

# Building Surrogates for Materials Properties with Adaptive Learning

A. Diaw,<sup>1</sup> M. McKerns,<sup>1</sup> I. Sagert,<sup>1</sup> L. G. Stanton,<sup>2</sup> and M. S. Murillo<sup>3</sup>

<sup>1</sup>*Los Alamos National Laboratory, Los Alamos, New Mexico 87545*

<sup>2</sup>*Department of Mathematics and Statistics, San José State University, San José, California 95192*

<sup>3</sup>*Department of Computational Mathematics, Science and Engineering,*

*Michigan State University, East Lansing, Michigan 48824*

(Dated: June 5, 2021)

The rapid and accurate calculation of microscopic simulation data for materials is computationally expensive. For this reason, parametric fits and interpolated tables are typically used to avoid repeat simulations. Learning from precomputed data obviates the need to recompute data or prescribe a particular functional form. We present an adaptive learning method to automate the efficient production of surrogates for computationally expensive microscopic model calculations. Our adaptive approach uses directed sampling, in a reinforcement machine learning workflow, where the learning of new surrogates is performed in an optimization loop with an objective function that scores a candidate surrogate’s quality. The hyperparameter optimization is augmented by an ability to request new model evaluations *on-the-fly* and adapt to the resulting new simulation data. Applications to various thermodynamic properties are presented, demonstrating that adaptive learning coupled to feedback from directed sampling can be used to produce accurate surrogate models.

## I. INTRODUCTION

In material science [1], large scale macroscopic simulations rely on closure information based on well-devised mathematical model that correctly reflects essential microphysical processes [2–5]. A fundamental challenge in this field is building microscopic models valid for modelling a wide variety of phenomena, including phase transition, complex mixtures, and shock waves, with a level of confidence similar to that provided by high-fidelity microscopic simulations [6]. For complex systems, the use of expensive microscopic data such as molecular dynamics [6–8] or Monte Carlo [9, 10] simulations is essential for an accurate description of complex flow phenomena. This approach however requires a large amount of microscopic simulations therefore can quickly become impractical. A possible strategy is to use surrogate models as substitutes to the expensive microscopic simulations.

In previous works [11, 12], we used active learning to iteratively build surrogate models of a fine scale response. Roehm et al. [13] used kriging to construct surrogates of stress fields and communicate the results to a fine-scale code that solves the macro-scale conservation laws for elastodynamics. In both of these examples, the surrogate models are not guaranteed to be valid across the parameter space of the simulations. Thus, at every time step the surrogate models are checked for validity, and new fine scale simulations are launched whenever the uncertainty metric exceeds a threshold. Therefore, the performance of these multiscale approaches can be significantly hampered by the quality of the sampling approach.

Beyond the choice of the learning model, the sampling approach is critical to the quality of the surrogate model. Standard sampling methods known as one-shot where the space is sampled to generate all points at once often under or over-sampled the space. In contrast, the direct sampling approach aims to place sample points in highly nonlinear space, targeting critical points on the response

surface. In this paper, we demonstrate that using directed sampling to discover the critical points of the response surface will rapidly drive learned surrogates to be valid for any yet-unforseen data. We demonstrate that, if our training set includes all the critical points of the response surface, we can produce a surrogate that has minimal model error. If we use a very small number of optimizers to perform our critical point search, we may not find all of the critical points in a single feedback loop, and may need a second, or third, loop to discover all of the critical points. We demonstrate the benefits of a combined approach of directed sampling and adaptive learning in the efficient on-the-fly learning of valid surrogate models for synthetic data. We integrate our adaptive sampling algorithm in *mystic* [14], a software providing robust uncertainty quantification in the context of sampling, optimization, interpolation, and machine learning.

The paper is organized as follows. Section II introduces the different components of our adaptive learning framework, and briefly discusses methods we use to construct our objective function, interpolation method, and validation strategy. In Section III, we demonstrate the utility of our framework in the construction and evaluation of surrogates for three different materials applications. Finally, some conclusions are presented in Section IV.

## II. ADAPTIVE LEARNING METHODOLOGY

### A. Interpolation: radial basis function (RBF)

To train and learn a valid surrogate for the data generated by the simulations, we use radial basis functions (RBFs) [15]. The utility of RBFs arises from their universal function approximation capabilities [16], and their connection to single hidden-layer feed-forward neural networks with non-sigmoidal nonlinearities [17].

Let us assume  $y(\mathbf{x})$  is an arbitrary function of vector  $\mathbf{x}$

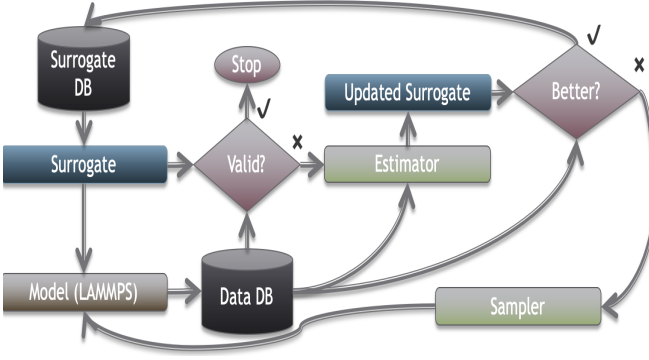


FIG. 1. Adaptive learning workflow designed to produce a valid surrogate for data from expensive microscopic simulations. Execution starts after evaluation of new points are requested. When a surrogate is evaluated, check for validity against simulation results stored in the data database. If the surrogate is valid, then stop. Otherwise, interpolate to produce a new surrogate. If the new surrogate reduces the model error, then store the new surrogate in the surrogate database and check for validity. If the new surrogate does not reduce the model error, then use a sampler to request new simulations and store the new points in the data database. Repeat until a valid surrogate is found.

represented on a subset of  $\mathbb{R}^n$ , and that the value of  $y$  at  $N$  input vectors  $\{\mathbf{x}^j\}$  ( $j = 1, \dots, N$ ), is known. We seek to find a surrogate function  $\hat{y}(\mathbf{x})$  with the lowest possible number of the evaluations satisfying [18–20]

$$\hat{y}(\mathbf{x}^j) = y(\mathbf{x}^j) \text{ for all } j = 1, \dots, N. \quad (1)$$

Using a RBF  $\phi(r)$ , the interpolated function can be written as:

$$\hat{y}(\mathbf{x}) = \sum_{j=1}^N \beta_j \phi(d(\mathbf{x}, \mathbf{x}^j)), \quad (2)$$

where  $\beta_j$  are coefficients to be determined, and  $d(\mathbf{x}, \mathbf{x}^j) = \|\mathbf{x} - \mathbf{x}^j\|$  is the Euclidean distance between an arbitrary vector  $\mathbf{x}$  and  $\mathbf{x}^j$ . The values of the coefficient vector  $\boldsymbol{\beta} = [\beta_1, \beta_2, \dots, \beta_N]^T$  are determined using the interpolation condition Eq. (1) by solving the linear system,  $\mathbf{M}\boldsymbol{\beta} = \mathbf{Y}$ , where  $\mathbf{M}$  is an  $N \times N$  symmetric matrix with elements  $M_{ij} = \phi(\|\mathbf{x}^i - \mathbf{x}^j\|)$ , and  $\mathbf{Y} = [y(\mathbf{x}^1), y(\mathbf{x}^2), \dots, y(\mathbf{x}^N)]^T$ .

Many options exist for the choice of RBF  $\phi$  and are readily available within *mystic*. In this work, we use thin-plate RBF to interpolate the data. To prevent issues due to singular matrix  $\mathbf{M}$ , we add a very small amount of Gaussian noise to the input data. For case of interest here, we will assume that the centers are given at the data points to enforce desirable features in the interpolated function, such as the value of a peak or a value at a boundary. This choice also has the desirable numerical advantage of only requiring a matrix inversion to obtain the weights.

## B. Selection of new points: adaptive sampling

The adaptive learning workflow designed to produce a valid surrogate for data from an expensive simulation is shown in Fig. 1. When the surrogate is requested to evaluate one or more new points, the surrogate is retrieved from the surrogate archive, and then the points are evaluated by calling **surrogate**. Concurrently, the function **estimate** checks the surrogate for validity against the point stored in the database. In our applications, we will implement **Valid** as a calculation of the model error between a function of the known data  $\theta(\text{data}|\gamma)$  and a single candidate **surrogate**  $\Phi$ , as :

$$\min_{\gamma} (\theta(\text{data}|\gamma) - \Phi(g^*))^2 \quad (3)$$

where  $\gamma$  are hyperparameters of the learned surrogate  $\theta(\text{data}|\gamma)$ , and  $\Phi(g^*)$  is an instance of the model with a fixed candidate response function  $g^*$ .

A surrogate is deemed valid when:  $\max[\text{distance}] \leq 10^{-6}$ ,  $\sum[\text{distance}] \leq 10^{-3}$ . If the surrogate is determined to be valid, then we stop. However, if the surrogate does not meet the conditions for validity, we interpolate the data in the data database to produce an updated surrogate. If the updated surrogate has a smaller distance compared to the current surrogate, then we store the updated surrogate in the surrogate database, and keep iteratively improving until we find a valid surrogate or we hit the maximum number of attempts.

If we exceed the maximum number of attempts to use interpolation to produce a valid surrogate, then we request more data from the model. A sampler is used to request new model evaluations, which are populated into the data database. Note that adding new data points to the data database does not ensure that an existing surrogate will become *more* valid; however, if the new data causes the surrogate to fail the validity criteria, the entire process starts over and repeats until a valid surrogate has been found.

The active learning process will “pause” whenever a surrogate is deemed valid, regardless of the amount of data in the database. When the data database is sparsely populated, we expect that any new data point will likely trigger a surrogate update. To reduce the number of sampling iterations required, we generally chose a large number of samplers used in each sampling step. We will discuss different sampling strategies in Section II B, highlighting the impact of the strategy on the efficiency of the calculation.

We build a sampler to provide directed sampling of the **model** response surface  $G$ . We distinguish *directed sampling* from *undirected sampling* in that undirected sampling follows the same sampling strategy for every point (e.g., a sample from a given distribution), while directed sampling uses an optimizer to direct the sampling toward a goal. We argue that while generating a set of undirected samples is faster, using directed sampling can yield a valid surrogate much more quickly. As we

are interested in generating an accurate surrogate for the model response surface  $G$ , we hypothesize that knowing the critical points of  $G$  would enable a highly accurate approximate response surface  $g$  to be generated. As a first approximation, we will use an ensemble of optimizers to search for minima and maxima of  $G$ . We extend the ensemble optimizers from *mystic* to sampling, producing sampler objects that configure an ensemble of optimizers to generate new points to run the model at.

We use a lattice sampler which initializes **npts** optimizers, each starting at the center of a grid point in parameter space. The sampler has the ability to sample until some condition is met, among other methods. To **sample**, is to take a minimal step with the underlying optimizer, effectively providing an asynchronous interface to sampling.

In our adaptive learning workflow, the sampler launches **npts** optimizers to seek the minima of  $G$ , and **npts** optimizers to seek the maxima of  $G$ , in parallel. While sampling can run continuously and is fundamentally independent of the interpolation, we request a new interpolation in our workflow whenever the sampler resets (as determined by the settings described above). The interpolator always interpolates using the data contained in the data database at the time the interpolation is requested.

### III. CASE STUDIES

In this section, we will assess the performance of our method. We then use our framework to find accurate surrogate models for two types of problems: equation of state (EOS) calculations of dense nuclear matter and RDFs from expensive MD simulations. For the latter type of problem regarding RDFs, we will consider three different systems with increasing complexity of the parameter space: the OCP model with a two parameter space, the LJ fluid with a three parameter space and finally an extension of the OCP, the binary-ionic mixture (BIM) model with a five parameter space.

#### A. Performance tests with Benchmark functions

We now test the performance of our directed sampling method against non-directed sampling. First, we explore the execution time for each of these methods to find accurate surrogate models for standard benchmark functions (see **mystic.models**). Features of these functions (name, dimensionality) and inputs used for the test are described in Table I. Simulations were performed on the Darwin cluster at Los Alamos National Laboratory on a 36 core Skylake microarchitecture. The last two columns of Table I show the average CPU time obtained for each function over 10 runs for both sampling strategies. We observe that our directed sampling method is 9 to 25 times faster than the more standard undirected sampling, even

Function	ndim	Bounds	Undirected [s]	Powell	Nelder Mead
Ackley	2	[-5,5]	134.19	0.2	0.1
Branins	2	[-10,20]	142.48	0.4	0.06
Rosenbrock	3	[-3, 3]	156.59	0.46	0.1
Michalewicz	5	[0, 3]	141.08	0.52	0.11
Hartmann	6	[-1, 1]	235.74	0.57	0.08
Rosenbrock	8	[-6, 6]	457.56	0.44	0.04

TABLE I. Comparison between directed sampling (DS) and undirected sampling (US), using benchmark functions, showing that DS is 9 to 25 times faster than US. We present the CPU time, averaged over 10 runs, to find a surrogate model that satisfies our quality metric of  $\text{max\_distance} < 10^{-6}$  and  $\text{sum\_distance} < 10^{-3}$  using US. The time to solution is given in seconds for US, while for DS, it is given as a fraction of the US time. The sampler for DS is configured to run until all of the solvers in the ensemble have terminated.

when the optimizers are not tuned. This result holds regardless of the dimensionality of the objective function.

To assess the performance of our framework, we used the Hartmann 6-Dimensional function:

$$f(\mathbf{x}) = - \sum_{i=1}^4 \alpha_i \exp \left( - \sum_{j=1}^6 A_{ij} (x_j - P_{ij})^2 \right), \quad (4)$$

where the coefficients  $\alpha_i$ ,  $A_{ij}$  and  $P_{ij}$  can be found in Appendix C. We evaluated this function on the hypercube  $-1 < x_i < 1$ , for all  $i = 1, \dots, 6$ . The function has a global minimum,  $f(x) = -3.322$ , that occurs at  $x = [0.20169, 0.15001, 0.4768, 0.2753, 0.311, 0.6573]$ . Figures 2 (a) and (b) show convergence of the parameters and cost function, per iteration, for the Hartmann 6-dimensional function using adaptive learning with directed sampling. The plots show that convergence to  $10^{-4}$  is generally reached in less than two iterations. The results depicted in Table I used a termination condition that required convergence to  $10^{-4}$  over 10 iterations, where the sampler waited until all optimizers have terminated. We estimate that with some tuning of the sampler and the underlying optimizer termination conditions (*e.g.*, requiring convergence at  $10^{-2}$  over 2 iterations), we could potentially gain another order of magnitude improvement in speed over undirected sampling.

#### B. Dense nuclear matter EOS

Our first application to test the framework is modeling an equation of state (EOS) for dense nuclear matter that contains a phase transition (PT). Reliable hadronic models for nuclear matter exist up to baryon number densities  $n_b$  of about twice nuclear saturation density  $n_0 \sim 0.16 \text{ fm}^{-3}$  and at asymptotically high densities of  $n_b \gg 40 n_0$  [21, 22]. While at low densities and temperatures  $T$ , nuclear matter is composed of neutrons and protons, for high values of  $n_b$  and  $T$ , it is expected to

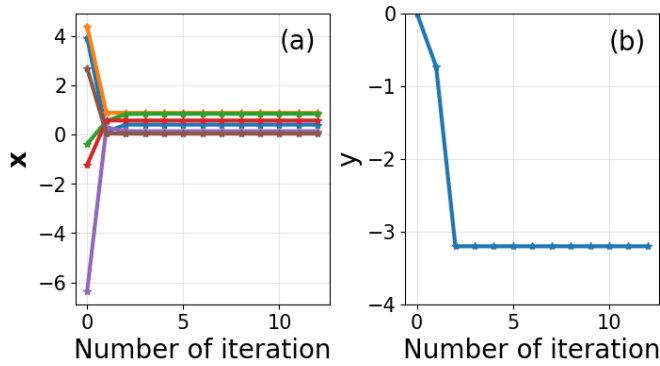


FIG. 2. Plotting convergence of the parameters (a) and cost convergence (b) as function of the iteration number, for the Hartmann 6 test function Eq. (2), shows that convergence is reached within the first two iterations. Thus, with some tuning to the sampler, we expect an additional order of magnitude efficiency can be achieved for directed sampling. The left plot shows the input parameters  $\mathbf{x} = [x_i]$ , where  $i = 1, \dots, 6$ , as function of number of iteration, and the right plot shows the cost as function of number of iterations.

undergo a PT to a phase composed of deconfined quarks and gluons. This quark-hadron PT is studied experimentally in heavy-ion collisions on Earth and might be present in the interior of neutron stars [23]. However, there are large uncertainties regarding the critical temperatures and densities for the onset of the PT.

Numerical studies of matter during heavy-ion collisions, core-collapse supernovae, neutron star merger events and within the interior of neutron stars, require the usage of a nuclear EOS [24]. The most common approach to create an EOS over a large density range is to select realistic hadronic and quark models and connect them either via a Maxwell or a Gibbs construction to describe the PT [25]. The Gibbs construction assumes that conservation laws are fulfilled globally in the quark-hadron mixed phase which results in pressure being a smooth function of the density. For the Maxwell construction, only baryon number is conserved globally, while other conservation laws like electric charge neutrality are fulfilled locally in the quark and hadronic phases. [25–27]. Neither of these constructions is currently ruled out, but the Maxwell construction usually leads to a more extreme behavior during the PT, leading to a pressure plateau in the so-called quark-hadron mixed phase.

For astrophysical simulations of core-collapse supernovae and neutron star mergers, nuclear matter EOS tables are used with thermodynamic quantities being functions of  $n_b$ , the proton fraction  $y_p$  and the temperature  $T$  [24]. The construction of these tables is time-intensive with many points in the  $(n_b, y_p, T)$  space, while their implementation in computational fluid dynamics (CFD) problems requires interpolation and inversion routines. A new approach to facilitate the usage of EOS tables is the construction of analytic fitting functions which can then be included in the numerical simulations instead of

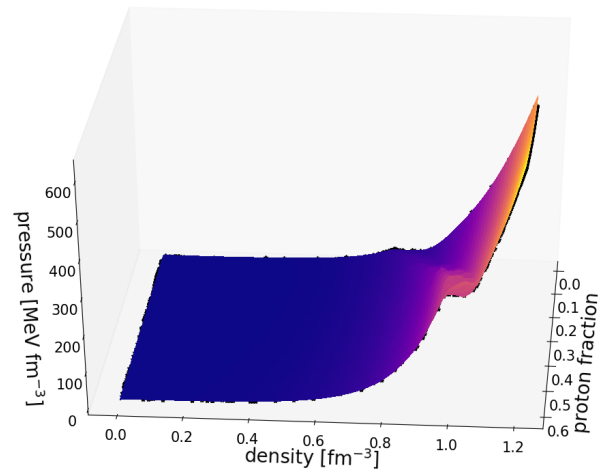


FIG. 3. Simulations (dots) and predicted values (surface) of the equation of state for quark matter. The initial search domain chosen as  $0.04 < \text{density} [\text{fm}^{-3}] < 1.2$  and  $0. < \text{proton fraction} < 0.6$  was sampled with a lattice sampler. We used 40 Nelder Mead solvers and the tolerance for a valid surrogate were set to  $\text{max\_distance} < 10^{-6}$  and  $\text{sum\_distance} < 10^{-3}$ . A valid surrogate model of the EOS that correctly describes the plateau region was found after 2839 function evaluations.

an EOS table [28]. While promising, this method has yet to be shown to be able to accurately model extreme features like high-density PTs.

To demonstrate our framework, we implement a quark-hadron PT into a simple nucleonic model that is frequently used in astrophysics and nuclear physics [29, 30]. Here, the nucleonic EOS is derived from the Skyrme-Hartree-Fock self-consistent mean-field model [31]. The energy of the system is determined as the expectation value of an effective nuclear Hamiltonian which contains the zero-range Skyrme nuclear interaction [32]. For high-density neutron star interiors at zero or low-temperatures, nuclear matter can be treated as degenerate and infinite with constant density. This greatly simplifies the expression for the Skyrme energy density functional. In addition, the many-body state of the system can be expressed as a Slater determinant of uncorrelated plane wave states from lowest momentum up to the Fermi momentum. As a result, the energy per baryon of nuclear matter composed of neutrons and protons with densities  $\rho_n$  and  $\rho_p$ , respectively, can be written in some an analytic form as shown A.

Given the energy or energy per baryon, other thermodynamic properties, such as pressure, can then be determined by standard relations [31]. The PT is modeled by the Maxwell construction, while quark matter is described by the MIT Bag model, where quarks are non-interacting fermions with a negative confinement pressure, the so-called bag constant [27, 33]. The latter model ensures that quarks are confined into neutrons and protons at low densities. Quark matter in our approach uses a bag constant of 170 MeV. It is composed of up, down,



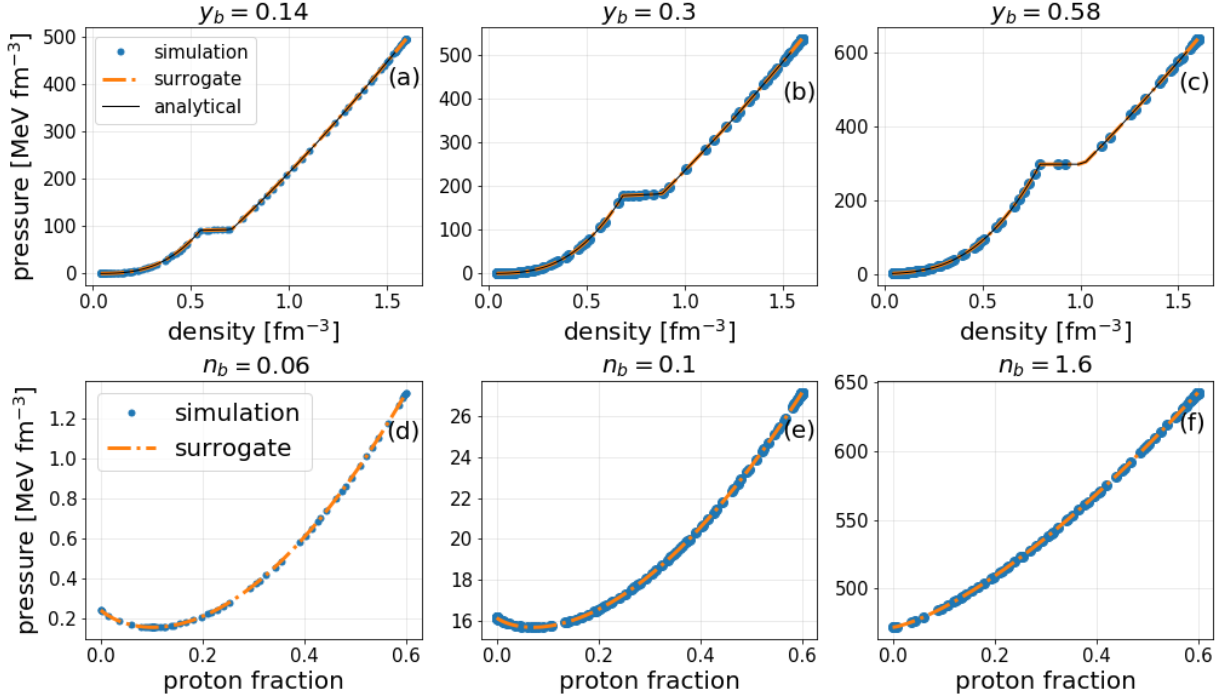


FIG. 4. Quark matter pressure as function of density and proton fraction. The dotted curves were obtained using the adaptive learning method with 40 Nelder Mead solvers and RBFs as the interpolator with a tolerance for a valid model set to  $\text{max\_distance} < 10^{-6}$  and  $\text{sum\_distance} < 10^{-3}$ . The dots are the results from more expensive simulation and the black line corresponds to EOS analytical formula. We found a good agreement between the simulations and the surrogate models across wide range of the parameter space.

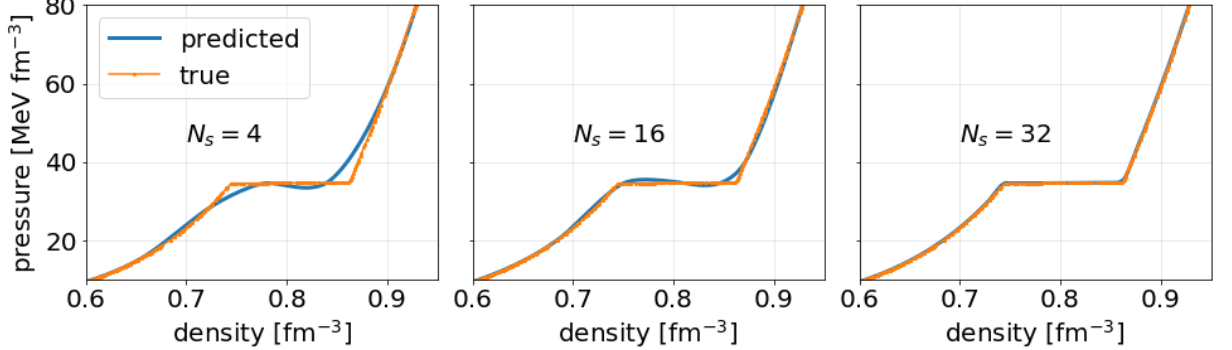


FIG. 5. Quark matter pressure as function of density for a given proton fraction. The orange curve shows the results from more expensive simulation and the blue curve represents prediction using our methodology. As we increase the number of solvers ( $N_s$ ) directed to finding the critical points in the discontinue, the accuracy of the surrogate obtained with thin-plate RBF improves.

and strange quarks, where we assume that the masses of up and down quarks are negligible in comparison to their chemical potential, while the strange quark mass is set to 150 MeV. Although nucleonic and quark matter are given by simple models, our intention here is to demonstrate the ability of our framework to use expensive EOS data, even in the presence of a PT, to find a surrogate model that can be directly used in high-fidelity CFD codes. EOS tables in astrophysical simulations usually have three degrees of freedom:  $n_b$ ,  $y_p$  and  $T$  [24], but

for simplicity, we will model a system where the pressure is a function of  $n_b$  and  $y_p$  only, and temperature effects are negligible, which is a reasonable assumption for systems such as neutron stars interiors. We take the quark-hadron EOS for  $0.04 \text{ fm}^{-3} \leq n_b \leq 1.6 \text{ fm}^{-3}$  and  $0 \leq y_p \leq 0.6$  as a model in the adaptive learner to find a surrogate function. The results of this test is illustrated in Figure 3, which shows the entire  $n_b$ - $y_p$  plane. The pressure plateau of the PT is clearly visible with critical density for the onset of the mixed phase moving to

higher values for increasing proton fraction as discussed in [27]. Figures 4 (a)-(f) give a more detailed view of the EOS by showing the pressure profiles for fixed  $y_p$  and  $n_b$ , respectively. It is seen from the figure that no systematic errors arise in the predicted values of the pressure with either proton fraction or density. These simulations were performed using 40 solvers, and a surrogate model obtained with thin-plate RBF.

The RBFs will reproduce the phase transition accurately if enough of the inflexion points of the response surface are sampled. As the optimizers discover all the critical points in the surface (here, the points in the discontinuity), the smooth functions also begin to reproduce these discontinuities. We illustrated this point in Fig. 5.

Let us now move to the second test problem. Here, we use the large-scale molecular dynamics code LAMMPS to generate dataset; radial distribution function (RDF) of neutral and charged liquids.

### C. Surrogate models of MD simulations: RDF

RDF describes interparticle correlations within isotropic materials by averaging over relevant atomic or molecular coordinates; it is an important fluid characteristic, necessary for analyzing x-ray Thomson scattering experiments [34], investigating protein interactions with cell membranes [35], and examining shock-induced phase transitions [3]. In addition, the RDF plays a key role in perturbative fluids theories, as many aspects of thermodynamic properties in fluids can be expressed in terms of it. The RDF is also useful in investigating some inhomogeneous fluids structure [36], for example in dense plasmas, mixing phenomena [6], and enhancement of nuclear reaction rates [37]. Furthermore, recent developments in kinetic theory [38] and hydrodynamics [39] models for dense plasmas highlight the importance for RDF models. Because of the fundamental role of RDFs in our understanding of physical phenomena in fluids, it has been the focus of numerous studies.

The Lennard-Jones (LJ) model is among the most widely used models for neutral liquids, and numerous fits to its RDF has been given. Early work by Goldman [40] argues for as few parameters as possible for ease of use and to avoid overfitting to noise in the data; fits were made to 87 tables with 108 parameters. Later, Matteoli and Mansoori [41] suggested a considerably simplified form with only 21 parameters and two forms (small and large separation  $r$ ), allowing for the possibility of extending the fits to mixtures. Later, Morsali *et al.* [42], reiterate that fits should not have too many parameters, and with improved simulation data, provide a fit with only 11 parameters, again with two functional forms valid at small and large separations.

For strongly coupled plasma studies, the one-component plasma (OCP) is the simplest model used. The OCP is a system of particles with charge  $Ze$  inter-

acting through a repulsive Coulomb potential

$$V(r) = \frac{Z^2 e^2}{r}, \quad (5)$$

and a uniform, neutralizing background. Here,  $Z$  is the charge number of a given particle, and  $e$  is the fundamental charge. The statistical properties of the OCP can be described in terms of a single parameter, the Coulomb coupling parameter  $\Gamma = (Z^2 e^2)/(aT)$ , where  $a = (4\pi n/3)^{-1/3}$  is the Wigner-Seitz radius with  $n$  being the total ionic number density, and  $T$  is the temperature in energy units (note that the charge  $q$  is represented here in Gaussian-cgs units). Rogers *et al.* [43] employed Monte Carlo data to generate the OCP structure factor in tabular form, which requires a special request (and cost) for distribution. Brettonet and Derouiche [44] provided a fit to the OCP structure factor in terms of only two parameters, one of which was found to be a constant. However, this fit lacked the accuracy of other approaches. Desbiens *et al.* [45] have parametrized the RDF of the OCP using molecular dynamics (MD) data, where their fit model was motivated by the Matteoli and Mansoori form [41]. To achieve high accuracy, it was necessary to fit weak and strong coupling functional forms separately, with five and nine parameters, respectively. Because there is a Fisher-Widom (FW) line [46–49], which delineates the transition to oscillatory behaviour in the RDF, different functional forms are needed. These examples illustrate the challenges with tabulating or fitting data to high accuracy, and because these challenges are for a model system with a single parameter  $\Gamma$ , it is not clear that they generalize to more complex systems such as binary ionic mixtures (BIM), ternary ionic mixtures (TIM) or Lennard-Jones mixtures where we have additional parameter space.

RDFs were computed with MD for three systems to explore both the type (*i.e.*, range and repulsive/attractive) of interaction and the role of dimensionality. In all cases, the MD code LAMMPS [50] was used to produce the data with standard techniques. The equations of motion were first integrated in the canonical ensemble, with constant particle number, volume, and temperature maintained using a Nosé-Hoover thermostat, over  $10^5$  timesteps, to establish thermodynamic equilibrium at the desired temperature. Then, the production runs were carried out in the micro-canonical ensemble with  $t = 1000 \omega_p^{-1}$ , where  $\omega_p^{-1}$  is a characteristic oscillation period for the system. In order to obtain good surrogate models, it is crucial to reduce statistical fluctuations, which was handled with large particle numbers ( $N = 2048$ ) and long runs. The RDFs  $g(r; \mathbf{x})$  were calculated with 1024 bins in the range  $0 < r < L/2$ , where  $L$  is the simulation system size, and  $\mathbf{x}$  are the input parameters of system. Note that for binary mixtures, we have three different RDFs  $g_{ij}(r; \mathbf{x})$  where  $i$  and  $j$  are the species indexes.

The execution of our implementation works as follows. First, we configure the optimizer with the model (LAMMPS) as the objective that takes the param-

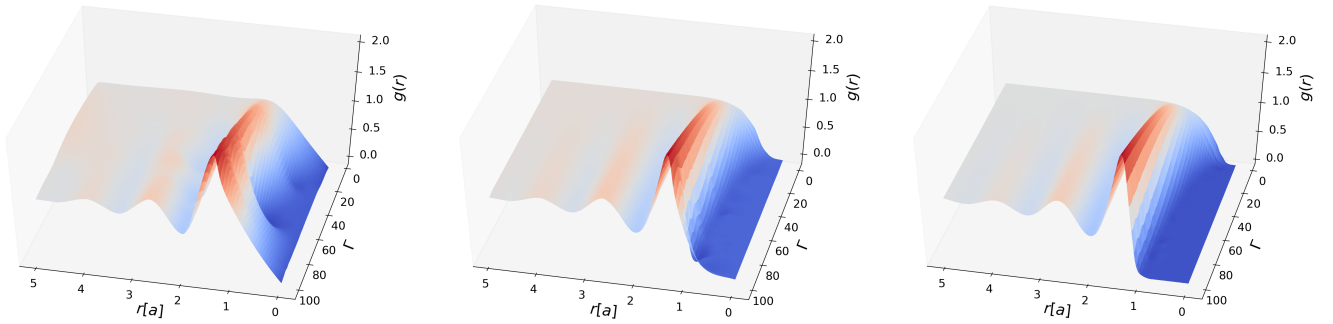


FIG. 6. Illustration of using adaptive learning to refine a surrogate for a radial distribution of OCP obtained from MD using an ensemble of Nelder Mead optimizers. The optimizers search for critical points, while an estimator learns the surface from the sampled points. The tolerance for a valid surrogate were set to  $\text{max\_distance} < 10^{-6}$  and  $\text{sum\_distance} < 10^{-3}$ . The surrogates below are generated with a single pass of 4, 8, and 16 solvers. A similar surrogate model can obtained with lower number of solvers (4) with multiple passes.

eter space, performs an MD simulation and returns some outputs values (RDFs) of the system. We use a `LatticeSampler` to sample points from a Gaussian distribution function within the prescribed bounds on each of the parameter space variables. The objective is then called to evaluate the outputs for each of the sampled points. The model evaluations are then archived to the model evaluation database. At each iteration, an ensemble of optimizers npts, are launched in search of extrema. When an extremum is found, it is archived to the extrema database. We keep sampling until all `NelderMeadSimplexSolver` instances are terminated, as defined by `NCOG(tolerance=10-4, generations=10)`. The extremum searches run continuously, where a new solver is started when one finishes.

When tolerances are met, a signal is sent to stop spawning new function evaluations. A hypercube of interpolation points that contain sampled points is built on an irregularly spaced grid with a cut-plane at each sampled point. Any duplicate of the inputs (and corresponding outputs) is dropped, and the remaining values of the inputs are sorted. We then finally interpolate using the function evaluation database to build an interpolated surrogate model.

The workflow to improve the surrogate is terminated when the quality metric is satisfied everywhere. We define the quality metric to be a function of the largest distance ( $\text{max\_distance}$ ) between each of points from the surrogate and the data such that:  $\text{max}(\text{distance}) \leq \text{maxtol}$  and  $\text{sum}(\text{distance}) \leq \text{sumtol}$ .

### 1. One-Component Plasma

The one-component plasma (OCP) is a model that assumes a system of point ions with charge  $q$  at a temperature  $T$  embedded in uniform, neutralizing background. The thermodynamic state of the OCP system is entirely determined by the coupling parameter  $\Gamma$ , so each MD

simulation computes the RDF with the form  $g(r, \Gamma)$ . Since the RDF enters into the calculations of many quantities in many forms, we also choose to calculate the related Salpeter screening potential, which is defined in terms of the RDF as:

$$\beta H(r) = \frac{\Gamma}{r} + \log [g(r)], \quad (6)$$

where  $\beta = 1/T$ , and  $r$  is in units of the Wigner-Seitz radius  $a$ . The screening potential  $H(r)$  is used to estimate the enhancement factor of nuclear reaction rates in dense plasmas.

For the OCP, the radial coordinate  $r$  was defined to be in the range  $[0, L/2]$ , where  $L$  is the simulation box size, and the coupling parameter was taken be in the range of  $[0.1, 100]$ , which spans both weakly and strongly coupled regimes. Figure 6 illustrates the sampling process of the framework with the surface of the OCP RDF  $g(r, \Gamma)$ , mapped using an ensemble of local optimizations. The surrogate surface is interpolated on a uniform grid, and as we increase the sampling, the surface resolution improves considerably, and the surrogate captures details very accurately across physical regimes. Next, Figure 7 shows both the predicted RDF and the screening potential in the  $(\Gamma, r)$  space. The blue dots are the MD results, and the green lines are the surrogate surfaces. We observe excellent agreement between the MD and the models across the parameter space. To drive the optimization throughout these simulations, we set the metric to stop where both the graphical distance is  $10^{-6}$  for the largest acceptable distance and the largest acceptable sum of all distances is  $10^{-3}$ .

To further assess the capabilities of our adaptive learning methodology and the transferability of the surrogate models, thermodynamic quantities such as the internal energy and pressure of the OCP system obtained from the cost function (MD simulations) are compared against the predicted values by the surrogate models. For OCP,

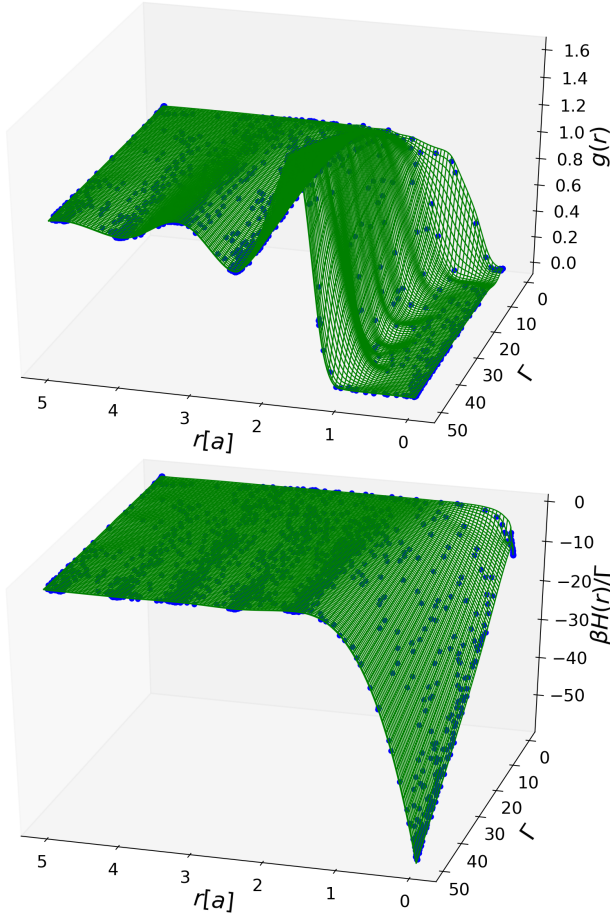


FIG. 7. MD data (blue dots) compared to the surrogate model (green surface) for the OCP system in the  $(r, \Gamma)$  space. The initial search domain was defined as  $0.1 < \Gamma < 50$  and  $0 < r[a] < 5$ , covering both the weakly and the strongly coupled regime. We used a lattice sampler with 40 Nelder Meader solvers. (a) RDF  $g(r)$  (b) screening potential  $H(r)$ . The tolerance for a valid surrogate was set to  $\text{max\_distance} < 10^{-6}$  and  $\text{sum\_distance} < 10^{-3}$ . We found accurate surrogates of the radial distribution function and screening function using 40 Nelder Mead solvers. These functions and data stored in SQLite database and can be directly called in coarse-grained codes.

the internal energy and pressure are given:

$$\frac{\beta U}{N} = \frac{3}{2} \Gamma \int_0^\infty r [g(r) - 1] dr, \quad (7)$$

$$\beta \frac{P}{n} = 1 + \frac{\Gamma}{2} \int_0^\infty r [g(r) - 1] dr, \quad (8)$$

where  $N$  is the number of particles, and  $n$  the density.

The optimization is an essential component of this framework as it drives the search for critical points (extrema) in the surface. The numbers of optimizers used for the search, the termination condition and finally the prescribed tolerance for goodness of interpolation are also very important. Figure 8 shows the results of several simulations for the energy and pressure where we vary the numbers of optimizers and the tolerance. In all these

simulations, the surrogate obtained with our framework is used in Eqs. (7)-(8) to calculate the pressure and energy. Also shown in dashed line, the results obtained with MD. Strong oscillations are observed in the predictions of both the pressure and energy in Figs. 8 (a) and (b). This can be explained by the low number of optimizers used which were not able to find all the critical points in the RDF before the termination condition was reached. Note that the number of peaks in the RDF and their amplitude are more substantial for high values of  $\Gamma$ , and it is exactly this high coupling region where the surrogates are failing. However, when we increase the number of optimizers (to 10), the surrogate functions of the pressure and internal energy agree very well with the simulations results with a score close to one. The relative deviations of the surrogate results from the ground truth (MD) are smaller than  $10^{-4}$ . Note that this trends holds across the full range of coupling parameter examined.

## 2. Lennard-Jones fluid

Increasing the complexity of the interpolation, we next consider a LJ liquid in 2 spatial dimensions. The system is composed of  $N$  particles of mass  $m$  randomly distributed in the simulation box and interacting through the LJ potential given by:

$$V(r) = 4\epsilon \left[ \left( \frac{\sigma}{r} \right)^{12} - \left( \frac{\sigma}{r} \right)^6 \right], \quad (9)$$

where  $\epsilon$  and  $\sigma$  are the two parameters of the LJ potential. Here,  $\epsilon$  measures the depth of the potential, while  $\sigma$  is the relative distance at which the interaction between pairs of two particles is zero. The LJ potential is known to give a reasonable description of interactions of atoms in rare-gas systems, which is the limit in which the attractive component is derived. For the LJ potential, we used the reduced units given by:  $T^* = T/\epsilon$ ,  $r^* = r/\sigma$  and  $\rho^* = \rho\sigma^3$ , where  $\rho^*$  is the dimensionless density. The resulting RDF for the LJ system then has the form:  $g(r^*, T^*, \rho^*)$ .

Figure 9 compares the LJ RDF from MD against the interpolated surface from thin-plate RBFs for three different values of temperature and density. It is seen from these figures that the surrogate models predict the MD data very well.

## 3. Binary Ionic Mixture

Finally, we consider the BIM model, which is an extension of the OCP allowing for two separate species embedded in a neutralizing electronic background. The ion-ion interaction potentials are modeled with the Coulomb potential

$$V_{ij}(r) = \frac{Z_i Z_j e^2}{r}, \quad (10)$$



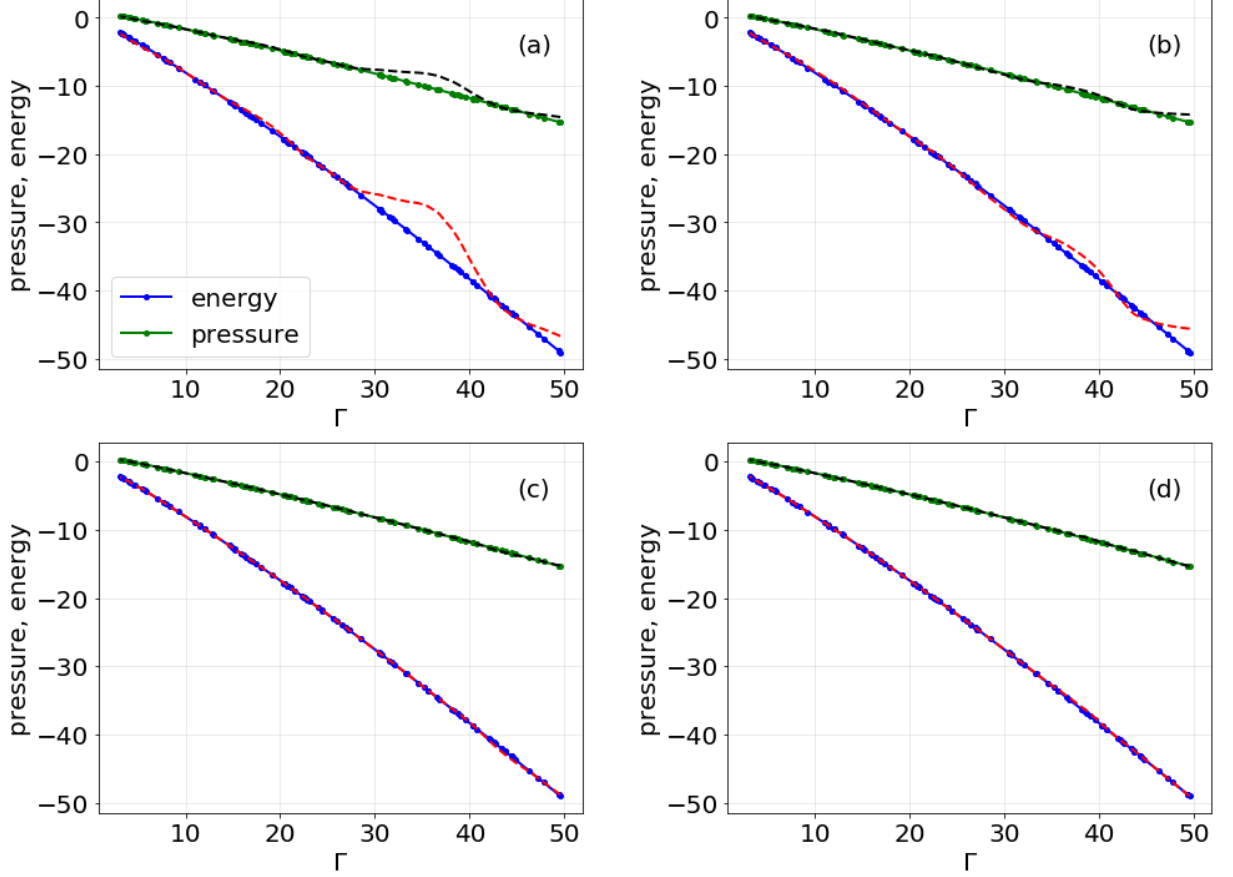


FIG. 8. Energy ( $\beta U/N$ ) and Pressure ( $\beta P/\rho$ ) for OCP system predicted by the adaptive sampler using Nelder Mead as searcher, various number of solvers ( $N_s$ ) and tolerance. (a)  $N_s=4$ ,  $\text{maxtol}=10^{-6}$ ,  $\text{sumtol}=10^{-3}$ ; (b)  $N_s=4$ ,  $\text{maxtol}=10^{-6}$ ,  $\text{sumtol}=10^{-4}$ ; (c)  $N_s=10$ ,  $\text{maxtol}=10^{-6}$ ,  $\text{sumtol}=10^{-3}$ ; (d)  $N_s=10$ ,  $\text{maxtol}=10^{-6}$ ,  $\text{sumtol}=10^{-4}$ . The dotted lines correspond to the MD simulations results while the dashed-lines are the predictions. Increasing  $N_s$  increases the likelihood that all critical points will be discovered by dynamic sampling before the termination condition is reached.

where the  $Z_i$  is the charge number of the  $i^{\text{th}}$  species. As with the OCP simulations, the long-range Coulomb forces are handled through an Ewald summation algorithm. For the binary mixture, we introduce the more general coupling parameter

$$\Gamma_{ij} = \frac{Z_i Z_j e^2}{aT}. \quad (11)$$

Letting  $\Gamma = e^2/(aT)$  be the proton-proton coupling parameter, the RDFs of the BIM model will subsequently have the form:  $g_{ij}(r; \Gamma, c, Z_i, Z_j)$ , where  $c = n_1/n$  is the concentration of one of the species.

Figure 10 shows the RDFs and the corresponding screening potential from MD simulations for three different mixtures of chemical elements present in white dwarfs [2, 51, 52], Al-Fe, Na-Mg, C-O, and the surrogate functions obtained with the adaptive learning framework. Excellent agreement was found despite the higher dimensions of the RDF parameter space. The condition for a valid surrogate was set to  $\text{max\_distance} < 10^{-6}$  and  $\text{sum\_distance} < 10^{-3}$ .

#### IV. CONCLUSIONS

We present a learning method to build an on-the-fly, highly accurate, surrogate models for expensive function evaluation. Our approach combines directed sampling and adaptive learning in a feedback loop, that automates the improvement of the model against a quality metric. With model error as our metric, we demonstrate the efficient on-the-fly learning of valid surrogate models for synthetic data generated with expensive molecular dynamics simulations. We extended *mystic* [14], a software that provides uncertainty quantification in the context of sampling, optimization, interpolation, and machine learning, to include feedback-driven adaptive controls. We applied the framework to two test problems. First, we used the equation of state for nuclear matter as cost function to learn showing excellent agreement between the predicted and model results with few number of function evaluations. For the second test, we used LAMMPS as an objective function to produce valid surrogate models of RDFs for neutral and charged systems across a

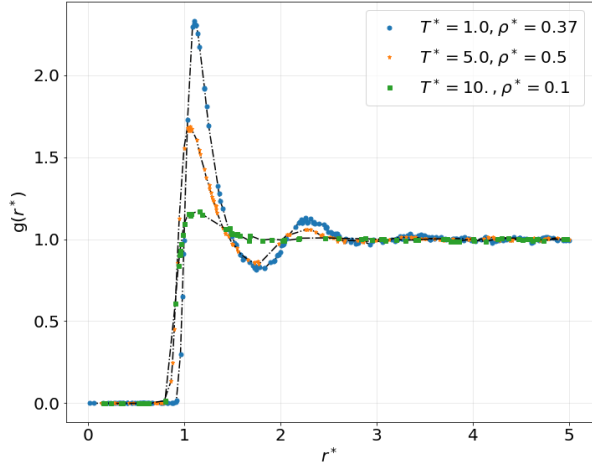


FIG. 9. RDF  $g(r)$  of 2D Lennard-Jones liquid. The dots and squares correspond to the MD data. The dot-dashed lines show the predicted results. The physical quantities are in Lennard-Jones units. For the one fluid LJ system, we observed the same agreement as to the OCP and BIM for various thermodynamic state points.

large range of thermodynamic conditions. We showed that this approach can produce high-quality surrogates at least one magnitude faster than methods utilizing standard sampling techniques. The surrogate models generated by our framework can be efficiently implemented

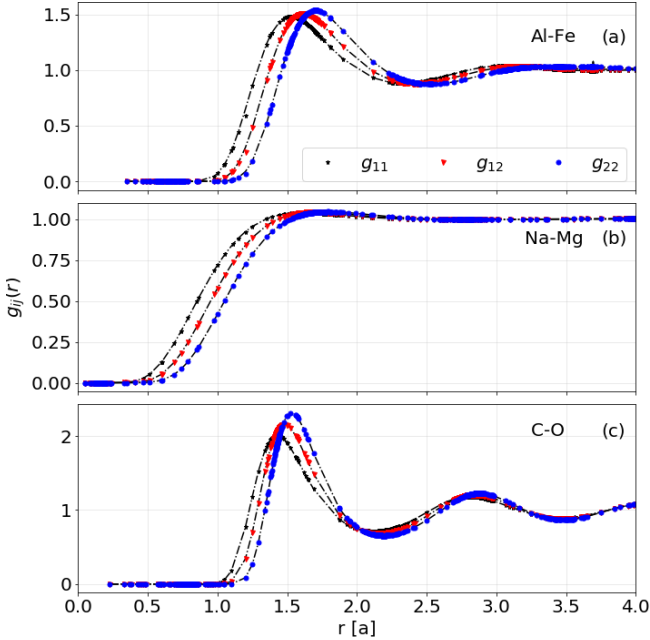


FIG. 10. RDFs  $g_{ij}(r)$  of various binary ionic mixtures: Na-Mg, Al-Fe and C-O. The dots correspond to the MD data. The lines show the interpolated results. (a)  $\Gamma = 0.1, c = 0.8, Z_1 = 11, Z_2 = 12$ ; (b)  $\Gamma = 1, c = 0.6, Z_1 = 13, Z_2 = 2$ ; (c)  $\Gamma = 2.5, c = 0.5, Z_1 = 6, Z_2 = 8$ .

in coarse-grained codes [13]. Recently, we have used active learning to demonstrate the iterative refinement of fast surrogates for molecular dynamics (MD) simulations in the context of material mixing in warm dense matter [12]. Finally, in dynamical density functional theory [53] the knowledge of fast and accurate RDFs are crucial to understanding dynamical properties of problems in the statistical mechanics of non-uniform fluids [36], spinodal decomposition in softer matter [54], cancer science [35] and plasma physics [55, 56].

In this work, we define validity of a surrogate as the model error, as in Eq. 3. This definition, while a standard one in machine learning, is sub-optimal, as it assesses the quality of the surrogate by determining the distance of the surrogate from the observed data. Thus, if one has only a small set of observed data that is not a representative sampling of all the possible scenarios (inputs and corresponding outputs of the model), then any learned surrogate will very likely become invalidated with the addition of new data. It is a weakness of traditional machine learning methods, which score purely against observed (i.e. training) data, that learned solutions are either fragile (due to overfitting of the surrogate to the training data) or inaccurate (due to averaging and other techniques to weaken the coupling of the surrogate and the training data). A better assessment of model quality considers training a surrogate with a statistical metric, such as the *expected* model error. The expected model error can be defined to take into account any knowledge about the data-generating distributions (for input and output values) as well as any uncertainty in the input and output parameters of the model. Commonly, models are non-deterministic, and thus an appropriate choice would be to either find a surrogate that is guaranteed to be accurate under uncertainty or a surrogate that is guaranteed to be robust under uncertainty. In future work, we will apply our active learning framework to produce surrogates that are guaranteed to be either accurate or robust under uncertainty, where the origin of the uncertainty is either in the model, in the input parameters, or in the representative nature of the observed data.

The code used for this work, data, surrogate functions, and tools to use them are under review for public release.

## ACKNOWLEDGMENTS

Research presented in this article was supported by Los Alamos National Laboratory under the Laboratory Directed Research and Development program (project numbers 20190005DR and 20200410DI), by the Department of Energy Advanced Simulation and Computing under the Beyond Moore's Law Program, and by the Uncertainty Quantification Foundation under the Statistical Learning program. Los Alamos National Laboratory is operated by Triad National Security, LLC, for the National Nuclear Security Administration of U.S. Department of Energy (Contract No. 89233218CNA000001).

The Uncertainty Quantification Foundation is a non-profit dedicated to the advancement of predictive science through research, education, and the development and dissemination of advanced technologies. The authors would like to thank Jeff Haack and Vikram Dharodi for very useful feedback on the manuscript. This document is LA-UR-20-24947.

### Appendix A:

The energy per baryon of nuclear matter composed of neutrons and protons with densities  $\rho_n$  and  $\rho_p$  are given by, respectively, form see :

$$\begin{aligned} \frac{E}{A}(y_p, \rho) = & \frac{3}{5} \left( \frac{\hbar^2}{2m} \rho \right)^{\frac{2}{3}} F_{5/3} \\ & + \frac{1}{8} t_0 \rho [2(x_0 + 2) - (2x_0 + 1)F_2] \\ & + \frac{1}{48} t_3 \rho^{\alpha+1} [2(x_3 + 2) - (2x_3 + 1)F_2] \\ & + \frac{3}{40} \left( \frac{3\pi^2}{2} \right)^{\frac{2}{3}} \rho^{\frac{5}{3}} \left[ [t_1(x_1 + 2) + t_2(x_2 + 1)] F_{\frac{5}{3}} \right. \\ & \left. + \frac{1}{2} [t_2(2x_2 + 1) - t_1(2x_1 + 1)] F_{\frac{8}{3}} \right], \end{aligned} \quad (\text{A1})$$

$$F_m(y_p) = 2^{m-1} [y_p^m + (1 - y_p)^m], \quad (\text{A2})$$

$$\rho = \rho_n + \rho_p, \quad y_p = \rho_p / \rho. \quad (\text{A3})$$

The parameters  $x_{1...3}$ ,  $t_{1...3}$  and  $\alpha$  are fitted to reproduce properties of nuclei, such as binding energy, and neutron stars, for example neutron star radii.

### Appendix B: Radial basis function

Examples of existing RBFs are shown in Table II.

RBF	$\phi(r)$	property
multiquadric	$\sqrt{1 + (r/\epsilon)^2}$	increasing
inverse	$(1 + (r/\epsilon)^2)^{-1/2}$	localized
gaussian	$\exp(-(r/\epsilon)^2)$	localized
linear	$r$	increasing
cubic	$r^3$	increasing
quintic	$r^5$	increasing
thin plate	$r^2 \ln(r)$	increasing

TABLE II. Some of the RBFs implemented in *mystic*.  $r$  is the Euclidean distance and  $\epsilon > 0$  is a scalar parameter.

### Appendix C: Hartman 6-D function coefficients

The coefficients of the Hartman 6-D function are given by:  $\alpha = (1.0, 1.2, 3.0, 3.2)^T$ ,

$$A = \begin{vmatrix} 10 & 3 & 17 & 3.50 & 1.7 & 8 \\ 0.05 & 10 & 17 & 0.1 & 8 & 14 \\ 3 & 3.5 & 1.7 & 10 & 17 & 8 \\ 17 & 8 & 0.05 & 10 & 0.1 & 14 \end{vmatrix}$$

and

$$P = 10^{-4} \begin{vmatrix} 1312 & 1696 & 5569 & 124 & 8283 & 5886 \\ 2329 & 4135 & 8307 & 3736 & 1004 & 9991 \\ 2348 & 1451 & 3522 & 2883 & 3047 & 6650 \\ 4047 & 8828 & 8732 & 5743 & 1091 & 381 \end{vmatrix}$$

### Appendix D: How to use the surrogate models

Data and surrogate functions generating during this work are stored in SQL database. These results can be used without needing to run the expensive simulator: For the binary mixture, the surrogate models can be loaded using:

```
import mystic.cache.function as func
g = func.read('BIM.db')
g11=g(r,Gamma,c,Z1,Z2)[0]
g12=g(r,Gamma,c,Z1,Z2)[1]
g22=g(r,Gamma,c,Z1,Z2)[2]
```

where  $r$  is the radial coordinate in Wigner-Seitz radius,  $\Gamma$  is the proton-proton coupling parameter,  $c$  is the concentration of species 2 with nuclear charge  $Z_1$  and  $Z_2$  is the nuclear charge of species 2.

- 
- [1] P. V. Coveney, J. P. Boon, and S. Succi, *Philosophical Transactions of the Royal Society of London Series A* **374**, 20160335 (2016).  
[2] B. Paxton, J. Schwab, E. B. Bauer, L. Bildsten, S. Blinnikov, P. Duffell, R. Farmer, J. A. Goldberg, P. Marchant, E. Sorokina, A. Thoul, R. H. D. Townsend, and F. X. Timmes, *ApJS* **234**, 34 (2018),

[arXiv:1710.08424](https://arxiv.org/abs/1710.08424) [astro-ph.SR].

- [3] K. Kadau, T. C. Germann, P. S. Lomdahl, and B. L. Holian, *Science* **296**, 1681 (2002).  
[4] C. L. Fryer, A. Diaw, C. J. Fontes, A. L. Hungerford, J. Kline, H. Johns, N. E. Lanier, S. Wood, and T. Urbatsch, *High Energy Density Physics* **35**, 100738 (2020), [arXiv:1912.10534](https://arxiv.org/abs/1912.10534) [physics.app-ph].

- [5] C. Ticknor, J. D. Kress, L. A. Collins, J. Cl  rouin, P. Arnault, and A. Decoster, *Phys. Rev. E* **93**, 063208 (2016).
- [6] L. G. Stanton, J. N. Glosli, and M. S. Murillo, *Phys. Rev. X* **8**, 021044 (2018).
- [7] L. A. Collins, J. D. Kress, S. F. Mazevet, and M. P. Desjarlais, *IEEE Transactions on Plasma Science* **33**, 586 (2005).
- [8] J. D. Kress, J. S. Cohen, D. A. Horner, F. Lambert, and L. A. Collins, *Phys. Rev. E* **82**, 036404 (2010).
- [9] E. W. Brown, B. K. Clark, J. L. DuBois, and D. M. Ceperley, *Phys. Rev. Lett.* **110**, 146405 (2013).
- [10] T. Dornheim, J. Vorberger, S. Groth, N. Hoffmann, Z. A. Moldabekov, and M. Bonitz, *J. Chem. Phys.* **151**, 194104 (2019), [arXiv:1907.08473](https://arxiv.org/abs/1907.08473) [physics.plasm-ph].
- [11] N. Lubbers, A. Agarwal, Y. Chen, S. Son, M. Mehana, Q. Kang, S. Karra, C. Junghans, T. C. Germann, and H. S. Viswanathan, *Scientific Reports* **10**, 13312 (2020).
- [12] A. Diaw, K. Barros, J. Haack, C. Junghans, B. Keenan, Y. W. Li, D. Livescu, N. Lubbers, M. McKerns, R. S. Pavel, D. Rosenberger, I. Sagert, and T. C. Germann, *Phys. Rev. E* **102**, 023310 (2020).
- [13] D. Roehm, R. S. Pavel, K. Barros, B. Rouet-Leduc, A. L. McPherson, T. C. Germann, and C. Junghans, *Computer Physics Communications* **192**, 138 (2015).
- [14] M. M. McKerns, L. Strand, T. Sullivan, A. Fang, and M. A. G. Aivazis, *arXiv e-prints*, [arXiv:1202.1056](https://arxiv.org/abs/1202.1056) (2012), [arXiv:1202.1056](https://arxiv.org/abs/1202.1056) [cs.MS].
- [15] J.-L. Coulomb, A. Kobetski, M. Caldora Costa, Y. Mar  chal, and U. Jonsson, *COMPEL-The international journal for computation and mathematics in electrical and electronic engineering* **22**, 616 (2003).
- [16] J. Park and I. W. Sandberg, *Neural computation* **3**, 246 (1991).
- [17] Y. Wu, H. Wang, B. Zhang, and K.-L. Du, *ISRN Applied Mathematics* **2012** (2012).
- [18] R. Schaback and H. Wendland, *Numerical Algorithms* **24**, 239 (2000).
- [19] H. Rocha, *Applied Mathematical Modelling* **33**, 1573 (2009).
- [20] A. S. Dervlo, J. A. Jervase, and A. Al-Lawati, *Applied Energy* **71**, 307 (2002).
- [21] D. Lonardon, I. Tews, S. Gandolfi, and J. Carlson, *Phys. Rev. Research* **2**, 022033 (2020).
- [22] E. Annala, T. Gorda, A. Kurkela, J. N  ttil  , and A. Vuorinen, *Nature Physics* (2020), [10.1038/s41567-020-0914-9](https://doi.org/10.1038/s41567-020-0914-9).
- [23] V. Dexheimer, *Publications of the Astronomical Society of Australia* **34** (2017), [10.1017/pasa.2017.61](https://doi.org/10.1017/pasa.2017.61).
- [24] S. Typel, M. Oertel, and T. Kl  hn, *Phys. Part. Nucl.* **46**, 633 (2015).
- [25] M. Hempel, G. Pagliara, and J. Schaffner-Bielich, *Phys. Rev. D* **80**, 125014 (2009).
- [26] N. K. Glendenning, *Compact Stars: Nuclear Physics, Particle Physics and General Relativity*, Astronomy and Astrophysics Library (Springer New York, 1997).
- [27] T. Fischer, I. Sagert, G. Pagliara, M. Hempel, J. Schaffner-Bielich, T. Rauscher, F.-K. Thielemann, R. K  ppeli, G. Mart  nez-Pinedo, and M. Liebend  rfer, *The Astrophys. Journal Supplement Series* **194**, 39 (2011).
- [28] C. A. Raithel, F.   zel, and D. Psaltis, *Astrophys. Journal* **875**, 12 (2019).
- [29] E. Chabanat, P. Bonche, P. Haensel, J. Meyer, and R. Schaeffer, *Nuclear Physics A* **635**, 231 (1998).
- [30] A. S. Schneider, L. F. Roberts, and C. D. Ott, *Phys. Rev. C* **96** (2017), [10.1103/physrevc.96.065802](https://doi.org/10.1103/physrevc.96.065802).
- [31] J. Stone and P.-G. Reinhard, *Progress in Particle and Nuclear Physics* **58**, 587 (2007).
- [32] T. Skyrme, *Nuclear Physics* **9**, 615 (1958).
- [33] A. Chodos, R. L. Jaffe, K. Johnson, C. B. Thorn, and V. F. Weisskopf, *Phys. Rev. D* **9**, 3471 (1974).
- [34] S. H. Glenzer and R. Redmer, *Reviews of Modern Physics* **81**, 1625 (2009).
- [35] F. Di Natale, H. Bhatia, T. S. Carpenter, C. Neale, S. Kokkila-Schumacher, T. Oppelstrup, L. Stanton, X. Zhang, S. Sundram, T. R. W. Scogland, G. Dharuman, M. P. Surh, Y. Yang, C. Misale, L. Schneidenbach, C. Costa, C. Kim, B. D'Amora, S. Gnanakaran, D. V. Nissley, F. Streitz, F. C. Lightstone, P.-T. Bremer, J. N. Glosli, and H. I. Ing  lfsson, in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '19 (Association for Computing Machinery, New York, NY, USA, 2019).
- [36] R. Evans, *Advances in Physics* **28**, 143 (1979).
- [37] S. Ichimaru, *Rev. Mod. Phys.* **65**, 255 (1993).
- [38] S. D. Baalrud and J. Daligault, *Physics of Plasmas* **26**, 082106 (2019), [arXiv:1904.09208](https://arxiv.org/abs/1904.09208) [physics.plasm-ph].
- [39] A. Diaw and M. S. Murillo, *Phys. Rev. E* **99**, 063207 (2019).
- [40] S. Goldman, *The Journal of Physical Chemistry* **83**, 3033 (1979), <https://doi.org/10.1021/j100486a020>.
- [41] E. Matteoli and G. A. Mansoori, *J. Chem. Phys.* **103**, 4672 (1995).
- [42] A. Morsali, E. K. Goharshadi, G. Ali Mansoori, and M. Abbaspour, *Chemical Physics* **310**, 11 (2005).
- [43] F. J. Rogers, D. A. Young, H. E. Dewitt, and M. Ross, *Phys. Rev. A* **28**, 2990 (1983).
- [44] J.-L. Bretonnet and A. Derouiche, *Phys. Rev. B* **38**, 9255 (1988).
- [45] N. Desbiens, P. Arnault, and J. Cl  rouin, *Physics of Plasmas* **23**, 092120 (2016), [arXiv:1606.04675](https://arxiv.org/abs/1606.04675) [physics.plasm-ph].
- [46] J. G. Kirkwood, *Chem. Rev.* **19**, 275–307 (1936).
- [47] M. E. Fisher and B. Wiodm, *J. Chem. Phys.* **50**, 3756 (1969).
- [48] C. Vega, L. F. Rull, and S. Lago, *Phys. Rev. E* **51**, 3146 (1995).
- [49] P. Hopkins, A. J. Archer, and R. Evans, *Phys. Rev. E* **71**, 027401 (2005).
- [50] S. Plimpton, *J. Comput. Phys.* **117**, 1 (1995).
- [51] A. Diaw and M. S. Murillo, *Astrophys. J.* **829**, 16 (2016).
- [52] E. B. Bauer and L. Bildsten, *Astrophys. J.* **872**, 96 (2019), [arXiv:1812.09602](https://arxiv.org/abs/1812.09602) [astro-ph.SR].
- [53] M. te Vrugt, H. L  wen, and R. Wittkowski, *arXiv e-prints*, [arXiv:2009.07977](https://arxiv.org/abs/2009.07977) (2020), [arXiv:2009.07977](https://arxiv.org/abs/2009.07977) [cond-mat.soft].
- [54] A. J. Archer and R. Evans, *J. Chem. Phys.* **121**, 4246 (2004), [arXiv:cond-mat/0405665](https://arxiv.org/abs/cond-mat/0405665) [cond-mat.stat-mech].
- [55] A. Diaw and M. S. Murillo, *Phys. Rev. E* **92**, 013107 (2015).
- [56] A. Diaw and M. S. Murillo, *The Astrophysical Journal* **829**, 16 (2016).