

Fast Bayesian Force Fields from Active Learning: Study of Inter-Dimensional Transformation of Stanene

Yu Xie,¹ Jonathan Vandermause,¹ Lixin Sun,¹ Andrea Cepellotti,¹ and Boris Kozinsky^{1,2,*}

¹*John A. Paulson School of Engineering and Applied Sciences,
Harvard University, Cambridge, MA 02138, USA*

²*Robert Bosch LLC, Research and Technology Center,
Cambridge, Massachusetts 02142, USA*

Abstract

We present a way to dramatically accelerate Gaussian process models for interatomic force fields based on many-body kernels by mapping both forces and uncertainties onto functions of low-dimensional features. This allows for automated active learning of models combining near-quantum accuracy, built-in uncertainty, and constant cost of evaluation that is comparable to classical analytical models, capable of simulating millions of atoms. Using this approach, we perform large scale molecular dynamics simulations of the stability of the stanene monolayer. We discover an unusual phase transformation mechanism of 2D stanene, where ripples lead to nucleation of bilayer defects, densification into a disordered multilayer structure, followed by formation of bulk liquid at high temperature or nucleation and growth of the 3D bcc crystal at low temperature. The presented method opens possibilities for rapid development of fast accurate uncertainty-aware models for simulating long-time large-scale dynamics of complex materials.

* bkoz@seas.harvard.edu

I. INTRODUCTION

Density functional theory (DFT) is one of the most successful methods for simulating condensed matter thanks to a reasonable accuracy for a wide range of systems. Ab-initio molecular dynamics (AIMD) offers a way to simulate the atomic motion using forces computed at the DFT level. Unfortunately, computational requirements limit the time-scale and size of AIMD simulations to a few hundred atoms for a few hundred picoseconds of time, precluding investigation of phase transitions and heterogeneous reactions. Such large-scale molecular dynamics (MD) simulation must resort to empirically derived analytical interatomic force fields with fixed functional form [6, 16, 30, 49], trading accuracy and transferability for larger length and time scales. Classical analytical force fields often do not match the accuracy of ab-initio results, limiting simulations to describing results qualitatively at best or, at worst, deviating from the correct behavior. In order to broaden the reach of computational simulations, it would be desirable to compute forces with ab-initio accuracy at the same cost of classical interatomic force fields.

In recent years, machine learning (ML) algorithms emerged as powerful tools in regression and classification problems. This interest has inspired several works to develop ML algorithms for interatomic force fields, for example neural-networks (NN) [7–11, 32, 44], MTP [24, 38, 43], FLARE [50], GAP [2–5], SNAP [48], SchNet [40–42], DeePMD [23, 51, 55], among others. All these machine learning potentials can make predictions at near ab-initio accuracy, while greatly reducing the computational cost compared to DFT.

However, most machine learning models have no predictive distributions which provide uncertainty for energy/force predictions. Without uncertainty, the training data is generally selected from ab-initio calculations via a manual or random scheme. Determining the reliability of the force fields then becomes difficult, which could result in untrustworthy configurations in the molecular dynamics simulations. Therefore, uncertainty quantification is a highly desirable capability [34, 36]. Neural network potentials, e.g. ANI [44, 45] can use ensemble disagreement as an uncertainty measure. This statistical approach is, however, not guaranteed to yield reliable calibrated uncertainty. In addition, NN approaches usually require tens of thousands of data for training, and are a few orders of magnitude slower than analytical force fields. Bayesian models are promising for uncertainty quantification in atomistic simulations since they have an internal principled uncertainty quantification

mechanism, the variance of the posterior prediction distribution, which can be used to keep track of the error on forces during a molecular dynamics (MD) run. For instance, Jinnouchi *et al.* [26, 27] used high-dimensional SOAP descriptors with Bayesian linear regression [4]. Gaussian Process (GP) regression [4, 5, 20–22] is a Bayesian method that has been shown to learn accurate forces with relatively small training data sets. Bartok *et al.* [3] used GP uncertainty with the GAP/SOAP framework to obtain only qualitative estimates of the force fields accuracy. Recently Vandermause *et al.* [50] demonstrated a GP-based Bayesian active learning scheme in the FLARE framework, utilizing rigorously principled and quantitatively calibrated uncertainty, applying it to a variety of multi-component materials.

In the most common form, however, GP models require using the whole training data set for prediction, meaning that the computational cost of prediction grows linearly with the size of the training set, and so accuracy increases together with computational cost. In complex multi-component structures, $O(10^3) \sim O(10^4)$ training data points are typically required to construct an accurate GP model, which makes predictions slow. To address this issue, Glielmo *et al.* [20–22] noted that, for a suitable choice of the n -body kernel function, it is possible to decompose GP force/energy prediction into a summation of low-dimensional functions. As a result, one can construct a parametric mapping of the n -body kernel function combined with a fixed training set. This approach reaches a constant scaling, which increases the speed of the GP model without accuracy loss.

In order to accelerate force prediction as well as the uncertainty quantification, enabling on-the-fly training for GP force fields, it is necessary to also map uncertainty onto low dimensional functions. However, mapping the prediction of uncertainty was not attempted in [20–22], since the mathematical form of uncertainty results in decomposition with twice the dimension of the corresponding mean functions, dramatically increasing the computational cost of their evaluation with spline interpolations. Thus, to date there is no method that combines high accuracy, modest training data requirement and fast prediction of both forces and their principled Bayesian uncertainty.

To address these issues, here we develop the *Mapped Gaussian Process (MGP)* method that implements an essentially lossless mapping of both force (mean) and uncertainty (variance) onto low-dimensional functions of atomic positions. Using dimensionality reduction techniques, we increase the efficiency of the MGP to make it applicable for large-scale (million atom) simulations, achieving speeds comparable to classical analytical force fields, several

orders of magnitude faster than available NN or full GP approaches. Quantified principled uncertainty allows us to implement a Bayesian active learning workflow, which automates and accelerates the training process and increases the efficiency of phase space exploration. As a result, the MGP method benefits from the capability of quantifying its uncertainty, while at the same time retaining cost independent of training size. To illustrate the benefits of this method, we incorporated the MGP force field in the parallel MD simulation code LAMMPS [37], and applied it to simulations of large-scale phase transformation dynamics.

As a test application, we focus on stanene, a 2D material that has recently gathered attention as a quantum spin Hall insulator [47]. Moreover, the only published force field for stanene is, to the best of our knowledge, the bond-order potential by Cherukara *et al.*[17], which is fitted to capture stanene’s low-temperature crystalline characteristics but, as common to many empirical interatomic potentials, suffers in accuracy near the melting temperature. In particular, we show that MGP is capable of rapidly learning to describe a wide range of temperatures, below and around the melting temperature, and that we can efficiently monitor the uncertainty on forces at each time step of the molecular dynamics, and use this capability to iteratively increase the accuracy of the force field by hierarchical active training. Using parallel simulations of large-scale structures, we characterize the unusual phase transition, where the 2D monolayer transforms to bcc bulk Sn at the temperature of 200K, and melts to ultimately form a 3D liquid phase at the temperature of 500K.

The application demonstrates that the MGP force field approaches the computational cost of an empirical analytical force field, retains nearly ab-initio accuracy, quantitative Bayesian uncertainty, and is capable of describing large-scale systems with complex physical processes such as phase transitions.

II. RESULTS

A. Bayesian Active Learning

In a MD simulation, it is likely that the system will evolve to atomic configurations that have not been seen before, and are therefore far from those in the training set. In this situation, the uncertainties of the predictions will grow to large values which may be considered unsatisfactory. Therefore, it is desirable to obtain an accurate estimate of the

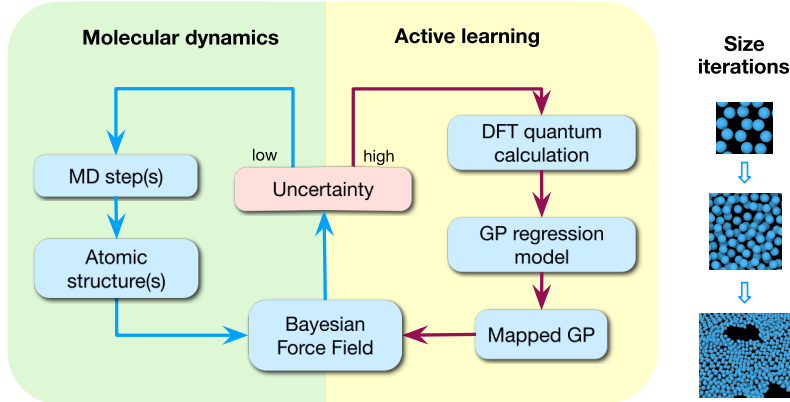


FIG. 1: Bayesian active learning (BAL) workflow with MGP. Iterations of BAL can be run for different system sizes to refine the force field.

forces for such configurations using a relatively expensive first principles calculation, and add this new information to the training set of the GP regression model.

This procedure is referred to as Bayesian Active Learning (BAL), where training examples are added on the fly as more information is obtained about the problem. Bayesian machine learning models such as GP are particularly well-suited to such uncertainty-based active learning approaches, as they provide a well-defined probability distribution over model outputs which can be used to extract rigorous estimates of model uncertainty for any test point [12, 33].

Here we adopt BAL to achieve automatic training of models for atomic forces, expanding on our earlier workflow [50]. This way the accuracy of the GP model increases with time, particularly in the configuration regions that are less explored and likely to be inaccurate.

As mentioned above, GP regression cost scales linearly with the training set size, so the prediction of forces and uncertainties becomes more expensive as the BAL algorithm keeps adding data to the training set, hindering simulations of complex materials systems. The MGP approach is based on an essentially exact mapping both mean and variance of the predicted forces from a full GP model for a fixed training set to a low-dimensional spline model. Mathematical formalism is detailed in the Methods section below. The novelty of this approach is the ability to accelerate prediction of not only the mean, but also GP’s internal uncertainty without loss of accuracy. MGP is incorporated into the BAL workflow as depicted in Fig.1. In this scheme, the mapping is constructed every time the GP model is updated. Atomic forces and their uncertainties for MD are produced by MGP, reducing the

computational cost of the full original GP prediction by orders of magnitude. We stress that not only is the MGP faster than the original GP, but also its speedup is more pronounced as the training data set size N_T increases. This is especially critical in complex systems with multiple atomic and molecular species and phases, where more training configurations are needed.

In addition, fast evaluation of forces and their variances enables training the force field model on larger system sizes. Therefore, we can train the MGP force field using a hierarchical scheme, i.e. for different system sizes, we can run several iterations of active learning. First, a small system size is used to train an MGP force field model using the BAL workflow. Then, we perform BAL on a larger system size, where DFT calculations are more expensive but are needed less frequently, in order to capture potentially new unexplored configurations such as defects that are automatically identified as high uncertainty by the model of the previous step. We note that DFT calculations scale as the 3rd power of the number of electrons in the system, so it is efficient to perform most of the training at smaller sizes, and to only require re-training on fewer large-scale structures. This way the hierarchical training scheme helps to overcome finite size effects and explore phenomena that cannot be captured with small system sizes, such as phase transformations.

B. Case Study: 2D to 3D Transformation of Stanene

To illustrate our method, we apply the mapped on-the-fly workflow (Fig.1) to study the phase transformation of stanene, a two-dimensional slightly buckled honeycomb lattice of tin. This structure has received much attention recently due to its unique topological electronic properties. Stanene can be synthesized by molecular beam epitaxy [56] and has been shown to have topologically protected states in the band gap [18]. Moreover, tri-layer stanene is a superconductor [29]. Here, we want to focus on the less studied aspects of thermodynamic stability and phase transition mechanisms of free standing stanene.

Despite the intense interest, capability is still lacking for accurate modeling of thermodynamic properties and phase stability of stanene, and most other 2D materials, for which \sim nm scale supercells are required. Such simulations are beyond the ability of DFT, and to the best of our knowledge, the only published classical force field for stanene is a bond-order potential (BOP) [17]. Our focus is on developing an accurate potential to model the 2D melting

process, which is a notoriously difficult problem to tackle with existing classical force fields. In fact, describing the melting mechanism of a 2D material is a highly non-trivial problem, since it can be characterized by specific behaviors such as defect formation and nucleation. For example, the melting of graphene is thought to occur in a temperature range between 4000K and 6000K [19, 31, 54], and seems to be characterized by the formation of defects, while above 6000K the structure melts directly into a totally disordered configuration. Since bulk tin melts at a much lower temperature ($\sim 500\text{K}$) than graphite ($\sim 4300\text{K}$), we expect that stanene will transform at a relatively low temperature. However, little is known about its melting mechanism and transition temperature. The MGP method is uniquely suited to describe the transformation processes, as it is scalable to large systems and capable of accurately describing any configuration that is sufficiently close to the training data.

1. Model Training

To train the force field for stanene, we start by training the MGP using the BAL loop on a small system size, and gradually increase the size of the simulation, iteratively improving upon the force field developed for the smaller size.

The DFT settings and the GP model hyperparameter settings are shown in the section 2.1 of supplementary material [53].

The first iteration of training is started with a $4 \times 4 \times 1$ supercell of stanene (32 atoms in total) and a 20 ps long MD simulation at a temperature of 50K. Subsequently, we increase the temperature of the system at 300K, 500K and 800K by velocity rescaling, and let the system evolve for 20, 30 and 30 ps respectively. This way we efficiently augment the training set with relevant configurations.

Fig.2b shows the mean square displacement (MSD) of atoms during the on-the-fly BAL training of the MGP model. Red dots indicate MD steps where DFT is called and new data is added to the training set, whereas black dots are steps where the MGP model is used to perform the MD steps. As one would expect, the DFT is called multiple times at the beginning of the MD simulation, since the MGP needs to build an initial training data set to make force predictions. Additionally, the DFT is called whenever temperature is increased, as the trajectory explores new configurations that have not been visited before. Clearly, the variance increases rapidly when temperature is increased.

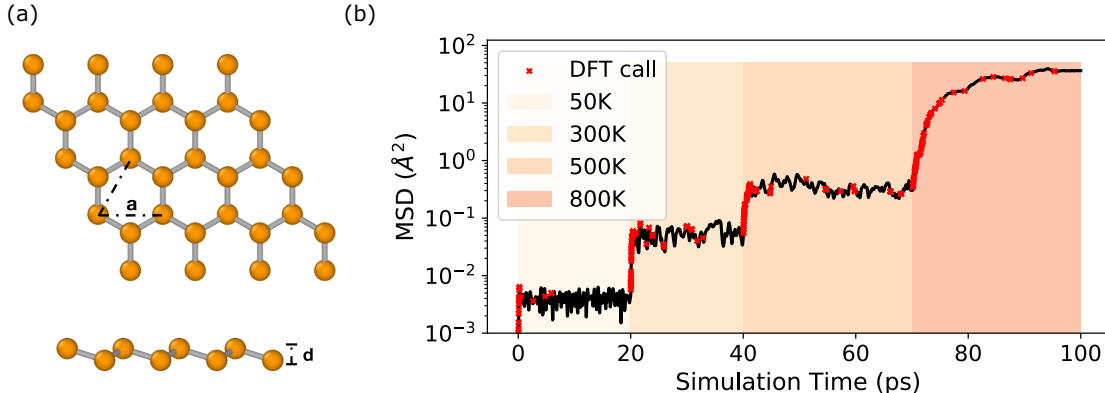


FIG. 2: (a) Pristine stanene, top view and front view. (b) Atomic mean square displacement (MSD) of the on-the-fly training trajectory. The solid black line shows the MSD of the 32 atoms, the color blocks indicate time intervals corresponding to different temperatures, and the red dots represent events when DFT calculations are called as the uncertainty predictions exceed the data acquisition decision threshold.

In this small simulation, the stanene monolayer decomposes at 800K, where Sn atoms start moving around the simulation cell. This allows us to augment the training set with such disordered configurations. However, the small size of the simulation cell does not yet allow us to draw conclusions about the melting process of stanene: for this, we need a larger simulation cell.

We continue with our hierarchical active learning scheme to refine the force field, but retaining the training data from the previous smaller MD run. We now increase the number of atoms in the simulation, which allows us to explore additional atomic configurations and improve the quality of the model. Since the system becomes larger and more computationally expensive, instead of updating the model at each learning step, we update it at the end of each MD simulation. We note that this "batching" approach is one of many possible training schemes and is chosen for reasons of efficiency. The procedure is as below:

1. Run MD simulation on a system of a given size with MGP;
2. Predict mapped uncertainties for the frames from the MD trajectory;
3. Select frames of highest uncertainties in different structures (crystal, transition state and disordered phase) for additional DFT calculations;

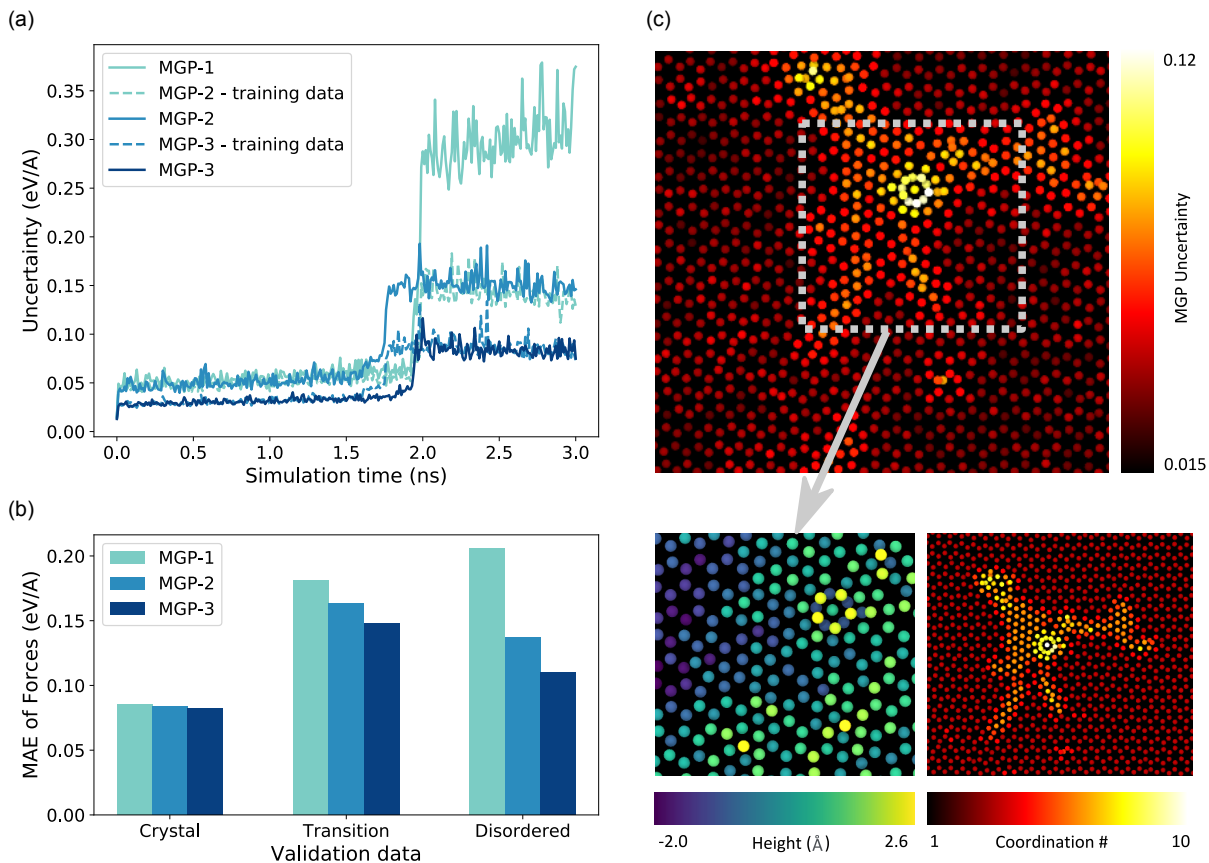


FIG. 3: (a) Uncertainty of the force predictions in the whole MD trajectory decreases in three iterations. (b) Mean absolute errors (MAE) of MGP forces against DFT in three iterations, validated on atomic structures of different phases. (c) Upper: A snapshot of 2048 atoms from MD simulation at 500K, colored by MGP uncertainty. The outlined area has highest predicted uncertainty, since it contains a defect and a compact region not known to the first MGP model. Lower-left: Zoomed-in structure of the region of high uncertainty, colored by atomic heights (z -coordinate) to show the bilayer structure of the defect. Lower-right: The same snapshot colored by coordination number, which shows strong correlation with the uncertainty measure.

4. Add atomic environments from those frames based on uncertainty to the training set of GP;
5. Construct a new MGP model using the updated GP;
6. Repeat the process for same or larger simulation using the new model.

The above steps form one iteration, and can be run several times to refine the force field. In our force field training for stanene, we performed one iteration of 32 atoms as described before, and two iterations all on the system size of 200 atoms at 500K, a relatively high temperature to explore diverse structures in the configuration phase space. We denote the MGP force field from the on-the-fly training of 32 atoms as MGP-1, the ones from the two iterations of training on 200 atoms as MGP-2 and MGP-3. MD are simulated by MGP-1, 2, 3 and the predicted uncertainties of the frames from the three MD trajectories predicted by MGP are shown in Fig.3a. First, we note that there is a drastic increase of uncertainty within each trajectory at a certain time. This increase corresponds to the transition from crystal lattice to a disordered phase, so the uncertainty acts as an automatic indicator of the structural change. Next, we note that the uncertainties of MGP-1, 2, 3 decrease iteration by iteration, indicating our model becomes more confident as it collects more data. The mean absolute error (MAE) of force predictions of MGP against DFT is investigated, as presented in Fig.3b, showing that MGP becomes also more accurate with each subsequent training iterations.

It is worth noting that in the transition process of the simulation by MGP-1, in the transition process there appear a bi-layer stacked defect and a specific compact triangular lattice around the defect. It is a visibly new ordered structure different from the hexagonal stanene, and is thus missing in the initial training set, as shown in Fig.3c. The uncertainty is low in honeycomb lattice, as expected, and becomes high around the bi-layer defect, exhibiting also a strong correlation with the coordination number. Thus, the mapped uncertainty is an accurate automatic indicator of the novelty of the new configurations. We note that after iterative model refinement by the hierarchical training scheme, the triangular lattice no longer appears in the simulation. This example illustrates how the hierarchical active learning scheme helps avoid qualitatively incorrect results that arise due to the a priori unknown relevant configurations. In contrast, a single-pass training of a force field based on manually chosen configurations is very likely to result in non-physical predictions.

2. Phase Transition Simulations

To investigate the full phase transition mechanism of stanene, we consider low temperature (below 200K) and high temperature (above 500K) values, corresponding to the crystal

and liquid phase for bulk tin. In studying the graphite-diamond transition, Khaliullin et al.[28] identified that nucleation and growth of the disordered phase play key roles in the phase transition process. We also find that in stanene the phase transition begins with the appearance of defects and the consequent nucleation and growth of the disordered phase that eventually transforms the structure.

From DFT ab-initio calculations, the ground state energy of honeycomb monolayer is ~ 0.38 eV/atom higher than the bcc bulk, so the 2D monolayer structure is expected to be metastable. The full transformation described below is therefore a sequence of 2D melting, followed by a kinetically controlled transition to the bulk phase of lower free energy. We set up the simulation of stanene monolayer of size $15.6\text{nm} \times 18.1\text{nm}$ (3000 atoms), and run MD in the NPT ensemble for 3 ns at the temperature of 200K. We discover that the ordered honeycomb monolayer loses long-range order and undergoes densification, and finally rearranges into the bcc bulk structure spontaneously, following three transformation steps. At first, the atoms vibrate around the honeycomb lattice sites, and the monolayer remains in its ordered 2D phase (Fig.4a - 0.3 ns). The main deviation from the ideal lattice is the out-of-plane rippling due to membrane fluctuations. The fluctuating ripples are similar to what was reported in simulations of graphene [19, 31, 54]. Next, double-layer stacking defects appear due to buckling and densification. These defect regions grow and quickly expand to the whole system, indicative of a first order phase transition in 2D. The atomic configuration appears to be a “disordered” phase (Fig.4a - 0.8 ns). In this stage, the structure becomes denser and increasingly three-dimensional driven by energetic gains from increasing Sn atom coordination, with the unit cell dimensions shrinking. Finally, patches of bcc lattice nucleate during the atomic rearrangement (Fig.4a - 1.6 ns), and those patches grow until the whole system forms the crystal bcc slab and atoms start vibrating around the lattice sites (Fig.4a - 2 ns).

The stable phase of bulk tin below 286.35K is expected to be the α -phase with the diamond structure, which is different from what we observe here. The reason for the formation of the bcc phase is kinetic rather than thermodynamic, caused by a lower free energy barrier separating the 3D disordered structure from the bcc phase than the α -phase, making the former phase more kinetically accessible. One well-known example is the graphite-diamond transition, in which the meta-stable hexagonal diamond (HD) is obtained from hexagonal graphite (HG) in most laboratory synthesis, instead of the more thermodynamically stable

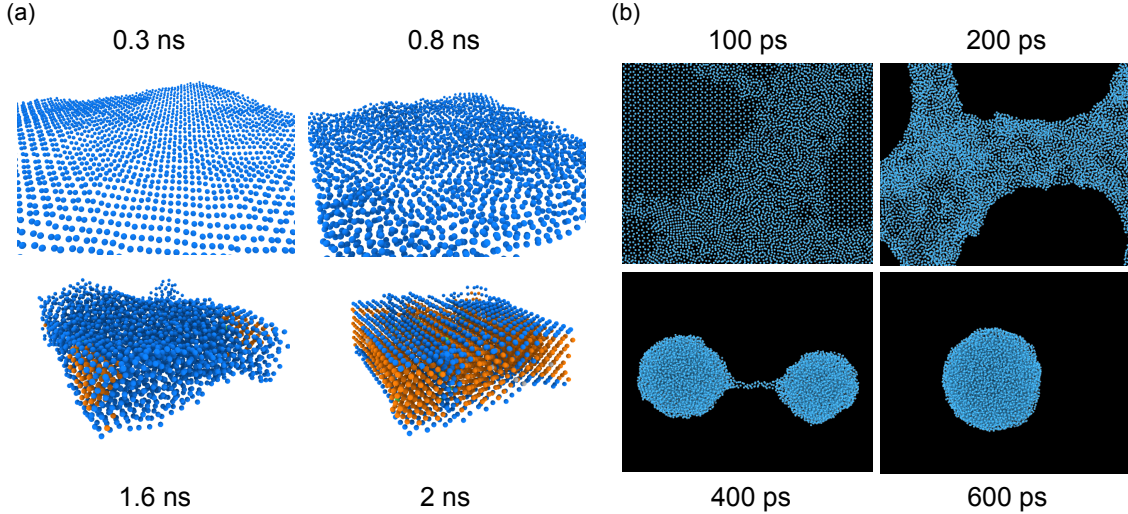


FIG. 4: (a) The monolayer - bulk transition at 200K. Using common neighbor analysis, the orange atoms are identified as bcc sites, and atoms of other colors are identified as other sites. The frame at 0.3 ns: the atoms are vibrating around the honeycomb lattice sites with out-of-plane ripples. The frame at 0.8 ns: the atoms arrange in the “disordered” way. The frame at 1.6 ns: some patches of bcc lattice structure appear. The frame at 2 ns: the atoms form and vibrate around the bcc slab sites. (b) Frames from 500K NPT simulation with 10,000 atoms ($39.4\text{nm} \times 22.7\text{nm}$) at 100, 200, 400 and 600 ps, showing the melting process of the monolayer.

cubic diamond (CD) [13–15, 25, 35]. Computational study [28] reveals that the higher in-layer distortions result in higher energy barrier for HG - CD than HG - HD transformation, under most of the experimental conditions. We observe that the 2D stacking order-disorder and the 3D crystallization transformation steps proceed via nucleation and growth, where the defect nuclei grow and turn the whole system into the intermediate disordered dense phase, while in the second step the bcc nuclei then appear from the disordered configuration and grow, turning the system into the bcc crystal phase.

To investigate the melting behavior at higher temperature, we focus on a 600 ps long MD run that starts from a pristine sample of stanene with size $39.4\text{nm} \times 22.7\text{nm}$ (10,000 atoms) at 500K. A few frames of this simulations are shown in Fig.4b. The melting process can also be identified as three stages. The first stage is the development of large out-of-plane ripples in the free-standing 2D lattice. In the second stage, at around 100 ps, double-layer defects in the monolayer form and agglomerate, causing densification. After 200 ps, densification

due to large patches of disordered structures lead to the formation of holes, and a large void area grows. We observe breaking of the 2-dimensionality of the crystal and formation of stacked configurations where atoms climb over other atoms of 2D stanene. In the third stage, the disordered stacked configurations shrink into spherical liquid droplets connected by a neck (400 ps). Subsequently the droplets merge to form a large sphere, minimizing the surface energy (600 ps).

In the supplementary material [53], we include more discussions and animations of the monolayer-bulk transition and the melting process.

C. Mapped Gaussian Process Performance

In this section, we discuss the computational cost of the MGP model in application to large scale MD simulations, and its advantages and limitations. Timing results are shown in Fig.5a, comparing different models including DFT, full GP (without mapping; implemented in Python), MGP (with mapped force and uncertainty; Python), MGP-F (MGP with force prediction only; Python), MGP (force only; implemented in LAMMPS) and BOP (LAMMPS force field). For DFT, we tested prediction time for system sizes of 32 atoms and 200 atoms, where computational cost growing as the cube of the system size. The other algorithms' cost scales linearly with system size, since they rely on local environments with finite cutoff radius, we measure the same *per atom* computational cost for both system sizes.

It is important to highlight the acceleration of the MGP relative to the original full GP. We present the timings of GP and MGP built from different training set sizes (100 and 400 training data) in Fig.5, where the GP scales linearly. MGP-F (without uncertainty) and MGP (with uncertainty, fixed rank) are independent of the training set size. The numerical result shows that the speedup of the MGP relative to the GP increases linearly with more data added to the training set, as reported in the Fig.S3 of the supplementary material [53]. In the stanene system discussed in Sec.II, MGP is two orders of magnitude faster than GP with $O(10^2)$ training data points when both force and variance are predicted, and the speedup becomes more significant as the GP collects more training data. We also note that although the prediction of MGP gets rid of the linear scaling with training set size, the construction of the mappings does scale linearly with training set size, since it requires GP to make prediction on each grid point.

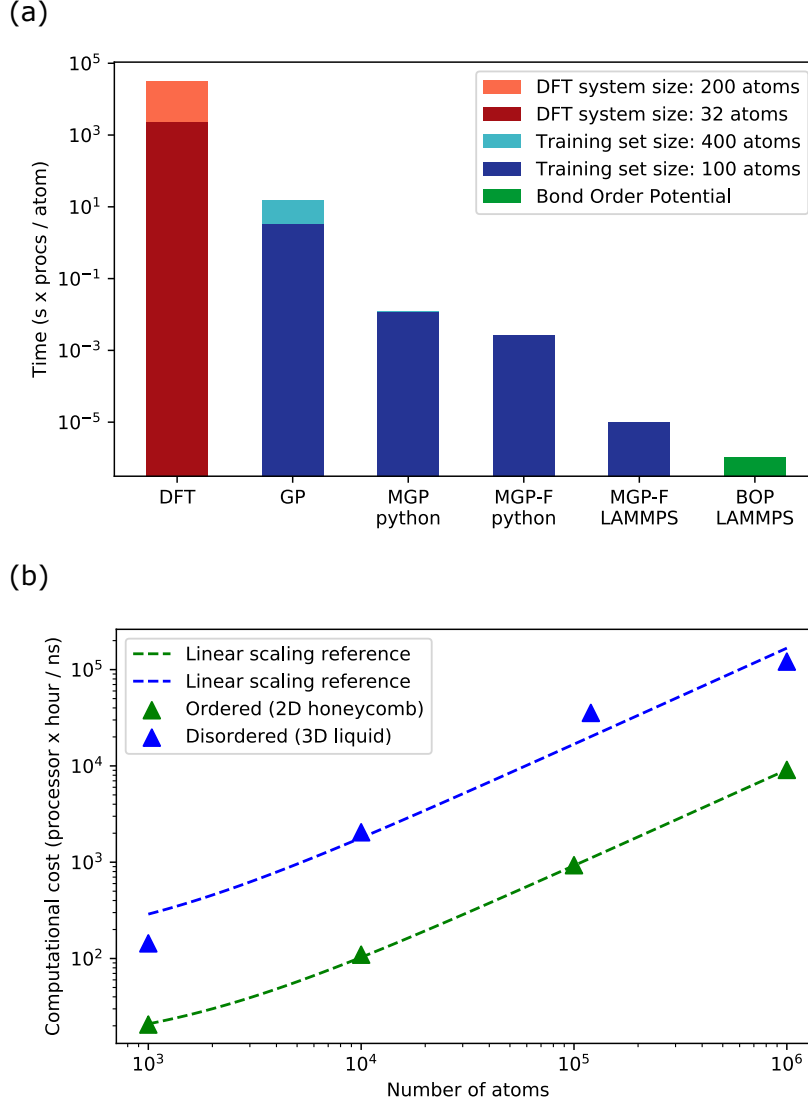


FIG. 5: (a) The comparison of prediction time (s \times processors/atom) between DFT, full GP, MGP (Python, with force and uncertainty), MGP-F (Python, with force only), MGP (LAMMPS), BOP (LAMMPS). (b) The scaling of MGP in LAMMPS with respect to different system sizes, in different phases. Green: 2D honeycomb lattice. Blue: 3D dense liquid.

The cost of the MGP force field is comparable to that of empirical interatomic BOP force field, since both BOP and MGP models consider interatomic distances within the local environment of a central atom and thus have similar computational cost at the same cutoff radius. For accurate simulations, however, the MGP uses a larger cutoff radius (7.2\AA) than what is used by the BOP (6\AA), and thus a simulation with our MGP potential for stanene

System size (atoms)	100,000
CPU number	256
Thermostat	NVT
Temperature (K)	100
BOP cutoff (Å)	6.0
MGP cutoff (Å)	7.2
BOP speed (atom · timestep/processor/s)	$\sim 2.3 \times 10^6$
MGP speed (atom · timestep/processor/s)	$\sim 5.0 \times 10^5$

TABLE I

is still slower by a factor of $4 \sim 7$ approximately, see our test of 1 million atoms in Table I. The computational cost of the MGP force field in LAMMPS as a function of the system size is shown in Fig.5b. The prefactor of the linear dependence depends on the number of pairs and triplets within a local environment. The scaling factor for the 2D honeycomb lattice structure is thus lower than for the dense 3D liquid. This is because each atom in the 3D liquid has more neighbors than the honeycomb lattice within the same cutoff radius.

The hierarchical active learning scheme adopted here is an efficient path to systematically improve the force field. However, it must be noted that the training is still limited by the system sizes that can be tackled with a DFT calculation. In fact, DFT calculations with thousands of atoms are unaffordable. To circumvent this problem, we note that regions of high uncertainties may be cropped out of the large MGP simulation cell, in order to perform DFT calculation on a smaller subset of atoms with a relevant structure. In this direction, we note that uncertainty is a local predicted property in MGP, indicating how each atom contributes to the total uncertainty. In the Fig.3c we show that the largest uncertainty in an MD frame of 2048 atoms comes from regions close to defects, since these configurations are missing in the training set for that particular model. Therefore, one may use the uncertainty to automatically select a section of the crystal close to the high-uncertainty region, and use a DFT simulation to add a training data point to the MGP. Automatically constructing such structures remains an open challenge in complex materials. However, access to the local uncertainty for each force prediction already allows for assessment of reliability of simulations, and is central to the new paradigm of Bayesian force fields.

Finally, we mention that the descriptive power of MGP is limited due to the lack of full many-body interactions in GP kernel, since the n -body kernel only uses low-dimensional functions of crystal coordinates. In this work, we have implemented the MGP for 2- and 3-body interactions, which, for the mapping problem, translate to interpolation problems in 3 dimensions. The approach may be extended to higher interaction orders at the expense of computational efficiency, but it becomes expensive in terms of both computation and storage requirements for splines on uniform grids to reach a satisfactory accuracy. Many-body descriptors can be taken into account to increase the descriptive power. To map it in the same way as the methodology we introduced, it is necessary to have a descriptor such that it can be decomposed into low-dimensional functions. For example, the SOAP approach of Refs. [26, 27] cannot be mapped using the methodology followed in this work, since it relies on a high-dimensional descriptor and, as a result, the MGP approach is not readily applicable. In particular, the interpolation procedure may be too computationally expensive to be a viable solution.

III. CONCLUSION

This work resolves a key computational cost scaling limitation of Gaussian Process regression of interatomic force fields in the case of many-body kernels and provides a new approach for Bayesian active learning, enabling robust automatic training of high accuracy interatomic force fields. We present an extended method for mapping the Gaussian Process regression model, where we accurately parameterize both the force and its variance obtained from the original Bayesian model as functions of atomic configurations. The mapped mean forces and their variance are then utilized in an active learning workflow to obtain a machine learned force field that includes its own uncertainty, the first realization of a fast *Bayesian force field*. The resulting model has near-quantum accuracy and computational cost comparable to empirical analytical interatomic potentials. Large scale simulations are used as a demonstration to investigate the microscopic details of inter-dimensional transformation behavior of 2D stanene into 3D bulk tin. We discover that the transformation proceeds by nucleation of bilayer defects, densification due to continued disordered multilayer stacking, and finally conversion into either crystalline or molten bulk tin, depending on temperature. This application shows that we can actively monitor the uncertainties of force predictions

during molecular dynamics simulations and iteratively improve the model as the material undergoes reorganization, exploring diverse structures in the configuration phase space. The ability to reach simulation sizes of over 1 million atoms is promising for application of machine learning force fields to study phase diagrams and transformation dynamics of complex systems, while automated active learning of force fields opens a way to accelerate wide-range material design and discovery.

IV. METHODS

The goal of the MGP method is to map the predictive mean and variance of n -body GP models onto low-dimensional splines, making the cost of evaluating atomic forces and uncertainty independent of the training set size, as also studied in previous works [20–22]. As mentioned in Ref. [22], decomposition of the GP kernel into low-dimensional functions exists for the GP predictive mean and variance. However, the resulting variance functions have twice the dimension of the corresponding mean functions, dramatically increasing the computational cost of their evaluation. In order to make such a mapping feasible in practice, we developed a dimension reduction technique that reduces the variance map to the same dimensions as mean, making it possible to use splines to evaluate both the mean and variance of the GP.

A. Background

1. Gaussian Process Force Field

In this section, we summarize the GP model introduced in Ref. [50]. From here on, we use bold letters to denote matrices/vectors and regular letters for scalars/numbers.

An *atomic configuration* is defined by atomic coordinates \mathbf{R}_α , a vector of dimension equal to the number of atoms N_{atoms} , and where α labels the three Cartesian directions $\alpha = x, y, z$. While in general the crystal may contain atoms of different species, we here consider only a single species case for the sake of simplicity. The results can be extended to the multi-component case, which is discussed in the section 1.1 of supplementary material [53].

The objective of the GP regression model is to predict the forces \mathbf{F}_α for an given atomic configuration \mathbf{R}_α . To this aim, we associate the forces of an atom i , $F_{i,\alpha}$, to the local environment around this atom ρ_i . $F_{i,\alpha}$ can then be computed by comparing its environment with a training data set of atomic environments with known forces, which is quantified by distance d between the target environment and the training set.

The *atomic environment* ρ_i of the atom i is defined as a set of atoms, including the atom i itself and all the other atoms that are within a sphere of a cutoff radius r_{cut} centered at the atom i . We proceed by partitioning this atomic environment ρ_i into a set of n -atom subsets, denoted by $\rho_i^{(n)}$. For each type of subset, a distance $d^{(n)}$ can be defined to quantify the difference between environments. We show examples of $n = 2$ and $n = 3$ on the construction of $\rho_i^{(n)}$ and $d^{(n)}$. For $n = 2$, $\rho_i^{(2)}$ is a set of pairs p , between central atom i and another atom i' in ρ_i ,

$$\rho_i^{(2)} = \{p = (i, i'); i' \in \rho_i, i' \neq i\}; \quad (1)$$

To each atomic pair p we associate an interatomic distance r_p between the two atoms. The distance metric between two pairs p and q is then defined as $(r_p - r_q)^2$. And the distance metric between two atomic environments $\rho_i^{(2)}$ and $\rho_j^{(2)}$ is defined as

$$(d_{ij}^{(2)})^2 = \sum_{p \in \rho_i^{(2)}, q \in \rho_j^{(2)}} (r_p - r_q)^2. \quad (2)$$

This distance $d^{(2)}$ will be used to determine the two-body contribution to the atomic forces, for the purposes of constructing the kernel function.

This construction can be extended to higher order interatomic interactions. In this work, we also include three-body interactions, corresponding to $n = 3$. Similar to the two-body case, the set $\rho_i^{(3)}$ of triplets p between the central atom i and two other atoms i' and i'' in ρ_i is defined as,

$$\rho_i^{(3)} = \{p = (i, i', i''); i', i'' \in \rho_i, i' \neq i, i'' \neq i, i' \neq i''\}. \quad (3)$$

This time, each triplet p is characterized by a vector $\mathbf{r}_p = \{r_{p,1}, r_{p,2}, r_{p,3}\}$, which represents the three interatomic distances of this atomic triplet p . Including permutation, the distance metric between two triplets p and q can be defined as $\sum_{u,v} (r_{p,u} - r_{q,v})^2$. And the 3-body distance metric between two atomic environments $\rho_i^{(3)}$ and $\rho_j^{(3)}$ is

$$(d_{ij}^{(3)})^2 = \sum_{p \in \rho_i^{(3)}, q \in \rho_j^{(3)}} \sum_{u,v} (r_{p,u} - r_{q,v})^2. \quad (4)$$

Having now defined a distance between atomic environments, we can build a GP regression model for atomic forces. We define the equivariant force-force kernel function to compare two atomic environments as

$$k_{\alpha\beta}(\rho_i, \rho_j) = \sum_n k_{\alpha\beta}^{(n)}(\rho_i^{(n)}, \rho_j^{(n)}) = \sum_n \frac{\partial^2 k^{(n)}(\rho_i^{(n)}, \rho_j^{(n)})}{\partial R_{i\alpha} \partial R_{j\beta}}, \quad (5)$$

where α and β label Cartesian coordinates, and $\frac{\partial}{\partial R_{i\alpha}}$ indicates the partial derivative of the kernel with respect to the position of atom i . The n -body energy kernel function $k^{(n)}(\rho_i^{(n)}, \rho_j^{(n)})$ is first expressed in terms of the partitions of n atoms of the two atomic environments ρ_i and ρ_j , and is defined as:

$$k^{(n)}(\rho_i^{(n)}, \rho_j^{(n)}) = \sum_{\substack{p \in \rho_i^{(n)} \\ q \in \rho_j^{(n)}}} \tilde{k}^{(n)}(p, q) = \sum_{\substack{p \in \rho_i^{(n)} \\ q \in \rho_j^{(n)}}} \sigma_{\text{sig}}^{(n)} \exp \left[- \frac{(d_{ij}^{(n)})^2}{2(l^{(n)})^2} \right] \varphi_{\text{cut}}^{(n)}, \quad (6)$$

where $\tilde{k}^{(n)}(p, q)$ is a Gaussian kernel function, built using the definition of distance between n -atoms subsets as described before. The $\sigma_{\text{sig}}^{(n)}$ and $l^{(n)}$ are hyperparameters independent of p and q , that set the signal variance related to the maximum uncertainty and the length scale, respectively. The smooth cutoff function $\varphi_{\text{cut}}^{(n)}$ dampens the kernel to zero as interatomic distances approach r_{cut} . In this work, we chose a quadratic form for the cutoff function,

$$\varphi_{\text{cut}}^{(2)}(p, q) = (r_q - r_{\text{cut}})^2 (r_p - r_{\text{cut}})^2, \quad (7)$$

$$\varphi_{\text{cut}}^{(3)}(p, q) = \prod_s (r_p^s - r_{\text{cut}})^2 \prod_t (r_q^t - r_{\text{cut}})^2. \quad (8)$$

The GP regression model uses training data as a reference set of N_T atomic environments. In particular, predictions of force values and variances require a $3N_T \times 3N_T$ covariance matrix

$$\Sigma = \Sigma_{i\alpha, j\beta} = k_{\alpha\beta}(\rho_i, \rho_j) + \sigma_n^2 \delta_{ij} \delta_{\alpha\beta} \quad (9)$$

where σ_n is a noise parameter (indices $i\alpha$ and $j\beta$ are grouped together). We define an auxiliary $3N_T$ dimensional vector $\boldsymbol{\eta}$ with components

$$\eta_{i\alpha} := (\Sigma^{-1})_{i\alpha, j\beta} y_{j\beta}, \quad (10)$$

where y contains the training labels, i.e., the forces $F_{j\beta}$ acting on the atom at the center of environment j in the Cartesian direction β .

In order to make a prediction of forces and their errors on a target structure ρ_i , we compute the Gaussian kernel vector $\bar{\mathbf{k}}_{i\alpha}$ for atom i and a direction α ($\alpha = x, y, z$), where the components of the vector are kernel distances $(\bar{k}_{i\alpha})_{j\beta} := k_{\alpha\beta}(\rho_i, \rho_j)$ between the prediction $i\alpha$ and the training set points $j\beta$, $j = 1, \dots, N_T$, $\beta = x, y, z$. Finally, the predictions of the mean value of the Cartesian component α of the force $F_{i\alpha}$ acting on the atom at the center of the atomic environment ρ_i and its variance $V_{i\alpha}$ are given by [39]:

$$F_{i\alpha} = \bar{\mathbf{k}}_{i\alpha}^\top \boldsymbol{\eta}, \quad (11)$$

$$V_{i\alpha} = k_{\alpha\alpha}(\rho_i, \rho_i) - \bar{\mathbf{k}}_{i\alpha}^\top \boldsymbol{\Sigma}^{-1} \bar{\mathbf{k}}_{i\alpha}. \quad (12)$$

We notice that this formulation has two major computational bottlenecks, that we intend to address in this work. First, the vector $\bar{\mathbf{k}}_{i\alpha}$ needs to be constructed for every force prediction; for each atom, the evaluation of $\bar{\mathbf{k}}_{i\alpha}$ requires the evaluation of the kernel between the atomic environment and the training data set, i.e. N_T calls to the Gaussian kernel. We discuss an approach to speed up the evaluation of $\bar{\mathbf{k}}_{i\alpha}$. Second, every variance calculation requires a matrix-vector multiplication that is computationally expensive, and thus will be further discussed in Sec.IV C.

B. Mapped Force Field

We start by observing that the GP model constructs a latent “weight field” with a fixed set of training data. In GP regression, a kernel function is used to build weights that depend on the distance from training data: training data that are similar (close) to the test data contribute with a larger weight, and vice versa, dissimilar (distant) training data contributes less. To make a prediction on a data point (force on an atom in its local environment), the GP regression model requires the evaluation of the kernel between such point and all the N_T training set data points. The result, *i.e.* the GP mean prediction, is a weighted average of the training labels, using the distances as measured by the kernel function as *weights*. In Eq. 11, we notice that the predictive mean requires the evaluation of a $3N_T$ -dimensional vector $\bar{\mathbf{k}}_{i\alpha}$, which quantitatively weighs the contribution to the test atomic environment ρ_i coming from the training atomic environments ρ_j .

In the following discussion, we consider a specific subset size n , capturing n -body atomic structure information, *i.e.* $(\bar{k}_{i\alpha}^{(n)})_{j\beta} = k_{\alpha\beta}^{(n)}(\rho_i^{(n)}, \rho_j^{(n)})$ without loss of generality. The same

treatment below can be implemented on each n and summed over the mappings of all " n "s to get the final prediction if we are using a kernel function of the form of Eq.5.

Starting with the idea introduced in Ref. [21], we can construct the weight field by considering the special form of the kernel at Eq. 6. In fact, such kernel definition allows us to decompose the force prediction into contributions from n -atom subsets, and to decompose the variance into contributions from pairs of n -atom subsets.

To see this, we first note that the kernel vector of atomic environments ρ_i can be decomposed into the kernel of all the n -atom subsets in ρ_i , and the kernel of an n -atom subset p can be further decomposed into the contributions from all the $n - 1$ neighbor atoms (except for the center atom i). We denote $\mathbf{r}_p^s := \mathbf{R}^s - \mathbf{R}^i$ as a bond in p , where s labels the remaining $n - 1$ neighbor atoms, and introduce the unit vector $\hat{r}_p^s := \mathbf{r}_p^s / r_p^s$ denoting the direction of the bond. Then the force kernel has decomposition

$$\bar{\mathbf{k}}_{i\alpha}^{(n)} = \sum_{p \in \rho_i} \tilde{\mathbf{k}}_{i\alpha}^{(n)}(p) = \sum_p \sum_s \boldsymbol{\mu}_s^{(n)}(p) \hat{r}_{p,\alpha}^s, \quad (13)$$

where

$$\left[\tilde{\mathbf{k}}_{i\alpha}^{(n)}(p) \right]_{j\beta} := \frac{\partial^2 \tilde{k}^{(n)}(p, q)}{\partial R_\alpha^i \partial R_\beta^j} \quad (14)$$

and

$$\left[\boldsymbol{\mu}_s^{(n)}(p) \right]_{j\beta} := \sum_{q \in \rho_j} \sum_t \frac{\partial^2 \tilde{k}^{(n)}(p, q)}{\partial r_p^s \partial r_q^t} \hat{r}_{q,\beta}^t \quad (15)$$

are vectors in \mathbb{R}^{3N_T} space. In the first equality of Eq.13 we used the decomposition of Eq. 6, and in the second equality we first convert the derivative from atomic coordinates to the relative distances of the n -atom subset, then take advantage of the fact that the kernel only depends on relative distances between atoms, i.e. $r_p^s = \|\mathbf{r}_p^s\|$.

Therefore, forces may be written as

$$F_{i\alpha}^{(n)} = \sum_{p \in \rho_i} \sum_s f_s^{(n)}(p) \hat{r}_{p,\alpha}^s. \quad (16)$$

where

$$f_s^{(n)}(p) := \boldsymbol{\mu}_s^{(n)\top}(p) \boldsymbol{\eta} \quad (17)$$

is the scalar "weight field" function and $\boldsymbol{\eta}$ is define in Eq.10.

As noted in Ref. [21], we can achieve a better scaling by constructing the weight field using a computationally efficient parametric method, i.e. an approximate way for constructing such functions f . In fact, a parametric function can be used to quickly yield the weight field value without the need of evaluating kernel distances from all the GP training points. As a result, the cost of evaluating the weight field using this parametric model is independent on the size of the training set, removing a computational bottleneck of the full GP implementation.

In order to achieve an effective parametrization of these weight functions, we use a cubic spline interpolation on a *uniform grid* of points, which benefits from having an analytic form and a continuous second derivative. By increasing the number of points in the uniform grid, the MGP prediction of mean and variance can be made arbitrarily close to the original functions.

The construction of the mapped force field includes the following steps:

Step 1: Define a uniform grid of points $\{x_k \in [a, b]^{\binom{n}{2}}\}$ in the space of distances given by the vectors \mathbf{r}_p . The lower bound $a \geq 0$ is chosen to be smaller than the minimal interatomic distance, and the upper bound b can be set to the cutoff radius of atomic neighborhoods, above which the prediction vanishes (using the zero prior of GP), and $\binom{n}{2} = n(n-1)/2$ is the number of interatomic distances in the n -atom subset.

Step 2: A GP model with a fixed training set yields the values $\{f_s^{(n)}(x_k)\}$ for each grid point x_k .

Step 3: Interpolate with a cubic spline function $\hat{f}_s^{(n)}$ through the points $\{(x_k^{(n)}, f_s^{(n)}(x_k))\}$.

In prediction, contributions from all n -bodies are added up, so the force of local environment ρ is predicted as:

$$\hat{F}_\alpha(\rho) = \sum_n \sum_{p \in \rho^{(n)}} \sum_s \hat{f}_s^{(n)}(p) \hat{r}_{p,\alpha}^s .$$

C. Mapped Variance Field

Using a derivation similar to that of the force decomposition (Eq.16), we derive the decomposition of the variance

$$V_{i\alpha} = \sum_{n,n'} \sum_{\substack{p \in \rho^{(n)} \\ q \in \rho^{(n')}}} \sum_{s,t} v_{s,t}^{(n,n')}(p, q) \hat{r}_{p,\alpha}^s \hat{r}_{q,\alpha}^t, \quad (18)$$

where

$$v_{s,t}^{(n,n')}(p, q) := \delta_{nn'} \tilde{k}_{\alpha\alpha}^{(n)}(p, q) - \boldsymbol{\mu}_s^{(n)}(p)^\top (\Sigma)^{-1} \boldsymbol{\mu}_t^{(n')}(q). \quad (19)$$

with the covariance matrix Σ defined in Eq.9.

We note that the domain of the variance function v has dimensionality twice that of f , so that when the interaction order $n > 3$, the number of grid points becomes prohibitively large to efficiently map v . Even for a 3-body kernel, when v is a 6-D function, we find that the evaluation of the variance, both in terms of computational time and memory footprint, limits its usage to simple benchmarks.

An efficient evaluation of the variance is critical for adopting Bayesian active learning. We now discuss a key result of this work and introduce an accurate yet efficient approximate mapping field for the variance. In particular, we focus on simplifying the vector $\boldsymbol{\mu}_s^{(n)}(p)$, which, from our tests, is the most computationally expensive term for predicting variance. In particular, since $\boldsymbol{\mu}_s^{(n)}(p)$ evaluates the kernel function between the test point p and all the training points, it scales linearly with the training set size. By implementing a mapping of the variance weight field, the cost of variance prediction depends only on the rank of our dimension reduction approach, and is independent of the training data set size.

To this aim, we first define the vector $\boldsymbol{\psi}_s^{(n)}(p) := \mathbf{L}^{-1} \boldsymbol{\mu}_s^{(n)}(p) \in \mathbb{R}^{3N_T}$, where \mathbf{L} is the Cholesky decomposition of Σ , i.e. $\Sigma = \mathbf{L}\mathbf{L}^\top$, so that Eq. 19 can be written as

$$v_{s,t}^{(n,n')}(p, q) = \delta_{nn'} \tilde{k}_{\alpha\alpha}^{(n)}(p, q) - \boldsymbol{\psi}_s^{(n)}(p)^\top \boldsymbol{\psi}_t^{(n')}(q). \quad (20)$$

The domain of vector function $\boldsymbol{\psi}_s^{(n)}(p)$ is half the dimensionality of that of $v_{s,t}^{(n,n')}(p, q)$.

As a first idea, since each component of $\boldsymbol{\psi}^{(n)}$ has domain of lower dimensionality, one may think of building $3N_T$ spline functions to interpolate the $3N_T$ components separately, which means the number of spline functions needed grows with the training set size.

To achieve a more efficient mapping of $\boldsymbol{\psi}_s^{(n)}$, we take advantage of the principal component analysis (PCA), a common dimensionality reduction method.

The construction of the mapped variance field includes the following steps:

Step 1: We start by evaluating the function values on the uniform grid $\{x_k \in [a, b]^{(n)}, k = 1, \dots, G\}$ as introduced in the section above. At each grid point x_k , we evaluate the vectors $\boldsymbol{\psi}_s^{(n)}(x_k)$ and use them to build the matrix $\mathbf{M}_s^{(n)} := (\boldsymbol{\psi}_s^{(n)}(x_1), \boldsymbol{\psi}_s^{(n)}(x_2), \dots, \boldsymbol{\psi}_s^{(n)}(x_G))^\top \in \mathbb{R}^{G \times 3N_T}$.

Step 2: In PCA we perform singular value decomposition (SVD) on $\mathbf{M}_s^{(n)}$, such that $\mathbf{M}_s^{(n)} = \mathbf{U}_s^{(n)} \boldsymbol{\Lambda}_s^{(n)} (\mathbf{V}_s^{(n)})^\top$, where $\mathbf{U}_s^{(n)} \in \mathbb{R}^{G \times 3N_T}$ and $\mathbf{V}_s^{(n)} \in \mathbb{R}^{3N_T \times 3N_T}$ have orthogonal columns, and $\boldsymbol{\Lambda}_s^{(n)} \in \mathbb{R}^{3N_T \times 3N_T}$ is diagonal.

Step 3: We reduce the dimensionality by picking the m largest eigenvalues of $\boldsymbol{\Lambda}_s^{(n)}$ and their corresponding eigenvectors from $\mathbf{U}_s^{(n)}$ and $\mathbf{V}_s^{(n)}$ (here we assume that eigenvalues are ordered). Then, we readily obtain a low-rank approximation as

$$\mathbf{M}_s^{(n)} \approx \bar{\mathbf{M}}_s^{(n)} = \bar{\mathbf{U}}_s^{(n)} \bar{\boldsymbol{\Lambda}}_s^{(n)} (\bar{\mathbf{V}}_s^{(n)})^\top, \quad (21)$$

with $\bar{\boldsymbol{\Lambda}}_s^{(n)} \in \mathbb{R}^{m \times m}$, $\bar{\mathbf{U}}_s^{(n)} \in \mathbb{R}^{G \times m}$, and $\bar{\mathbf{V}}_s^{(n)} \in \mathbb{R}^{3N_T \times m}$.

Step 4: Define $\boldsymbol{\phi}_s^{(n)} := (\bar{\mathbf{V}}_s^{(n)})^\top \boldsymbol{\psi}_s^{(n)}$, which is a vector in \mathbb{R}^m , and interpolate this vector between its values at the grid points $\{x_k\}$ (in detail, $\phi_{s,i}^{(n)}(x_k) = \bar{U}_{s,ij}^{(n)} \bar{\Lambda}_{s,j}^{(n)}$, for $i, j = 1, \dots, m$). As done previously for the force mapping, spline interpolation can now be used for approximating the $\boldsymbol{\phi}_s^{(n)}$ vector as well.

Therefore, the MGP prediction of the variance is given by

$$\hat{v}_{s,t}^{(n,n')}(p, q) := \delta_{nn'} \tilde{k}_{\alpha,\alpha}^{(n)}(p, q) - \boldsymbol{\phi}_s^{(n)}(p)^\top (\bar{\mathbf{V}}_s^{(n)})^\top \bar{\mathbf{V}}_t^{(n')} \boldsymbol{\phi}_t^{(n')}(q). \quad (22)$$

In this form, the variance would still require a multiplication between two matrices of size

$m \times 3N_T$. However, we note that the computational cost of total variance can be reduced

$$\hat{V}_\alpha(\rho) = k_{\alpha,\alpha}(\rho, \rho) - \left(\sum_n \sum_{p \in \rho^{(n)}} \sum_s \bar{\mathbf{V}}_s^{(n)} \phi_s^{(n)}(p) \hat{r}_{p,\alpha}^s \right)^2. \quad (23)$$

Nominally, the calculation of the second term in the variance still scales with the data set size N_T , since the size of $\bar{\mathbf{V}}^{(n)}$ depends on N_T . However, we find the cost of the vector-matrix multiplication to be negligible compared to evaluations of the self-kernel function (the 1st term in Eq.22). Therefore, we have formulated a way to significantly speed up the calculation of the variance, which only weakly depends on the training set size. In addition, by tuning the PCA rank m , we can optimize the balance between accuracy and efficiency, increasing the flexibility of this model.

V. ACKNOWLEDGEMENT

We thank Steven Torrisi and Simon Batzner for help with the development of the FLARE code. We thank Anders Johansson for the development of the KOKKOS GPU acceleration of the MGP LAMMPS pair style. We thank Dr. Nicola Molinari, Dr. Aldo Glielmo, Dr. Claudio Zeni and Dr. Sebastian Matera for useful discussions. Y.X. is supported by the US Department of Energy (DOE) Office of Basic Energy Sciences under Award No. DE-SC0020128. L.S. was supported by the Integrated Mesoscale Architectures for Sustainable Catalysis (IMASC), an Energy Frontier Research Center funded by the US Department of Energy (DOE) Office of Basic Energy Sciences under Award No. DE-SC0012573. A.C. is supported by the Harvard Quantum Initiative. J.V. is supported by Robert Bosch LLC and the National Science Foundation (NSF), Office of Advanced Cyberinfrastructure, Award No. 2003725.

VI. COMPETING INTERESTS

The authors declare no competing financial or non-financial interests.

VII. CODE AVAILABILITY

The algorithm for the MGP and mapped Bayesian active learning has been implemented in the open-source package FLARE [50] and is publicly available online [1].

VIII. DATA AVAILABILITY

The code used to generate results in this paper, and the related data are published in Materials Cloud Archive [46] with DOI: 10.24435/materialscloud:cs-tf [52].

IX. AUTHOR CONTRIBUTIONS

Y.X. developed the MGP variance method. J.V. implemented the FLARE Bayesian active learning framework, and Y.X. implemented MGP in the training workflow. L.S. and Y.X. developed the LAMMPS pair style for MGP. Y.X. trained the MGP models for stanene and performed molecular dynamics and analysis. A.C. contributed to the DFT calculations. B.K. initiated and supervised the work and contributed to algorithm development. Y.X. and B.K. worked on the manuscript, with contributions from all authors.

[1] <https://github.com/mir-group/flare>.

[2] Albert P Bartók and Gábor Csányi. Gaussian approximation potentials: A brief tutorial introduction. *International Journal of Quantum Chemistry*, 115(16):1051–1057, 2015.

[3] Albert P Bartók, James Kermode, Noam Bernstein, and Gábor Csányi. Machine learning a general-purpose interatomic potential for silicon. *Physical Review X*, 8(4):041048, 2018.

[4] Albert P Bartók, Risi Kondor, and Gábor Csányi. On representing chemical environments. *Physical Review B*, 87(18):184115, 2013.

[5] Albert P Bartók, Mike C Payne, Risi Kondor, and Gábor Csányi. Gaussian approximation potentials: The accuracy of quantum mechanics, without the electrons. *Physical review letters*, 104(13):136403, 2010.

[6] MI Baskes. Application of the embedded-atom method to covalent materials: a semiempirical potential for silicon. *Physical review letters*, 59(23):2666, 1987.

- [7] Jörg Behler. Atom-centered symmetry functions for constructing high-dimensional neural network potentials. The Journal of chemical physics, 134(7):074106, 2011.
- [8] Jörg Behler. Neural network potential-energy surfaces in chemistry: a tool for large-scale simulations. Physical Chemistry Chemical Physics, 13(40):17930–17955, 2011.
- [9] Jörg Behler. Constructing high-dimensional neural network potentials: A tutorial review. International Journal of Quantum Chemistry, 115(16):1032–1050, 2015.
- [10] Jörg Behler. Perspective: Machine learning potentials for atomistic simulations. The Journal of chemical physics, 145(17):170901, 2016.
- [11] Jörg Behler and Michele Parrinello. Generalized neural-network representation of high-dimensional potential-energy surfaces. Physical review letters, 98(14):146401, 2007.
- [12] Christopher M Bishop. Pattern recognition and machine learning. springer, 2006.
- [13] Viktor F Britun, Alexander V Kurdyumov, and Igor A Petrusha. Diffusionless nucleation of lonsdaleite and diamond in hexagonal graphite under static compression. Powder Metallurgy and Metal Ceramics, 43(1-2):87–93, 2004.
- [14] FP Bundy. Direct conversion of graphite to diamond in static pressure apparatus. The Journal of Chemical Physics, 38(3):631–643, 1963.
- [15] FP Bundy, WA Bassett, MS Weathers, RJ Hemley, HU Mao, and AF Goncharov. The pressure-temperature phase and transformation diagram for carbon; updated through 1994. Carbon, 34(2):141–153, 1996.
- [16] Kimberly Chenoweth, Adri CT Van Duin, and William A Goddard. Reaxff reactive force field for molecular dynamics simulations of hydrocarbon oxidation. The Journal of Physical Chemistry A, 112(5):1040–1053, 2008.
- [17] Mathew J Cherukara, Badri Narayanan, Alper Kinaci, Kiran Sasikumar, Stephen K Gray, Maria KY Chan, and Subramanian KRS Sankaranarayanan. Ab initio-based bond order potential to investigate low thermal conductivity of stanene nanostructures. The journal of physical chemistry letters, 7(19):3752–3759, 2016.
- [18] Jialiang Deng, Bingyu Xia, Xiaochuan Ma, Haoqi Chen, Huan Shan, Xiaofang Zhai, Bin Li, Aidi Zhao, Yong Xu, Wenhui Duan, et al. Epitaxial growth of ultraflat stanene with topological band inversion. Nature materials, 17(12):1081, 2018.
- [19] Eric Ganz, Ariel B Ganz, Li-Ming Yang, and Matthew Dornfeld. The initial stages of melting of graphene between 4000 k and 6000 k. Physical Chemistry Chemical Physics, 19(5):3756–

- 3762, 2017.
- [20] Aldo Glielmo, Peter Sollich, and Alessandro De Vita. Accurate interatomic force fields via machine learning with covariant kernels. Physical Review B, 95(21):214302, 2017.
- [21] Aldo Glielmo, Claudio Zeni, and Alessandro De Vita. Efficient nonparametric n-body force fields from machine learning. Physical Review B, 97(18):184307, 2018.
- [22] Aldo Glielmo, Claudio Zeni,  Fekete, and Alessandro De Vita. Building nonparametric n -body force fields using gaussian process regression. arXiv preprint arXiv:1905.07626, 2019.
- [23] Jiequn Han, Linfeng Zhang, Roberto Car, et al. Deep potential: A general representation of a many-body potential energy surface. arXiv preprint arXiv:1707.01478, 2017.
- [24] Max Hodapp and Alexander Shapeev. In operando active learning of interatomic interaction during large-scale simulations. arXiv preprint arXiv:2004.13158, 2020.
- [25] Tetsuo Irifune, Ayako Kurio, Shizue Sakamoto, Toru Inoue, and Hitoshi Sumiya. Correction: ultrahard polycrystalline diamond from graphite. Nature, 421(6925):806–806, 2003.
- [26] Ryosuke Jinnouchi, Ferenc Karsai, and Georg Kresse. On-the-fly machine learning force field generation: Application to melting points. arXiv preprint arXiv:1904.12961, 2019.
- [27] Ryosuke Jinnouchi, Jonathan Lahnsteiner, Ferenc Karsai, Georg Kresse, and Menno Bokdam. Phase transitions of hybrid perovskites simulated by machine-learning force fields trained on the fly with bayesian inference. Physical Review Letters, 122(22):225701, 2019.
- [28] Rustam Z Khaliullin, Hagai Eshet, Thomas D Khne, Jrg Behler, and Michele Parrinello. Nucleation mechanism for the direct graphite-to-diamond phase transition. Nature materials, 10(9):693–697, 2011.
- [29] Menghan Liao, Yunyi Zang, Zhaoyong Guan, Haiwei Li, Yan Gong, Kejing Zhu, Xiao-Peng Hu, Ding Zhang, Yong Xu, Ya-Yu Wang, et al. Superconductivity in few-layer stanene. Nature Physics, 14(4):344, 2018.
- [30] L Lindsay and DA Broido. Optimized tersoff and brenner empirical potential parameters for lattice dynamics and phonon thermal transport in carbon nanotubes and graphene. Physical Review B, 81(20):205441, 2010.
- [31] J. H. Los, K. V. Zakharchenko, M. I. Katsnelson, and Annalisa Fasolino. Melting temperature of graphene. Phys. Rev. B, 91:045415, Jan 2015.
- [32] Jonathan P Mailoa, Mordechai Kornbluth, Simon Batzner, Georgy Samsonidze, Stephen T Lam, Jonathan Vandermause, Chris Ablitt, Nicola Molinari, and Boris Kozinsky. A fast neural

- network approach for direct covariant forces prediction in complex multi-element extended systems. Nature Machine Intelligence, 1(10):471–479, 2019.
- [33] Kevin P Murphy. Machine learning: a probabilistic perspective. MIT press, 2012.
- [34] Felix Musil, Michael J Willatt, Mikhail A Langovoy, and Michele Ceriotti. Fast and accurate uncertainty estimation in chemical machine learning. Journal of chemical theory and computation, 15(2):906–915, 2019.
- [35] Hiroaki Ohfuji and Kiyoshi Kuroki. Origin of unique microstructures in nano-polycrystalline diamond synthesized by direct conversion of graphite at static high pressure. Journal of Mineralogical and Petrological Sciences, 104(5):307–312, 2009.
- [36] Andrew A Peterson, Rune Christensen, and Alireza Khorshidi. Addressing uncertainty in atomistic machine learning. Physical Chemistry Chemical Physics, 19(18):10978–10985, 2017.
- [37] Steve Plimpton. Fast parallel algorithms for short-range molecular dynamics. Journal of computational physics, 117(1):1–19, 1995.
- [38] Evgeny V Podryabinkin and Alexander V Shapeev. Active learning of linearly parametrized interatomic potentials. Computational Materials Science, 140:171–180, 2017.
- [39] Carl Edward Rasmussen. Gaussian processes in machine learning. In Summer School on Machine Learning, pages 63–71. Springer, 2003.
- [40] Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Saucedo Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. In Advances in Neural Information Processing Systems, pages 991–1001, 2017.
- [41] Kristof T Schütt, Huziel E Saucedo, P-J Kindermans, Alexandre Tkatchenko, and K-R Müller. Schnet—a deep learning architecture for molecules and materials. The Journal of Chemical Physics, 148(24):241722, 2018.
- [42] KT Schutt, Pan Kessel, Michael Gastegger, KA Nicoli, Alexandre Tkatchenko, and K-R Muller. Schnetpack: A deep learning toolbox for atomistic systems. Journal of chemical theory and computation, 15(1):448–455, 2018.
- [43] Alexander V Shapeev. Moment tensor potentials: A class of systematically improvable interatomic potentials. Multiscale Modeling & Simulation, 14(3):1153–1173, 2016.
- [44] Justin S Smith, Olexandr Isayev, and Adrian E Roitberg. Ani-1: an extensible neural network potential with dft accuracy at force field computational cost. Chemical science, 8(4):3192–

- 3203, 2017.
- [45] Justin S Smith, Ben Nebgen, Nicholas Lubbers, Olexandr Isayev, and Adrian E Roitberg. Less is more: Sampling chemical space with active learning. The Journal of chemical physics, 148(24):241733, 2018.
- [46] Leopold Talirz, Snehal Kumbhar, Elsa Passaro, Aliaksandr V Yakutovich, Valeria Granata, Fernando Gargiulo, Marco Borelli, Martin Uhrin, Sebastiaan P Huber, Spyros Zoupanos, et al. Materials cloud, a platform for open computational science. arXiv preprint arXiv:2003.12510, 2020.
- [47] Peizhe Tang, Pengcheng Chen, Wendong Cao, Huaqing Huang, Seymour Cahangirov, Lede Xian, Yong Xu, Shou-Cheng Zhang, Wenhui Duan, and Angel Rubio. Stable two-dimensional dumbbell stanene: A quantum spin hall insulator. Physical Review B, 90(12):121408, 2014.
- [48] Aidan Patrick Thompson, Laura Painton Swiler, Christian Robert Trott, Stephen Martin Foiles, and Garritt J. Tucker. Snap: Automated generation of quantum-accurate interatomic potentials. Journal of Computational Physics, 9 2014.
- [49] Adri CT Van Duin, Siddharth Dasgupta, Francois Lorant, and William A Goddard. Reaxff: a reactive force field for hydrocarbons. The Journal of Physical Chemistry A, 105(41):9396–9409, 2001.
- [50] Jonathan Vandermause, Steven B Torrisi, Simon Batzner, Alexie M Kolpak, and Boris Kozinsky. On-the-fly bayesian active learning of interpretable force-fields for atomistic rare events. Preprint at <https://arxiv.org/abs/1904.02042>, 2019.
- [51] Han Wang, Linfeng Zhang, Jiequn Han, and E Weinan. Deepmd-kit: A deep learning package for many-body potential energy representation and molecular dynamics. Computer Physics Communications, 228:178–184, 2018.
- [52] Yu Xie, Jonathan Vandermause, Lixin Sun, Andrea Cepellotti, and Boris Kozinsky. Fast bayesian force fields from active learning and mappedgaussian processes: Application to structural phase transition of stanene. <https://archive.materialscloud.org/record/2020.99>.
- [53] Yu Xie, Jonathan Vandermause, Lixin Sun, Andrea Cepellotti, and Boris Kozinsky. Supplementary material.
- [54] Kostyantyn V Zakharchenko, Annalisa Fasolino, JH Los, and MI Katsnelson. Melting of graphene: from two to one dimension. Journal of Physics: Condensed Matter, 23(20):202202, 2011.

- [55] Linfeng Zhang, Jiequn Han, Han Wang, Roberto Car, and E Weinan. Deep potential molecular dynamics: a scalable model with the accuracy of quantum mechanics. Physical review letters, 120(14):143001, 2018.
- [56] Feng-feng Zhu, Wei-jiong Chen, Yong Xu, Chun-lei Gao, Dan-dan Guan, Can-hua Liu, Dong Qian, Shou-Cheng Zhang, and Jin-feng Jia. Epitaxial growth of two-dimensional stanene. Nature materials, 14(10):1020, 2015.

Fast Bayesian Force Fields from Active Learning: Study of Inter-Dimensional Transformation of Stanene

Yu Xie,¹ Jonathan Vandermause,^{1,2} Lixin Sun,¹ Andrea Cepellotti,¹ and Boris Kozinsky^{1,*}

¹*John A. Paulson School of Engineering and Applied Sciences,
Harvard University, Cambridge, MA 02138, USA*

²*Department of Physics, Harvard University, Cambridge, MA 02138, USA*

Supplementary Materials

1. Methodology	2
1.1. Multi-Component Systems	2
2. Case Study: Stanene	2
2.1. Training Settings	2
2.2. Accuracy Validation	4
2.3. Phase Transition Process	5
2.3.1. Radial Distribution Function	5
2.3.2. Video of the Transition Process	5
3. Performance	6
3.1. MGP's Acceleration of Original GP	6
References	7

* bkoz@seas.harvard.edu

1. METHODOLOGY

1.1. Multi-Component Systems

In the main text, we derived the mathematical formulation in the context of single component. We note that our GP and MGP are natural to extend to multiple components, by differentiating combinations of species. As described in the main text, the n -body kernel compares two n -atom subsets $p \in \rho_i^{(n)}$, $q \in \rho_j^{(n)}$. When there are multiple species in the system, we define the kernel function $\tilde{k}^{(n)}(p, q)$ to be non-zero only when p and q have the same species in their corresponding atoms. To formulate more rigorously, denote \mathbf{s}_p is the sequence of species of atom is in p , we define the multi-component kernel as

$$\tilde{k}^{(n)}(p, q) = \delta_{\mathbf{s}_p, \mathbf{s}_q} \exp\left(-\frac{d^{(n)}(p, q)^2}{2(\lambda^{(n)})^2}\right) \varphi_{\text{cut}}^{(n)}(p, q),$$

Therefore, the mapped force and variance can be extended to multi-component system in the way that for each possible \mathbf{s}_p , we construct $f^{(n)}(p)$ and $\phi^{(n)}(p)$ the same as the single-component system that is described in the main text. In prediction, the force and variance contribution from n -atom subsets in the testing atomic environment are predicted by the spline functions corresponding to their species.

2. CASE STUDY: STANENE

2.1. Training Settings

We start constructing a $4 \times 4 \times 1$ supercell of stanene (32 atoms in total), a size that can be tackled with first-principles techniques. The lattice constant $a = 4.58\text{\AA}$ and buckling height $d = 0.83\text{\AA}$ are determined from the DFT structural optimization. The vacuum size of $c = 20\text{\AA}$ is used to make sure that the stanene monolayer is well isolated so that the periodic boundary condition in the z-direction has little influence on the result. We perform MD with the MGP force field using a time step of 1 fs, and the Velocity Verlet algorithm to integrate Newtons equations of motion, in the NVE ensemble.

We use Quantum-ESPRESSO [6], a plane-wave implementation of DFT, using an ultra-soft pseudo-potential with PBEsol[4] exchange-correlation functional for Sn. The k-point mesh is $4 \times 4 \times 1$, and the energy cutoff is 50 Ry, with charge density cutoff of 400 Ry.

Python is used in all our code except for DFT. FLARE [10] is our base framework of Bayesian active learning with GP, and the newly developed MGP is inserted into the workflow. The radial cutoffs for local environments are chosen as 7.2\AA , the right end of the fourth peak in the radial distribution function (RDF). The cutoffs are tuned from maximizing the objective function (maximize likelihood and minimize redundancy).

The initialization of signal strength (σ), length scale l and noise σ_n are shown in the table below, and are optimized to maximize the marginal likelihood function during the training, using the BFGS algorithm [2, 5, 7, 9]. More details on the hyperparameter optimization are explained in our earlier work [10].

Kernel	Cutoff (\AA)	Initial Hyps			Optimized Hyps		
		σ_{sig} (eV/ \AA)	l (\AA)	σ_n (eV/ \AA)	σ_{sig} (eV/ \AA)	l (\AA)	σ_n (eV/ \AA)
2-body	7.2	0.2	1.0	0.05	3.27e-2	0.518	0.140
3-body	7.2	1.0e-4	1.0	0.05	3.75e-5	0.691	0.140

TABLE I: GP hyperparameters

For MGP, the hyperparameters are shown in Table.SII. To interpolate a function, we need to choose a closed interval and determine the grid points in this interval. The number of grid points for the mapped spline models is chosen based on the trade-off between accuracy and efficiency of the interpolation. For 2-body, the interval is chosen as $[a, b]$, where a should be smaller than the minimal interatomic distance in MD simulation, and b is set to be the same as the 2-body cutoff of GP. For 3-body, we choose a “cube” space, where we find a closed interval in each dimension. Since 3-body describes triplets, i.e. the triangle formed by three atoms, that can be determined by the two bonds connected to the center atom, r_1 , r_2 , and the angle $a_{12} = \langle \hat{r}_1, \hat{r}_2 \rangle$ between r_1 and r_2 , we use (r_1, r_2, a_{12}) as grid points to build up interpolation function. So the interval for r_1 and r_2 should be chosen the same as 2-body, only the upper bound is changed to the 3-body cutoff of GP. For the angle, the lower bound and upper bound are fixed to be 0 and π , which are not varied in different systems and not chosen by users.

The 1-D and 3-D cubic spline functions come from Github repository *interpolation.py* [1].

	grid number	lower bound	upper bound
2-body	144	2.35 Å	7.2 Å
3-body	$128 \times 128 \times 128$	2.35 Å, 2.35 Å, 0 rad	7.2 Å, 7.2 Å, π rad

TABLE II: MGP hyperparameters

2.2. Accuracy Validation

After obtaining the refined MGP model from the above iterative training process, it is necessary to validate the accuracy of the force field and the uncertainty prediction.

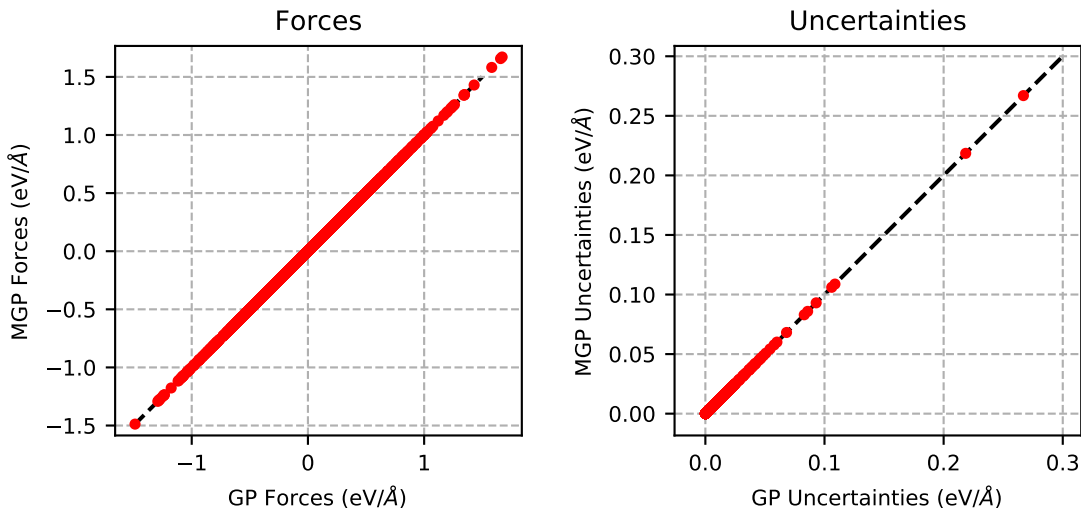


FIG. 1: MGP prediction versus GP

First, because the MGP is an approximation to the original GP, we must verify that MGP force and uncertainty prediction match those of the GP. In Fig.S1, we show that the mean absolute error (MAE) for force and uncertainty are both in the order 10^{-6} eV/Å, and can therefore conclude that the MGP has a negligible loss of accuracy compared to the GP.

Next, we compare the performance of the MGP against the BOP force field of Ref. [3]. In particular, we validate results using frames of different atomic structures taken from an independent MD trajectory, and study the mean absolute error of forces with respect to the DFT prediction. Since the MGP force field is trained with PBEsol exchange-correlation functional, ultra-soft pseudopotential, of the platform Quantum Espresso (labelled as DFT “I”), while the BOP potential is trained with PBE functional, PAW pseudopotential of

VASP (labelled as DFT “II”), we make a comparison of the mean absolute error of force prediction with both DFT “I” and “II” as the ground truth. In Table.SIII we show that the MGP achieves a better accuracy than the BOP in all of these frames, by at least 54%. As temperature is increased, and as the deviation from the crystal phase increases, both force fields have larger MAE. This is because the system at high temperature explores many more different configurations, likely different from the training set, nevertheless, the results show that the MGP has uniformly better accuracy.

	Crystal		Transition		Molten	
	I	II	I	II	I	II
DFT						
BOP	0.254	0.241	0.322	0.376	0.258	0.305
MGP	0.082	0.120	0.148	0.206	0.110	0.139

TABLE III: Mean absolute error (MAE) of the force prediction compared to DFT, in units of eV/Å.

2.3. Phase Transition Process

2.3.1. Radial Distribution Function

The radial distribution function (RDF) of the slab configuration from 200K MD simulation is shown in Fig.2 (top), where the black dash line is the RDF of the perfect bcc lattice of tin [8].

The radial distribution function of 600 ps frame in the melting process simulation is shown in Fig.2 (bottom), which shows one peak at $\sim 3 \text{ \AA}$ and a tail that converges to a uniform density. This indicates that the final cluster becomes a 3D disordered dense liquid phase.

2.3.2. Video of the Transition Process

The videos can be downloaded from DOI: 10.24435/materialscloud:cs-tf [11].

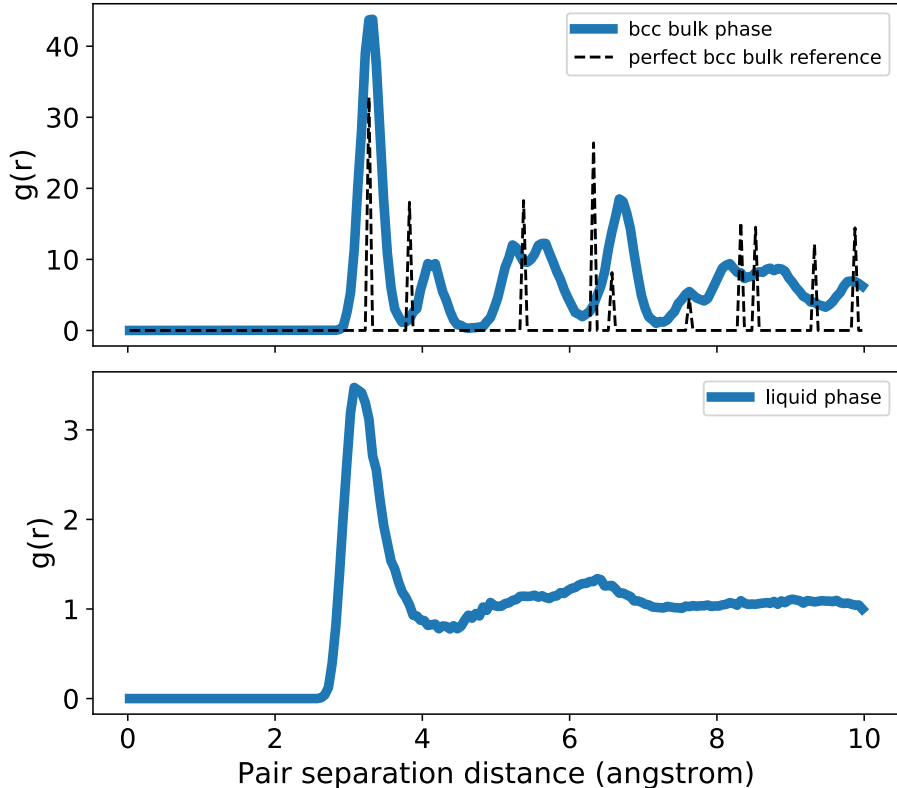


FIG. 2: Radial distribution function (RDF) of 3D slab and liquid

3. PERFORMANCE

3.1. MGP's Acceleration of Original GP

Since the original GP is a non-parametric method, it depends on the training data set completely, thus its prediction cost scales linearly with respect to the training data set size. After the force field is trained and mapped, MGP is obtained as a parametric model, whose basis is a group of cubic spline functions, and can be directly employed independent of the original GP regression model as well as training data set, thus reaches constant scaling of computational cost at prediction.

The major computational bottleneck in the prediction of the current scheme lies in two parts. One is the construction of the local atomic environment, which requires calculating interatomic distances and identifying all n -atom subsets. The other is the calculation of the first term in the variance formulation, the self-kernel function $k_{\alpha\alpha}(\rho, \rho)$ which scales quadratically with the number of n -atom subsets in ρ , while the second term, the vectorized

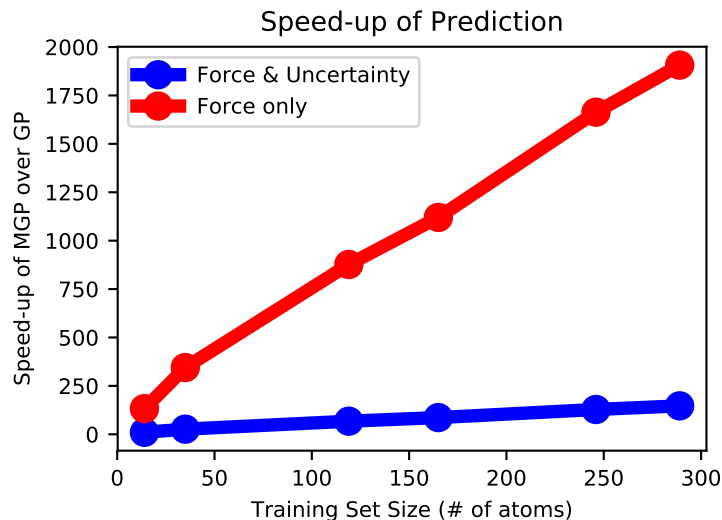


FIG. 3: Speed up of MGP compared to GP, which grows linearly w.r.t. the training set size. Prediction of both force and uncertainty (blue line) gets 2 orders of magnitude of acceleration when GP has $O(10^2)$ training data, while force prediction without uncertainty is 3 orders of magnitude faster.

spline function ϕ , scales linearly. We note the situation is different in the original GP, where the construction of atomic environment and the evaluation of the self-kernel are negligible compared to the evaluation of the kernel vector. In this way the present mapping scheme for both forces and uncertainties eliminated the main computational bottleneck of the full GP model.

-
- [1] <https://github.com/EconForge/interpolation.py>.
 - [2] Charles G Broyden. The convergence of a class of double-rank minimization algorithms: 2. the new algorithm. *IMA journal of applied mathematics*, 6(3):222–231, 1970.
 - [3] Mathew J Cherukara, Badri Narayanan, Alper Kinaci, Kiran Sasikumar, Stephen K Gray, Maria KY Chan, and Subramanian KRS Sankaranarayanan. Ab initio-based bond order potential to investigate low thermal conductivity of stanene nanostructures. *The journal of physical chemistry letters*, 7(19):3752–3759, 2016.
 - [4] Gábor I Csonka, John P Perdew, Adrienn Ruzsinszky, Pier HT Philipsen, Sébastien Lebègue,

- Joachim Paier, Oleg A Vydrov, and János G Ángyán. Assessing the performance of recent density functionals for bulk solids. *Physical Review B*, 79(15):155107, 2009.
- [5] Roger Fletcher. A new approach to variable metric algorithms. *The computer journal*, 13(3):317–322, 1970.
- [6] Paolo Giannozzi, Stefano Baroni, Nicola Bonini, Matteo Calandra, Roberto Car, Carlo Cavazzoni, Davide Ceresoli, Guido L Chiarotti, Matteo Cococcioni, Ismaila Dabo, et al. Quantum espresso: a modular and open-source software project for quantum simulations of materials. *Journal of physics: Condensed matter*, 21(39):395502, 2009.
- [7] Donald Goldfarb. A family of variable-metric methods derived by variational means. *Mathematics of computation*, 24(109):23–26, 1970.
- [8] Materials Project. Crystal structure of bcc sn (mp-7162). <https://materialsproject.org/materials/mp-7162>.
- [9] David F Shanno. Conditioning of quasi-newton methods for function minimization. *Mathematics of computation*, 24(111):647–656, 1970.
- [10] Jonathan Vandermause, Steven B Torrisi, Simon Batzner, Alexie M Kolpak, and Boris Kozinsky. On-the-fly bayesian active learning of interpretable force-fields for atomistic rare events. *Preprint at <https://arxiv.org/abs/1904.02042>*, 2019.
- [11] Yu Xie, Jonathan Vandermause, Lixin Sun, Andrea Cepellotti, and Boris Kozinsky. Fast bayesian force fields from active learning and mappedgaussian processes: Application to structural phase transition of stanene. <https://archive.materialscloud.org/record/2020.99>.