a_{BW}
Run DPD simulation
Convert topology and coordinate files into a format compatible with the post-processing routine
Post-processing routine:

- Remove water beads from the coordinate file
- Identify aggregates:
  - Define the minimum distance threshold for two beads from different macromolecules to be considered part of the same aggregate
  - Establish the minimum number of contacts required between two macromolecules to classify them as belonging to the same aggregate
- Compute:
  - Aggregation number distribution
  - Average aggregation number
  - Equilibration time
  - Aggregation number distribution and average aggregation number (A_s) discarding equilibration steps
  - Critical Micelle Concentration (CMC)
  - Gyration tensor
  - Shape descriptors
  - Gyration radius (R_g)

Output: CMC, A_s, R_g

Each model captures the relationship between the respective output property and the input parameters (a_{AW}, a_{BW}). The same process is applied to all four Pluronic systems (L64, P65, P104, F68). To establish these input-output relationships, we employ GPR, a Bayesian machine learning technique that leverages non-parametric modeling through Gaussian processes. It excels in handling small datasets, provides uncertainty estimates for predictions, and offers flexibility through its kernel-based approach, making it well-suited for our regression tasks. The estimated uncertainty, quantified by the confidence intervals 95%, can inform the required size of the training data to achieve a desired level of confidence.

For each Pluronic system, we construct three distinct GPR models of the form $y = GP(\mathbf{x})$, where the input vector $\mathbf{x} = [a_{\mathrm{AW}}, a_{\mathrm{BW}}]$ represents the water-interaction parameters. Each model predicts one of our target properties: R_g, A_s, or CMC. Using R_g as an example, we will demonstrate the modeling process that is subsequently applied to A_s and CMC. Consider a training set with N_{train} samples. We can represent the input and output as:

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_{N_{\mathrm{train}}} \end{bmatrix} \in \mathbb{R}^{N_{\mathrm{train}} \times 2} \quad \text{and} \quad \mathbf{Y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{N_{\mathrm{train}}} \end{bmatrix} \in \mathbb{R}^{N_{\mathrm{train}}}, \tag{19}$$

respectively. Here, $\mathbf{x}_i = [a_{\mathrm{AW},i}, a_{\mathrm{BW},i}]$ represents the input parameters for the i-th sample from simulation, and y_i is its corresponding output ($R_{g,i}$ in this example).

The Gaussian process assumes that the output values $\mathbf{Y}(\mathbf{X})$ in any given set of input points $\mathbf{X}$ are drawn from a multivariate normal distribution. The Gaussian process is fully characterized by a mean function $\mathbf{U}(\mathbf{X})$ and a covariance function, which describes the relationships between input features:

$$\mathbf{Y}(\mathbf{X}) \sim \mathrm{GP}(\mathbf{U}(\mathbf{X}), \mathbf{\Sigma}(\mathbf{X}, \mathbf{X})), \tag{20}$$

where $\mathbf{U}(\mathbf{X}) \in \mathbb{R}^{N_{\mathrm{train}}}$ is the mean function, and $\mathbf{\Sigma}(\mathbf{X}, \mathbf{X}) \in \mathbb{R}^{N_{\mathrm{train}} \times N_{\mathrm{train}}}$ is the covariance matrix. The covariance function is defined using the Radial Basis Function (RBF) kernel, which measures the similarity between two input points $\mathbf{x}_i$ and $\mathbf{x}_j$. The RBF kernel is given by:

$$\Sigma(\mathbf{x}_i, \mathbf{x}_j) = \theta_a^2 \exp\left(-\frac{1}{2}\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\theta_l^2}\right), \tag{21}$$

where the θ_a is the amplitude parameter, and θ_l is the length scale, which control the smoothness of the mean function.

The GPR model training involves optimizing the kernel hyperparameters by maximizing the log-likelihood function:

$$\log p(\mathbf{Y}|\mathbf{X}; \mathbf{\Theta}) = -\frac{1}{2}\left[\mathbf{Y}^\top \tilde{\mathbf{\Sigma}}^{-1}\mathbf{Y} + \log|\tilde{\mathbf{\Sigma}}| + N_{\mathrm{train}}\log(2\pi)\right], \tag{22}$$

where $\tilde{\mathbf{\Sigma}} = \mathbf{\Sigma}(\mathbf{X}, \mathbf{X}) + \sigma^2\mathbf{I}$ and $|\tilde{\mathbf{\Sigma}}|$ is the determinant of matrix $\tilde{\mathbf{\Sigma}}$. Here, $\mathbf{\Theta} = [\sigma, \theta_a, \theta_l]$, which include the noise level σ and kernel parameters θ_a and θ_l, are optimized using the L-BFGS Quasi-Newton optimization algorithm[35].

Once the GPR model is trained, for any new input $\mathbf{x}'$, the posterior distribution of the output can be obtained as:

$$y(\mathbf{x}')|\mathbf{Y}(\mathbf{X}) \sim \mathrm{GP}(\hat{\mathbf{u}}, \hat{\mathbf{\Sigma}}), \tag{23}$$

where

$$\begin{aligned} \hat{\mathbf{u}} &= \mathbf{\Sigma}(\mathbf{x}', \mathbf{X})\left[\mathbf{\Sigma}(\mathbf{X}, \mathbf{X}) + \sigma^2\mathbf{I}\right]^{-1}[\mathbf{Y} - \mathbf{U}(\mathbf{X})] + \mathbf{u}(\mathbf{x}'), \\ \hat{\mathbf{\Sigma}} &= \mathbf{\Sigma}(\mathbf{x}', \mathbf{x}') - \mathbf{\Sigma}(\mathbf{x}', \mathbf{X})\left[\mathbf{\Sigma}(\mathbf{X}, \mathbf{X}) + \sigma^2\mathbf{I}\right]^{-1}\mathbf{\Sigma}(\mathbf{x}', \mathbf{X})^\top. \end{aligned} \tag{24}$$

Here, $\hat{\mathbf{u}}$ denotes the GPR model's predicted mean value of the output, while the covariance matrix $\hat{\mathbf{\Sigma}}$ indicates the prediction confidence and reveals the associated uncertainty of this prediction. The 95% confidence interval is determined from the mean prediction and the standard deviation as: [mean_prediction $-1.96\times$ std_prediction, mean_prediction $+1.96\times$ std_prediction], where the standard deviation is calculated as the square root of each diagonal element of the covariance matrix.

3.3 SHapley Additive exPlanations (SHAP) analysis

In addition to achieving good accuracy, interpretability, understanding why each prediction is made, is also important when building a predictive surrogate model. ML-derived models, like neural networks or GPR models, are often characterized as "black boxes" due to their intricate internal structures, making it difficult to discern the relationships they learn between inputs and outputs. This lack of interpretability can hinder the extraction of actionable knowledge from the constructed surrogate model. To overcome this challenge, a post-hoc analysis via SHAP is employed, which seeks an interpretable approximation of the original model through evaluating the additive feature attribution[25].

Let $f(\mathbf{x})$ denote the original model that we attempt to interpret, with $\mathbf{x}$ the input vector that can belong to a high dimensional space. The goal herein is to explain f locally at a given $\mathbf{x}$. For this, an explanation model $g(\mathbf{x}')$ is sought, whose input vector $\mathbf{x}'$ can be mapped to $\mathbf{x}$ using a mapping function h_x, i.e., $\mathbf{x} = h_x(\mathbf{x}')$. The mapping may reduce the dimensionality; $\mathbf{x}'$ can have the same

or a lower dimension than $\mathbf{x}$. Mathematically, we can say that g successfully explains the original model f, if we ensure $g(\mathbf{z}') \approx f(h_x(\mathbf{z}'))$, whenever $\mathbf{z}' \approx \mathbf{x}'$. This gets more intuitive when g is an additive feature attribution model which is defined as:

$$g(\mathbf{z}') = \phi_0 + \sum_{i=1}^{M} \phi_i z_i' \,, \tag{25}$$

where $\mathbf{z}'$ is a binary vector with dimension M, i.e., $\mathbf{z}' = \{0,1\}^M$ indicating the presence or absence of a feature; $\phi_i \in \mathbb{R}$ represents the contribution of feature i to the output $f(h_x(\mathbf{z}'))$; and ϕ_0 is the base value of the model output without any feature's contribution. In particular for our analysis, $f(\mathbf{x})$ is the GPR surrogate model with the input vector $\mathbf{x} = [a_{\mathrm{AW}}, a_{\mathrm{BW}}]$; and $M = 2$, i.e., $\mathbf{z}'$ and $h_x(\mathbf{z}')$ are both two-dimensional vectors. The attribution values ϕ_i are referred to as Shapley values (or SHAP values). By the definition and additive nature of g, Shapley values can effectively quantify how much each input feature contributes to the value of the output.

It has been proved using the co-operative game theory[25] that for a given machine learning model f and the additive attribution model g as defined in Eq. (25), if f and g satisfy all three properties namely the local accuracy, missingness, and consistency, there exists a unique solution to g, and Shapley values ϕ_i are given by:

$$\phi_i = \sum_{S \subseteq \{1,2,\ldots,M\} \setminus \{i\}} \frac{|S|!(M-|S|-1)!}{M!} [f_x(S \cup \{i\}) - f_x(S)] \,, \tag{26}$$

where S denotes the set of non-zero indexes in $\mathbf{z}'$, and hence $|S|$ indicates the number of non-zero indexes in $\mathbf{z}'$. Here, $f(h_x(\mathbf{z}'))$ is denoted by $f_x(S)$ and called the value function given by $f_x(S) = f(h_x(\mathbf{z}')) = E[f(\mathbf{x})|\mathbf{x}_S]$, where $E[f(\mathbf{x})|\mathbf{x}_S]$ is the expected value of the function conditioned on a subset S of the input features. As an example, consider a machine learning model f with 4 input features $\mathbf{x} = (x_1, x_2, x_3, x_4)$ and let the corresponding interpretation model g also have 4 feature inputs $\mathbf{z}' = (z_1', z_2', z_3', z_4')$. Let $S = \{1,3\}$ which implies $\mathbf{z}' = (1,0,1,0)$. If we denote the i^{th} component of $h_x(\mathbf{z}')$ by h_{xi} then $f_x(S)$ is given by

$$f_x(S) = E[f(\mathbf{x})|x_1 = h_{x1}, x_2 = h_{x2}] \,. \tag{27}$$

Computing Shapley values directly using Eq. (26) is intractable. Inspired from the Local Interpretable Model-agnostic Explanations (LIME) method[36], the same Shapely values can be computed by finding an additive feature attribution model g which minimizes the loss function L given by[25]:

$$L(f, g, \pi_x) = \sum_{\mathbf{z}' \in Z} [f(h_x(\mathbf{z}')) - g(\mathbf{z}')]^2 \pi_x(\mathbf{z}') \,, \tag{28}$$

where Z is the set of all possible M-dimensional binary vectors $\mathbf{z}'$; $|\mathbf{z}'|$ is the number of non-zero elements in $\mathbf{z}'$; and the function $\pi_x(\mathbf{z}')$ is defined as:

$$\pi_x(\mathbf{z}') = \frac{(M-1)}{\binom{M}{|\mathbf{z}'|} |\mathbf{z}'| (M - |\mathbf{z}'|)} \,. \tag{29}$$

By such, it treats estimating the function g in Eq. (25) essentially as a weighted linear regression problem. This approach of determining the Shapley values is referred to as Kernel SHAP method[25], which is adopted in this work for our estimation of Shapley values for the two features $[a_{\mathrm{AW}}, a_{\mathrm{BW}}]$.

3.4 Adaptive grid-search for parameters optimization

The GPR-based surrogate model and SHAP analysis are employed to perform an adaptive grid-search optimization of the parameters. This optimization process addresses a multi-target objective, defined as:

$$\begin{aligned} \boldsymbol{\theta}^{\mathrm{opt}} &= \arg\min_{\boldsymbol{\theta}} \mathscr{P}(\boldsymbol{\theta}) \\ \mathscr{P}(\boldsymbol{\theta}) &= \sqrt{\sum_i \left(1 - \frac{A_i(\boldsymbol{\theta})}{A_i^T}\right)^2} \end{aligned} \tag{30}$$

where $\boldsymbol{\theta} = [a_{AW}, a_{BW}]$ represents the input parameters, and $\boldsymbol{A} = [\mathrm{CMC}, R_g, A_s]$ denotes the output properties of interest with $\boldsymbol{A}^T$ representing their target experimental values.

Four different Pluronics, as reported in Tab. 3, are analyzed in this work. The target experimental values for the output properties are taken from the literature and summarized in Tab. 5.

Table 5 Experimental values[12,13] for the Critical Micellar Concentration (CMC), radii of spherical aggregates (R_g), and aggregation numbers (A_s) of the considered Pluronics at 30°C and low concentrations.

Pluronic Name	CMC [wt. %]	R_g [nm]	A_s
L64	2.5	2.28	10
P65	3.2	2.27	3.8
P104	0.002	5.36	81
F68	10	2.30	3.5

R2C1

The grid-search systematically evaluates combinations of parameter values across a predefined search space to identify the configuration that minimizes the objective function $\mathscr{P}(\boldsymbol{\theta})$. Each evaluation involves querying the GPR surrogate models to estimate the output properties for a given set of input parameters. The optimization employs an adaptive grid-search strategy, which iteratively refines the search space based on the results of previous evaluations. Unlike a traditional grid-search that evaluates all combinations of parameter values in a predefined grid, the adaptive approach dynamically adjusts the resolution and boundaries of the grid to focus on regions of interest where the objective function $\mathscr{P}(\boldsymbol{\theta})$ is minimized. Initially, a coarse grid is defined over the range identified by SHAP analysis in the parameter space, and then the GPR models are used to evaluate $\mathscr{P}(\boldsymbol{\theta})$ at each grid point. Based on these results, regions with low objective values as well as high sensitivity (as revealed from SHAP analysis) are identified, and the grid resolution in these regions is refined iteratively. Each evaluation of $\mathscr{P}(\boldsymbol{\theta})$ involves querying the GPR surrogate models to estimate the output properties for a given set of input parameters. The process continues until the optimal parameter configuration $\boldsymbol{\theta}^{\mathrm{opt}} = [a_{AW}^{\mathrm{opt}}, a_{BW}^{\mathrm{opt}}]$ that minimizes the objective function is identified.

4 Results and discussion

4.1 Assessment of the GPR model

In this study, we employ GPR as an effective surrogate model for DPD simulations. GPR demonstrates its capability to provide accurate predictions, successfully capturing the relationships between DPD model input parameters (a_{AW},a_{BW}) and the resulting properties of the system (CMC, R_g and A_s). Three different GPR surrogate models are employed, each tailored to predict one of the specific properties of the system based on the two input parameters.

R2C5, R3C3

In this section, we assess the training convergence and predictive accuracy of each GPR model, using the Pluronic system L64 for demonstration. To ensure a systematic evaluation, we employ a K-fold cross-validation approach[37]. Specifically, with $K = 8$, we randomly generate eight distinct training-validation splits, each comprising 35 training data points and 5 validation data points. For each fold, we conduct a convergence analysis to evaluate the model's accuracy as the training data size increases. The detailed procedure is as follows: 1) For the first fold, a GPR model is trained using an initial subset of 5 randomly selected data points from the 35 available training samples. The trained model's predictive performance is then quantified by computing the relative L_2 error on the 5 validation data points of that fold. This error serves as the initial validation error. 2) Subsequently, a new GPR model is trained using an expanded training set of 10 data points, consisting of the previously used 5 and an additional 5 randomly sampled from the remaining 30 training data points. The validation error is computed again using the same 5 validation data points. 3) This iterative process continues, with 5 new training data points added at each step, until all 35 training data within the fold are utilized. This procedure yields a convergence curve illustrating the evolution of the validation error as the training data size grows. 4) The aforementioned convergence analysis is performed independently for each of the 8 folds. For each training data size considered (5, 10, ..., 35), the mean and standard deviation of the validation errors across the 8 folds are then calculated, as shown in Fig. 5b, 6b, and 7b. This rigorous approach provides a comprehensive evaluation of how the trained GPR model's accuracy improves with increasing training data, as evidenced by the consistent decrease in the mean relative error observed with larger training sets. Finally, the ultimate surrogate model is trained using the entire dataset of 40 available data points, leading to further enhanced accuracy. Fig. 5a illustrates the validation result for one of the eight cross-validation folds using the GPR model trained on 35 data points for CMC. The CMC values predicted by the trained GPR model are displayed using a color gradient, with yellow shades indicating higher values and pink shades indicating lower values. The red circles indicate the 35 data points used to train the GPR model, while the continuous surface displays the model predictions across the a_{AW} and a_{BW} input space. Additionally, blue triangles mark the 5 validation data points, allowing for an assessment of the model's predictive accuracy. For the final model trained using all 40 available data, Fig. 5c and 5d provide detailed views of specific sections of the two-dimensional input space. These plots show slices along the a_{AW} and a_{BW} directions, respectively, to visualize the confidence intervals of the final GPR model. The plots display the predicted CMC values with green regions representing the 95% confidence intervals, where the very thin green regions indicate that the uncertainty of the GPR model is very low and the model represents a reliable regression. By examining these one-dimensional sections, the model's accuracy and robustness are further examined, demonstrating the capability of GPR to effectively capture variations and uncertainties within the input domain. Fig. 6 and 7 present similar analyses and findings for the GPR models of R_g and A_s, respectively.

4.2 Interpretation

After assessing the training convergence and predictive accuracy of each GPR surrogate model, we next interpret the output of a GPR model through the results of the SHAP analysis. SHAP values quantify the individual contribution of each DPD model input parameter (a_{AW} or a_{BW}) to a specific prediction of the polymer property (CMC, R_g, or A_s), effectively revealing how each parameter influences the predicted property. They provide a deeper understanding of the GPR model's behavior and the underlying relationships it captures, moving beyond simple observation to provide insights into the causal mechanisms driving the predicted properties. Using the results of L64 Pluronic system as a demonstration, the SHAP summary plots in Fig. 8 visualize the quantitative impact of each input parameter on the predicted property (the GPR model's output) for all sampled points in the space of (a_{AW},a_{BW}). In total 400 data points uniformly sampled (as lattice grids) in the space of (a_{AW},a_{BW}) are used for this calculation. The color gradient represents the magnitude of the input parameter (red for high values, blue for low). The SHAP values not only indicate whether an input parameter pushes the prediction above (positive SHAP value) or below (negative SHAP value) the base value, but also quantify the extent. Here, the base value corresponds to the desired experimental value for each property (see Tab. 5). Later, it will become evident how this choice of base value greatly enhances the efficiency of the optimization process. The summary plots reveal the varying influences of a_{AW} and a_{BW} on CMC, A_s, and R_g, respectively. For CMC, higher values of a_{AW} or a_{BW} are associated with negative SHAP values, implying that the DPD model with higher a_{AW} or a_{BW} tends to predict lower CMC values than the target. In contrast, for A_s, higher a_{AW} or a_{BW} correspond to positive SHAP values, suggesting a tendency to predict higher values A_s. The impact on R_g is more complicated. Although higher a_{BW} generally leads to higher R_g, the influence of a_{AW} on R_g can depend on other parameters. High a_{AW} typically leads to R_g values exceeding the target, while low a_{AW} results in R_g values both below and above the desired value. This suggests a substantial interaction effect between the two parameters in determining R_g, which will be discussed further later. Furthermore, by taking the average absolute value of the SHAP values for each input parameter, we obtain a bar graph, as shown in Fig. 9. This plot reveals the relative, global importance of each input parameter in determining the predicted property by comparing the two input parameters' overall influences on the model's output. More

R1C1

R1C1

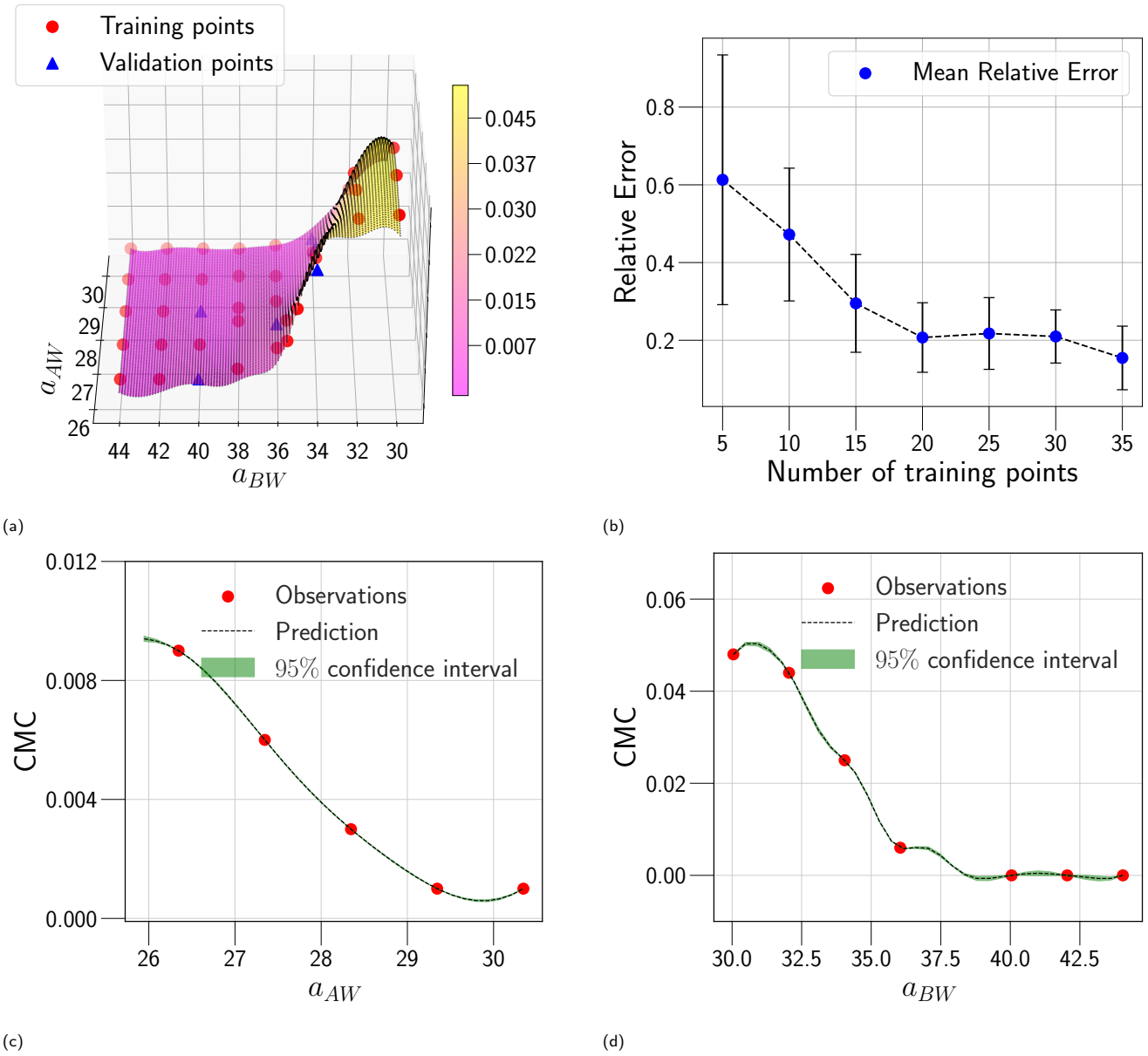


Fig. 5 GPR surrogate model predictions of CMC. (a) 3D representation of the model predictions as a function of a_{AW} and a_{BW}; (b) Convergence plot showing the mean relative error (with standard deviation as error bars) across 8-fold cross-validation, as the number of training points increases. Each fold's convergence analysis was performed independently, and the final values represent the average behavior across folds.(c) Prediction slice along a_{BW} at fixed $a_{AW} = 27.346$; (d) Prediction slice along a_{AW} at fixed $a_{BW} = 36.046$.

R2C2, R1C3, R1C4, R3C2

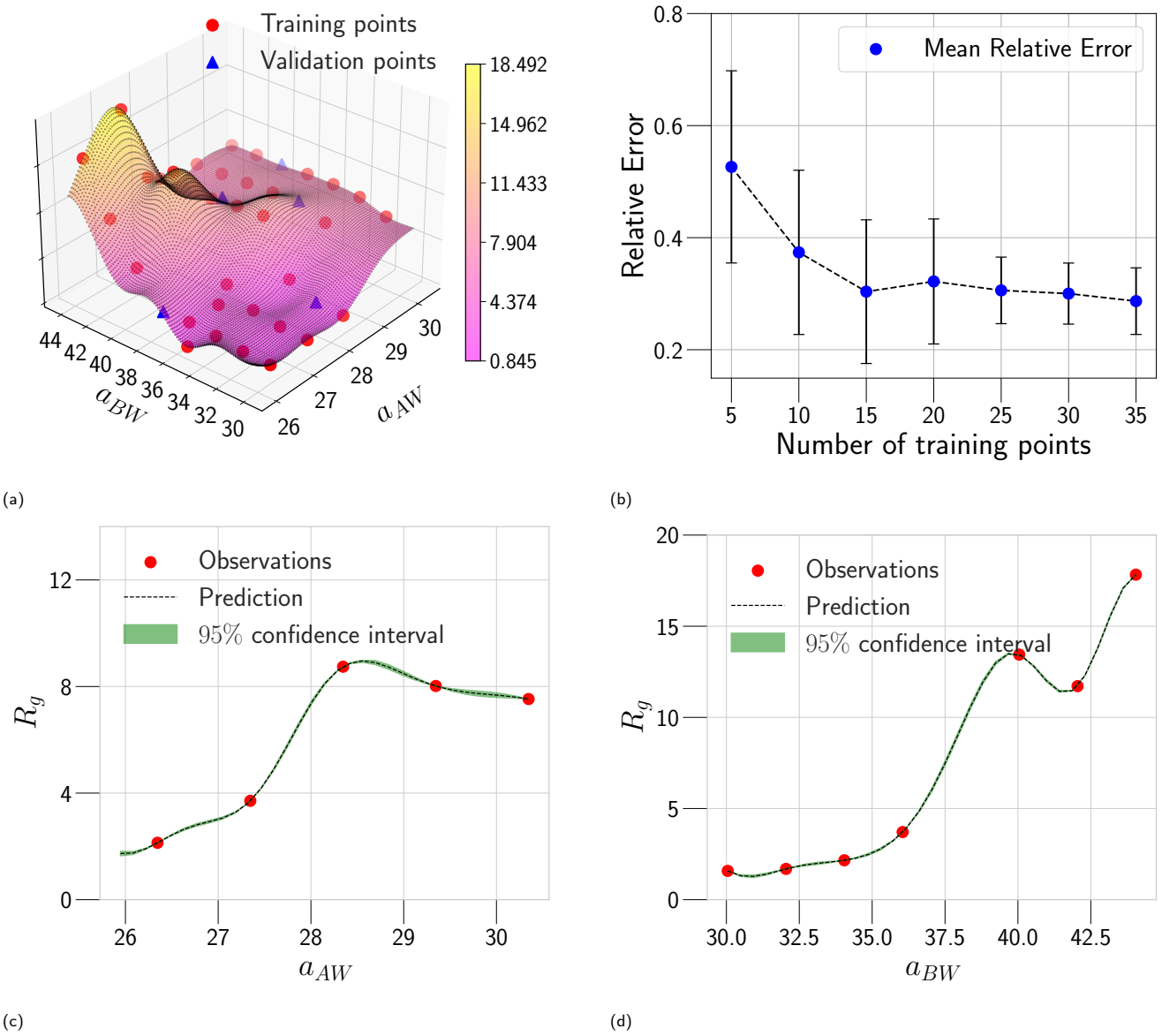
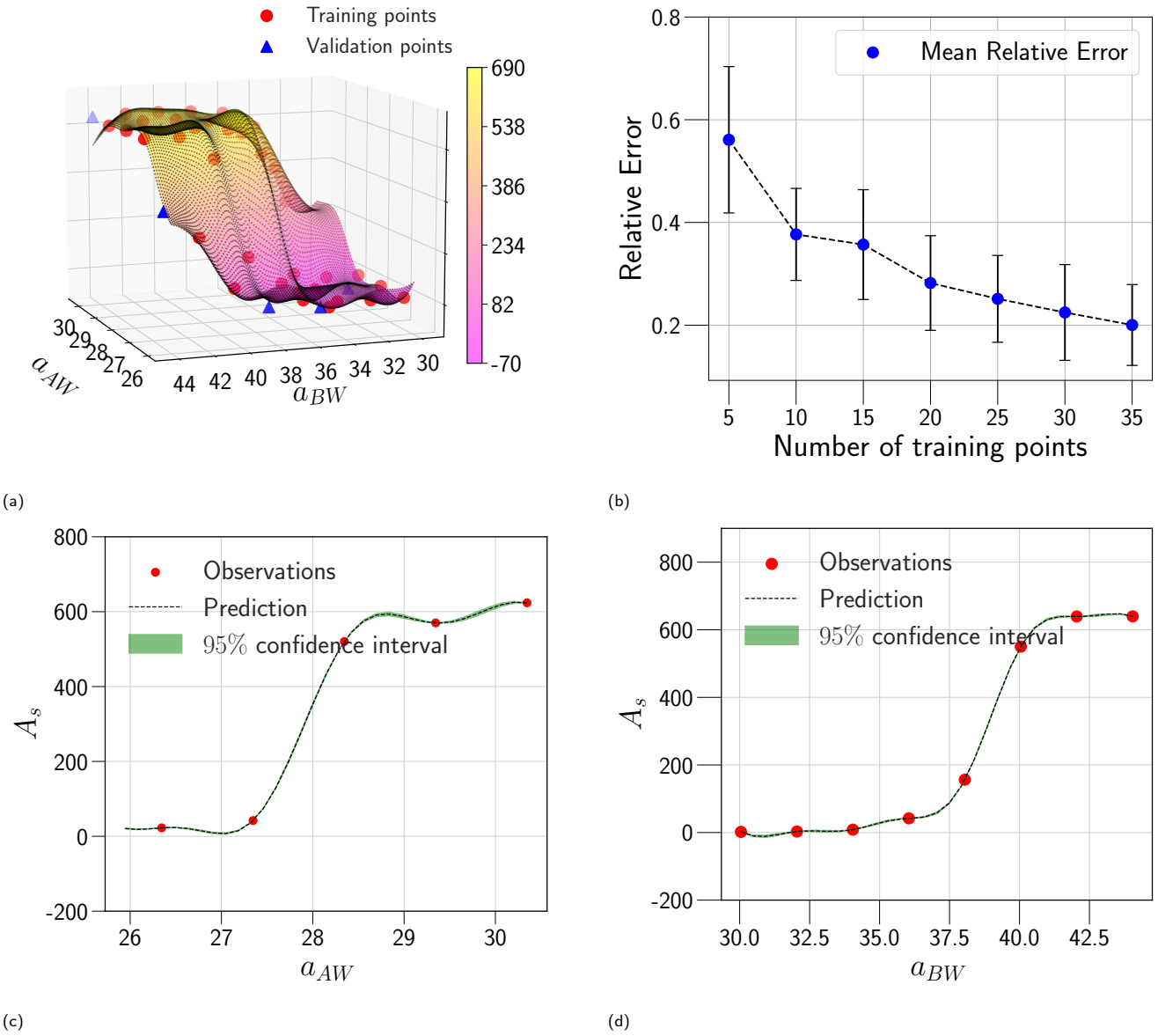


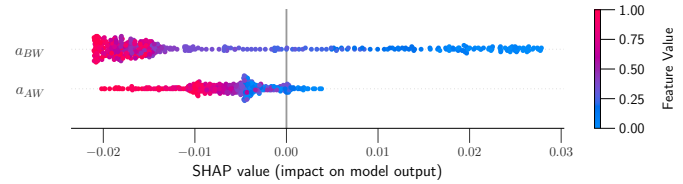
Fig. 6 GPR surrogate model predictions of R_g. (a) 3D representation of the model predictions as a function of a_{AW} and a_{BW}; (b) Convergence plot showing the mean relative error (with standard deviation as error bars) across 8-fold cross-validation, as the number of training points increases. Each fold's convergence analysis was performed independently, and the final values represent the average behavior across folds.(c) Prediction slice along a_{BW} at fixed $a_{AW} = 27.346$; (d) Prediction slice along a_{AW} at fixed $a_{BW} = 36.046$.

R2C2, R1C3, R1C4, R3C2

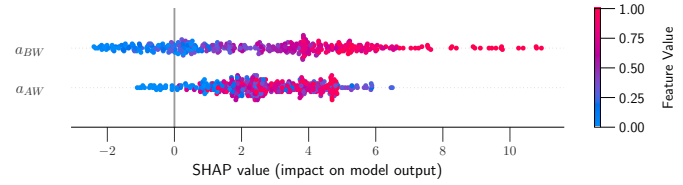


R2C2, R1C3, R1C4, R3C2

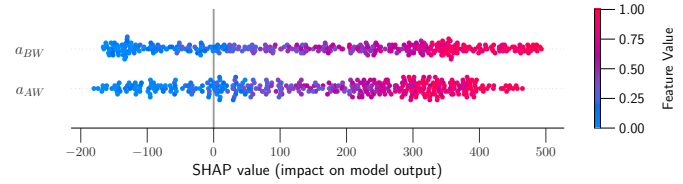
Fig. 7 GPR surrogate model predictions of A_s. (a) 3D representation of the model predictions as a function of a_{AW} and a_{BW}; (b) Convergence plot showing the mean relative error (with standard deviation as error bars) across 8-fold cross-validation, as the number of training points increases. Each fold's convergence analysis was performed independently, and the final values represent the average behavior across folds.(c) Prediction slice along a_{BW} at fixed $a_{AW} = 27.346$; (d) Prediction slice along a_{AW} at fixed $a_{BW} = 36.046$.



(a) CMC



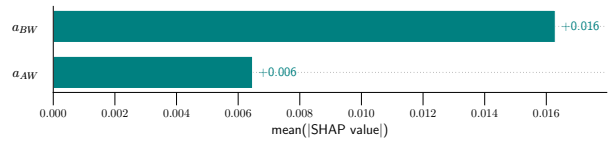
(b) R_g



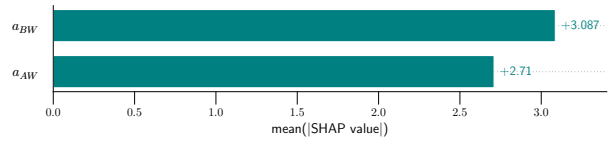
(c) A_s

Fig. 8 SHAP summary plot for each predicted property. The color bar indicates the scaled feature (input parameter) value readings.

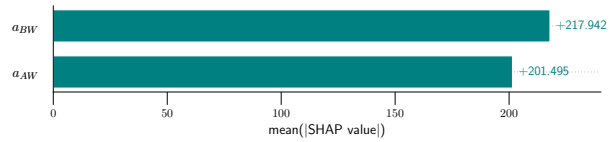
specifically, a_{BW} plays a significantly more prominent role in determining CMC values compared to a_{AW}. However, both a_{AW} and a_{BW} influence R_g and A_s, with a_{BW} slightly more influential.



(a) CMC



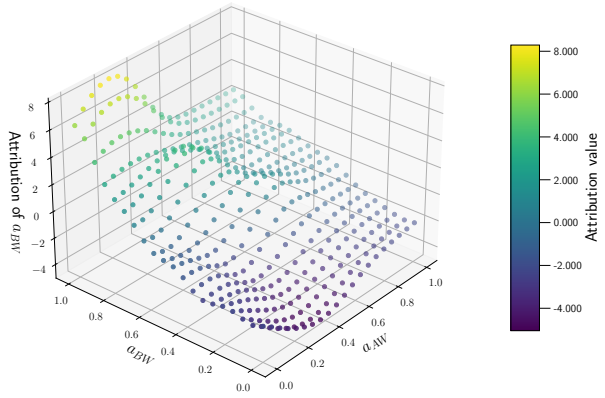
(b) R_g



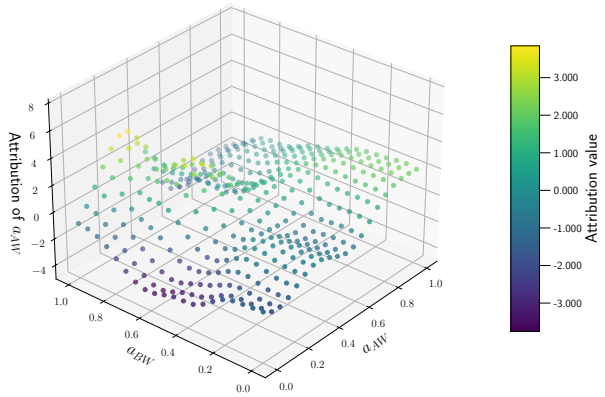
(c) A_s

Fig. 9 The relative importance of a_{AW} and a_{BW} for each predicted property, assessed using the average absolute value of the SHAP values, i.e., mean(|SHAP value|).

Finally, to obtain a deeper insight into the interactions between the two input parameters, we present the SHAP partial dependence plot. Based on the foregoing discussion, a potentially significant interaction between the two input parameters was identified in the determination of R_g. Thus, the SHAP partial dependence plot for R_g is examined, as shown in Fig. 10, from which we can see how the attribution of each parameter depends on the magnitudes of the other parameters. As depicted in Fig. 10a, although high values of a_{BW} can push R_g above its target value, the influence of a_{BW} on R_g is more pronounced when a_{AW} is lower. As a_{AW} decreases, its contribution to R_g shifts from a more monotonic dependence on a_{BW} to a more sensitive and non-linear one, as indicated by Fig. 10b.



(a) Variation of the attribution of a_{BW} with respect to a_{AW} and a_{BW}



(b) Variation of the attribution of a_{AW} with respect to a_{AW} and a_{BW}

Fig. 10 SHAP partial dependence plots for R_g.

These findings validate our intuitive understanding of how the different parameters can affect the predicted property. For instance, a_{BW} is expected to be the most influent parameter in the CMC, since it controls the repulsive force between the hydrophobic block, PPO, and water. The PPO block is the most important factor in the formation of micelles. The tendency of the Pluronic copolymers to associate spontaneously when present in dilute aqueous solutions emanates from the minimization of the contact between their hydrophobic parts and the aqueous polar environment. The micellization of Pluronic copolymers in water

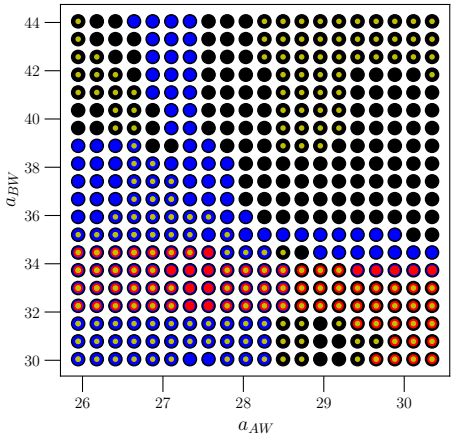
is strongly endothermic; it is an entropic gain, which can be attributed to the increased entropy of the water molecules as they are released from the hydrating of the PPO blocks[38]. In the DPD model, this tendency is quantified by the repulsive parameter a_{BW}. The higher a_{BW}, the greater the tendency to self-organize into micelles. Furthermore, the presence of PEO makes this effect more pronounced. In addition, a_{AW} has an impact, as it controls the repulsive force between PEO and water. Thus, the higher a_{AW}, the more pronounced the effect of a_{BW} on the CMC value. Specifically, the higher a_{AW}, the lower the predicted CMC. The effects of a_{AW} and a_{BW} on A_s, are strictly related to the same considerations made for CMC. The higher a_{AW} and a_{BW}, the greater the tendency of macromolecules to self-assemble, and thus the higher the average aggregation number of clusters. In contrast, the effects of a_{AW} and a_{BW} on R_g are less straightforward. Although R_g scales with the aggregation number, as larger aggregates tend to form larger micelles, it is also influenced by the compactness of the structure. The compactness, in turn, depends not only on the number of macromolecules within the aggregate but also on other model parameters and their interactions.

The relationships and causal mechanisms between the input parameters and the predicted properties revealed by the SHAP analysis can also inform the optimization process to efficiently identify optimal parameter values for target properties. Using the analysis presented in Fig. 8, we can narrow down the regions in the parameter space of a_{AW} and a_{BW} where the optimal values may reside. Consider, for example, Fig. 8b. Firstly, we note that the base value corresponding to the experimentally observed R_g from Tab. 5 is 2.28 nm. This means that at any sampled data point (a^*_{AW}, a^*_{BW}), the value of R_g is given by: $R_g(a^*_{AW}, a^*_{BW}) = 2.28 +$ (SHAP value of a^*_{AW}) + (SHAP value of a^*_{BW}). Then we observe in Fig. 8b that the negative arm of a_{AW} extends to about -1. If the SHAP value of $a_{BW} > 1$, then R_g cannot reach the base value 2.28. Hence, the optimal value should occur where $a_{BW} \leq 1$. Similarly, the negative arm of a_{BW} extends to about -2.1, so the optimal value cannot occur in the domain where the SHAP value of $a_{AW} > 2.1$. The data points within these identified regions for a_{AW} and a_{BW} are highlighted as small yellow circles in Fig. 11a. Following the same analysis for CMC and A_s, we then identify the overlap region where (a_{AW}, a_{BW}) may simultaneously achieve the target values for all three properties, as indicated by green circles in Fig. 11b. It is worth highlighting that this approach assumes that the regions in the (a_{AW}, a_{BW})-space, where the functions CMC(a_{AW}, a_{BW}), $R_g(a_{AW}, a_{BW})$, and $A_s(a_{AW}, a_{BW})$ match experimental values, are either overlapping or sufficiently close to each other. Given the close relationship between these functions, it is indeed expected that an optimal set of parameters (a_{AW}, a_{BW}) would allow for the simultaneous reproduction of all three target experimental values. For example, it is conceivable that micelles (or supramolecular structures, in general) may exhibit the same CMC and A_s while differing in compactness, leading to a radius of gyration (R_g) that is either more or less compact than the experimental value, as discussed later in 4.3. Furthermore, there may be situations where these regions do not overlap. In such cases, the absence of overlap between the target regions suggests that the parameter space should be expanded, for example, by including other interaction parameters, or that the current model resolution is insufficient to accurately reproduce the experimental behavior.

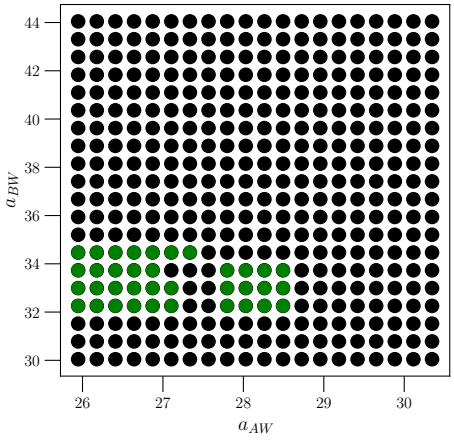
R1C1

R2C6

As such, SHAP analysis can substantially confine the initial search space for the subsequent grid-search optimization. Instead of searching the full parameter space, i.e., $25.9 < a_{AW} < 30.3$ and $30 < a_{BW} < 44$, we can start our search only in $25.9 < a_{AW} < 29$ and $32 < a_{BW} < 36$. Besides confining the initial search space, SHAP analysis also guides the refinement of grid searches. In particular, for where SHAP values are sensitive to the other parameter, e.g., at low a_{AW} as indicated by Fig. 10, finer grid searches are needed.



(a) Regions where the optimal values of a_{AW} and a_{BW} may occur for the target CMC, A_s, and R_g, respectively.



(b) Regions where the optimal values of a_{AW} and a_{BW} may occur for achieving the target CMC, A_s, and R_g simultaneously.

Fig. 11 Confining the search space for a_{AW} and a_{BW} using SHAP analysis. Here, black circles represent the data points sampled for performing the SHAP analysis. Red, blue, and yellow circles represent the data points falling into the regions identified by SHAP analysis for CMC, A_s, and R_g, respectively. Green circles mark the overlap of these three regions.

4.3 Optimization results

By leveraging the availability of GPR surrogate models for CMC, R_g and A_s based on the input parameters of the DPD model, the optimization step is performed using an adaptive grid search method. The average execution time of GPR is $0.001s$. The total

execution time to identify the minimum of the penalty function is contingent on the grid resolution. In this analysis, we used an adaptive resolution approach that progressively increases within the area of interest to improve efficiency, as shown in Figs. 12. With the aid of SHAP analysis, this optimization process can be further accelerated, as shown in Fig. 13, where the grid search starts from a smaller space determined by SHAP analysis (as discussed in the previous section). Compared with Fig. 12, we can see that both the iteration count and the execution time per iteration are substantially reduced, highlighting the practical utility of this interpretable ML technique. **R1C1** Finally, the optimal parameter values identified for all Pluronic systems are summarized in Tab. 6 and visually represented in Fig. 14. The optimal value of a_{AW} increases with increasing percentage content of EO weight and molecular weight of PO. This trend is even more pronounced for a_{BW}. This behavior in the model parameters, aimed at capturing the experimental target, aligns with the considerations discussed in Sect. 2.2. At a fixed EO content, a higher PO content corresponds to a lower experimental CMC. To achieve a smaller CMC, both a_{AW} and a_{BW} must be increased, with a more significant adjustment required for a_{BW}. Similarly, with a fixed PO content to achieve a smaller CMC, both a_{AW} and a_{BW} must be increased. A comparison of the experimental and computed targets is provided in Fig. 15. It demonstrates a strong correlation between the experimental and simulated CMC values, while the agreement remains good for A_s. For R_g, some points align well, while others show minor deviations. This suggests that optimization is more effective for CMC and A_s. The results indicate that the CMC is highly sensitive to the chosen model parameters, which allows us to capture its variation by only adjusting a_{AW} and a_{BW}. In contrast, R_g is influenced by additional parameters that were kept constant in this study, which could be considered in future work.

Table 6 Optimized DPD model parameters for the considered Pluronics.

Pluronic Name	Composition	DPD model composition	a_{AW}	a_{BW}
L64	$EO_{13}PO_{30}EO_{13}$	$A_3B_9A_3$	25.946	35.072
P65	$EO_{20}PO_{30}EO_{20}$	$A_5B_9A_5$	26.341	35.120
P104	$EO_{27}PO_{63}EO_{27}$	$A_4B_{18}A_4$	26.472	40.894
F68	$EO_{82}PO_{31}EO_{82}$	$A_{18}B_9A_{18}$	27.152	35.228

5 Conclusions

This study demonstrated that the top-down parameterization of Pluronic systems significantly benefited from the application of interpretable machine learning techniques. The presented workflow outlined a successful data-driven strategy for accurately determining model parameters tailored to different Pluronic systems. In particular, we developed a GPR model as an effective surrogate for DPD simulations. GPR's strength lies in its ability to handle small datasets and provide uncertainty estimates, informing the process of creating sufficient training data for desired confidence levels. By modeling the relationship between input parameters and output properties, the GPR model significantly reduced the need for extensive DPD simulations, enabling fast predictions on the properties of various Pluronic systems and facilitating parameter optimization. Furthermore, a post-hoc analysis via SHAP on the GPR model, which seeks an interpretable approximation of the GPR model through evaluating the additive feature attribution, granted deeper insights on how each input parameter impacts the output properties. This in turn allowed for a more informed exploration of the parameter space and, combined with GPR, significantly accelerated the optimization process to determine the optimal DPD model parameters that yield experimentally desired properties for diverse Pluronic systems. **R2C3** The model parameters obtained are specifically tuned to reproduce spherical micelles at certain temperature and concentration, ensuring accurate prediction of key properties such as the critical micelle concentration (CMC), radius of gyration, and aggregation number. It is not intended to capture the full phase diagram or to be directly transferable across different systems. This work laid the groundwork for a general framework aiming to streamline the parameterization process across various Pluronic systems (such as with different EO/PO block lengths and molecular weights) and conditions (such as differing temperatures). The goal is to generalize the parameterization process by incorporating additional DPD model input parameters and extending it to various Pluronic systems and conditions, including differing temperatures. GPR is capable of multi-variate regression, and its performance for high-dimensional surrogate modeling can be further enhanced by coupling it with dimension reduction techniques, such as proper orthogonal decomposition (POD)[39,40], to reduce training and prediction costs. The interpretability provided by SHAP analysis will become increasingly valuable for high-dimensional models. **R1C2**

Exploring alternative ML techniques, such as deep neural networks (DNNs), could also be advantageous. DNNs generally demand more training data, and hence careful design of the network architecture and training process is needed to reduce this demand. For instance, inspired by multi-task learning strategies in natural language processing[41], we could partition DNN hidden layers into fixed layers, which learn universal features, and tunable layers, which adapt to specific systems or conditions. This approach would significantly reduce the amount of data needed for training on new systems or conditions. Another promising avenue involves leveraging Graph Neural Networks (GNNs), drawing inspiration from the literature[42,43]. By representing the DPD system explicitly as a graph, where individual beads and their pairwise interactions are encoded as nodes and edges (with edge attributes describing interaction parameters), respectively. Such a GNN-based surrogate model could potentially learn more complex relationships and generalize more effectively to unseen Pluronic systems and/or conditions. We can achieve interpretability for the constructed surrogate model by either applying the model-agnostic post-hoc SHAP analysis or by building the neural network with intrinsic explainability through generalized additive models[44] with structured interactions[45,46]. **R3C1, R3C2, R3C4**

Author contributions

Nunzia Lauriello: Conceptualization (equal); Data curation (lead); Formal analysis (equal); Investigation (equal); Methodology (equal); Software (lead); Writing – original draft (lead). **Deekshith Naidu Ponnana**: Formal analysis (supporting); Investigation (supporting); Methodology (equal); Software (equal);

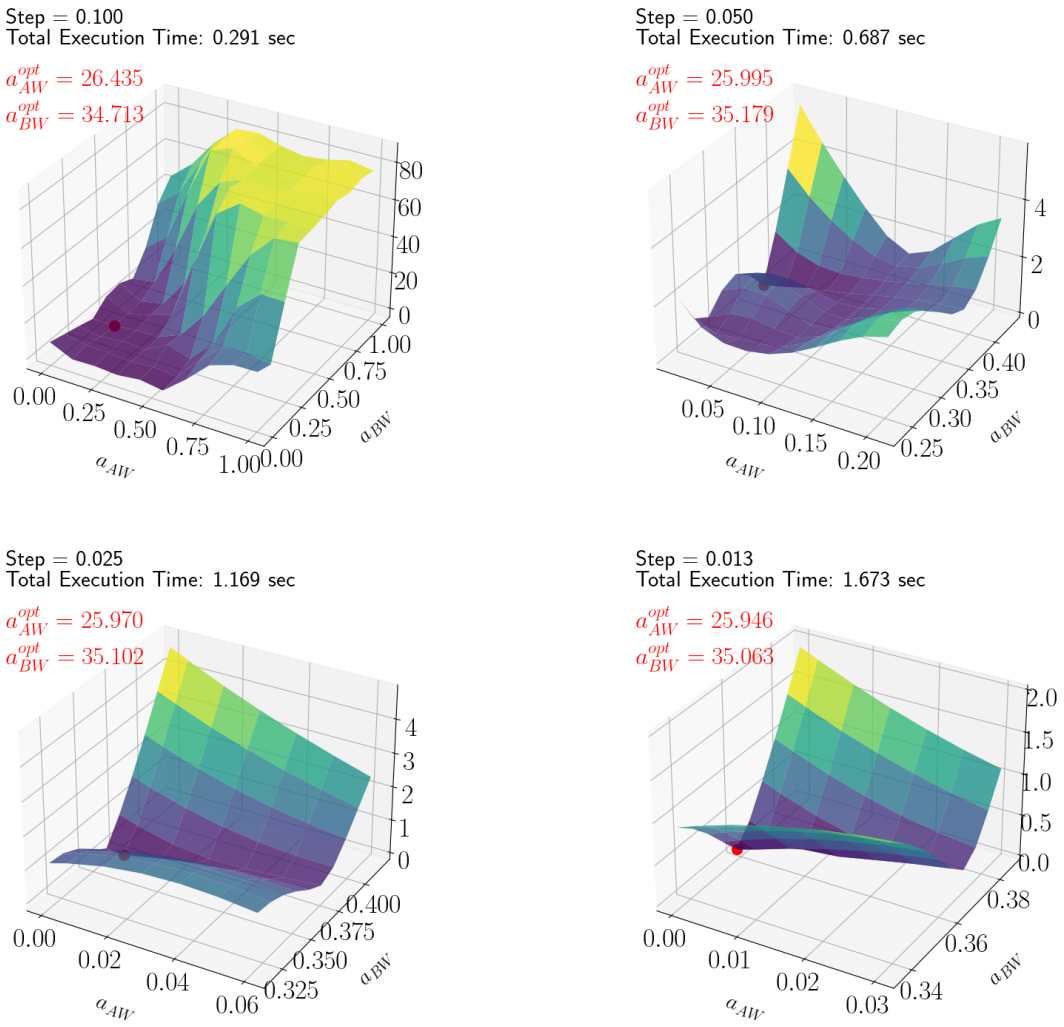


Fig. 12 Adaptive grid search results for optimizing L64 parameters, illustrating the progression across different refinement steps.

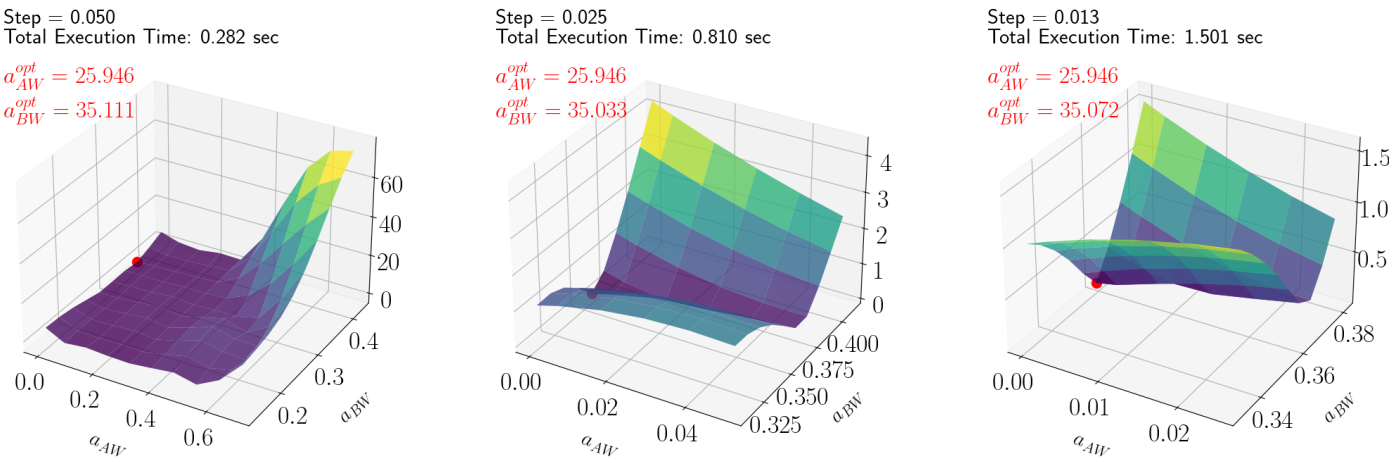


Fig. 13 SHAP assisted adaptive grid search for optimizing L64 parameters, showing results at different step sizes.

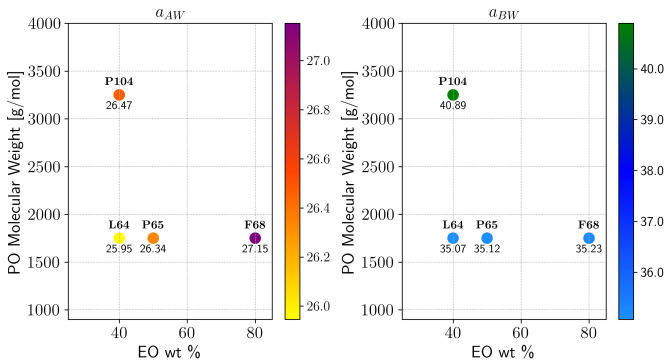


Fig. 14 Optimized DPD model parameters for the considered Pluronics systems, defined by PO molecular weight and EO content.

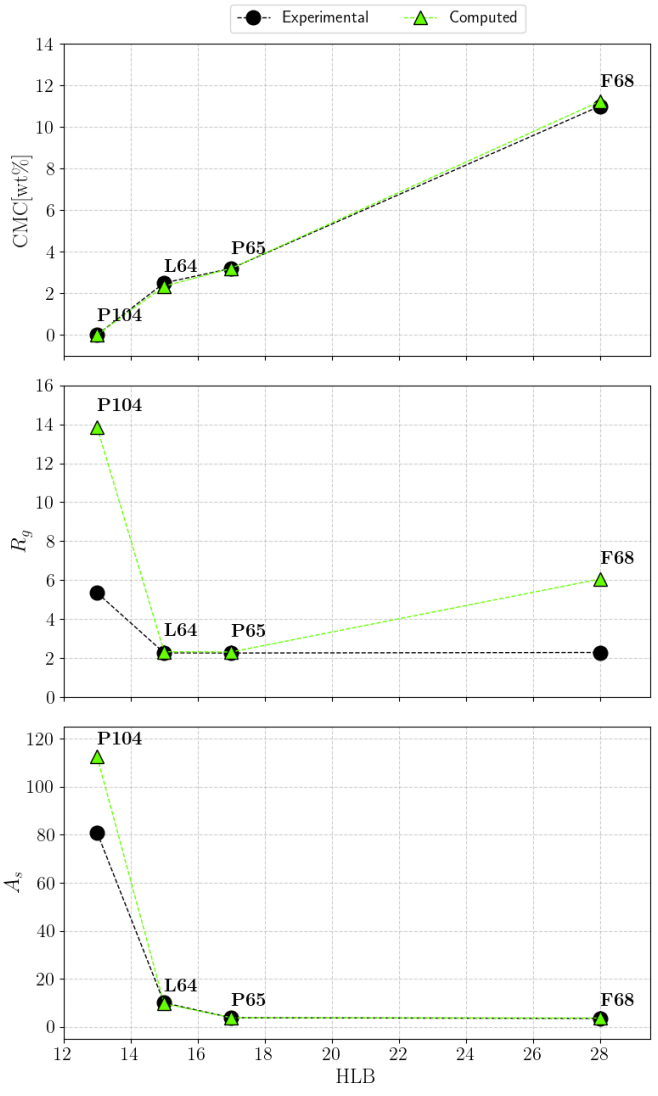


Fig. 15 Comparison of experimental targets and computed results obtained through optimized parameters, for the Critical Micellar Concentration (CMC), radii of spherical aggregates (R_g), and aggregation numbers (A_s) of the considered Pluronics at 30°C and low concentrations.

Writing – original draft (supporting). **Zhan Ma**: Conceptualization (equal); Methodology (equal); Writing - review & editing (equal). **Karel Šindelka**: Methodology (equal); Software (equal); Writing – original draft (supporting). **Antonio Buffo**: Conceptualization (equal); Supervision (equal); Writing - review & editing (equal). **Gianluca Boccardo**: Conceptualization (equal); Supervision (equal); Writing - review & editing (equal). **Daniele Marchisio**: Conceptualization (equal); Funding acquisition (lead); Project administration (lead); Supervision (equal); Writing - review & editing (equal). **Wenxiao Pan**: Conceptualization (equal); Funding acquisition (lead); Supervision (equal); Writing - original draft (supporting), review & editing (equal).

Conflicts of interest

There are no conflicts to declare.

Data availability

The code used can be found online at https://github.com/mulmopro/InterpML-DPD-Param and the corresponding dataset can be accessed https://doi.org/10.5281/zenodo.15040138.

Acknowledgements

Computational resources were provided by HPC@POLITO. We acknowledge the CINECA award under the ISCRA initiative, for the availability of high-performance computing resources and support. The financial support from ICSC (Centro Nazionale di Ricerca in High Performance Computing, Big Data and Quantum Computing, funded by European Union - NextGenerationEU) is also gratefully acknowledged. This study was carried out within the «Non-equilibrium self-assembly of structured fluids: a multiscale engineering problem» project – funded by European Union – Next Generation EU within the PRIN 2022 program (D.D. 104 - 02/02/2022 Ministero dell'Università e della Ricerca). W. P., D. N. P., and Z. M. acknowledge the funding support provided by the US Army Research Office Grant No. W911NF2310256, Defense Established Program to Stimulate Competitive Research (DEPSCoR) Grant No. FA9550-20-1-0072, and Army Research Laboratory contract No. W911NF2320073 under the Center for Extreme Events in Structurally Evolving Materials. This manuscript reflects only the authors' views and opinions and the Ministry cannot be considered responsible for them. The authors also wish to thank the three anonymous reviewers for their constructive comments and suggestions, which helped improve the quality of the manuscript.

Notes and references

1 J. Herzberger, K. Niederer, H. Pohlit, J. Seiwert, M. Worm, F. R. Wurm, and H. Frey, Chemical Reviews **116**, 2170 (2016).

2 A. V. Kabanov, E. V. Batrakova, and V. Y. Alakhov, Journal of Controlled Release **82**, 189 (2002).

3 N. A. D. Spirito, N. Grizzuti, and R. Pasquino, Physics of Fluids **36**, 111302 (2024).

4 M. J. Newman, M. Balusubramanian, and C. W. Todd, Advanced Drug Delivery Reviews **32**, 199 (1998).

5 D. Y. Alakhova and A. V. Kabanov, Molecular Pharmaceutics **11**, 2566 (2014).

6 K. C. de Castro, J. C. Coco, Érica Mendes dos Santos, J. A. Ataide, R. M. Martinez, M. H. M. do Nascimento, J. Prata, P. R. M. L. da Fonte, P. Severino, P. G. Mazzola, A. R. Baby, E. B. Souto, D. R. de Araujo, and A. M. Lopes, Journal of Controlled Release **353**, 802 (2023).

7 P. Alexandridis and T. A. Hatton, Colloids and Surfaces A **96**, 1 (1995).

8 P. Alexandrilis and B. Lindman, *Amphiphilic Block Copolymers - Self-Assembly and Applications* (Elsevier, 2000).

9 N. A. Di Spirito, N. Grizzuti, M. Casalegno, F. Castiglione, and R. Pasquino, International Journal of Pharmaceutics **644**, 37647976 (2023).

10 A. Pitto-Barry and N. P. E. Barry, Polymer Chemistry **5**, 3281 (2014).

11 N. U. Khaliq, J. Lee, S. Kim, D. Sung, and H. Kim, Pharmaceutics **15**, 2102 (2023).

12 G. Wanka, H. Hoffmann, and W. Ulbricht, Macromolecules **27**, 4145 (1994).

13 A. V. Kabanov, I. R. Nazarova, I. V. Astafieva, E. V. Batrakova, V. Y. Alakhov, A. A. Yaroslavov, and V. A. Kabanov, Macromolecules **28**, 2303 (1995).

14 R. D. Groot and P. B. Warren, The Journal of Chemical Physics **107**, 4423 (1997).

15 K. Procházka, Z. Limpouchová, M. Štěpánek, K. Šindelka, and M. Lísal, Polymers **14**, 404 (2022).

16 S. Trément, B. Schnell, L. Petitjean, M. Couty, and B. Rousseau, The Journal of Chemical Physics **140**, 134113 (2014).

17 S. Izvekov and B. M. Rice, The Journal of Chemical Physics **140**, 104104 (2014).

18 H. Mori, Progress of Theoretical Physics **33**, 423–455 (1965).

19 R. Zwanzig, The Journal of Chemical Physics **33**, 1338 (1965).

20 H. Droghetti, I. Pagonabarraga, P. Carbone, P. Asinari, and D. Marchisio, The Journal of Chemical Physics **149**, 184903 (2018).

21 R. Pasquino, H. Droghetti, P. Carbone, S. Mirzaagha, N. Grizzuti, and D. Marchisio, Soft Matter **15**, 1396 (2019).

22 A. Prhashanna, S. A. Khan, and S. B. Chen, The Journal of Physical Chemistry B **119**, 572 (2015).

23 J. B. Avalos, M. Lísal, J. P. Larentzos, A. D. Mackie, and J. K. Brennan, Physical Chemistry Chemical Physics **21**, 24891 (2019).

24 P. Flory, *Principles of Chemistry Cap.12* (Cornell University Press, Itacha, NY, 1953).

25 S. M. Lundberg and S.-I. Lee, in *Advances in Neural Information Processing Systems*, Vol. 30 (Curran Associates, Inc., 2017).

26 P. J. Hoogerbrugge and J. M. V. A. Koelman, Europhysics Letters (EPL) **19**, 155 (1992).

27 P. Español and P. Warren, Europhysics Letters (EPL) **30**, 191 (1995).

28 M. P. Allen and F. Schmid, Molecular Simulation **33**, 21 (2006).

29 P. Durchschlag and H. Zipper, Progress in Colloid and Polymer Science **94**, 20 (1994).

30 B. A. C. van Vlimmeren, N. M. Maurits, A. V. Zvelindovsky, G. J. A. Sevink, , and J. G. E. M. Fraaije, ACS Publications , 646 (1999).

31 N. Lauriello, M. Lísal, G. Boccardo, D. Marchisio, and A. Buffo, The Journal of Chemical Physics **161**, 034112 (2024).

32 P. Vanya, J. Sharman, and J. A. Elliott, The Journal of Chemical Physics **150**, 064101 (2019).

33 H. Rezaei and H. Modarress, Chemical Physics Letters **620**, 114 (2015).

34 K. Šindelka and M. Lísal, Molecular Physics **119**, e1857863 (2021).

35 D. C. Liu and J. Nocedal, Mathematical Programming **45**, 503 (1989).

36 M. T. Ribeiro, S. Singh, and C. Guestrin, in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16 (Association for Computing Machinery, New York, NY, USA, 2016) p. 1135–1144.

37 G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning: With Applications in R*, 1st ed., Springer Texts in Statistics (Springer, New York, 2013).

38 P. Alexandridis, Current Opinion in Colloid Interface Science **1**, 490 (1996).

39 Z. Ma, S. Wang, M. Kim, K. Liu, C.-L. Chen, and W. Pan, Soft Matter **17**, 5864 (2021).

40 Z. Ma and W. Pan, Computer Methods in Applied Mechanics and Engineering **373**, 113495 (2021).

41 S. Chen, Y. Zhang, and Q. Yang, ACM Computing Surveys **56**, 1 (2024).

42 Z. Ma, Z. Ye, and W. Pan, Computer Methods in Applied Mechanics and Engineering **400**, 115496 (2022).

43 A. Aminimajd, J. Maia, and A. Singh, Soft Matter **21**, 2826 (2025).

44 R. Agarwal, L. Melnick, N. Frosst, X. Zhang, B. Lengerich, R. Caruana, and G. E. Hinton, Advances in neural information processing systems **34**, 4699 (2021).

45 Z. Yang, A. Zhang, and A. Sudjianto, Pattern Recognition **120**, 108192 (2021).

46 S. Ibrahim, G. Afriat, K. Behdin, and R. Mazumder, Advances in Neural Information Processing Systems **36**, 61158 (2023).