MRS Advances
https://doi.org/10.1557/s43580-024-00852-x

REVIEW

A snapshot review on soft materials assembly design utilizing machine learning methods

Maya M. Martirossyan[1] · Hongjin Du[1] · Julia Dshemuchadse[1] · Chrisy Xiyu Du[2]

Received: 20 March 2024 / Accepted: 15 April 2024

Abstract

Since the surge of data in materials-science research and the advancement in machine learning methods, an increasing number of researchers are introducing machine learning techniques into the next generation of materials discovery, ranging from neural-network learned potentials to automated characterization techniques for experimental images. In this snapshot review, we first summarize the landscape of techniques for soft materials assembly design that do not employ machine learning or artificial intelligence and then discuss specific machine learning and artificial-intelligence-based methods that enhance the design pipeline, such as high-throughput crystal-structure characterization and the inverse design of building blocks for materials assembly and properties. Additionally, we survey the landscape of current developments of scientific software, especially in the context of their compatibility with traditional molecular-dynamics engines such as LAMMPS and HOOMD-blue.

Introduction

The design of soft materials assemblies with targeted structures and properties requires the engineering of building blocks and interactions that can spontaneously assemble a target material. Before the upsurge of computational capabilities, many studies of soft materials assemblies followed a similar framework: identify a few parameters (building-block properties, densities, etc.), run forward simulations varying the parameters, outline phase diagrams based on these parameters, and iterate. This "forward approach" has provided researchers with valuable insights and tools for exploring soft materials systems: phase diagrams for systems of hard spheres, anisotropic particles with polyhedral shapes, and block copolymers; rare-event sampling techniques; and local bond-order parameters to identify crystal motifs and structures. In recent decades, the exponential growth of computational power has widened the parameter space that can feasibly be searched, and researchers are incorporating machine learning and artificial intelligence (ML/AI) techniques to enhance their materials assembly pipelines (Fig. 1). Not only do these advanced tools enable us to more thoroughly probe broad questions and challenges in the field—for example, the competing nature of enthalpy and entropy in determining structure formation, dynamics, and materials properties in physical systems—but they also allow for the pursuit of reverse- or inverse-design approaches enabled by numerical optimization. Moreover, the study of soft materials (i.e., composed of mesoscopic building blocks, e.g., nanoparticles, colloids, or block copolymers) serves as a coarse-grained version of nano- or atomic-scale phenomena and can aid in understanding how to manipulate and design significantly more complicated building blocks (e.g., macromolecules, such as proteins).

Many prior review articles provide an overview of different ML/AI techniques that have been applied in soft materials design, such as active and transfer learning [1] or neural networks for structural representation [2] and property design [3]. These reviews focus heavily on novel ML algorithms and their application to soft matter. By contrast, this snapshot review will discuss the physical

✉ Julia Dshemuchadse
jd732@cornell.edu

✉ Chrisy Xiyu Du
xiyudu@hawaii.edu

Maya M. Martirossyan
mmm457@cornell.edu

Hongjin Du
hd329@cornell.edu

[1] Department of Materials Science and Engineering, Cornell University, Ithaca 14853, NY, USA

[2] Department of Mechanical Engineering, University of Hawai'i at Mānoa, Honolulu 96822, HI, USA

Published online: 03 May 2024

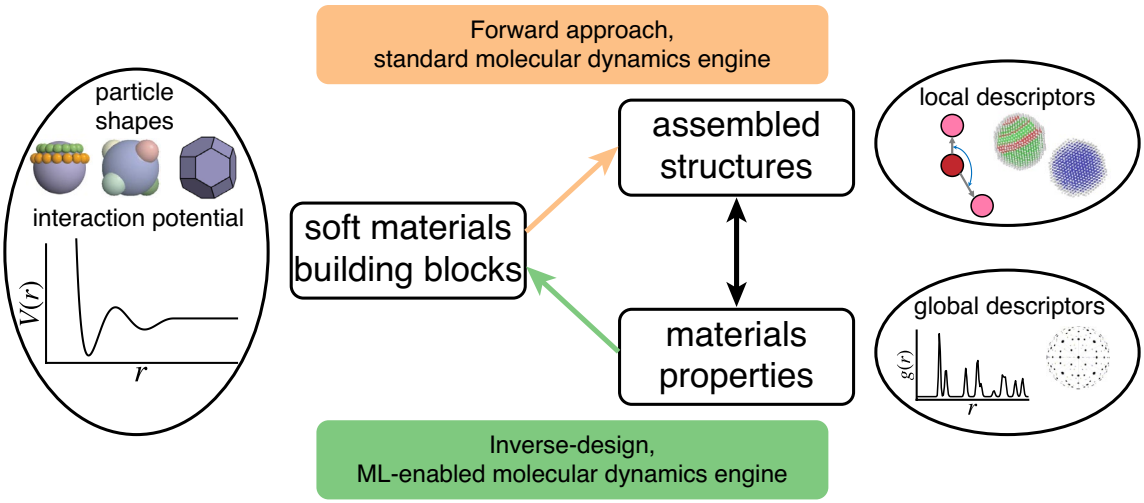

Fig. 1 Soft materials design pipeline. Input parameters for building blocks can be patchy particles, sphere unions, and polyhedral shapes with any arbitrary pair potential functions. To quantify materials structures and properties, a variety of descriptors can be used. Here, we depict bond-order parameters and OVITO's adaptive-CNA for local descriptors, radial distribution functions, and bond-orientational order diagrams (BOODs) as examples for global descriptors

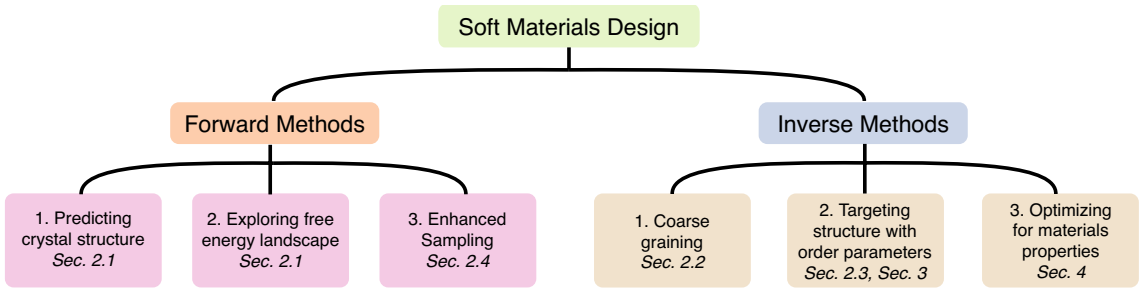

Fig. 2 Overview of different forward and inverse methods for soft materials design

inspiration and insights that can be gleaned from adding ML/AI approaches to the quest for designing self-assembled soft materials. Given the modular nature of the soft-matter design pipeline, various ML/AI strategies can be applied to different stages of the process, and a combination of ML/AI and ML/AI-free strategies can be used to strike a balance between high predictive power and limited computational resources.

Firstly, we discuss the current state of ML-free techniques developed over many decades to study soft matter in both simulation and experiment. Secondly, we describe ML/AI-aided methods for different facets of materials assembly (also shown in Fig. 2): novel descriptors for quantifying local or global structure, an inverse-design framework aided by automatic differentiation, and materials property design aided by ML/AI. Lastly, we discuss the capabilities of various molecular-dynamics (MD) engines in incorporating ML tools and summarize existing ML-based descriptors by their software, methods, and their accessibility to researchers based on the computing resources needed. We intend to elucidate the state of the available methods in the field, give context for the development of the plethora of new tools created in the last few decades, and chart out how we can use these in the study of soft materials design in the future.

ML/AI-free materials design

Crystal structure prediction

In computer simulations of soft materials assembly, an approximation of interparticle interactions is created and employed to predict the structure and properties of the materials system. In such *forward* approaches, building blocks and interactions may also be tuned experimentally. While interactions among all components in a system can be well-defined, *a-priori* knowledge of the stable or metastable

crystal structures that form is not straightforwardly obtained. Crystal structure prediction stands as one of the central challenges in materials systems and is necessary for controlling polymorphism in, for example, pharmaceutical research [4]. Here, we discuss general structure-prediction methods used in modeling—not only of atomistic systems—but in particular of soft-matter systems.

Systems explore their free-energy landscape through dynamics, yet despite the ergodic hypothesis, molecular simulations may not be able to access the entirety of their phase space within a finite time frame. Several methods—simulated annealing, genetic algorithms, and enhanced sampling—have been utilized to answer questions about the global minimum (i.e., stable) structures that may be difficult to access via computational methods.

Simulated annealing [5]—derived from the analogy to physical annealing—is a computational technique that aims to locate the global minimum of a cost (or energy) function and was developed as one of the earliest global optimization techniques. This is achieved through a gradual cooling that leads the system from an initially random configuration to an equilibrium crystal structure. An example of its application to soft materials is the prediction of binary crystal structures of oppositely charged spherical colloids [6].

The Monte-Carlo-based basin hopping method [7, 8] explores rugged energy landscapes by hopping among the local minima (i.e., basins) using a Metropolis criterion, and it has been employed to determine the global energy minimum of size-selected clusters in two distinct hierarchical self-assemblies of triblock patchy particles [9].

The genetic algorithm used for atomistic structure prediction [10, 11] mimics concepts from Darwinian evolution and selects an optimal structure from a set of candidates through a process akin to procreation: structural features from pairs of candidate structures are combined through a crossover algorithm, and new features are introduced to individual structures with a mutation algorithm. Eventually, good features are preserved during 'procreation' through a defined cost function. Genetic algorithms have also been used to predict stable candidate structures of patchy particles [12] and DNA-grafted particles [13]. There are many other global optimization algorithms, such as metadynamics, particle swarm optimization, and landscape paving, that we do not address here.

Coarse-grained models

Coarse-grained (CG) models are developed as reduced-resolution descriptions of a system to perform simulations on a larger time- and length-scale at the cost of fine-grained details. Upon treating groups of atoms as single CG particles, the subsequent challenge is to model interactions between these CG particles. Generally, CG potentials can be derived by: (1) fitting parameters of given potential functions to reproduce target structures or thermodynamic properties, derived from atomistic simulations or empirical measurements; (2) calculating them from the direct interactions between the grouped atoms [14]. Coarse graining has wide applications in studying soft-matter systems (with relevant reviews included in the SI). Below we briefly review three categories of coarse-graining techniques, serving as essential conceptual foundations that underpin the development of ML-based approaches in optimization and parameterization.

Iterative Boltzmann inversion & inverse Monte Carlo

Both the iterative Boltzmann inversion (IBI) [15] and inverse Monte Carlo (IMC) [16] methods use a figure of merit computed directly from the structure to iteratively refine the free-energy surface of the system. The radial distribution function (RDF) of pairwise interparticle distances is a common method in materials science for fingerprinting a crystal structure and can serve as a figure of merit for both IBI and IMC. IBI iteratively refines the potential of mean force (PMF) using Boltzmann inversion until the RDF measured in the system converges to that of the target structure. IMC (or reverse Monte Carlo—RMC) is an iterative procedure that is very similar to IBI, but derives pair potentials differently during the iteration using an exact update scheme with the Jacobian matrix of the RDF with respect to the potential, instead of the empirical update scheme used in IBI. Since IMC takes into account correlations of observables in multi-component systems, it has a higher computational cost than IBI which can lead to convergence problems. Detailed comparisons of these two methods are discussed elsewhere [17]. Note that the Henderson theorem states that only one pair potential is uniquely determined by a given RDF under given conditions of temperature and density [18], yet the accuracy required to distinguish RDFs produced by two different pair potentials is beyond what is needed in practical use. Therefore, additional thermodynamic properties (such as pressure [19]) can be integrated into the optimization process alongside the RDF.

Force matching & multiscale coarse graining

In contrast to the aforementioned structure-based methods (IBI, IMC), the force matching (FM) method does not aim at reproducing target distributions of structural descriptors such as the RDF. Instead it fits potentials by minimizing the difference between the CG forces and the forces in the underlying fine-grained system [20]. The parameterization of the CG model is realized in a non-iterative way: the force

of each atom in a CG particle is taken into account in calculating the force on that CG particle, and the minimization of force difference can be described as a least-squares problem given a sufficiently large number of snapshots (i.e., configurations) from the atomistic trajectory. Force matching was further extended to the multiscale coarse-graining method, wherein the multibody potential of mean force is approximated by deriving effective pair potentials directly from the underlying atomistic potentials [21].

Relative entropy

The relative entropy S_{rel}—also known as the Kullback–Leibler (KL) divergence—is adopted from information theory and is a type of statistical distance that measures the disparity—or *relative entropy*—between two probability distributions. For coarse graining, S_{rel} measures the information loss using the probability density distributions of atomistic (P_{A}) and CG models (P_{CG}): $S_{\text{rel}} = \sum_i P_{\text{A}} \ln \frac{P_{\text{A}}(i)}{P_{\text{CG}}(i)}$, where $P(i)$ is the probability of configuration i in a given ensemble. The minimization of the relative entropy has been applied to the quantification of phase-space overlap between two molecular ensembles [22], CG model development [23, 24], calculation of free-energy differences [25], and inverse design of isotropic interactions that promote self-assembly of structures including multi-component crystals [26] and colloidal strings [27]. The relative entropy formalism is connected to other coarse-graining approaches insofar as they can lead to the same results depending on how potentials are modeled [24]. While IBI and IMC are limited to optimizing pair potentials, relative entropy provides a more general framework for handling many-body CG potentials [14].

Inverse Design

Coarse graining and inverse methods share the goal of identifying a set of parameters of a model that best reproduces the target distribution. In fact, we can view the development of CG models as solving an inverse-design problem where the target properties are the forces from the respective fine-grained systems. Furthermore, both coarse graining and inverse methods are fundamentally rooted in the pursuit of a more systematic framework for materials design and discovery.

A multitude of inverse methods for soft-matter self-assembly and design have been discussed in a recent review [28]. In particular, here we highlight the methods used in the inverse design of isotropic pair potentials that define short-ranged forces only by interparticle distance. Counterintuitively, the simplicity of these interactions does not compromise the structural diversity exhibited by systems that interact with such forces [29], and they can provide insight into the underlying mechanisms of self-assembly. Isotropic interactions are experimentally realizable by tuning, for example, the interactions of the isotropic DNA shell of functionalized nanoparticles [30].

The concept of tailoring potentials to maximize the difference in the ground-state energy between the target structure and its competitors has been successfully applied to the inverse design of structures in multiple systems, including the square and honeycomb lattices in 2D [31, 32], and simple cubic [33], diamond [34], and wurtzite structures [35].

Relative entropy minimization [23, 24] has also been used as a design principle for isotropic pair potentials to control the formation of pores for the assembly of porous mesophases [36], and to promote self-assembly of 2D and 3D crystals [26, 37], colloidal strings [27], as well as size-specific cluster fluids [38]. This "on-the-fly" approach uses structures generated during each optimization step of the particle interactions, thereby promoting the self-assembly of the target structure from a disordered state. This optimization process was also employed in combination with Fourier-space filters to design simple interactions that could be more experimentally feasible [39].

All these approaches to modifying interactions or building blocks can be encompassed by "digital alchemy," which was first introduced as a statistical-thermodynamics method to inversely design anisotropic particle shapes that favor the self-assembly of a target structure with Monte Carlo simulations [40]. The general framework of describing particle attributes as thermodynamic variables—allowing them to fluctuate, and as a result identifying attributes crucial for controlling self-assembly—has also been extended to MD simulations with success for a handful of structures [41, 42].

Enhanced sampling

Enhanced sampling encompasses a class of methods that enables the simulation of hard-to-reach states. There are many different flavors of enhanced-sampling methods: umbrella sampling, replica exchange, metadynamics, and simulated annealing, to name a few. Most enhanced-sampling methods apply a bias force or potential to drive the system to explore the region of phase space containing states of interest. These states are often described by a set of collective variables (also referred to as reaction coordinates, order parameters, or structural descriptors in other contexts). We refer the reader to the SI for many in-depth reviews on different aspects of enhanced sampling.

Descriptors for self-assembly studies

There is a rich history of using order parameters to define and study phase transitions in physical systems, allowing for the most important variables or degrees of freedom to be captured. Reducing a physical system's $3N$ spatial dimensions to a more "natural" low-dimensional representation extracts the *most relevant* characteristics of the system's behavior. In the study of self-assembly and growth, order parameters—i.e., structural descriptors—vary widely in their physical basis and in the behavior of interest of the physical system for whose study they are being used.

Specific variable choices are often necessary to define an order metric, but they can also prove limiting or insufficiently descriptive when confronted with an increasing variety of behaviors or motifs in a single system. For example, how can a descriptor be designed to study a growing crystal with multiple crystalline environments, each with a different kind of crystalline symmetry?

Here, we highlight conventional approaches using physically inspired descriptors and how coupling these methods with machine learning techniques—well-suited for leveraging and interpreting high-dimensional data—allows for a more complete picture of self-assembly to emerge across a variety of physical systems. In our discussion, we place significant emphasis on the physical basis of descriptors rather than on the specific ML tools utilized, in part because these physical descriptors should be tailored to the given system or behavior being studied, and in part because of evidence suggesting that the optimization schema used does not significantly change the outcome of an ML-based analysis approach [43]. Later, in Sect. "Software and methods overview for descriptors", we highlight the technical ML specifics for many of the methods discussed in this section.

Local descriptors that accurately quantify structural motifs have been used to develop atomistic machine-learned interatomic potentials (MLIAPs) and led to marked improvements over potentials calculated from electronic structure alone [44]. The difference in applying descriptors to soft matter is the lack of atomic or energetic data to train on, in order to predict resultant properties; this, in part, explains why solving the inverse-design question represents such a "holy grail" for the field. Consequently, good structural descriptors are critical for capturing and optimizing system behavior.

As is true for MLIAPs, a good descriptor should be immutable upon equivalent configurations generated by translations, rotations, and permutations to a motif or crystal structure. This mathematical property with respect to a symmetry operation is referred to as invariance (equivariance or covariance also satisfy the required criteria). While not required, differentiability is particularly useful for applications utilizing automatic differentiation methods such as JAX-MD [45] (see Sect. "JAX-MD" for a detailed description).

We discuss several structural descriptors, broadly grouped according to their physical basis (as shown in Fig. 3): (1) structure-only parameters including RDF-based, position-based, as well as "descriptor-free" (featureless) parameters that are exclusively machine-learned; (2) bond-orientational features utilizing spherical harmonics in a variety of approaches; and (3) graph-based or topological features. We aim to provide a comprehensive overview of the featurizations in the field, although inevitably we will be unable to cover all relevant work in the scope of this snapshot review.

We include in the SI an additional summary on thermodynamics-inspired features and methods, although these are less commonly used and not as effective compared to structure-based approaches. We will sidestep a common problem lurking among many of the discussed methods, which is exactly how neighbors or radial cutoffs are chosen—handled differently by each method. Finally, we largely ignore informatics approaches (for example, the Polymer Genome platform [46]) as they use hundreds of descriptors in a hierarchical manner to train models that target properties. This section will focus on work that uses a specific choice of descriptor and its appropriate use cases.

RDF-based and position-based features

Using positional data with minimal manipulation is a logical choice for a structural descriptor. The oldest criterion for melting is the Lindemann order parameter [47]—based on particle mean-squared displacement from equilibrium position—utilized in soft-matter studies of nucleation and growth dynamics [48]. In hard-disk or sphere systems, phase transitions were commonly detected using only sorted neighbor distances (i.e., in the first shell of the RDF) as features with unsupervised learning [49].

Behler and Parrinello [50] introduced radial and angular symmetry functions—representing potential-energy surfaces in atomic systems—that also bear similarity to the RDF but are localized to a particle's environment. Such symmetry functions have been utilized in the development of ML-based structural identification methods for complex phases in polymorphic systems such as ice [51] or the ML-based order metric "softness" for identifying particle susceptibility to rearrangements [52]. Softness has been used to predict glass dynamics [53] as well as to identify grain boundaries [54] and improve growth models [55] in atomistic MD simulations. Other position-based features include using bond angle, bond length, and interparticle separation distance as inputs for an unsupervised crystal-structure identification

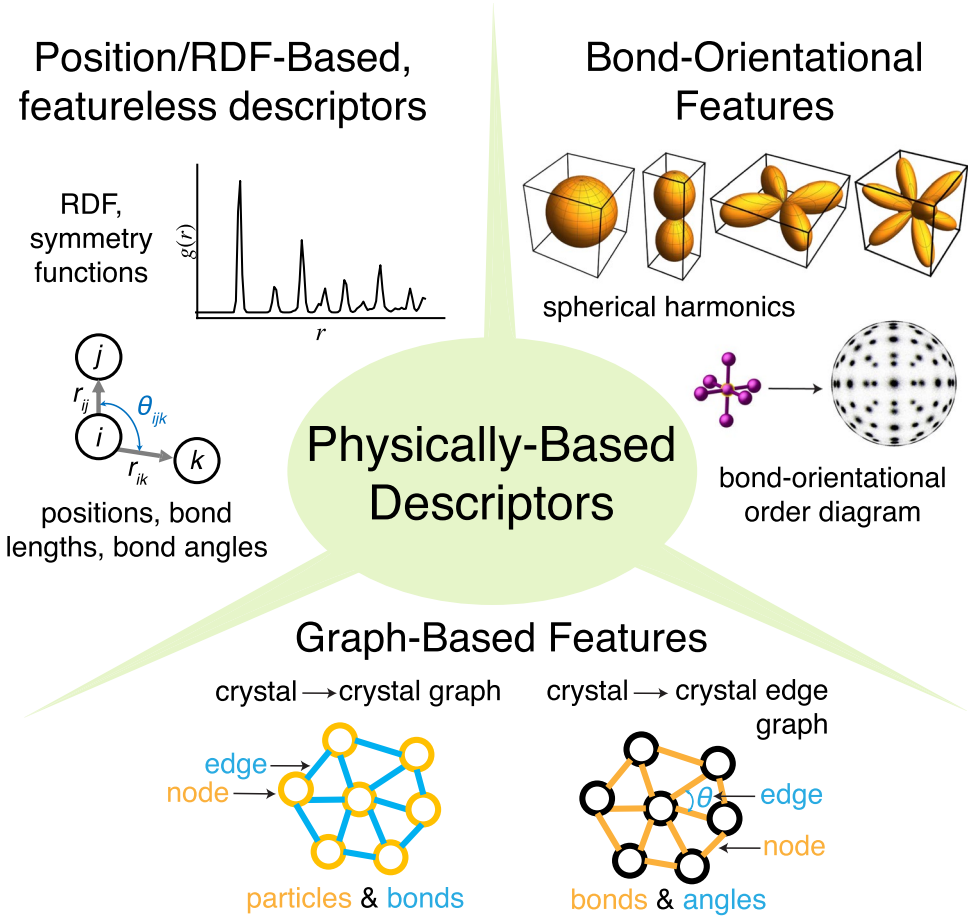

Fig. 3 The three broad classes of descriptors: position-/RDF-based and featureless descriptors, bond-orientational features, and graph-based features

method [56], defining a loss function based on a "stencil" used to target assembly of a specific polymorph using JAX-MD [57], or utilizing particle positions and particle-level features to build geometric algebra-based representations of structure with deep learning [58].

Featureless order parameters

"Descriptor-free" or featureless order parameters can be conceptualized as a subcategory of position-based features, but they differ in that they use entirely unmanipulated data that must be interpreted using statistical or machine-learned methods. Because these descriptors do not use representations that are invariant to translations, rotations, and permutations, they instead rely on data augmentation—that is, the model must learn these symmetries from an abundance of data in a variety of configurations, rather than invariance being built into the inputs for training.

Featureless unsupervised learning methods have so far been used to identify magnetization phase transitions in the canonical two-dimensional Ising model [59, 60] using entire Ising spin matrices as inputs. Unlike in the Ising model, a "descriptor-free" approach is more difficult to apply to systems where particle positions are variable, but this has been accomplished [61, 62] relying on sophisticated model architectures such as PointNet [63] in order to perform feature extraction. Other frameworks that use deep learning approaches with particle positions [64] (or, combined with atomic-level features as inputs [65]) could be extended to target properties in soft-matter systems.

Bond-orientational features

Bond-orientational features differ from those described above in that they enforce spherical symmetry in their representations of local structure. For two-dimensional structures, the Ψ_n order parameter is defined by the expectation of n-fold symmetry in the crystalline phase, and it has been utilized in the study of colloidal crystallization experiments [66]. The Steinhardt Q_l order parameter [67] and its neighbor-averaged version $\overline{Q}_l$ [68]—which are rotationally invariant representations of a particle's neighborhood using summations of spherical harmonics—have been used to identify local motifs or differentiate phases of matter, distinguish between simple sphere packings (*bcc*, *ccp*, *hcp*) [69], and study quasicrystal growth [70] in simulations of three-dimensional systems. Steinhardt's $\hat{W}_l$ parameter has

also been utilized to identify motifs in computational studies of pre-crystallization fluids [71].

The addition of machine learning methods to these bond-orientational approaches has allowed for the extension of order parameters to more complex crystal structures, polydisperse packings, and non-close-packed local environments in crystalline solids (i.e., expanding beyond icosahedral, *fcc*, *hcp*, or *bcc* local environments). Spherical harmonics-based descriptors have been used with unsupervised learning approaches to distinguish highly similar, complex structures [72], as well as to distinguish between local environments and phases during the self-assembly of structures with one or more crystalline motifs [73, 74]. Supervised approaches using spherical harmonics [72] or Steinhardt-based features have also successfully identified crystalline motifs in binary systems [75], and unsupervised approaches have similarly been employed with Steinhardt-based features to study local order in glasses and liquids [76] or at crystalline grain boundaries and binary systems [77].

A handful of other approaches using spherical harmonics-based descriptors have been formulated for the study of atomic materials and extended to the study of phase transitions or soft and molecular systems. The Smooth Overlap of Atomic Positions (SOAP) descriptor [78]—which utilizes spherical harmonics to represent Gaussian-smeared particle densities—has also been adapted for ML-based studies of materials: a Gaussian process with a SOAP kernel [79] or unsupervised methods with a SOAP descriptor [80] have been used to study the formation of (supra)molecular materials. Euclidean neural networks (e3nn) [81] use spherical harmonics to create irreducible representations that leverage equivariance to learn symmetry-based translations and rotations, and they can be used to define order parameters that identify the breaking of these symmetries (e.g., during a phase transition) [82].

Graph-based features

Yet another intuitive way to featurize inter-particle bonding structure is through graph-based features—not to be conflated with graph neural networks, although they can appear together. Graph-based features include particle connections, bond lengths and angles, and local neighborhood geometry in their representation of local structure. The most popular graph-based feature is referred to as Common Neighbor Analysis (CNA) [83, 84], a tool which classifies simple 3D motifs by the topology of particle neighborhoods and which is integrated (along with its variants) into the "Open Visualization Tool" (OVITO)[1]. CNA has been applied to numerous studies of crystallization, such as the simulation study of charge-stabilized colloidal suspensions [85]. Another commonly used method for simple crystal-structure or motif identification is polyhedral template matching (PTM) [86], which uses the convex hull of neighbors around a particle to create a planar graph and performs template matching to identify motifs. Given the success of these approaches, the addition of machine learning methods is highly sensible and allows a larger variety of local structures to be represented and identified as compared to CNA and PTM, particularly for systems where atom- or particle-level features are important (such as having two different particle sizes or components).

Graph neural networks (GNNs) are designed to take in graphs as inputs and perform convolutions to create embeddings of local structure. Crystals lend themselves naturally to representations as planar graphs, where nodes and edges represent particles and bonds. For example, a GNN is used to build local descriptors from graph-based features that can identify disorder such as in grain boundaries or interfaces [87]. GNNs have also been used with 'crystal edge graphs' rather than crystal graphs, where nodes represent bonds in the crystal and edges represent bond pairs (i.e., angles between bonds). Recent work has identified phase transitions by building global descriptors [88] with the Atomistic Line Graph Neural Network (ALIGNN) [89], which uses a GNN to create latent representations using message-passing between the crystal graph (interatomic bond graph) and the crystal edge graph (line graph corresponding to bond angles). In a similar vein, crystal edge graphs have been used to perform crystal identification tasks on individual particles [90].

However, GNNs are not the only types of ML approaches utilized with graph-based features. Convolutional neural networks have been applied to graph-based features for molecules [91], atomic structures [92], and glasses [93]. Other ML methods have also been applied with graph-based features—for example, diffusion maps for local environment identification including both amorphous and crystalline structures [94], or for building representations of chemical ordering in multi-component alloys for use in a relative entropy-based order metric [95].

Designing for properties

Frequently, the design of materials with specific properties has been handled separately from the design of assembly pathways to target particular structures [28]. Inverse design for the properties of a material requires the use of the property itself as the figure of merit of the computation, enabling the use of any property that can be computed from a material's structure. Machine learning facilitates the accelerated evaluation of complex structure–property relationships that

[1] https://www.ovito.org.

would otherwise be prohibitive as a computational figure of merit [28]. This can be done by reducing the dimensionality of the order parameter or by applying supervised machine learning directly to the structure–property relationship [3].

Many advances have been made in the field of atomistic ("hard") materials such as alloys, and electronic materials, in which atomic structure can directly be related to bulk properties of a material [89, 92]. In this snapshot review, we concentrate on mesoscopic ("soft") matter. While bulk properties can often be computed directly in hard-matter systems—because the relevant structural features occur on only one length scale—soft-matter systems exhibit multiple salient length scales, making predictive modeling of a variety of properties more challenging. For length consideration, we provide our discussion on soft materials properties in the SI, where we discuss five different kinds of materials properties: (1) mechanical properties, (2) thermodynamic and phase properties, (3) electronic and optical properties, (4) transport properties, and (5) chemical properties.

Software developments

Many well-known MD packages were developed before the popularity of ML-enhanced materials research, so integrating ML methods with traditional MD simulations can prove challenging. In this section, we briefly review the compatibility of current MD engines with ML methods, how software for descriptors can be used, and we showcase a new MD engine that is intrinsically compatible with the current ML/AI software packages.

Integration with ML methods for traditional MD engines

Traditional MD packages such as LAMMPS[2] and HOOMD-blue[3] are extremely powerful engines that can perform MD simulations very effectively. Although primarily written in C++, there are now tools available to integrate ML methods with these MD platforms.

LAMMPS hosts a well-documented webpage[4] providing a list of software packages that are either external—and built on top of LAMMPS—or standalone—either providing input parameters for LAMMPS or other MD engines, or incorporating LAMMPS as one of their MD engines to produce simulation trajectories. Within the list, there are packages linking PyTorch with LAMMPS and several packages for ML-based interaction potentials.

Similarly, HOOMD-TF[5] was developed to link TensorFlow with HOOMD-blue (currently compatible with HOOMD-blue 2.6+ but not 3.x, etc., due to a major API change[6]).

Software and methods overview for descriptors

There is a large variety of methods for quantifying structural order that are applied to study crystal growth and assembly—which are equally as diverse as the open questions in the field. With many ML-based methods being developed for different use cases and specific physical systems, we highlight methods in Table 1 with the most important software and architecture details as well as computing resources needed for each.

JAX-MD

Computing derivatives or gradients is a crucial component of many machine learning techniques. Utilizing general-purpose automatic differentiation [96] implementations is standard in many different machine learning packages such as PyTorch, JuliaDiff, and MatLab's Deep Learning Toolbox. Similarly, various materials-science applications also require the computation of gradients ranging from force computation in MD to evaluating stress tenors for materials properties.

Following the release of JAX[7] in 2018—a Python-based software package that enables end-to-end differentiation—various packages were developed utilizing JAX's new ability to differentiate through complicated functions. JAX-based materials-science software packages are not limited to JAX-MD, and include JAX-AM[8], JAX-FEM[9], and GradDFT[10]. Given the scope of this snapshot review, which concentrates on assembly design, we will only highlight work related to JAX-MD.

The molecular-dynamics engine JAX-MD currently features simulation environments to model isotropic pair potentials and anisotropic particles using rigid-body constructions with standard integrators such as NVE, NVT, NVP, Brownian dynamics, and Langevin dynamics. As JAX-MD is written fully in Python, the overhead for any user to define a

[2] https://www.lammps.org.

[3] https://glotzerlab.engin.umich.edu/hoomd-blue.

[4] https://www.lammps.org/external.html.

[5] https://github.com/ur-whitelab/hoomd-tf.

[6] A—to-date unmerged—branch exists on the HOOMD-TF GitHub page, allowing to make the code compatible with HOOMD-blue 3.x.

[7] http://github.com/google/jax.

[8] https://github.com/tianjuxue/jax-am.

[9] https://github.com/deepmodeling/jax-fem.

[10] https://github.com/XanaduAI/GradDFT.

Table 1 ML-based descriptors for materials assembly and design

References	Features	Models	Software	Compute
Geiger and Dellago [51]	Symmetry functions	ANN	–	GPU
Cubuk et al. [52]	Symmetry functions	SVM	LIBSVM[1]	–
Wang [59]	Ising spin matrix	PCA+k-means clustering	–	–
Wetzel [60]	Ising spin matrix	kernel PCA/DBSCAN/variational autoencoder	–	–
Jadrich et al. [49]	Sorted neighbor distances	Incremental PCA	Sklearn	–
Reinhart [56]	Neighbor distances, bond angles & lengths, particle-level features	UMAP+Random Forest Classifier	UMAP, Sklearn	–
DeFever et al. [61]	Particle positions	PointNet[2]	TensorFlow	–
Wang et al. [62]	Particle positions	Autoencoder+GMM	TensorFlow, Sklearn	–
Schütt et al. [65]	Atomic nuclear charges & positions	Filter-generating network	TensorFlow, SchNet[3]	CPU-intensive/GPU
Swanson et al. [64]	Particle positions	CNN/message-passing neural network	TensorFlow/PyTorch, "glassML"[4]	GPU
Spellings [58]	Multivectors (geometric products of particle positions) & particle-level features	Attention mechanism	Keras, TensorFlow, GAlA [5]	GPU
Spellings and Glotzer [72]	Spherical harmonics (pythia) [6]	PCA+GMM/ANN	Sklearn/ Keras	CPU
Adorf et al. [73]	Bispectrum spherical harmonics (pythia)[6]	PCA+UMAP+HDBSCAN*	Sklearn, UMAP, HDBSCAN*	–
Boattini et al. [77]	Steinhardt parameters	Autoencoder+GMM	Sklearn	–
Coli and Dijkstra [75]	Steinhardt parameters	ANN	Keras, TensorFlow	–
Grisafi et al. [79]	SOAP[7]	Gaussian process	SciPy, SA-GPR[8]	CPU-intensive
Gardin et al. [80]	SOAP[7]	PCA+PAMM/Hierarchical clustering	Sklearn, PAMM[9]	–
Geiger and Smidt [81]	Irreps (tensor products of spherical harmonics)	CNN	JAX/PyTorch, e3nn[10]	GPU
Duvenaud et al. [91]	Molecular graphs	CNN	SciPy, Autograd, "Neural fingerprint"[11]	–
Bapst et al. [93]	Crystal graphs	GNN	TensorFlow/TF-Replicator, JAX, "Glassy dynamics"[12]	–
Chapman et al. [87]	Crystal graphs	GNN	PyTorch, SODAS/graphite[13]	GPU
Choudhary and DeCost [89]	Crystal/line graphs+radial basis functions	Message-passing GNN	ALIGNN[14]	GPU
Aroboto et al. [88]	ALIGNN[14]	UMAP+GNN	UMAP, PyTorch, SODAS++[15]	–
Reinhart et al. [94]	CNA-based crystal graph	Diffusion maps	Neighborhood Graph Analysis (NGA)	–
Xie and Grossman [92]	Atom-level features & crystal graphs	CNN	Sklearn, PyTorch, CGCNN[16]	–
Banik et al. [90]	Crystal edge graphs	Attention mechanism	PyTorch, Sklearn, CEGANN[17]	GPU
Sheriff et al. [95]	Crystal graphs & particle-level features	–	e3nn[10]	–

Compute resources reflect those used or reported by the respective authors

SVM support vector machine, *PCA* principal component analysis, *DBSCAN* density-based spatial clustering of applications with noise, *GMM* Gaussian mixture model, *ANN* artificial neural network, *CNN* convolutional neural network, *GNN* graph neural network

[1] https://github.com/cjlin1/libsvm

[2] https://github.com/charlesq34/pointnet

[3] https://github.com/atomistic-machine-learning/SchNet

[4] https://github.com/ks8/glassML

[5] https://github.com/klarh/geometric_algebra_attention

[6] https://github.com/glotzerlab/pythia

[7] https://singroup.github.io/dscribe/latest/

[8] https://github.com/lab-cosmo/SA-GPR

[9] https://github.com/lab-cosmo/pamm

Table 1 (continued)

[10] https://github.com/e3nn

[11] https://github.com/HIPS/neural-fingerprint

[12] https://github.com/google-deepmind/deepmind-research/tree/master/glassy_dynamics

[13] https://github.com/LLNL/graphite

[14] https://github.com/usnistgov/alignn

[15] https://github.com/Materials-Informatics-Laboratory/SODAS

[16] https://github.com/txie-93/cgcnn

[17] https://github.com/sbanik2/CEGANN

new pair potential or external field—or interface with other ML/AI methods—is minimal. Moreover, when implementing a new pair potential, no additional force implementation is needed as gradients (i.e., derivatives) of the interaction potential can be retrieved directly to update quantities such as particle velocity and acceleration.

So far, JAX-MD has been used to design assembly and transition rates for colloidal systems [57], anisotropic building blocks for bulk and finite assembly [97], controlled disassembly of colloidal clusters [98], error-free polymer growth [99], and minimal-work pathways in non-equilibrium systems [100]. These papers showcase the breadth and versatility of the physical systems and properties that JAX-MD can model and design. Generally, the optimization regime in JAX-MD is system-agnostic as long as the user can provide a loss function / order parameter with meaningful gradients to describe the simulated system.

Figure 4 illustrates a schema for the use of JAX-MD for inverse design, but this is not the only way to implement such a workflow. For example, one can update building-block properties after a fixed number of simulation steps instead of at the end of one round of forward simulation. Here, we want to provide a working example as a starting point for interested researchers to explore. Apart from using JAX-MD to inversely design assembly, yet another unexplored territory for JAX-MD is to combine it with enhanced-sampling methods. Computing forces for a bias potential in MD can be challenging to implement, but with the help of automatic differentiation, no explicit force implementation would be needed.[11]

Accessibility

In this section, we highlight a few barriers to accessing some of the methods discussed in this snapshot review. Computing resources are vital to those who may want to train models or utilize inverse-design methods for their research. Methods that run on GPUs *can* be run on CPUs as well, but the difference in wall time can amount to orders of magnitude—especially for tasks that require backpropagation. Moreover, tasks that require backpropagation or automatic differentiation can be GPU-memory intensive, sometimes requiring the most advanced GPUs with 80 GB of memory. Therefore, we encourage including computational resources either used (or utilizable) for training models and describing the associated computational costs for new methods being published. In a similar vein, sharing code on open-source platforms like GitHub is increasingly common and can function as a "plug and play" tool for non-experts to utilize.

Another possible barrier is the need to transmute training data (from simulation or experiment) into the expected data format for a specific method. For example, many of the approaches highlighted in Table 1 rely on specific file formats (typically only used for data output by a particular MD engine). Tools such as the *garnett* software[12] can help with reading/writing to/from different simulation file formats, although not all common formats are included (e.g., the `.xyz` file format). A similar issue arises with ML-specific backends: often methods are developed for only one of the three—Keras/TensorFlow, JAX, or PyTorch. These are just a few of the "language barriers" that arise from the diversity of computing tools that researchers use.

Finally, access to large volumes of data for training models is usually straightforward for simulators, and there are already databases hosting services for more various materials datasets[13]—but this is not necessarily the case for soft-matter experimentalists. While we do not necessarily endorse the publishing of trained models as a solution, we urge consideration of how models can behave for low data-volume cases. These considerations could be especially important in developing simulation–experiment pipelines for training models or inverse-design approaches.

[11] See https://colab.research.google.com/drive/1eOBqUlRxhUvPsxfl9hGcVwHTleN2GNBJ for an example.

[12] https://garnett.readthedocs.io

[13] https://www.materialsdatafacility.org

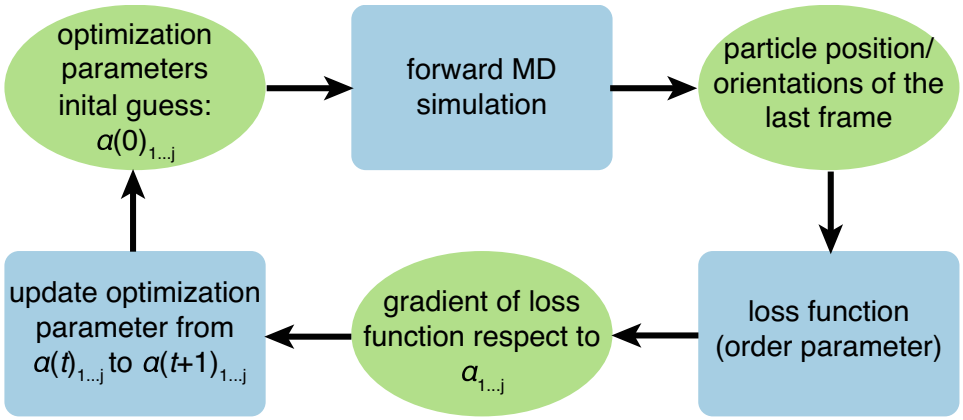

Fig. 4 Inverse-design framework based on JAX-MD

Conclusion

In this snapshot review, we discuss many approaches used in the optimization and design of soft materials such as structure prediction, coarse-graining, enhanced sampling, and how these approaches are not only compatible with but enhanced by ML methods, as well as a variety of software that can be used to target specific structures or materials properties. We discuss both forward approaches—critical for the study of phase behavior and self-assembly—as well as more targeted inverse approaches that are used specifically for design.

Through our survey of methods in the field, we emphasize the importance of the physical basis of methods and features. We also include relevant methods that are developed for atomic systems as these approaches can be extended to soft or mesoscale materials. We hope that this snapshot review can serve as a guide for those looking to apply (or create) ML-based methods for scientific questions.

We also offer a few reflections on how we believe the methods we review can be best used going forward. Given the multitude of descriptors and inverse-design tools developed in just the last decade, the fields of both atomistic and soft materials are ripe for employing new methods to conduct scientific research. That is, the "low-hanging fruit" of ML-based approaches are being or have been picked, and further development or use of methods should be tailored to answering open questions in the field or addressing specific design principles. Bridging the gap between the tool-makers and tool-users will be imperative in order to address open scientific questions and to connect theory, simulation, and experiments: these range from the need for robust descriptors that can handle particle-locating in experiments, to more fundamental questions such as the effect of interaction and structure in particle-based systems. We are optimistic that with the newly available avenues—provided by the power of machine learning and the multitude of new computational approaches built upon decades of progress—we can answer fundamental questions regarding structure formation and design of matter across various length scales in the future.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1557/s43580-024-00852-x.

Author contributions MMM, HD, JD, and CXD wrote the manuscript.

Funding This material is based upon work supported by the National Science Foundation under Grant No. DMR-2144094, the National Science Foundation AI Institute in Dynamic Systems under Grant No. CBET-2112085, as well as the Camille and Henry Dreyfus Foundation through a Machine Learning in the Chemical Sciences and Engineering Award (ML-22-038). M. M. M. acknowledges support from the National Science Foundation Graduate Research Fellowship Grant No. DGE-2139899 (2021–2024).

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

1. A.L. Ferguson, K.A. Brown, Data-driven design and autonomous experimentation in soft and biological materials engineering. Annu. Rev. Chem. Biomol. Eng. **13**(1), 25–44 (2022). https://doi.org/10.1146/annurev-chembioeng-092120-020803
2. P.S. Clegg, Characterising soft matter using machine learning. Soft Matter **17**(15), 3991–4005 (2021). https://doi.org/10.1039/D0SM01686A
3. S. Kadulkar, Z.M. Sherman, V. Ganesan, T.M. Truskett, Machine learning-assisted design of material properties. Annu. Rev. Chem. Biomol. Eng. **13**(1), 235–254 (2022). https://doi.org/10.1146/annurev-chembioeng-092220-024340
4. M.A. Neumann, J. Streek, F.P.A. Fabbiani, P. Hidber, O. Grassmann, Combined crystal structure prediction and high-pressure crystallization in rational pharmaceutical polymorph screening. Nat. Commun. **6**(1), 7793 (2015). https://doi.org/10.1038/ncomms8793

5. S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, Optimization by simulated annealing. Science **220**(4598), 671–680 (1983). https://doi.org/10.1126/science.220.4598.671
6. A.-P. Hynninen, C.G. Christova, R. Roij, A. Blaaderen, M. Dijkstra, Prediction and observation of crystal structures of oppositely charged colloids. Phys. Rev. Lett. **96**(13), 138308 (2006). https://doi.org/10.1103/PhysRevLett.96.138308
7. D.J. Wales, J.P.K. Doye, Global optimization by basin-hopping and the lowest energy structures of Lennard-Jones clusters containing up to 110 atoms. J. Phys. Chem. A **101**(28), 5111–5116 (1997). https://doi.org/10.1021/jp970984n
8. D.J. Wales, H.A. Scheraga, Global optimization of clusters, crystals, and biomolecules. Science **285**(5432), 1368–1372 (1999). https://doi.org/10.1126/science.285.5432.1368
9. D. Morphew, J. Shaw, C. Avins, D. Chakrabarti, Programming hierarchical self-assembly of patchy particles into colloidal crystals via colloidal molecules. ACS Nano **12**(3), 2355–2364 (2018). https://doi.org/10.1021/acsnano.7b07633
10. A.R. Oganov, C.W. Glass, Crystal structure prediction using ab initio evolutionary techniques: principles and applications. J. Chem. Phys. **124**(24), 244704 (2006). https://doi.org/10.1063/1.2210932
11. D.C. Lonie, E. Zurek, XtalOpt: an open-source evolutionary algorithm for crystal structure prediction. Comput. Phys. Commun. **182**(2), 372–387 (2011). https://doi.org/10.1016/j.cpc.2010.07.048
12. E. Bianchi, G. Doppelbauer, L. Filion, M. Dijkstra, G. Kahl, Predicting patchy particle crystals: variable box shape simulations and evolutionary algorithms. J. Chem. Phys. **136**(21), 214102 (2012). https://doi.org/10.1063/1.4722477
13. B. Srinivasan, T. Vo, Y. Zhang, O. Gang, S. Kumar, V. Venkatasubramanian, Designing DNA-grafted particles that self-assemble into desired crystalline structures using the genetic algorithm. Proc. Natl. Acad. Sci. USA **110**(46), 18431–18435 (2013). https://doi.org/10.1073/pnas.1316533110
14. E. Brini, E.A. Algaer, P. Ganguly, C. Li, F. Rodríguez-Ropero, N.F.A. Vegt, Systematic coarse-graining methods for soft matter simulations - a review. Soft Matter **9**(7), 2108–2119 (2013). https://doi.org/10.1039/C2SM27201F
15. D. Reith, M. Pütz, F. Müller-Plathe, Deriving effective mesoscale potentials from atomistic simulations. J. Comput. Chem. **24**(13), 1624–1636 (2003). https://doi.org/10.1002/jcc.10307
16. A.P. Lyubartsev, A. Laaksonen, Calculation of effective interaction potentials from radial distribution functions: a reverse Monte Carlo approach. Phys. Rev. E **52**(4), 3730–3737 (1995). https://doi.org/10.1103/PhysRevE.52.3730
17. D. Rosenberger, M. Hanke, N.F.A. Vegt, Comparison of iterative inverse coarse-graining methods. Eur. Phys. J. Spec. Top. **225**(8), 1323–1345 (2016). https://doi.org/10.1140/epjst/e2016-60120-1
18. R.L. Henderson, A uniqueness theorem for fluid pair correlation functions. Phys. Lett. A **49**(3), 197–198 (1974). https://doi.org/10.1016/0375-9601(74)90847-0
19. F. Müller-Plathe, Coarse-graining in polymer simulation: from the atomistic to the mesoscopic scale and back. ChemPhysChem **3**(9), 754–769 (2002)
20. F. Ercolessi, J.B. Adams, Interatomic potentials from first-principles calculations: the force-matching method. Europhys. Lett. **26**(8), 583 (1994). https://doi.org/10.1209/0295-5075/26/8/005
21. S. Izvekov, G.A. Voth, A multiscale coarse-graining method for biomolecular systems. J. Phys. Chem. B **109**(7), 2469–2473 (2005). https://doi.org/10.1021/jp044629q
22. D. Wu, D.A. Kofke, Phase-space overlap measures. I. Fail-safe bias detection in free energies calculated by molecular simulation. J. Chem. Phys. **123**(5), 054103 (2005). https://doi.org/10.1063/1.1992483
23. M.S. Shell, The relative entropy is fundamental to multiscale and inverse thermodynamic problems. J. Chem. Phys. **129**(14), 144108 (2008). https://doi.org/10.1063/1.2992060
24. A. Chaimovich, M.S. Shell, Coarse-graining errors and numerical optimization using a relative entropy framework. J. Chem. Phys. **134**(9), 094112 (2011). https://doi.org/10.1063/1.3557038
25. I. Bilionis, P.S. Koutsourelakis, Free energy computations by minimization of Kullback–Leibler divergence: an efficient adaptive biasing potential method for sparse representations. J. Comput. Phys. **231**(9), 3849–3870 (2012). https://doi.org/10.1016/j.jcp.2012.01.033
26. W.D. Piñeros, B.A. Lindquist, R.B. Jadrich, T.M. Truskett, Inverse design of multicomponent assemblies. J. Chem. Phys. **148**(10), 104509 (2018). https://doi.org/10.1063/1.5021648
27. D. Banerjee, B.A. Lindquist, R.B. Jadrich, T.M. Truskett, Assembly of particle strings via isotropic potentials. J. Chem. Phys. **150**(12), 124903 (2019). https://doi.org/10.1063/1.5088604
28. Z.M. Sherman, M.P. Howard, B.A. Lindquist, R.B. Jadrich, T.M. Truskett, Inverse methods for design of soft materials. J. Chem. Phys. **152**(14), 140902 (2020). https://doi.org/10.1063/1.5145177
29. J. Dshemuchadse, P.F. Damasceno, C.L. Phillips, M. Engel, S.C. Glotzer, Moving beyond the constraints of chemistry via crystal structure discovery with isotropic multiwell pair potentials. Proc. Natl. Acad. Sci. USA **118**(21), 2024034118 (2021). https://doi.org/10.1073/pnas.2024034118
30. R. Mao, B. Minevich, D. McKeen, Q. Chen, F. Lu, O. Gang, J. Mittal, Regulating phase behavior of nanoparticle assemblies through engineering of DNA-mediated isotropic interactions. Proc. Natl. Acad. Sci. USA **120**(52), 2302037120 (2023). https://doi.org/10.1073/pnas.2302037120
31. M. Rechtsman, F. Stillinger, S. Torquato, Designed interaction potentials via inverse methods for self-assembly. Phys. Rev. E **73**(1), 011406 (2006). https://doi.org/10.1103/PhysRevE.73.011406
32. E. Marcotte, F.H. Stillinger, S. Torquato, Optimized monotonic convex pair potentials stabilize low-coordinated crystals. Soft Matter **7**(6), 2332–2335 (2011). https://doi.org/10.1039/C0SM01205J
33. M.C. Rechtsman, F.H. Stillinger, S. Torquato, Self-assembly of the simple cubic lattice with an isotropic potential. Phys. Rev. E **74**(2), 021404 (2006). https://doi.org/10.1103/PhysRevE.74.021404
34. A. Jain, J.R. Errington, T.M. Truskett, Inverse design of simple pairwise interactions with low-coordinated 3D lattice ground states. Soft Matter **9**(14), 3866–3870 (2013). https://doi.org/10.1039/C3SM27785B
35. M.C. Rechtsman, F.H. Stillinger, S. Torquato, Synthetic diamond and wurtzite structures self-assemble with isotropic pair interactions. Phys. Rev. E **75**(3), 031403 (2007). https://doi.org/10.1103/PhysRevE.75.031403
36. B.A. Lindquist, S. Dutta, R.B. Jadrich, D.J. Milliron, T.M. Truskett, Interactions and design rules for assembly of porous colloidal mesophases. Soft Matter **13**(7), 1335–1343 (2017). https://doi.org/10.1039/C6SM02718K
37. B.A. Lindquist, R.B. Jadrich, T.M. Truskett, Communication: inverse design for self-assembly via on-the-fly optimization. J. Chem. Phys. **145**(11), 111101 (2016). https://doi.org/10.1063/1.4962754
38. B.A. Lindquist, Inverse design of equilibrium cluster fluids applied to a physically informed model. J. Chem. Phys. **154**(17), 174907 (2021). https://doi.org/10.1063/5.0048812
39. C.S. Adorf, J. Antonaglia, J. Dshemuchadse, S.C. Glotzer, Inverse design of simple pair potentials for the self-assembly of complex structures. J. Chem. Phys. **149**(20), 204102 (2018). https://doi.org/10.1063/1.5063802

40. G. Anders, D. Klotsa, A.S. Karas, P.M. Dodd, S.C. Glotzer, Digital alchemy for materials design: colloids and beyond. ACS Nano **9**(10), 9542–9553 (2015). https://doi.org/10.1021/acsnano.5b04181
41. P. Zhou, J.C. Proctor, G. Van Anders, S.C. Glotzer, Alchemical molecular dynamics for inverse design. Mol. Phys. **117**(23–24), 3968–3980 (2019). https://doi.org/10.1080/00268976.2019.1680886
42. P. Zhou, S.C. Glotzer, Inverse design of isotropic pair potentials using digital alchemy with a generalized Fourier potential. Eur. Phys. J. B **94**(12), 243 (2021). https://doi.org/10.1140/epjb/s10051-021-00250-4
43. J. Mao, I. Griniasty, H.K. Teoh, R. Ramesh, R. Yang, M.K. Transtrum, J.P. Sethna, P. Chaudhari, The training process of many deep networks explores the same low-dimensional manifold. arXiv:2305.01604 [cond-mat] (2023). https://doi.org/10.48550/arXiv.2305.01604
44. J. Behler, Perspective: machine learning potentials for atomistic simulations. J. Chem. Phys. **145**(17), 170901 (2016). https://doi.org/10.1063/1.4966192
45. S.S. Schoenholz, E.D. Cubuk, Jax, M.D. A framework for differentiable physics. J. Stat. Mech.: Theory Exp. **2021**(12), 124016 (2021). https://doi.org/10.1088/1742-5468/ac3ae9
46. C. Kim, A. Chandrasekaran, T.D. Huan, D. Das, R. Ramprasad, Polymer genome: a data-powered polymer informatics platform for property predictions. J. Phys. Chem. C **122**(31), 17575–17585 (2018). https://doi.org/10.1021/acs.jpcc.8b02913
47. F.A. Lindemann, Über die Berechnung molekularer Eigenfrequenzen. Physikalische Zeitschrift **11**, 609–612 (1910)
48. Y. Peng, F. Wang, Z. Wang, A.M. Alsayed, Z. Zhang, A.G. Yodh, Y. Han, Two-step nucleation mechanism in solid-solid phase transitions. Nat. Mater. **14**(1), 101–108 (2015). https://doi.org/10.1038/nmat4083
49. R.B. Jadrich, B.A. Lindquist, T.M. Truskett, Unsupervised machine learning for detection of phase transitions in off-lattice systems. I. Foundations. J. Chem. Phys. **149**(19), 194109 (2018). https://doi.org/10.1063/1.5049849
50. J. Behler, M. Parrinello, Generalized neural-network representation of high-dimensional potential-energy surfaces. Phys. Rev. Lett. **98**(14), 146401 (2007). https://doi.org/10.1103/PhysRevLett.98.146401
51. P. Geiger, C. Dellago, Neural networks for local structure detection in polymorphic systems. J. Chem. Phys. **139**(16), 164105 (2013). https://doi.org/10.1063/1.4825111
52. E.D. Cubuk, S.S. Schoenholz, J.M. Rieser, B.D. Malone, J. Rottler, D.J. Durian, E. Kaxiras, A.J. Liu, Identifying structural flow defects in disordered solids using machine-learning methods. Phys. Rev. Lett. **114**(10), 108001 (2015). https://doi.org/10.1103/PhysRevLett.114.108001
53. S.S. Schoenholz, E.D. Cubuk, D.M. Sussman, E. Kaxiras, A.J. Liu, A structural approach to relaxation in glassy liquids. Nat. Phys. **12**(5), 469–471 (2016). https://doi.org/10.1038/nphys3644
54. T.A. Sharp, S.L. Thomas, E.D. Cubuk, S.S. Schoenholz, D.J. Srolovitz, A.J. Liu, Machine learning determination of atomic dynamics at grain boundaries. Proc. Natl. Acad. Sci. USA **115**(43), 10943–10947 (2018). https://doi.org/10.1073/pnas.1807176115
55. R. Freitas, E.J. Reed, Uncovering the effects of interface-induced ordering of liquid on crystal growth using machine learning. Nat. Commun. **11**(1), 3260 (2020). https://doi.org/10.1038/s41467-020-16892-4
56. W.F. Reinhart, Unsupervised learning of atomic environments from simple features. Comput. Mater. Sci. **196**, 110511 (2021). https://doi.org/10.1016/j.commatsci.2021.110511
57. C.P. Goodrich, E.M. King, S.S. Schoenholz, E.D. Cubuk, M.P. Brenner, Designing self-assembling kinetics with differentiable statistical physics models. Proc. Natl. Acad. Sci. USA **118**(10), 2024083118 (2021). https://doi.org/10.1073/pnas.2024083118
58. M. Spellings, Geometric algebra attention networks for small point clouds. arXiv:2110.02393 [cs] (2022). https://doi.org/10.48550/arXiv.2110.02393
59. L. Wang, Discovering phase transitions with unsupervised learning. Phys. Rev. B **94**(19), 195105 (2016). https://doi.org/10.1103/PhysRevB.94.195105
60. S.J. Wetzel, Unsupervised learning of phase transitions: from principal component analysis to variational autoencoders. Phys. Rev. E **96**(2), 022140 (2017). https://doi.org/10.1103/PhysRevE.96.022140
61. R.S. DeFever, C. Targonski, S.W. Hall, M.C. Smith, S. Sarupria, A generalized deep learning approach for local structure identification in molecular simulations. Chem. Sci. **10**(32), 7503–7515 (2019). https://doi.org/10.1039/C9SC02097G
62. Y. Wang, W. Deng, Z. Huang, S. Li, Descriptor-free unsupervised learning method for local structure identification in particle packings. J. Chem. Phys. **156**(15), 154504 (2022). https://doi.org/10.1063/5.0088056
63. C.R. Qi, H. Su, K. Mo, L.J. Guibas, PointNet: deep learning on point sets for 3D classification and segmentation. arXiv:1612.00593 [cs] (2017). https://doi.org/10.48550/arXiv.1612.00593
64. K. Swanson, S. Trivedi, J. Lequieu, K. Swanson, R. Kondor, Deep learning for automated classification and characterization of amorphous materials. Soft Matter **16**(2), 435–446 (2020). https://doi.org/10.1039/C9SM01903K
65. K.T. Schütt, H.E. Sauceda, P.-J. Kindermans, A. Tkatchenko, K.-R. Müller, SchNet - a deep learning architecture for molecules and materials. J. Chem. Phys. **148**(24), 241722 (2018). https://doi.org/10.1063/1.5019779
66. U. Gasser, E.R. Weeks, A. Schofield, P.N. Pusey, D.A. Weitz, Real-space imaging of nucleation and growth in colloidal crystallization. Science **292**(5515), 258–262 (2001). https://doi.org/10.1126/science.1058457
67. P.J. Steinhardt, D.R. Nelson, M. Ronchetti, Bond-orientational order in liquids and glasses. Phys. Rev. B **28**(2), 784–805 (1983). https://doi.org/10.1103/PhysRevB.28.784
68. W. Lechner, C. Dellago, Accurate determination of crystal structures based on averaged local bond order parameters. J. Chem. Phys. **129**(11), 114707 (2008). https://doi.org/10.1063/1.2977970
69. C.X. Du, G. Anders, R.S. Newman, S.C. Glotzer, Shape-driven solid-solid transitions in colloids. Proc. Natl. Acad. Sci. USA **114**(20), 3892–3899 (2017). https://doi.org/10.1073/pnas.1621348114
70. A.S. Keys, S.C. Glotzer, How do quasicrystals grow? Phys. Rev. Lett. **99**(23), 235503 (2007). https://doi.org/10.1103/PhysRevLett.99.235503
71. Y.-C. Hu, H. Tanaka, Revealing the role of liquid preordering in crystallisation of supercooled liquids. Nat. Commun. **13**(1), 4519 (2022). https://doi.org/10.1038/s41467-022-32241-z
72. M. Spellings, S.C. Glotzer, Machine learning for crystal identification and discovery. AIChE J. **64**(6), 2198–2206 (2018). https://doi.org/10.1002/aic.16157
73. C.S. Adorf, T.C. Moore, Y.J.U. Melle, S.C. Glotzer, Analysis of self-assembly pathways with unsupervised machine learning algorithms. J. Phys. Chem. B **124**(1), 69–78 (2020). https://doi.org/10.1021/acs.jpcb.9b09621
74. M.M. Martirossyan, M. Spellings, H. Pan, J. Dshemuchadse, Local structural features elucidate crystallization of complex structures. arXiv:2401.13765 [cond-mat] (2024). http://arxiv.org/abs/2401.13765

75. G.M. Coli, M. Dijkstra, An artificial neural network reveals the nucleation mechanism of a binary colloidal AB crystal. ACS Nano **15**(3), 4335–4346 (2021). https://doi.org/10.1021/acsnano.0c07541
76. E. Boattini, S. Marín-Aguilar, S. Mitra, G. Foffi, F. Smallenburg, L. Filion, Autonomously revealing hidden local structures in supercooled liquids. Nat. Commun. **11**(1), 5479 (2020). https://doi.org/10.1038/s41467-020-19286-8
77. E. Boattini, M. Dijkstra, L. Filion, Unsupervised learning for local structure detection in colloidal systems. J. Chem. Phys. **151**(15), 154901 (2019). https://doi.org/10.1063/1.5118867
78. S. De, A.P. Bartók, G. Csányi, M. Ceriotti, Comparing molecules and solids across structural and alchemical space. Phys. Chem. Chem. Phys. **18**(20), 13754–13769 (2016). https://doi.org/10.1039/C6CP00415F
79. A. Grisafi, D.M. Wilkins, G. Csányi, M. Ceriotti, Symmetry-adapted machine learning for tensorial properties of atomistic systems. Phys. Rev. Lett. **120**(3), 036002 (2018). https://doi.org/10.1103/PhysRevLett.120.036002
80. A. Gardin, C. Perego, G. Doni, G.M. Pavan, Classifying soft self-assembled materials via unsupervised machine learning of defects. Commun. Chem. **5**(1), 1–15 (2022). https://doi.org/10.1038/s42004-022-00699-z
81. M. Geiger, T. Smidt, e3nn: Euclidean neural networks. arXiv:2207.09453 [cs] (2022). https://doi.org/10.48550/arXiv.2207.09453
82. T.E. Smidt, M. Geiger, B.K. Miller, Finding symmetry breaking order parameters with Euclidean neural networks. Phys. Rev. Res. **3**(1), 012002 (2021). https://doi.org/10.1103/PhysRevResearch.3.L012002
83. J.D. Honeycutt, H.C. Andersen, Molecular dynamics study of melting and freezing of small Lennard–Jones clusters. J. Phys. Chem. **91**(19), 4950–4963 (1987). https://doi.org/10.1021/j10003a014
84. D. Faken, H. Jónsson, Systematic analysis of local atomic structure combined with 3D computer graphics. Comput. Mater. Sci. **2**(2), 279–286 (1994). https://doi.org/10.1016/0927-0256(94)90109-0
85. E. Urrutia Bañuelos, C. Contreras Aburto, A. Maldonado Arce, A common neighbor analysis of crystallization kinetics and excess entropy of charged spherical colloids. J. Chem. Phys. **144**(9), 094504 (2016). https://doi.org/10.1063/1.4943001
86. P.M. Larsen, S. Schmidt, J. Schiøtz, Robust structural identification via polyhedral template matching. Modell. Simul. Mater. Sci. Eng. **24**(5), 055007 (2016). https://doi.org/10.1088/0965-0393/24/5/055007
87. J. Chapman, T. Hsu, X. Chen, T.W. Heo, B.C. Wood, Quantifying disorder one atom at a time using an interpretable graph neural network paradigm. Nat. Commun. **14**(1), 4030 (2023). https://doi.org/10.1038/s41467-023-39755-0
88. B. Aroboto, S. Chen, T. Hsu, B.C. Wood, Y. Jiao, J. Chapman, Universal and interpretable classification of atomistic structural transitions via unsupervised graph learning. Appl. Phys. Lett. **123**(9), 094103 (2023). https://doi.org/10.1063/5.0156682
89. K. Choudhary, B. DeCost, Atomistic line graph neural network for improved materials property predictions. npj Comput. Mater. **7**(1), 1–8 (2021). https://doi.org/10.1038/s41524-021-00650-1
90. S. Banik, D. Dhabal, H. Chan, S. Manna, M. Cherukara, V. Molinero, S.K.R.S. Sankaranarayanan, CEGANN: crystal edge graph attention neural network for multiscale classification of materials environment. npj Comput. Mater. **9**(1), 1–12 (2023). https://doi.org/10.1038/s41524-023-00975-z
91. D.K. Duvenaud, D. Maclaurin, J. Aguilera-Iparraguirre, R. Gómez-Bombarelli, T. Hirzel, A. Aspuru-Guzik, R.P. Adams, Convolutional networks on graphs for learning molecular fingerprints. In: Advances in neural information processing systems, vol. 28. Curran Associates, Inc. (2015). https://papers.nips.cc/paper_files/paper/2015/hash/f9be311e65d81a9ad8150a60844bb94c-Abstract.html
92. T. Xie, J.C. Grossman, Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties. Phys. Rev. Lett. **120**(14), 145301 (2018). https://doi.org/10.1103/PhysRevLett.120.145301
93. V. Bapst, T. Keck, A. Grabska-Barwińska, C. Donner, E.D. Cubuk, S.S. Schoenholz, A. Obika, A.W.R. Nelson, T. Back, D. Hassabis, P. Kohli, Unveiling the predictive power of static structure in glassy systems. Nat. Phys. **16**(4), 448–454 (2020). https://doi.org/10.1038/s41567-020-0842-8
94. W.F. Reinhart, A.W. Long, M.P. Howard, A.L. Ferguson, A.Z. Panagiotopoulos, Machine learning for autonomous crystal structure identification. Soft Matter **13**(27), 4733–4745 (2017). https://doi.org/10.1039/C7SM00957G
95. K. Sheriff, Y. Cao, T. Smidt, R. Freitas, Quantifying chemical short-range order in metallic alloys. arXiv:2311.01545 [cond-mat] (2023). https://doi.org/10.48550/arXiv.2311.01545
96. A.G. Baydin, B.A. Pearlmutter, A.A. Radul, J.M. Siskind, Automatic differentiation in machine learning: a survey. J. Mach. Learn. Res. **18**, 1–43 (2018). https://jmlr.org/papers/v18/17-468.html
97. E.M. King, C.X. Du, Q.-Z. Zhu, S.S. Schoenholz, M.P. Brenner, Programmable patchy particles for materials design. arXiv:2312.05360 [cond-mat, physics:physics] (2023). https://doi.org/10.48550/arXiv.2312.05360
98. R. Krueger, E. King, M. Brenner, Tuning colloidal reactions. arXiv:2312.07798 [cond-mat, physics:physics] (2023). https://doi.org/10.48550/arXiv.2312.07798
99. Q.-Z. Zhu, C.X. Du, E.M. King, M.P. Brenner, Proofreading mechanism for colloidal self-assembly. arXiv:2312.08619 [cond-mat] (2023). https://doi.org/10.48550/arXiv.2312.08619
100. M.C. Engel, J.A. Smith, M.P. Brenner, Optimal control of nonequilibrium systems through automatic differentiation. Phys. Rev. X **13**(4), 041032 (2023). https://doi.org/10.1103/PhysRevX.13.041032