Exploring the parameter dependence of atomic minima with implicit differentiation

Ivan Maliyov,[1,*] Petr Grigorev,[1] and Thomas D Swinburne[1,†]

[1] *Aix-Marseille Université, CNRS, CINaM UMR 7325, Campus de Luminy, Marseille 13288, France*

Interatomic potentials are essential to go beyond *ab initio* size limitations, but simulation results depend sensitively on potential parameters. Forward propagation of parameter variation is key for uncertainty quantification, whilst backpropagation has found application for emerging inverse problems such as fine-tuning or targeted design. Here, the implicit derivative of functions defined as a fixed point is used to Taylor expand the energy and structure of atomic minima in potential parameters, evaluating terms via automatic differentiation, dense linear algebra or a novel sparse operator approach. The latter allows efficient forward and backpropagation through relaxed structures of arbitrarily large systems. The implicit expansion accurately predicts lattice distortion and defect formation energies and volumes with classical and machine-learning potentials, enabling high-dimensional uncertainty propagation without prohibitive overhead. We then show how the implicit derivative can be used to solve challenging inverse problems, minimizing an implicit loss to fine-tune potentials and stabilize solute-induced structural rearrangements at dislocations in tungsten.

I. INTRODUCTION

Atomic simulations employing interatomic potentials are an essential tool of computational materials science [1]. Classical models such as Lennard-Jones potentials are central to the study of glasses and polymer systems, whilst modern, data-driven models are becoming quantitative surrogates for *ab initio* calculations [2]. For solid-state materials, the energy landscape of relaxed atomic geometries is central to exploring thermodynamic, diffusive and mechanical properties [3–6].

Regardless of the interatomic potential employed, changing parameters will change any quantity of interest extracted from a simulation. For simple classical models, parameter variation is essential to explore the model's phenomenology [5, 7]. For modern data-driven models which target quantitative accuracy, parameter uncertainties can be estimated by Bayesian regression on training data [8, 9] and should be forward propagated to simulation results to bound quantities of interest [10].

Any forward propagation scheme must account for the strong correlation between individual energy or force evaluations when calculating e.g. formation energies or dynamical averages, as these will strongly affect (typically *reduce*) uncertainty on the final simulation result. Backpropagation of structural modifications to changes in parameters is finding application in training interatomic potentials from experimental data [11] or tuning simple interatomic potentials to reproduce desired self-assembly kinetics [7]. More recently, 'universal' machine learning potentials have shown near-quantitative accuracy across large portions of the periodic table [12, 13]. This has raised interest in back-propagation for fine-tuning universal models for specific applications [14], and opened the possibility of targeted design through navigation of a smooth latent composition representation [15].

Forward and backpropagation of parameter variation through complex simulations is typically achieved using reverse-mode automatic differentiation (AD) routines, which offer high arithmetic efficiency at the price of large memory requirements [16–18]. Whilst AD offers clear advantages in implementation, their significant memory burden complicates application to large atomic systems, especially when targeting higher derivatives beyond forces. Existing methods to propagate variation in parameters to variation in simulation results thus typically employ resampling: new parameters are drawn from some distribution [10] and simulations are repeated. Whilst conceptually straightforward, back-propagation is not possible, assessing convergence is challenging, and the cost can be potentially very large, especially for simulation results requiring geometry minimization for each potential sample.

In this paper, we explore an alternative approach, analytically expanding the structure of relaxed minima to first order in parameter variation, giving a second order expansion of the total energy. The expansion is achieved through evaluation of an *implicit* derivative [19], i.e. the derivative of a function defined as fixed point, in this case, the atomic structure of a local minimum (figure 1a). Our main results are that the implicit derivative enables 1) forward propagation of parameter uncertainties to simulation results for orders of magnitude less computational effort than resampling schemes, allowing rapid propagation of parameter uncertainties and 2) back-propagation of structural variations to target composition-induced structural rearrangements in multi-thousand-atom systems, a challenging task for any other approach (see illustration of these two ideas in figures 1b and 1c).

We implement and compare methods to evaluate the implicit derivative using AD [17, 18], dense linear algebra and a novel sparse operator approach, using the `jax-md` [16] and `LAMMPS` [20] simulation codes. We find

* ivan.maliyov@cnrs.fr
† thomas.swinburne@cnrs.fr

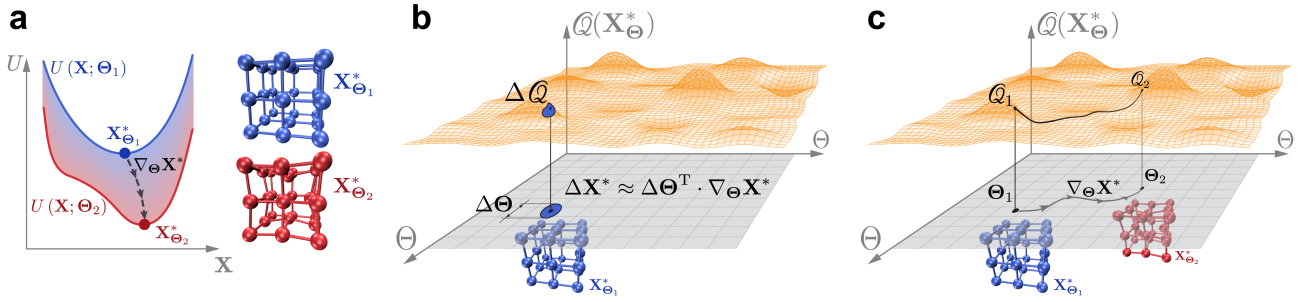

FIG. 1. **Schematic of the implicit derivative approach in atomic systems.** **a** A local minima $\mathbf{X}^*$ will change under changes in potential parameters $\boldsymbol{\Theta}$. The implicit derivative is the gradient of this change, allowing a first-order prediction of structural variations and a second-order prediction of energy variations. **b** Uncertainty quantification application. Having the uncertainty in the parameter space, $\Delta\boldsymbol{\Theta}$, implicit derivative allows one to compute the uncertainty of any molecular statics property $\mathcal{Q}$. **c** Inverse design application. With initial configuration $\mathbf{X}^*_{\boldsymbol{\Theta}_1}$ and potential $\boldsymbol{\Theta}_1$, one defines a target configuration $\mathbf{X}^*_{\boldsymbol{\Theta}_2}$. Implicit derivative allows one to effectively minimize the potential parameters that provide the target configuration.

AD routines for the implicit derivative reach GPU memory limits for ∼1000-atom systems even on best-in-class A100 hardware. In contrast, our sparse operator reduces to a constrained minimization in `LAMMPS`, allowing memory-efficient and highly parallelized evaluation. This innovation allows uncertainty quantification and inverse design studies with the large atomic simulations essential to capture realistic defect structures.

For the purposes of backpropagation, our expansion can be used for any form of interatomic potential. However, for the forward propagation in uncertainty quantification, the expansion captures parameter variation in the vicinity of some minimum of the loss. In this paper we therefore focus on classical [16, 21, 22] and linear-in-descriptor interatomic potentials [8, 23–26] whose loss typically has a well defined global minimum rather than the multi-modal loss landscape of neural network potentials [5, 12, 27].

The paper is structured as follows. We first define the implicit derivative, the Taylor expansion approximations used and their evaluation using automatic differentiation or linear algebra techniques. We then describe how the implicit derivative can be used in forward and backpropagation of parameter variations. Forward propagation of parameter variation is demonstrated using classical and machine learning potentials to explore lattice distortion and vacancy defect formation [28]. Finally, we demonstrate back-propagation of parameter variations, finding parameter variations which stabilize subtle solute-induced dislocation core reconstructions in tungsten [4].

II. RESULTS

A. Implicit derivative of atomic configurations

We consider a system of N atoms in a periodic supercell of volume V, with atomic coordinates $\mathbf{X} \in \mathbb{R}^{N\times 3}$ and a supercell matrix $\mathbf{C} \in \mathbb{R}^{3\times 3}$, $V = \mathrm{Det}(\mathbf{C})$. Changes to the supercell $\mathbf{C} \to \mathbf{C} + \delta\mathbf{C}$ are defined to induce homogeneous deformations, as in e.g. energy volume curves. As a result, we work with scaled atomic coordinates $\tilde{\mathbf{X}}$, such that $\mathbf{X} \equiv \tilde{\mathbf{X}}\mathbf{C}$ and the tuple $(\tilde{\mathbf{X}}, \mathbf{C})$ fully defines atomic configurations with periodic boundary conditions and fixed atomic species. With a vector of N_D potential parameters $\boldsymbol{\Theta}$, a potential energy model $U(\tilde{\mathbf{X}}, \mathbf{C}; \boldsymbol{\Theta})$, has stationary configurations $(\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}}, \mathbf{C}^*_{\boldsymbol{\Theta}})$ satisfying

$$\nabla_{\tilde{\mathbf{X}}} U(\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}}, \mathbf{C}^*_{\boldsymbol{\Theta}}; \boldsymbol{\Theta}) \equiv \mathbf{0}, \quad \nabla_{\mathbf{C}} U(\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}}, \mathbf{C}^*_{\boldsymbol{\Theta}}; \boldsymbol{\Theta}) \equiv 0 \quad (1)$$

where $(\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}}, \mathbf{C}^*_{\boldsymbol{\Theta}})$ is one of the exponentially many stationary points in the energy landscape [5]. In the following, we only consider minima; extension to the treatment of saddle points will be presented in future work. Under a parameter variation $\boldsymbol{\Theta} + \delta\boldsymbol{\Theta}$ the scaled positions and supercell matrix are defined to change as

$$\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}+\delta\boldsymbol{\Theta}} = \tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}} + \delta\boldsymbol{\Theta}\nabla_{\boldsymbol{\Theta}}\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}} + \mathcal{O}(\delta\boldsymbol{\Theta}^2), \quad (2)$$

$$\mathbf{C}^*_{\boldsymbol{\Theta}+\delta\boldsymbol{\Theta}} = \mathbf{C}^*_{\boldsymbol{\Theta}} + \delta\boldsymbol{\Theta}\nabla_{\boldsymbol{\Theta}}\mathbf{C}^*_{\boldsymbol{\Theta}} + \mathcal{O}(\delta\boldsymbol{\Theta}^2), \quad (3)$$

where $\nabla_{\boldsymbol{\Theta}}\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}} \in \mathbb{R}^{N_D\times N\times 3}$ and $\nabla_{\boldsymbol{\Theta}}\mathbf{C}^*_{\boldsymbol{\Theta}} \in \mathbb{R}^{N_D\times 3\times 3}$ are *implicit* derivatives that determine how a *stationary* configuration changes with the variation of potential parameters. In practical applications, variation $\mathbf{C}$ is typically constrained; for simplicity, we will only consider fixed-volume simulations or isotropic variations controlled by a homogeneous strain $\epsilon^*_{\boldsymbol{\Theta}} \in \mathbb{R}$ around a reference supercell $\mathbf{C}_0$

$$\mathbf{C}^*_{\boldsymbol{\Theta}} = [1 + \epsilon^*_{\boldsymbol{\Theta}}]\,\mathbf{C}_0, \quad \nabla_{\boldsymbol{\Theta}}\mathbf{C}^*_{\boldsymbol{\Theta}} = (\nabla_{\boldsymbol{\Theta}}\epsilon^*_{\boldsymbol{\Theta}})\mathbf{C}_0, \quad (4)$$

where $\nabla_{\boldsymbol{\Theta}}\epsilon^*_{\boldsymbol{\Theta}} \in \mathbb{R}^{N_D}$. Taylor expanding (1) to first order in $\delta\boldsymbol{\Theta}$, it is simple to show $\nabla_{\boldsymbol{\Theta}}\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}}, \nabla_{\boldsymbol{\Theta}}\epsilon^*_{\boldsymbol{\Theta}}$ solve the system of linear equations

$$\begin{bmatrix}\nabla_{\boldsymbol{\Theta}}\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}} & \nabla_{\boldsymbol{\Theta}}\epsilon^*_{\boldsymbol{\Theta}}\end{bmatrix}\begin{bmatrix}\nabla^2_{\tilde{\mathbf{X}}\tilde{\mathbf{X}}}U & \nabla^2_{\tilde{\mathbf{X}}\epsilon}U \\ \nabla^2_{\tilde{\mathbf{X}}\epsilon}U^\top & \nabla^2_{\epsilon\epsilon}U\end{bmatrix} = -\begin{bmatrix}\nabla^2_{\boldsymbol{\Theta}\tilde{\mathbf{X}}}U \\ \nabla^2_{\boldsymbol{\Theta}\epsilon}U\end{bmatrix}, \quad (5)$$

where $\nabla^2_{\tilde{\mathbf{X}}\tilde{\mathbf{X}}}U$ is the Hessian matrix in scaled coordinates, $\nabla^2_{\epsilon\epsilon}U$ is proportional to the bulk modulus of the

system and $-\nabla^2_{\epsilon\tilde{\mathbf{X}}}U$ is proportional to change in atomic forces under a homogeneous strain and $\nabla^2_{\boldsymbol{\Theta}\tilde{\mathbf{X}}}U$, $\nabla^2_{\boldsymbol{\Theta}\epsilon}U$ are mixed curvatures. Whilst solution of (5) in principle requires $\mathcal{O}(N^3)$ effort due to the Hessian, we introduce a novel Hessian-free solution method below, allowing application to large systems.

B. Taylor expansion of stationary energies and volumes using implicit derivatives

In our numerical experiments, we will compare three levels of approximate solution to the linear equations (5):

- *constant (c)*: $\nabla_{\boldsymbol{\Theta}}\epsilon^*_{\boldsymbol{\Theta}} = 0$, $\nabla_{\boldsymbol{\Theta}}\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}} = \mathbf{0}$
- *homogeneous (h)*: $\nabla_{\boldsymbol{\Theta}}\epsilon^*_{\boldsymbol{\Theta}} \neq 0$, $\nabla_{\boldsymbol{\Theta}}\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}} = \mathbf{0}$
- *inhomogeneous (ih)*: $\nabla_{\boldsymbol{\Theta}}\epsilon^*_{\boldsymbol{\Theta}} = 0$, $\nabla_{\boldsymbol{\Theta}}\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}} \neq \mathbf{0}$,

with the full expansion then given the shorthand *h+ih*. Under changes in parameters $\boldsymbol{\Theta}$, changes in the stationary energy and volume (or equivalently strain) admit the implicit Taylor expansions

$$\delta^{(\zeta)}U^* \equiv \delta\boldsymbol{\Theta}\nabla_{\boldsymbol{\Theta}}U + \delta\boldsymbol{\Theta}\mathbf{H}_\zeta\delta\boldsymbol{\Theta}^\top + \mathcal{O}(\delta\boldsymbol{\Theta}^3) \quad (6)$$

$$\delta^{(\zeta)}\epsilon^* \equiv \delta\boldsymbol{\Theta}\nabla_{\boldsymbol{\Theta}}\epsilon^*_\zeta + \mathcal{O}(\delta\boldsymbol{\Theta}^2), \quad (7)$$

where $\mathbf{H}_\zeta \in \mathbb{R}^{N_D\times N_D}$ is a generalized curvature within a given level of approximation and $\zeta = c, h, ih, h+ih$ is the level of approximation used. Expressions for $\nabla_{\boldsymbol{\Theta}}\epsilon^*_\zeta$ and $\mathbf{H}_\zeta$ are given in the SM.

Whilst all approaches predict changes in energy, only $\zeta = h, ih, h+ih$ predict changes in structure. For constant volume relaxations, the inhomogeneous $\zeta = ih$ expansion will be asymptotically exact. For variable volume relaxations, the full $\zeta = h+ih$ expansion will be asymptotically exact, but as we show below the cheaper homogeneous $\zeta = h$ expansion can also give accurate results if we are primarily interested in changes to the energy or volume, rather than structure.

C. Evaluation of the implicit derivative through sparse and dense linear algebra methods

In the linear equations (5), the $\nabla^2_{\epsilon\epsilon}U$ and $\nabla^2_{\epsilon\tilde{\mathbf{X}}}U$ derivatives in (5) require only a few $\mathcal{O}(N)$ force calls for evaluation. As a result, the *homogeneous* approximation requires minimal computational effort, but all knowledge of structural changes is missing as $\nabla_{\boldsymbol{\Theta}}\tilde{\mathbf{X}}^*$ is not evaluated. Evaluation of $\nabla_{\boldsymbol{\Theta}}\tilde{\mathbf{X}}^*$ for the *inhomogeneous* approach requires $\mathcal{O}(N^2)$ finite difference evaluation of the Hessian matrix $\nabla^2_{\tilde{\mathbf{X}}\tilde{\mathbf{X}}}U$ and $\mathcal{O}(N^3)$ solution of the dense linear equation (5). Whilst of reasonable cost for small systems ($N < 2000$), study of extended defects where $10^4 < N < 10^6$ requires significant, typically prohibitive, computational resources and careful use of shared memory parallel linear algebra techniques [6].

To overcome this limitation, we note that the Hessian matrix is highly sparse due to the strong locality of atomic forces. In this regime, efficient solutions of the linear equation (5) can be obtained using iterative algorithms such as `gmres`. In addition, such algorithms do not require access to every element of the Hessian at each iteration, only a linear operator which gives the action of the Hessian on some vector $\mathbf{V}$, i.e. $\mathcal{L}(\mathbf{V}) = \mathbf{V}\nabla^2_{\tilde{\mathbf{X}}\tilde{\mathbf{X}}}U$.

Avoiding direct Hessian evaluation can give a much faster time-to-solution. We define the operator

$$\mathcal{L}(\mathbf{V}) \equiv \lim_{\alpha\to 0} \nabla_{\tilde{\mathbf{X}}}U(\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}} + \alpha\mathbf{V}, \mathbf{C}^*_{\boldsymbol{\Theta}}; \boldsymbol{\Theta})/\alpha, \quad (8)$$

which clearly limits to $\mathcal{L}(\mathbf{V}) = \mathbf{V}\nabla^2_{\tilde{\mathbf{X}}\tilde{\mathbf{X}}}U$ as desired and only requires $\mathcal{O}(N)$ force calls for evaluation, repeated for each vector $[\nabla_{\boldsymbol{\Theta}}\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}}]_l \in \mathbb{R}^{N\times 3}$, $l \in [1, N_D]$ of the implicit derivative. Details of our $\mathcal{O}(NN_D)$ massively parallel method to compute $\nabla_{\boldsymbol{\Theta}}\tilde{\mathbf{X}}^*$ in `LAMMPS` [20] through the solution of (5) using (8) at finite values of α is described in methods section IV A. In section II H we apply the method to enable use of the implicit derivative in large atomic systems.

D. Evaluation of the implicit derivative with automatic differentiation methods

AD-enabled simulation schemes such as `jax-md` [16] can clearly evaluate all terms in equation (5) or the sparse operator approach (8). Recently, implicit differentiation schemes have been implemented in `jaxopt` [17, 18], allowing direct evaluation of e.g. $\nabla_{\boldsymbol{\Theta}}\tilde{\mathbf{X}}^*$ by differentiating through the minimization algorithm chosen in `jax-md`. We have implemented and tested all approaches in AD for the binary Lennard-Jones system described below. Despite the simplicity of the potential form, using AD to evaluate terms in (5) or using AD implicit derivative schemes incurs extremely large memory usage, reaching the 80GB limit on A100 GPUs for only a few thousand atoms, as we detail in the supplementary material (SM). We thus conclude that existing AD schemes for direct evaluation of the implicit derivative or Hessian matrices are ill-suited for application to the thousand-atom systems essential for many materials science problems. In contrast, our sparse operator (8) has the same memory usage as any structural minimization. Whilst still incurring a significantly greater memory burden than non-AD methods implemented in `LAMMPS`, our sparse operator is ideal for implicit derivative evaluation in AD-enabled schemes. Investigation of how (8) can be used with neural network-based interatomic potentials[12] is left for a future study.

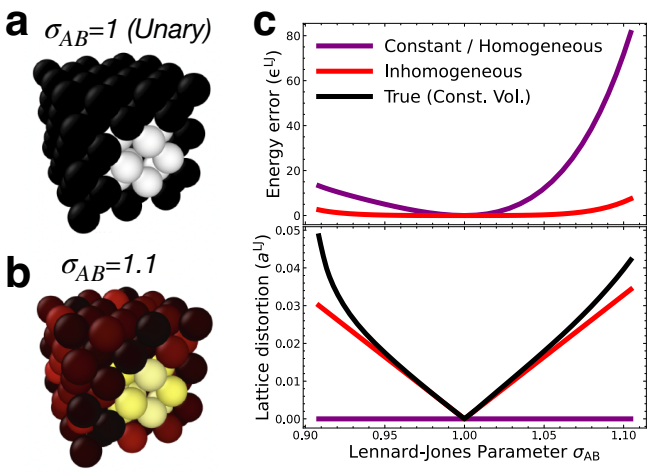

FIG. 2. **Automatic implicit differentiation with classical potentials. a**, **b** Atomic configurations of a single vacancy configuration with $\sigma_{\mathrm{AB}} = 1$ (unary) and $\sigma_{\mathrm{AB}} = 1.1$ (random binary). Atom color: fcc centrosymmetry $\in [0, 1.25]$. **c** Relaxed energy for $\sigma_{\mathrm{AB}} \in [0.9, 1.1]$ for a fixed volume, where *constant* and *homogeneous* approximations are identical. The *inhomogeneous* expansion accounts for changes in $\tilde{\mathbf{X}}$, giving much more accurate energy predictions.

E. Backpropagation of parameter variations for inverse design problems

Inverse design aims to produce materials with specified (desirable) properties through inverting structure-property relationships. However, this typically requires high-throughput searches which necessarily cannot afford to perform atomistic simulations of e.g. defect structures and mechanisms to predict mechanical properties. The implicit derivative can be used to make a first step towards gradient-led inverse design, allowing us to find interatomic parameters $\boldsymbol{\Theta}$ that stabilise some atomic configuration $\mathbf{X}^*$ observed in an *ab initio*-accurate simulation e.g. DFT or hybrid DFT-ML [4]. With $\mathbf{X}^*_{\boldsymbol{\Theta}}$ being the local minimum found when minimizing $U(\mathbf{X}, ; \boldsymbol{\Theta})$ starting from $\mathbf{X}^*$, we can write an implicit loss function

$$L(\boldsymbol{\Theta}) = \frac{1}{2}\left[\mathbf{X}^*_{\boldsymbol{\Theta}} - \mathbf{X}^*\right] : \left[\mathbf{X}^*_{\boldsymbol{\Theta}} - \mathbf{X}^*\right] \tag{9}$$

where we use the notation : to indicates summation over the N atomic sites and 3 spatial indices. The implicit derivative is necessary to compute the derivative of the loss via the chain rule:

$$\nabla_{\boldsymbol{\Theta}} L(\boldsymbol{\Theta}) = \nabla_{\boldsymbol{\Theta}} \mathbf{X}^*_{\boldsymbol{\Theta}} : \left[\mathbf{X}^*_{\boldsymbol{\Theta}} - \mathbf{X}^*\right] \in \mathbb{R}^{N_D}. \tag{10}$$

An immediate application of (9) is the ability to 'fine-tune', or retrain[4], interatomic potentials to reproduce important DFT minima. Fine-tuning has gained increasing interest following the rise of 'foundation model' machine learning potentials [13], which are beyond the scope of this study. In practice, one typically includes the loss against the original training database to regularize the fine-tuning fit. However, we have found that minimizing (9) in practice produces only very small perturbations to the final potential.

The ability to fine-tune interactions is of particular relevance for low energy structures such as dislocation lines[4, 8], which typically require careful weighting in potential fitting[13]. In section II H, we use implicit loss minimization to find solute substitutions which induce 'hard' screw dislocation core reconstruction in tungsten, and fine-tune a tungsten-beryllium potential to correctly reproduce *ab initio* observations [4].

F. Prediction of the total relaxed energy in classical potentials with automatic differentiation

The binary Lennard-Jones potential is a classical model for nanoclusters and glassy systems [5]. The model is defined by six parameters $\boldsymbol{\Theta} = [\epsilon^{\mathrm{LJ}}_{\mathrm{AA}}, \epsilon^{\mathrm{LJ}}_{\mathrm{AB}}, \epsilon^{\mathrm{LJ}}_{\mathrm{BB}}, \sigma_{\mathrm{AA}}, \sigma_{\mathrm{AB}}, \sigma_{\mathrm{BB}}]$, with a total energy

$$U(\tilde{\mathbf{X}}, \mathbf{C}; \boldsymbol{\Theta}) = \sum_i \sum_{j \in N_i} \epsilon^{\mathrm{LJ}}_{s_i s_j} \left(\frac{\sigma^{12}_{s_i s_j}}{r^{12}_{ij}} - \frac{\sigma^{6}_{s_i s_j}}{r^{6}_{ij}} \right), \tag{11}$$

where s_i is species i, $s_i \in [A, B]$, r_{ij} is the minimum image distance (as determined by $\mathbf{C}$) between atoms i, j and N_i is the set of neighbors of i. As discussed above, automatic differentiation enabled by `jax-md` was used to study this simple system, with all examples shown using the dense linear algebra approach to evaluate the implicit derivative at constant volume.

To simplify the problem, we set $\epsilon^{\mathrm{LJ}}_{\mathrm{AA}} = \epsilon^{\mathrm{LJ}}_{\mathrm{AB}} = \epsilon^{\mathrm{LJ}}_{\mathrm{BB}}$ and $\sigma_{\mathrm{AA}} = \sigma_{\mathrm{BB}} = 1$, leaving $\Theta = \sigma_{\mathrm{AB}}$ as the only varying parameter in this example. When $\sigma_{\mathrm{AB}} = 1$, all atoms are identical and the system is a unary fcc lattice; we additionally remove one atom to form a vacancy and promote additional deformation. When $\sigma_{\mathrm{AB}} \neq 1$ the system becomes a random fcc binary alloy, with lattice distortion in the bulk and around the vacancy (see Fig. 2a and 2b). Figure 2c shows the *inhomogeneous* implicit derivative around $\sigma_{AB} = 1$ that gives an excellent prediction of the total energy and lattice distortion for $\sigma_{AB} \in [0.95, 1.05]$, with mild disagreement as $|\sigma_{AB} - 1|$ grows. At constant volume both *constant* and *homogeneous* approximations are equivalent and predict no structural change, with significantly higher errors. However, in the remainder of this paper, we focus on modern machine learning potentials.

G. Prediction of defect formation energies and volumes with machine learning potentials

Almost all modern interatomic models use high-dimensional regression techniques from the machine learning community[8, 13, 29, 30]. A common first step is to represent atomic environments as N_D per-atom descriptor functions $D_l(\tilde{\mathbf{X}}, \mathbf{C}, i)$, $l \in [1, N_D]$ which represent the atomic environment around an atom of index

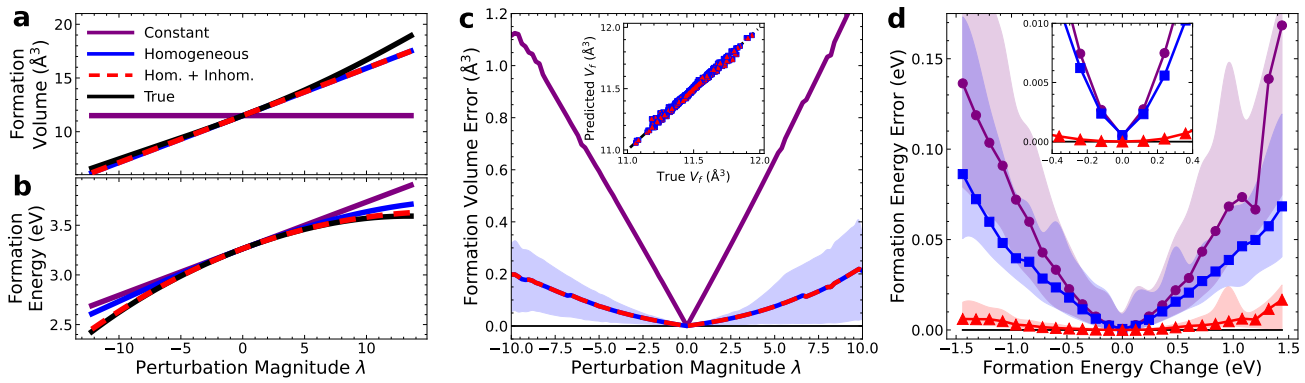

FIG. 3. **Implicit differentiation of vacancy defect formation with SNAP potentials.** **a**, **b** Formation volume and formation energy vs potential perturbation magnitude for one of 100 tungsten potential samples used in the study. **c** Average of the absolute value of vacancy formation volume vs perturbation magnitude. On the inset, predicted vacancy formation volume vs its true value obtained with the full relaxation. **d** Error vs change in vacancy formation energy averaged over the set of potentials and perturbation magnitudes. On the inset, zoom into a smaller formation energy range. In **c**,**d** the shaded region indicates the [16%,84%] percentile ($\pm 1\sigma$) range across all parameter variations. Line colors across the panels indicate the prediction method with the legend presented in **a**. In **d**, symbols are added at data points.

i. In practice, the descriptor functions also have species-dependent hyperparameters which must also be tuned, but in the following, we assume that these are fixed. We use the widely implemented SNAP Bispectrum descriptors [23] with $N_D = 55$ (see methods), giving a potential energy

$$U(\tilde{\mathbf{X}}, \mathbf{C}; \boldsymbol{\Theta}) = \boldsymbol{\Theta} \sum_i \mathbf{D}(\tilde{\mathbf{X}}, \mathbf{C}, i) \equiv \boldsymbol{\Theta} \cdot \mathbf{D}(\tilde{\mathbf{X}}, \mathbf{C}), \quad (12)$$

where $\boldsymbol{\Theta} \in \mathbb{R}^{N_D}$ is the potential parameter vector and $\mathbf{D}(\tilde{\mathbf{X}}, \mathbf{C}) \in \mathbb{R}^{N_D}$ is the total descriptor vector. The advantage of the linear functional form is that $\nabla_{\boldsymbol{\Theta}} U = \mathbf{D}$ and $\nabla^2_{\boldsymbol{\Theta}\tilde{\mathbf{X}}} U = \nabla_{\tilde{\mathbf{X}}} \mathbf{D}$, required for the solution of equation (5), are readily evaluated without numerical or automatic differentiation schemes.

We use DFT training data for tungsten from Goryaeva *et al.*[8], generating 100 samples $\boldsymbol{\Theta}_m$ from a parameter distribution using the approach described in [9]. The potential samples are suitable for multi-scale uncertainty quantification, for which a more detailed study will be presented elsewhere. Here, we are primarily interested in using the variations of parameter samples away from the reference potential $\bar{\boldsymbol{\Theta}}$ to test our implicit expansion method. To this end, we define an additional 'perturbation magnitude' λ and generate samples with

$$\boldsymbol{\Theta}(\lambda, m) = \bar{\boldsymbol{\Theta}} + \lambda(\boldsymbol{\Theta}_m - \bar{\boldsymbol{\Theta}}), \quad (13)$$

such that $\lambda = 0$ corresponds to the reference potential $\bar{\boldsymbol{\Theta}}$ and $\lambda = 1$ corresponds to the original sample $\boldsymbol{\Theta}_m$. We then generated very large (and thus often unphysical) perturbations with $\lambda \in [-25, 25]$, with a step $\Delta\lambda = 0.2$, truncating only when the bcc lattice became unstable. This yielded a total ensemble of around 20000 stable potentials. For each potential, we calculated the formation vacancy defect, allowing for relaxation of both structure and volume, meaning that only the full ih+h expansion is expected to be asymptotically exact. The diversity of the resultant dataset allows for a robust test of our implicit Taylor expansions (35) and (7).

Figures 3a and 3b illustrate this approach. As expected, this form of perturbation applied to all stable potentials produced a wide range of very strong perturbations. Fig. 3c shows the formation volume variation across the samples as a function of perturbation magnitude λ. Both homogeneous and inhomogeneous expansions provide a nearly-perfect predictions of vacancy formation volume. However, for the formation energy, the inhomogeneous approach is notably more accurate, with errors under 2% across a wide (3 eV) range of formation energies (see Fig. 3d). This indicates that whilst the efficient homogeneous expansion offers a very useful prediction of energy and volume changes, the inhomogeneous term allows for accurate prediction at small to moderate perturbations. This asymptotic accuracy is particularly important when using the implicit derivative to solve inverse problems, which we discuss in the next section.

H. Implicit loss minimization applied to solute-induced dislocation core reconstruction

In this final section we employ the implicit derivative concept to solve a challenging inverse problem: with a starting potential $\boldsymbol{\Theta}$, and some stationary configurations $\mathbf{X}^*_{\boldsymbol{\Theta}}$, we search for the potential parameters that stabilize a structure as close as possible to some target configuration $\mathbf{X}^*$. As discussed in section II E this has application for potential fine-tuning, as we demonstrate below. More generally, the ability to find parameters which yield certain desired structures represents a first step towards a range of inverse design strategies, in particular, given the

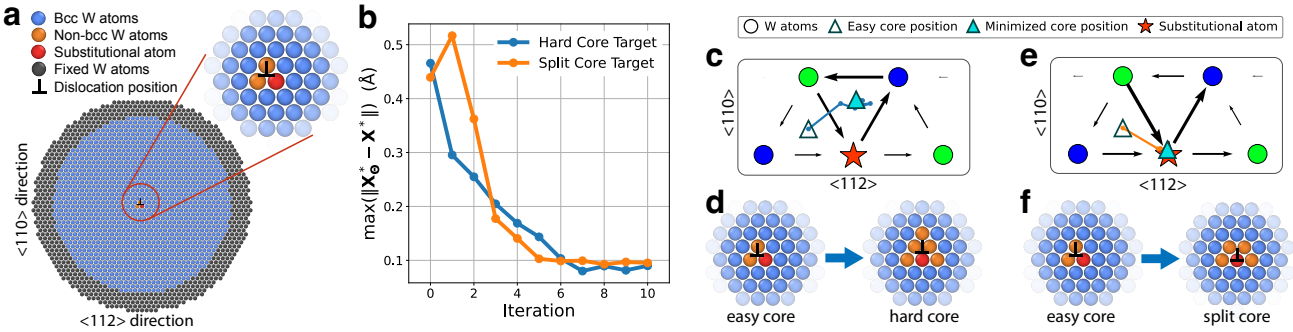

FIG. 4. **Targeting dislocation core reconstruction in tungsten.** **a** Dislocation cluster supercell. The ∼2000-atom cell has fixed boundary conditions in the ⟨110⟩ and ⟨112⟩ directions and periodic boundary conditions in the ⟨111⟩ direction with one Burgers vector within the cell, with one 'substitutional' atom at the dislocation core, with interaction parameters initially corresponding to tungsten. Around 500 atoms are fixed at the cluster border to stabilize the dislocation core structure. Non-bcc atoms are identified using the `OVITO` package [31]. **b** Maximal difference of atomic positions between a current and the target configurations during the minimization procedure. **c**, **d** Differential displacement maps [32]. The starting configuration for both simulations is the easy core and the target configurations are the hard core (**c**) and split core (**e**). The transient dislocation core positions as functions of the minimization iteration are presented with lines starting with empty and ending at solid triangles as initial and final core configurations. **d**, **f** Atomic configurations around the dislocation core with hard code (**d**) and split core (**f**) target configurations.

ability of emerging foundation models to smoothly interpolate across chemical space [15].

To demonstrate minimization of the implicit loss function 9, we selected a computationally challenging system of a ∼2000-atom tungsten disk with a ⟨111⟩/2 screw dislocation along the disk axis. As detailed in [4], the outer layers of atoms in the disk are fixed to displacement from elasticity theory and we impose periodic boundary conditions along the dislocation line direction. Using initial potential parameters $\boldsymbol{\Theta}_0$ for W from Goryaeva *et al.* [8], the dislocation core relaxes into the 'easy' core structure in agreement with *ab initio*, as illustrated in Fig. 4a. For the inverse problem, we assign one 'alchemical' atom (the red atom in Fig. 4a) with it's own independent set of parameters $\boldsymbol{\Theta}$, initially set to $\boldsymbol{\Theta}_0$. We use a simple form for the linear multi-specie potential, detailed in the methods (IV B). Modifying the alchemical potential parameters $\boldsymbol{\Theta}$ whilst keeping $\boldsymbol{\Theta}_0$ fixed, we aim to stabilize the 'hard' and 'split' core structures which are unstable for W [8]. We generated the target core configurations with the `matscipy.dislocation` [33] Python package. As the target structures are for pure, single-element W we do not expect the structure induced by the alchemical solute to give an exact match, but we can monitor the effective dislocation core position using the strain-matching approach detailed in [33]. The implicit loss minimization is achieved through the procedure presented in Algorithm 1. The loss gradient is computed according to equation (10). We employ an adaptive step size h as detailed in SM.

Figure 4b shows the maximal deviation of atomic positions at iteration k for two target structures. The minimization error decreases significantly at first steps and saturates at ∼10 iterations for both structures. The error does not reach zero, which we attribute to the target configurations being derived from pure tungsten systems, whereas our minimization involves systems that more closely represent substitutional defects. However, the minimization goals are clearly achieved as seen in Figure 4 panels c-f: the hard core (Fig. 4c,d) and split core (Fig. 4e,f) are located at expected positions denoted by solid triangles in Fig. 4c,e.

Algorithm1 Implicit Loss Minimization

$\boldsymbol{\Theta}^{(0)} \leftarrow$ initial potential
$\mathbf{X}^*_{\boldsymbol{\Theta}^{(0)}} \leftarrow$ initial relaxed positions
$k \leftarrow 0$
while $k \leq$ max_interations **do**
  Compute implicit derivative $\nabla_{\boldsymbol{\Theta}} \mathbf{X}^*_{\boldsymbol{\Theta}^{(k)}}$
  Compute $\nabla_{\boldsymbol{\Theta}} L(\boldsymbol{\Theta}^{(k)})$
  $\boldsymbol{\Theta}^{(k+1)} \leftarrow \boldsymbol{\Theta}^{(k)} - h \nabla_{\boldsymbol{\Theta}} L(\boldsymbol{\Theta}^{(k)})$
  $\mathbf{X}^*_{\boldsymbol{\Theta}^{(k+1)}} \leftarrow \mathbf{X}^*_{\boldsymbol{\Theta}^{(k)}} + (\boldsymbol{\Theta}^{(k+1)} - \boldsymbol{\Theta}^{(k)}) \nabla_{\boldsymbol{\Theta}} \mathbf{X}^*_{\boldsymbol{\Theta}^{(k)}}$
  if $\|\mathbf{X}^*_{\boldsymbol{\Theta}^{(k+1)}} - \mathbf{X}^*_{\boldsymbol{\Theta}^{(k)}}\|_\infty < \varepsilon$ **then**
    return $\mathbf{X}^*_{\boldsymbol{\Theta}^{(k+1)}}, \boldsymbol{\Theta}^{(k+1)}$
  end if
end while

As a final application, we show how the implicit loss minimization can be used to fine-tune an initial interatomic potential to match *ab initio* training data. Here, our target is a solute-induced reconstruction of the ⟨111⟩/2 screw dislocation core caused by an interstitial Be atom, using the same disk geometry as described above and shown in Fig. 5a. The target data was generated using the QM/ML simulation method which embeds a DFT region at core as detailed in methods IV C. As shown in Fig. 5c, we see that Be induces reconstruction to the 'hard' core structure, with the Be interstitial sitting at the center of the dislocation.

Using W-Be *ab initio* training data from Wood *et*

al. [34], we created an initial set of Be interaction parameters $\mathbf{\Theta}$ using the same linear multi-species interatomic potential as above (methods IV B) with W parameters $\mathbf{\Theta}_0$ set to those from Goryaeva *et al.* [8]. The relaxed structure using the initial potential fit is shown in Fig. 5b. It can be seen that in contrast to the QM/ML target, the dislocation core remains in the 'easy' configuration, with the Be atom lying outside of the central core region. Using the procedure described above, we performed implicit loss minimization using this initial relaxed configuration. As shown in the SM, the implicit loss achieved near-perfect reproduction of the core reconstruction in around 20 iterations. Future work will investigate more sophisticated minimization schemes than the simple gradient descent used here.

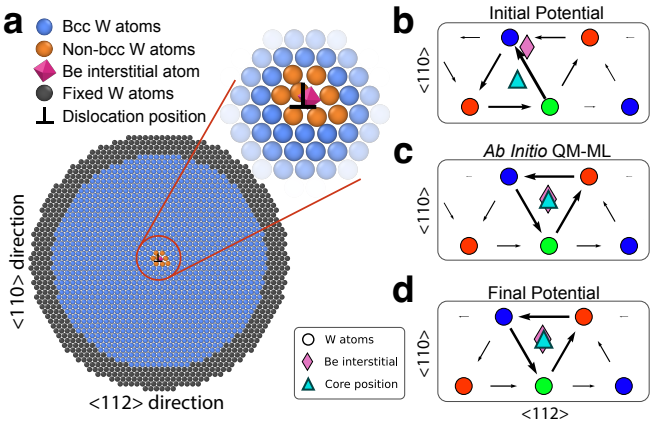

FIG. 5. **'Fine-tuning' Be-induced dislocation core reconstruction in tungsten. a** Dislocation cluster supercell similar to that shown in figure 4, but with a Be interstitial in the center of the cluster and no substitution. **b**, **c**, **d** differential displacement maps [32] with core and interstitial positions corresponding to the initial `SNAP` potential (panel **b**), target configuration obtained with QM-ML (methods IV C, panel **c**), and the minimum of the 'fine-tuned' potential (panel **d**).

III. DISCUSSION

In this paper, we have investigated the use of the implicit derivative of the relaxed atomic structure with interatomic potential parameters, giving a first-order implicit Taylor expansion for the relaxed structures and second order for relaxed energies. We detailed how the implicit derivative could be calculated using dense linear algebra, requiring Hessian evaluation, automatic differentiation, or a Hessian-free linear operator approach, which reduces to a constrained minimization in `LAMMPS`, allowing application to arbitrarily large systems.

The implicit expansion enables very efficient forward propagation of parameter uncertainties to simulation results, including the effect of geometry relaxation, essential to capture changes in structure such as strain. This was demonstrated on simple classical models and machine learning models for pure W [8]. The implicit expansion was able to capture a wide range of changes in energy and structure, far beyond typical variations associated with potential parameter uncertainty. A forthcoming publication will demonstrate the implicit expansion in a wide-ranging uncertainty quantification study. Beyond uncertainty quantification, our results show that the implicit expansion can also be used to rapidly explore the parameter space of high dimensional interatomic models, permitting parametric studies that would be intractable with standard methods. In future work, we will explore how the implicit expansion can be used in a correlative study of defect structures and implications for both uncertainty quantification and materials design goals.

In addition to the forward propagation enabled by the implicit expansion, we also investigated the use of the implicit derivative in backpropagation of structural changes to parameter changes. Our exploratory applications focused on solute-induced dislocation core reconstruction in W [4], a key feature in understanding plasticity and irradiation damage in bcc metals [35]. We showed how the implicit derivative could be used to 'fine-tune' parameters from an initial fit against training data for WBe [34], in order to stabilize the structure seen in *ab initio* calculations [4].

In a first effort towards targeted 'alchemical' design applications, we used the implicit derivative to find substitutional solute interaction parameters which stabilized 'hard' or 'split' dislocation cores in pure W. The success of this effort extends the scope of alchemical machine learning to large-scale simulations essential to for mechanistic studies of e.g. plasticity. We anticipate that both approaches will gain ever-increasing application with the advent of general-purpose 'universal' machine learning potentials, which will be a focus of future efforts.

IV. METHODS

A. Implementation of the sparse linear operator as a constrained minimization in `LAMMPS`

Equation (8) defines a linear operator which can be used in iterative solution of the linear equations (5) and thus to evaluate the implicit derivative $\nabla_{\mathbf{\Theta}}\tilde{\mathbf{X}}^*_{\mathbf{\Theta}}$. To fully exploit the efficiencies afforded by the Hessian sparsity, in this section, we detail how $\nabla_{\mathbf{\Theta}}\tilde{\mathbf{X}}^*_{\mathbf{\Theta}}$ can be implemented in the massively parallel `LAMMPS` simulation package. This is achieved through N_D constrained minimizations, one for each column $[\nabla_{\mathbf{\Theta}}\tilde{\mathbf{X}}^*_{\mathbf{\Theta}}]_l$, $l \in [1, N_D]$ of the implicit derivative. We have established, through a wide range of numerical tests using the full dense solution, that the off-diagonal terms $\nabla^2_{\tilde{\mathbf{X}}_\epsilon} U$ in (5) can be neglected when determining the homogeneous term $\nabla_{\mathbf{\Theta}}\epsilon^*_{\mathbf{\Theta}}$, meaning the inhomogeneous term $\nabla_{\Theta_l}\tilde{\mathbf{X}}^*_{\mathbf{\Theta}}$ satisfies

$$[\nabla_{\mathbf{\Theta}}\tilde{\mathbf{X}}^*_{\mathbf{\Theta}}]_l \nabla^2_{\tilde{\mathbf{X}}\tilde{\mathbf{X}}} U + [\mathbf{B}]_l, \quad l \in [1, N_D], \tag{14}$$

where $\mathbf{B} = \nabla_{\boldsymbol{\Theta}} \epsilon^*_{\boldsymbol{\Theta}} \otimes \nabla^2_{\epsilon \tilde{\mathbf{X}}} U + \nabla^2_{\boldsymbol{\Theta} \tilde{\mathbf{X}}} U \in \mathbb{R}^{N_D \times N \times 3}$. For the linear-in-descriptor `SNAP` potentials used here, the $\nabla^2_{\epsilon \boldsymbol{\Theta}} U$ term can be directly accessed as derivatives of descriptors [23] using `fix sna/atom` and related commands in `LAMMPS`. For the inverse design applications in this paper, which modified only the interaction parameters of a single solute atom without changes to the supercell strain, we have $\nabla_{\boldsymbol{\Theta}} \epsilon^*_{\boldsymbol{\Theta}} = \mathbf{0}$ and thus $\mathbf{B} = \nabla^2_{\boldsymbol{\Theta} \tilde{\mathbf{X}}} U$. With an initial parameter vector $\boldsymbol{\Theta}$, we then define the modified energy function for a parameter index $l \in [1, N_D]$

$$U(\tilde{\mathbf{X}}, \mathbf{C}; \boldsymbol{\Theta}) + \alpha [\mathbf{B}]_l \cdot [\tilde{\mathbf{X}} - \tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}}], \quad l \in [1, N_D]. \tag{15}$$

This modified energy is simple to implement in `LAMMPS` through the `fix addforce` function, with similar ease of implementation in any molecular dynamics package. It is straightforward to show that in the limit $\alpha \to 0$ the minimizer $\tilde{\mathbf{X}}^*_{l,\alpha}$ of (15) gives the lth vector $[\nabla_{\boldsymbol{\Theta}} \tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}}]_l$ of the implicit derivative $\nabla_{\boldsymbol{\Theta}} \tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}}$ through

$$[\nabla_{\boldsymbol{\Theta}} \tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}}]_l = \left(\tilde{\mathbf{X}}^*_{l,\alpha} - \tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}}\right)/\alpha. \tag{16}$$

Further details regarding hyperparameter scanning for suitable values of α and comparison against the full dense linear solution employing the Hessian matrix is provided in SM.

B. Multi-specie ML potentials

For systems with multiple atomic species $s_i \in [1, N_S]$, as investigated in sections II G,II H, we simply assign a new specie-dependent parameter vector as in the original SNAP paper [23], giving a total energy

$$U(\tilde{\mathbf{X}}, \mathbf{C}; \boldsymbol{\Theta}) = \sum_i [\boldsymbol{\Theta}]_{s_i} \mathbf{D}(\tilde{\mathbf{X}}, \mathbf{C}, i) \equiv \boldsymbol{\Theta} \cdot \mathbf{D}(\tilde{\mathbf{X}}, \mathbf{C}), \tag{17}$$

where $\boldsymbol{\Theta} \in \mathbb{R}^{N_S \times N_D}$ is the total parameter vector, $s_i \in [1, N_S]$ and $\mathbf{D}(\tilde{\mathbf{X}}, \mathbf{C}) \in \mathbb{R}^{N_S \times N_D}$ is the total descriptor vector. In typical usage practice, the descriptor functions have species-dependent hyperparameters which must also be tuned, but here we assume these are fixed.

C. *Ab initio* QM-ML calculations

The *ab initio* reference data for Be segregation to screw dislocations in W was calculated using QM-ML hybrid simulations [4], which couple *ab initio* and machine learning potentials. Initial structures were obtained with `matscipy.dislocation` module [33]. The Be segregation calculation used the exact same approach reported in [4] for He segregation. *Ab initio* forces were evaluated using `VASP` [36, 37] with 10 **k**-points along the periodic line direction, with a cutoff energy of 500 eV and a minimization force threshold of 0.01 eV/Å. The machine learning force field was a modified `SNAP/MILaDy` potential from [8], as detailed in [4]. The QM/ML coupling used a buffer radius of 10 Å, resulting in a total of 246 `VASP` atoms, of which 168 were in the buffer. We refer the reader to [4] for further details.

V. DATA AVAILABILITY

The implicit derivative implementations derived will be publicly available on GitHub following peer review.

VI. ACKNOWLEDGEMENTS

IM and TDS gratefully acknowledge support from an Emergence@INP grant from the CNRS. TDS thanks the Institute for Pure and Applied Mathematics at the University of California, Los Angeles (supported by NSF grant DMS-1925919) for their hospitality. TDS and PG gratefully acknowledge support from ANR grants ANR-19-CE46-0006-1 and ANR-23-CE46-0006-1, IDRIS allocation A0120913455.

VII. CONTRIBUTIONS

TDS designed the research program and derived the initial theoretical results. IM implemented the sparse operator, designed the implicit loss minimizer, and ran all simulations. PG generated the dislocation structures and performed the QM/ML calculations. IM and TDS wrote the paper.

VIII. COMPETING INTERESTS

The authors declare no competing interests.

[1] E. Van Der Giessen, P. A. Schultz, N. Bertin, V. V. Bulatov, W. Cai, G. Csányi, S. M. Foiles, M. G. Geers, C. González, M. Hütter, *et al.*, Roadmap on multiscale materials modeling, Modelling and Simulation in Materials Science and Engineering **28**, 043001 (2020).

[2] V. L. Deringer, M. A. Caro, and G. Csányi, Machine learning interatomic potentials as emerging tools for materials science, Advanced Materials **31**, 1902765 (2019).

[3] T. Swinburne and D. Perez, Automated calculation and convergence of defect transport tensors, npj Computational Materials **6**, 190 (2020).

[4] P. Grigorev, A. M. Goryaeva, M.-C. Marinica, J. R. Kermode, and T. D. Swinburne, Calculation of dislocation binding to helium-vacancy defects in tungsten using hybrid ab initio-machine learning methods, Acta Materialia **247**, 118734 (2023).

[5] D. J. Wales, *Energy Landscapes*, edited by C. U. Press (Cambridge, 2003).

[6] L. Proville, D. Rodney, and M.-C. Marinica, Quantum effect on thermally activated glide of dislocations, Nature materials **11**, 845 (2012).

[7] C. P. Goodrich, E. M. King, S. S. Schoenholz, E. D. Cubuk, and M. P. Brenner, Designing self-assembling kinetics with differentiable statistical physics models, Proceedings of the National Academy of Sciences **118**, e2024083118 (2021).

[8] A. M. Goryaeva, J. Dérès, C. Lapointe, P. Grigorev, T. D. Swinburne, J. R. Kermode, L. Ventelon, J. Baima, and M.-C. Marinica, Efficient and transferable machine learning potentials for the simulation of crystal defects in bcc Fe and W, Phys. Rev. Materials **5**, 103803 (2021).

[9] T. D. Swinburne and D. Perez, Parameter uncertainties for imperfect surrogate models in the low-noise regime (2024), arXiv:2402.01810 [stat.ML].

[10] F. Musil, M. J. Willatt, M. A. Langovoy, and M. Ceriotti, Fast and accurate uncertainty estimation in chemical machine learning, Journal of chemical theory and computation **15**, 906 (2019).

[11] S. Thaler, M. Stupp, and J. Zavadlav, Deep coarse-grained potentials via relative entropy minimization, The Journal of Chemical Physics **157** (2022).

[12] I. Batatia, D. P. Kovács, G. N. Simm, C. Ortner, and G. Csányi, Mace: Higher order equivariant message passing neural networks for fast and accurate force fields, arXiv preprint arXiv:2206.07697 (2022).

[13] I. Batatia, P. Benner, Y. Chiang, A. M. Elena, D. P. Kovács, J. Riebesell, X. R. Advincula, M. Asta, M. Avaylon, W. J. Baldwin, F. Berger, N. Bernstein, A. Bhowmik, S. M. Blau, V. Cărare, J. P. Darby, S. De, F. D. Pia, V. L. Deringer, R. Elijošius, Z. El-Machachi, F. Falcioni, E. Fako, A. C. Ferrari, A. Genreith-Schriever, J. George, R. E. A. Goodall, C. P. Grey, P. Grigorev, S. Han, W. Handley, H. H. Heenen, K. Hermansson, C. Holm, J. Jaafar, S. Hofmann, K. S. Jakob, H. Jung, V. Kapil, A. D. Kaplan, N. Karimitari, J. R. Kermode, N. Kroupa, J. Kullgren, M. C. Kuner, D. Kuryla, G. Liepuoniute, J. T. Margraf, I.-B. Magdău, A. Michaelides, J. H. Moore, A. A. Naik, S. P. Niblett, S. W. Norwood, N. O'Neill, C. Ortner, K. A. Persson, K. Reuter, A. S. Rosen, L. L. Schaaf, C. Schran, B. X. Shi, E. Sivonxay, T. K. Stenczel, V. Svahn, C. Sutton, T. D. Swinburne, J. Tilly, C. van der Oord, E. Varga-Umbrich, T. Vegge, M. Vondrák, Y. Wang, W. C. Witt, F. Zills, and G. Csányi, A foundation model for atomistic materials chemistry (2024), arXiv:2401.00096 [physics.chem-ph].

[14] B. Deng, Y. Choi, P. Zhong, J. Riebesell, S. Anand, Z. Li, K. Jun, K. A. Persson, and G. Ceder, Overcoming systematic softening in universal machine learning interatomic potentials by fine-tuning (2024), 2405.07105.

[15] J. Nam and R. Gomez-Bombarelli, Interpolation and differentiation of alchemical degrees of freedom in machine learning interatomic potentials (2024), arXiv:2404.10746 [cond-mat.mtrl-sci].

[16] S. Schoenholz and E. D. Cubuk, Jax md: a framework for differentiable physics, Advances in Neural Information Processing Systems **33**, 11428 (2020).

[17] P. Ablin, G. Peyré, and T. Moreau, Super-efficiency of automatic differentiation for functions defined as a minimum, in *International Conference on Machine Learning* (PMLR, 2020) pp. 32–41.

[18] M. Blondel, Q. Berthet, M. Cuturi, R. Frostig, S. Hoyer, F. Llinares-López, F. Pedregosa, and J.-P. Vert, Efficient and modular implicit differentiation, Advances in neural information processing systems **35**, 5230 (2022).

[19] S. G. Krantz and H. R. Parks, *The implicit function theorem: history, theory, and applications* (Springer Science & Business Media, 2002).

[20] S. Plimpton, Fast Parallel Algorithms for Short-Range Molecular Dynamics, Journal Computational Physics **117**, 1 (1995).

[21] S. R. Xie, M. Rupp, and R. G. Hennig, Ultra-fast interpretable machine-learning potentials, npj Computational Materials **9**, 162 (2023).

[22] A. Del Masto, J. Baccou, G. Tréglia, F. Ribeiro, and C. Varvenne, Insights on the capabilities and improvement ability of classical many-body potentials: Application to α-zirconium, Computational Materials Science **231**, 112544 (2024).

[23] A. P. Thompson, L. P. Swiler, C. R. Trott, S. M. Foiles, and G. J. Tucker, Spectral neighbor analysis method for automated generation of quantum-accurate interatomic potentials, J. Comp. Phys. **285**, 316 (2015).

[24] A. E. Allen, G. Dusson, C. Ortner, and G. Csányi, Atomic permutationally invariant polynomials for fitting molecular force fields, Machine Learning: Science and Technology **2**, 025017 (2021).

[25] E. V. Podryabinkin and A. V. Shapeev, Active learning of linearly parametrized interatomic potentials, Computational Materials Science **140**, 171 (2017).

[26] Y. Lysogorskiy, C. van der Oord, A. Bochkarev, S. Menon, M. Rinaldi, T. Hammerschmidt, M. Mrovec, A. Thompson, G. Csányi, C. Ortner, *et al.*, Performant implementation of the atomic cluster expansion (PACE) and application to copper and silicon, npj Computational Materials **7**, 1 (2021).

[27] S. Batzner, A. Musaelian, L. Sun, M. Geiger, J. P. Mailoa, M. Kornbluth, N. Molinari, T. E. Smidt, and B. Kozinsky, E (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials, Nature communications **13**, 1 (2022).

[28] L. Reali, M. Boleininger, M. R. Gilbert, and S. L. Dudarev, Macroscopic elastic stress and strain produced by irradiation, Nuclear Fusion **62**, 016002 (2021).

[29] A. P. Bartók, M. C. Payne, R. Kondor, and G. Csányi, Gaussian approximation potentials: The accuracy of quantum mechanics, without the electrons, Phys. Rev. Lett. **104**, 136403 (2010).

[30] A. P. Bartók, S. De, C. Poelking, N. Bernstein, J. R. Kermode, G. Csányi, and M. Ceriotti, Machine learning unifies the modeling of materials and molecules, Sci. Adv. **3**, e1701816 (2017).

[31] A. Stukowski, Visualization and analysis of atomistic simulation data with ovito–the open visualization tool, Modelling and Simulation in Materials Science and Engineering **18**, 015012 (2009).

[32] V. Vitek, Theory of the core structures of dislocations in body-centred-cubic metals., Cryst. Latt. Def. Amorp. (1974).
[33] P. Grigorev, L. Frérot, F. Birks, A. Gola, J. Golebiowski, J. Grießer, J. L. Hörmann, A. Klemenz, G. Moras, W. G. Nöhring, J. A. Oldenstaedt, P. Patel, T. Reichenbach, T. Rocke, L. Shenoy, M. Walter, S. Wengert, L. Zhang, J. R. Kermode, and L. Pastewka, matscipy: materials science at the atomic scale with python, Journal of Open Source Software **9**, 5668 (2024).
[34] M. A. Wood, M. A. Cusentino, B. D. Wirth, and A. P. Thompson, Data-driven material models for atomistic simulation, Phys. Rev. B **99**, 184305 (2019).
[35] G. Hachet, L. Ventelon, F. Willaime, and E. Clouet, Screw dislocation-carbon interaction in bcc tungsten: an ab initio study, Acta Materialia **200**, 481 (2020).
[36] G. Kresse and J. Furthmüller, Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set, Phys. Rev. B **54**, 11169 (1996).
[37] J. P. Perdew, K. Burke, and M. Ernzerhof, Generalized gradient approximation made simple, Phys. Rev. Lett. **77**, 3865 (1996).

Supplemental Material for: Exploring the parameter dependence of atomic minima with implicit differentiation

1. IMPLICIT DIFFERENTIATION

A. General expression derivation

Here we consider $\mathbf{X}^*_{\boldsymbol{\Theta}}$ as a stationary configuration that includes the scaled atomic coordinates $\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}}$ and supercell $\mathbf{C}^*_{\boldsymbol{\Theta}}$. A stationary configuration is defined with a zero-force condition:

$$\mathbf{F}(\mathbf{X}^*_{\boldsymbol{\Theta}}; \boldsymbol{\Theta}) = \mathbf{0}. \tag{18}$$

Under a parameter perturbation, $\boldsymbol{\Theta} + \delta\boldsymbol{\Theta}$, a new stationary configuration $\mathbf{X}^*_{\boldsymbol{\Theta}+\delta\boldsymbol{\Theta}}$ will satisfy:

$$\mathbf{F}(\mathbf{X}^*_{\boldsymbol{\Theta}+\delta\boldsymbol{\Theta}}; \boldsymbol{\Theta} + \delta\boldsymbol{\Theta}) = \mathbf{0}. \tag{19}$$

Hence, under a parameter perturbation, the deferential of the force is zero as well:

$$\mathrm{d}\mathbf{F}(\mathbf{X}^*_{\boldsymbol{\Theta}+\delta\boldsymbol{\Theta}}; \boldsymbol{\Theta} + \delta\boldsymbol{\Theta}) = \delta\mathbf{X}^*_{\boldsymbol{\Theta}} \nabla_{\mathbf{X}} \mathbf{F}(\mathbf{X}^*_{\boldsymbol{\Theta}}; \boldsymbol{\Theta}) + \delta\boldsymbol{\Theta} \cdot \nabla_{\boldsymbol{\Theta}} \mathbf{F}(\mathbf{X}^*_{\boldsymbol{\Theta}}; \boldsymbol{\Theta}) + \mathcal{O}(\delta\boldsymbol{\Theta}^2) = \mathbf{0}. \tag{20}$$

We express the variation in coordinates using the implicit derivative definition $\delta\mathbf{X}^*_{\boldsymbol{\Theta}} = \delta\boldsymbol{\Theta}\nabla_{\boldsymbol{\Theta}}\mathbf{X}^*_{\boldsymbol{\Theta}}$ and obtain

$$\delta\boldsymbol{\Theta} \cdot \left[\nabla_{\boldsymbol{\Theta}}\mathbf{X}^*_{\boldsymbol{\Theta}} \nabla_{\mathbf{X}} \mathbf{F}(\mathbf{X}^*_{\boldsymbol{\Theta}}; \boldsymbol{\Theta}) + \nabla_{\boldsymbol{\Theta}} \mathbf{F}(\mathbf{X}^*_{\boldsymbol{\Theta}}; \boldsymbol{\Theta})\right] = \mathbf{0} + \mathcal{O}(\delta\boldsymbol{\Theta}^2). \tag{21}$$

As this holds for any parameter variation $\delta\boldsymbol{\Theta}$, the term inside the square brackets is zero. Finally, we will express the force through energy $U(\mathbf{X}^*_{\boldsymbol{\Theta}}; \boldsymbol{\Theta}) = -\nabla_{\mathbf{X}}\mathbf{F}(\mathbf{X}^*_{\boldsymbol{\Theta}}; \boldsymbol{\Theta})$ and get

$$\nabla_{\boldsymbol{\Theta}}\mathbf{X}^*_{\boldsymbol{\Theta}} \nabla^2_{\mathbf{X}\mathbf{X}} U(\mathbf{X}^*_{\boldsymbol{\Theta}}; \boldsymbol{\Theta}) = -\nabla_{\boldsymbol{\Theta}\mathbf{X}} U(\mathbf{X}^*_{\boldsymbol{\Theta}}; \boldsymbol{\Theta}). \tag{22}$$

Expressing atomic coordinates as $\mathbf{X} = \tilde{\mathbf{X}}\mathbf{C}$ and splitting the position and mixed Hessians on the scale-coordinate and supercell blocks, one gets the equation (5) from the main text.

B. *Homogeneous* implicit derivative of strain

In this work, we split the full implicit derivative (equation (22)) onto the *inhomogeneous* and *homogeneous* contributions. As explained in the main text (Methods A), the *inhomogeneous* implicit derivative $\nabla_{\boldsymbol{\Theta}}\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}}$ can be computed in `LAMMPS` with constraint minimization. In this section, we detail our finite difference implementation of the *homogeneous* part of the implicit derivative, $\nabla_{\boldsymbol{\Theta}}\epsilon^*_{\boldsymbol{\Theta}}$.

A stationary configuration of a system of volume $V^*_{\boldsymbol{\Theta}}$ corresponds to zero pressure, i.e. $P(V^*_{\boldsymbol{\Theta}}) = -\partial U(\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}}, \mathbf{C}^*_{\boldsymbol{\Theta}}; \boldsymbol{\Theta})/\partial V = 0$. For isotropic supercell variations, $\mathbf{C}^*_{\boldsymbol{\Theta}} = [1 + \epsilon^*_{\boldsymbol{\Theta}}]\,\mathbf{C}_0$, this equation can be reformulated in terms of strain:

$$\left.\frac{\partial U(\tilde{\mathbf{X}}, \epsilon; \boldsymbol{\Theta})}{\partial \epsilon}\right|_{\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}}, \epsilon^*_{\boldsymbol{\Theta}}} = 0. \tag{23}$$

In this section, we neglect the *inhomogeneous* contribution to the strain derivative and we will omit $\tilde{\mathbf{X}}$ from the arguments of U for clarity. For linear-in-descriptor potentials (e.g. equation (12) in the main text), equation (23) writes

$$\boldsymbol{\Theta} \cdot \left.\frac{\partial \mathbf{D}(\epsilon)}{\partial \epsilon}\right|_{\epsilon^*_{\boldsymbol{\Theta}}} = 0. \tag{24}$$

In analogy with the previous section (equation (20)), we consider a system upon a parameter variation $\boldsymbol{\Theta} + \delta\boldsymbol{\Theta}$:

$$\left.\frac{\partial U(\epsilon; \boldsymbol{\Theta} + \delta\boldsymbol{\Theta})}{\partial \epsilon}\right|_{\epsilon^*_{\boldsymbol{\Theta}} + \delta\epsilon^*_{\boldsymbol{\Theta}}} = 0. \tag{25}$$

Applying the linear-in-descriptor form of potential energy, we get:

$$(\boldsymbol{\Theta} + \delta\boldsymbol{\Theta}) \cdot \left.\frac{\partial \mathbf{D}(\epsilon)}{\partial \epsilon}\right|_{\epsilon^*_{\boldsymbol{\Theta}} + \delta\epsilon^*_{\boldsymbol{\Theta}}} = 0. \tag{26}$$

Neglecting the term proportional to $\delta\boldsymbol{\Theta}\delta\epsilon^*_{\boldsymbol{\Theta}}$, we get

$$\delta\boldsymbol{\Theta} \cdot \left.\frac{\partial \mathbf{D}(\epsilon)}{\partial \epsilon}\right|_{\epsilon^*_{\boldsymbol{\Theta}}} + \delta\epsilon^*_{\boldsymbol{\Theta}}\boldsymbol{\Theta} \cdot \left.\frac{\partial^2 \mathbf{D}(\epsilon)}{\partial \epsilon^2}\right|_{\epsilon^*_{\boldsymbol{\Theta}}} = 0. \tag{27}$$

We then use the definition of the *homogeneous* implicit derivative $\delta\epsilon^*_{\boldsymbol{\Theta}} = \nabla_{\boldsymbol{\Theta}}\epsilon^*_{\boldsymbol{\Theta}}\delta\boldsymbol{\Theta}$:

$$\delta\boldsymbol{\Theta} \cdot \left[\frac{\partial \mathbf{D}(\epsilon)}{\partial \epsilon} + \nabla_{\boldsymbol{\Theta}}\epsilon^*_{\boldsymbol{\Theta}}\Big(\boldsymbol{\Theta} \cdot \frac{\partial^2 \mathbf{D}(\epsilon)}{\partial \epsilon^2}\Big)\right]_{\epsilon^*_{\boldsymbol{\Theta}}} = 0. \tag{28}$$

Since this equation is valid for any parameter variation $\delta\boldsymbol{\Theta}$, we can get the final expression for the *homogeneous* implicit derivative:

$$\nabla_{\boldsymbol{\Theta}}\epsilon^*_{\boldsymbol{\Theta}} = -\left.\frac{\partial \mathbf{D}(\epsilon)/\partial \epsilon}{\boldsymbol{\Theta} \cdot \partial^2 \mathbf{D}(\epsilon)/\partial \epsilon^2}\right|_{\epsilon^*_{\boldsymbol{\Theta}}}. \tag{29}$$

For numerical purposes, we evaluate the derivatives of the descriptor vector with finite differences:

$$\frac{\partial \mathbf{D}(\epsilon^*_{\boldsymbol{\Theta}})}{\partial \epsilon} \approx \frac{\mathbf{D}(\epsilon^*_{\boldsymbol{\Theta}} + \Delta\epsilon) - \mathbf{D}(\epsilon^*_{\boldsymbol{\Theta}} - \Delta\epsilon)}{2\Delta\epsilon}; \quad \frac{\partial^2 \mathbf{D}(\epsilon^*_{\boldsymbol{\Theta}})}{\partial \epsilon^2} \approx \frac{\mathbf{D}(\epsilon^*_{\boldsymbol{\Theta}} + \Delta\epsilon) + \mathbf{D}(\epsilon^*_{\boldsymbol{\Theta}} - \Delta\epsilon) - 2\mathbf{D}(\epsilon^*_{\boldsymbol{\Theta}})}{\Delta\epsilon^2}, \tag{30}$$

where $\Delta\epsilon$ is typically 10^{-3}Å.

C. Implicit derivative of strain including *homogeneous* and *inhomogeneous* terms

Let us take into account the *inhomogeneous* contribution in equation (31):

$$\frac{\partial U(\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}} + \delta\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}}, \epsilon^*_{\boldsymbol{\Theta}} + \delta\epsilon^*_{\boldsymbol{\Theta}}; \boldsymbol{\Theta} + \delta\boldsymbol{\Theta})}{\partial \epsilon} = 0. \tag{31}$$

Following a similar procedure as in the previous section, we get the *h+ih* level of approximation for the strain implicit derivative:

$$\nabla_{\boldsymbol{\Theta}}\epsilon^*_{\boldsymbol{\Theta}} = -\frac{\partial \mathbf{D}(\epsilon^*_{\boldsymbol{\Theta}})/\partial \epsilon + \nabla_{\boldsymbol{\Theta}}\mathbf{X}^*_{\boldsymbol{\Theta}} \cdot \partial \mathbf{F}/\partial \epsilon}{\boldsymbol{\Theta} \cdot \partial^2 \mathbf{D}(\epsilon^*_{\boldsymbol{\Theta}})/\partial \epsilon^2}, \tag{32}$$

where the force derivative is similarly evaluated with the finite difference approach.

D. Taylor expansion of energy

In this section, we derive the expansion of potential energy of a stationary system, $U(\mathbf{X}^*_{\boldsymbol{\Theta}}; \boldsymbol{\Theta})$, due to a perturbation in parameters $\boldsymbol{\Theta}$. For clarity we will omit $(\mathbf{X}^*_{\boldsymbol{\Theta}}; \boldsymbol{\Theta})$ from $U(\mathbf{X}^*_{\boldsymbol{\Theta}}; \boldsymbol{\Theta})$ in this section. One has to account for contributions

arising from the explicit dependence of U on parameters and changes in the stationary configurations $\mathbf{X}^*_{\boldsymbol{\Theta}}$. Here, we first consider the general atomic coordinates $\mathbf{X}^*$ and later split them on scaled coordinates $\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}}$ and supercell $\mathbf{C}^*_{\boldsymbol{\Theta}}$.

Under a parameter perturbation $\boldsymbol{\Theta} + \delta\boldsymbol{\Theta}$, the potential energy expansion reads

$$\begin{aligned} U(\mathbf{X}^*_{\boldsymbol{\Theta}} + \delta\mathbf{X}^*_{\boldsymbol{\Theta}}; \boldsymbol{\Theta} + \delta\boldsymbol{\Theta}) = U + \delta\mathbf{X}^*_{\boldsymbol{\Theta}}\nabla_{\mathbf{X}}U + \delta\boldsymbol{\Theta}\nabla_{\boldsymbol{\Theta}}U \\ + \frac{1}{2}\delta\mathbf{X}^*_{\boldsymbol{\Theta}}\nabla^2_{\mathbf{XX}}U\delta\mathbf{X}^{*\top}_{\boldsymbol{\Theta}} + \frac{1}{2}\delta\boldsymbol{\Theta}\nabla^2_{\boldsymbol{\Theta}\boldsymbol{\Theta}}U\delta\boldsymbol{\Theta}^\top + \delta\boldsymbol{\Theta}\nabla^2_{\boldsymbol{\Theta}\mathbf{X}}U\delta\mathbf{X}^{*\top}_{\boldsymbol{\Theta}} + \mathcal{O}(\delta\boldsymbol{\Theta}^3) \end{aligned} \tag{33}$$

The term proportional to $\nabla_{\mathbf{X}}U$ vanishes since $\mathbf{X}^*_{\boldsymbol{\Theta}}$ is a stationary atomic configuration. For the second-order terms, we express $\delta\mathbf{X}^*_{\boldsymbol{\Theta}}$ using the implicit derivative: $\delta\mathbf{X}^*_{\boldsymbol{\Theta}} = \delta\boldsymbol{\Theta}\nabla_{\boldsymbol{\Theta}}\mathbf{X}^*_{\boldsymbol{\Theta}}$. Referring to equation (22), we further substitute $\nabla_{\boldsymbol{\Theta}}\mathbf{X}^*_{\boldsymbol{\Theta}}$ and get: $\delta\mathbf{X}^*_{\boldsymbol{\Theta}} = -\delta\boldsymbol{\Theta}\nabla^2_{\boldsymbol{\Theta}\mathbf{X}}U\left[\nabla^2_{\mathbf{XX}}U\right]^+$. The final energy expansion up to terms $\mathcal{O}(\delta\boldsymbol{\Theta}^3)$ reads:

$$U(\mathbf{X}^*_{\boldsymbol{\Theta}} + \delta\mathbf{X}^*_{\boldsymbol{\Theta}}; \boldsymbol{\Theta} + \delta\boldsymbol{\Theta}) = U + \delta\boldsymbol{\Theta}\nabla_{\boldsymbol{\Theta}}U + \frac{1}{2}\delta\boldsymbol{\Theta}\left[\nabla^2_{\boldsymbol{\Theta}\boldsymbol{\Theta}}U + \nabla^2_{\boldsymbol{\Theta}\mathbf{X}}U(\nabla_{\boldsymbol{\Theta}}\mathbf{X}^*_{\boldsymbol{\Theta}})^\top\right]\delta\boldsymbol{\Theta}^\top. \tag{34}$$

In the main text, we outline the energy expansion as follows:

$$\delta^{(\zeta)}U^* \equiv \delta\boldsymbol{\Theta}\nabla_{\boldsymbol{\Theta}}U + \delta\boldsymbol{\Theta}\mathbf{H}_\zeta\delta\boldsymbol{\Theta}^\top + \mathcal{O}(\delta\boldsymbol{\Theta}^3), \tag{35}$$

where ζ represents the level of approximation. Considering $\mathbf{X} = \tilde{\mathbf{X}}\mathbf{C}$ and employing the implicit derivative definitions for scaled coordinates and strain, $\nabla_{\boldsymbol{\Theta}}\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}}$ and $\nabla_{\boldsymbol{\Theta}}\epsilon^*_{\boldsymbol{\Theta}}$ (equations (2)-(5) from the main text), we derive the expressions of $\mathbf{H}_\zeta$ corresponding to each level of approximation:

- *constant (c)*: $\mathbf{H}_c = \nabla^2_{\boldsymbol{\Theta}\boldsymbol{\Theta}}U$
- *homogeneous (h)*: $\mathbf{H}_h = \nabla^2_{\boldsymbol{\Theta}\boldsymbol{\Theta}}U + \nabla^2_{\boldsymbol{\Theta}\epsilon}U(\nabla_{\boldsymbol{\Theta}}\epsilon^*_{\boldsymbol{\Theta}})^\top$
- *inhomogeneous (ih)*: $\mathbf{H}_{ih} = \nabla^2_{\boldsymbol{\Theta}\boldsymbol{\Theta}}U + \nabla^2_{\boldsymbol{\Theta}\tilde{\mathbf{X}}}U(\nabla_{\boldsymbol{\Theta}}\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}})^\top$.

2. MEMORY AND TIME EFFICIENCY OF IMPLICIT DERIVATIVE EVALUATION

In this section, we provide the details on memory and time efficiency of the automatic differentiation (AD) and `LAMMPS` implementations of the implicit derivative. Given its computational complexity, our focus here will be on the *inhomogeneous* contribution, $\nabla_{\boldsymbol{\Theta}}\tilde{\mathbf{X}}^*_{\boldsymbol{\Theta}}$.

A. Automatic differentiation implementation

Here, we use the LJ random fcc alloy (presented in the main text, Results F) as a test system. We compute the implicit derivative with AD using three approaches: 1) Computing the pseudo-inverse of the Hessian matrix with dense linear algebra solution, called *dense*. 2) Finding $\mathbf{X}^*_{\boldsymbol{\Theta}+\delta\boldsymbol{\Theta}}$ with minimization (gradient descent method was used in this work) and applying AD to the entire pipeline of functions (potential energy and its derivatives) with `jaxopt` Python library, called *jaxopt*. 3) Sparse linear operator technique (Results C in the main text) within the AD framework, called *sparse*.

For this study, we used the best-in-class, `NVIDIA A100` 80GB graphic card tailored for scientific computations. To track the usage of the GPU memory for each solver and system size, we used the NVIDIA Management Library, `nvml` through the Python interface provided by the `pynvml` package. Figure 6 shows the time (panel **a**) and memory (panel **b**) required for *inhomogeneous* implicit derivative evaluation as a function of number of atoms in the system. The *inverse* approach shows the best time performance, however, it runs out of 80GB GPU memory at a system of 500 atoms. The second fastest, *jaxopt* method, allows one to achieve system sizes of up to 2000 atoms before saturating the memory. Lastly, *sparse* technique, is the most memory efficient and reaches the systems of up to 7000 atoms on a single `A100` GPU.

We would like to emphasize that while the methods based on AD provide unique advantages such as computational efficiency and ease of implementation, their substantial memory consumption significantly limits their applicability for large-scale simulations.

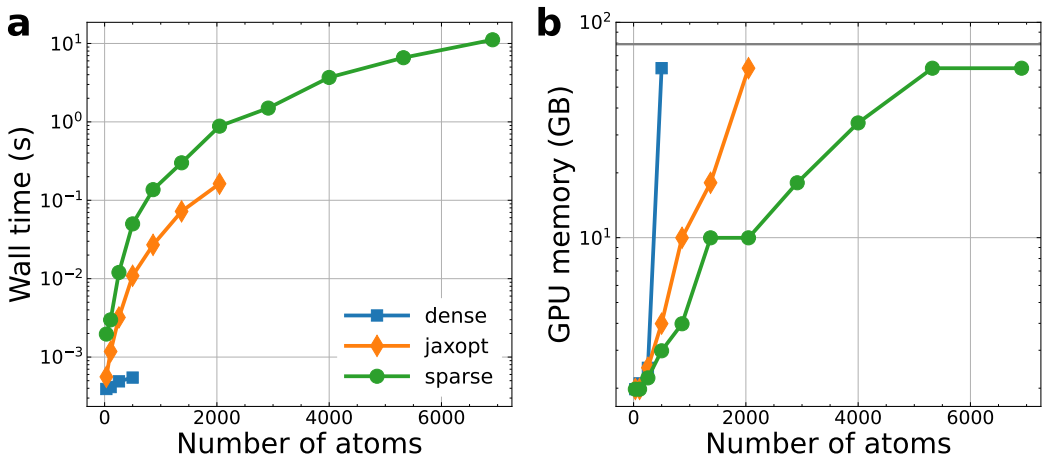

FIG. 6. **Efficiency of AD implicit derivative solvers.** **a** Wall time and and **b** GPU memory required to compute the *inhomogeneous* implicit derivative vs the number of atoms in an LJ random alloy system. The gray horizontal line on panel **b** indicates the GPU memory limit of `NVIDIA A100` 80GB graphic card.

B. Implicit derivative implementation using LAMMPS package

Here, we discuss the time and memory efficiency of the *inhomogeneous* implicit derivative implementation using the `LAMMPS` software. The test system is a vacancy in bcc tungsten with `SNAP` potential (Results G in the main text). We used four CPU nodes with two `AMD EPYC 7763` CPUs and 512 GB of RAM per node. We monitored RAM usage with the `vmstat` tool. We test the efficiency of the *dense* and *sparse* approaches presented above. Additionally, we explore the efficiency of the constraint energy minimization approach, called *energy*, that is described in the main text (Methods A). As seen from Figure 7, the energy approach is the most time- and memory-efficient for large systems. Due to the efficient massive parallelization of the `LAMMPS` software, this method can be effectively scaled and limited only by the amount of available compute resources.

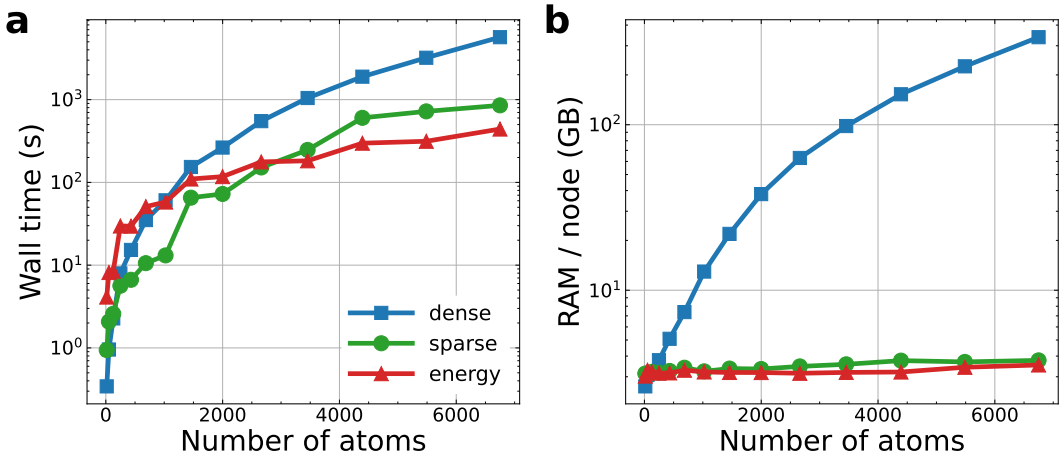

FIG. 7. **Efficiency of the LAMMPS implicit derivative implementation approaches.** **a** Wall time and **b** RAM required to compute the *inhomogeneous* implicit derivative as a function of the number of atoms in a vacancy in a tungsten system with `SNAP` potential.

3. LAMMPS IMPLEMENTATION: ACCURACY AND NUMERICAL DETAILS

A. Accuracy of position change predictions

In this section, we present a comparison of three methods for evaluating the *inhomogeneous* implicit derivative implemented within the `LAMMPS` package. The test system is a vacancy in bcc tungsten and `SNAP` potential parameters are set according to the Results G section of the main text, equation (13), $\boldsymbol{\Theta}(\lambda, m) = \bar{\boldsymbol{\Theta}} + \lambda(\boldsymbol{\Theta}_m - \bar{\boldsymbol{\Theta}})$, with a strong perturbation of $\lambda = 40$. The predicted position changes are computed as

$$\delta\tilde{\mathbf{X}}^{*\,\mathrm{pred}} = \tilde{\mathbf{X}}^* + \delta\boldsymbol{\Theta}\nabla_{\boldsymbol{\Theta}}\tilde{\mathbf{X}}^*_{\bar{\boldsymbol{\Theta}}}, \tag{36}$$

where the *inhomogeneous* implicit derivative $\nabla_{\boldsymbol{\Theta}}\tilde{\mathbf{X}}^*_{\bar{\boldsymbol{\Theta}}}$ is computed with *dense*, *sparse*, or *energy* methods, $\delta\boldsymbol{\Theta} = \lambda(\boldsymbol{\Theta}_m - \bar{\boldsymbol{\Theta}})$, and positions $\tilde{\mathbf{X}}^*$ are obtained through the energy minimization of the system with parameters $\bar{\boldsymbol{\Theta}}$. The true position changes are calculated as follows

$$\delta\tilde{\mathbf{X}}^{*\,\mathrm{true}} = \tilde{\mathbf{X}}^*(\boldsymbol{\Theta}) - \tilde{\mathbf{X}}^*(\bar{\boldsymbol{\Theta}}), \tag{37}$$

where $\tilde{\mathbf{X}}^*(\boldsymbol{\Theta})$ are the minimized positions of a system with parameters $\boldsymbol{\Theta}$.

As demonstrated in Fig. 8a, there is a remarkable agreement between the true and predicted positions, with negligible differences among the three computational methods. Given the low computational cost and memory requirements, the *energy* method stands out as the optimal choice for *inhomogeneous* implicit derivative evaluation.

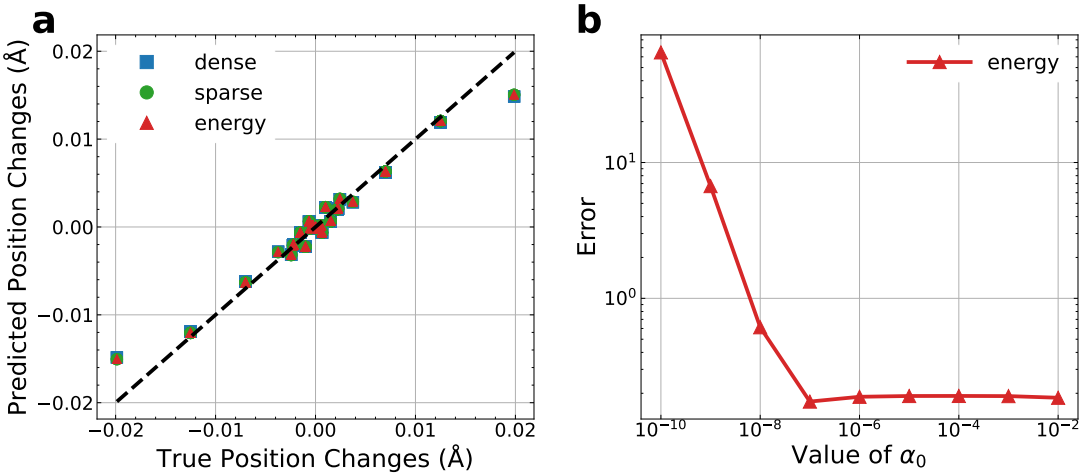

FIG. 8. **Accuracy of the *inhomogeneous* implicit derivative solvers implemented in LAMMPS. a** Predicted vs true position changes for the vacancy in the bcc tungsten system. **b** Error of the *inhomogeneous* implicit derivative prediction of position changes.

B. Numerical details on LAMMPS constraint minimization

Here, we detail our constraint energy minimization approach (*energy*) to compute the *inhomogeneous* implicit derivative implemented in `LAMMPS`. As explained in the main text, the implicit derivative corresponding to a parameter index l is obtained as $[\nabla_{\boldsymbol{\Theta}}\tilde{\mathbf{X}}^*_{\bar{\boldsymbol{\Theta}}}]_l = \left(\tilde{\mathbf{X}}^*_{l,\alpha} - \tilde{\mathbf{X}}^*_{\bar{\boldsymbol{\Theta}}}\right)/\alpha$. We have found that the optimal values of α should be computed for each parameter $\boldsymbol{\Theta}_l$ separately as follows

$$\alpha(l) = \frac{\alpha_0}{\max\left(\left|\nabla^2_{\boldsymbol{\Theta}\tilde{\mathbf{X}}}U\right|\right)}, \tag{38}$$

where α_0 is a constant.

Figure 8b presents the dependence of the error of the implicit derivative prediction as a function of the α_0. The error is computed as

$$\frac{\|\delta\tilde{\mathbf{X}}^{*\,\mathrm{true}} - \delta\tilde{\mathbf{X}}^{*\,\mathrm{pred}}\|}{\|\delta\tilde{\mathbf{X}}^{*\,\mathrm{true}}\|}. \tag{39}$$

For $\alpha_0 < 10^{-7}$, the error is large due to limitations in numerical precision. For larger α_0 values, the error does not change. However, the minimization time increases significantly for $\alpha_0 \geq 10^{-3}$. Therefore, we conclude that values of $\alpha_0 \in [10^{-6}; 10^{-4}]$ are optimal for the *energy* method. For the constraint energy minimization, we use the `FIRE` algorithm implemented in `LAMMPS`.

4. INVERSE DESIGN

A. Adaptive step for loss minimization

This section describes the adaptive step calculation for the inverse design applications. As explained in the main text (Results H), at iteration $k+1$ of the minimization procedure, the potential parameters are updated as $\boldsymbol{\Theta}^{(k+1)} = \boldsymbol{\Theta}^{(k)} - h\nabla_{\boldsymbol{\Theta}} L(\boldsymbol{\Theta}^{(k)})$ and positions as $\mathbf{X}^*_{\boldsymbol{\Theta}^{(k+1)}} = \mathbf{X}^*_{\boldsymbol{\Theta}^{(k)}} + (\boldsymbol{\Theta}^{(k+1)} - \boldsymbol{\Theta}^{(k)})\nabla_{\boldsymbol{\Theta}}\mathbf{X}^*_{\boldsymbol{\Theta}^{(k)}}$. Accordingly, the loss at iteration $k+1$ is

$$L(\boldsymbol{\Theta}^{(k+1)}) = \frac{1}{2}\|\mathbf{X}^*_{\boldsymbol{\Theta}^{(k)}} + h\Delta\mathbf{X}^{(0)}_{\boldsymbol{\Theta}^{(k)}} - \mathbf{X}^*\|^2, \tag{40}$$

where $\Delta\mathbf{X}^{(0)}_{\boldsymbol{\Theta}^{(k)}}$ is the change in atomic positions with step $h = 1$. Then, the *change* in loss at a given iteration is

$$\Delta L(\boldsymbol{\Theta}^{(k+1)}) \equiv L(\boldsymbol{\Theta}^{(k+1)}) - L(\boldsymbol{\Theta}^{(k)}) = h\Delta\mathbf{X}^{(0)\,\top}_{\boldsymbol{\Theta}^{(k)}}\left(\mathbf{X}^*_{\boldsymbol{\Theta}^{(k)}} - \mathbf{X}^*\right) + \frac{1}{2}h^2\|\Delta\mathbf{X}^{(0)}_{\boldsymbol{\Theta}^{(k)}}\|^2. \tag{41}$$

Finally, the step $h(k)$ that minimizes the loss at iteration k can be found as

$$h(k) = -\frac{\Delta\mathbf{X}^{(0)\,\top}_{\boldsymbol{\Theta}^{(k)}}\left(\mathbf{X}^*_{\boldsymbol{\Theta}^{(k)}} - \mathbf{X}^*\right)}{\|\Delta\mathbf{X}^{(0)}_{\boldsymbol{\Theta}^{(k)}}\|^2}. \tag{42}$$

B. W-Be POTENTIAL FINE-TUNING

Figure 9 presents the error minimization for the potential fine-tuning for the W-Be system presented in the main text (Results H).

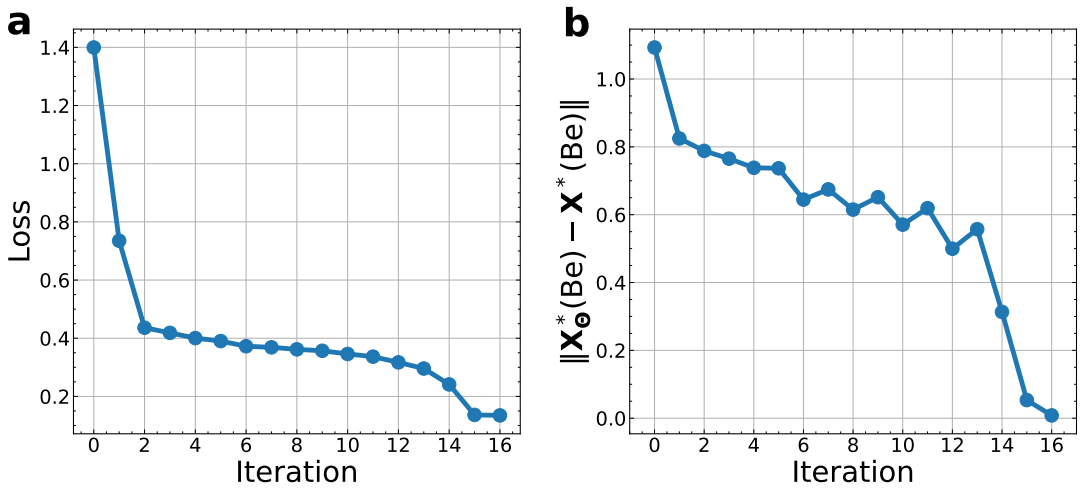

FIG. 9. **Loss minimization for W-Be potential fine-tuning problem. a** Implicit loss during the minimization procedure as defined in equation (9) of the main text. **b** Difference between a current and target Be atom positions during minimization.