Active learning workflows and integrable deep neural networks for representing the free energy functions of alloys

G.H. Teichert[a], A.R. Natarajan[c], A. Van der Ven[c], K. Garikipati[a,b,d,*]

[a]*Department of Mechanical Engineering, University of Michigan*
[b]*Department of Mathematics, University of Michigan*
[c]*Materials Department, University of California, Santa Barbara*
[d]*Michigan Institute for Computational Discovery & Engineering, University of Michigan*

Abstract

The free energy plays a fundamental role in descriptions of many systems in continuum physics. Notably, in multiphysics applications, it encodes thermodynamic coupling between different fields, such as mechanics and chemistry. It thereby gives rise to driving forces on the dynamics of interaction between the constituent phenomena. In mechano-chemically interacting materials systems, even consideration of only compositions, order parameters and strains can render the free energy to be reasonably high-dimensional. In proposing free energy functions as a paradigm for scale bridging, we have previously exploited neural networks for their representation of such high-dimensional functions. Specifically, we have developed an integrable deep neural network (IDNN) that can be trained to free energy derivative data obtained from atomic scale models and statistical mechanics, then analytically integrated to recover a free energy function. The motivation comes from the statistical mechanics formalism, in which certain free energy derivatives are accessible for control of the system, rather than the free energy itself in its entirety. Our current work combines the IDNN with an active learning workflow to improve sampling of the free energy derivative data in a high-dimensional input space. Treated as input-output maps, machine learning representations accommodate role reversals between independent and dependent quantities as the mathematical descriptions change across scale boundaries. As a prototypical material system we focus on Ni-Al. Phase field simulations using the resulting IDNN representation for the free energy of Ni-Al demonstrates that the appropriate physics of the material have been learned.

---

[*] Corresponding Author

Preprint submitted to Elsevier *February 7, 2020*

---

1. Introduction

Many continuum models are fundamentally based on an underlying material free energy. For example, the phase field dynamics described by the Cahn-Hilliard and Allen-Cahn equations have at their core, chemical potentials. These chemical potentials are variational derivatives of the total free energy with respect to composition and order parameters, respectively [1, 2, 3]. Another manifestation is seen in nonlinear elasticity, wherein hyperelastic material models are defined by a strain energy density. The first derivatives of this energy with respect to frame invariant strains define the stresses, and second derivatives give (generally) non-constant elastic moduli. The governing equations for quasi-static elasticity can be derived by extremization of the strain energy [4]. Furthermore, as is obvious, for mechano-chemically coupled material systems cross terms arise among the driving forces, and their correct representation is critical to resolving the dynamics. Due to these fundamental roles, it is important to have a mathematical description of the free energy that accurately reflects the physics. It is actually important to also control the accuracy of free energy derivatives, since differentiation tends to magnify errors.

Several challenges may arise in constructing such a free energy density function from data. One is rapid fluctuations that may exist in the free energy with respect to its arguments, that can be difficult to capture. As we have shown, while spline representations prove superior to various polynomial forms [5] they too can have limitations [6]. Additionally, the data that are calculated or measured are often the derivative of the free energy, rather than the free energy itself. This is typical for statistical mechanics approaches, where the chemical potential is the accessible variable rather than the free energy. In previous work, we introduced a variant on the standard deep neural network (DNN), which we termed an integrable deep neural network (IDNN), to train a chemical free energy function from chemical potential data, while maintaining the appropriate physics of the system [6]. This was done for the free energy as a function of two variables, namely, composition and an order parameter.

Another potential challenge to training a free energy density function comes from its high-dimensional inputs. DNNs are well suited to handling high dimensional input [7, 8, 9, 10, 11], therefore the greater difficulty lies in the creation of data that are well-sampled in the high-dimensional space. Depending on the method for computing or measuring the free energy or its derivatives, a “brute force” approach to sampling the space may be infeasible due to time and cost.

Furthermore, the foundations of theoretical descriptions such as statistical mechanics and continuum physics can prove to be at odds in a manner such that the notions of inputs to and outputs of relations can become reversed as the bridge between scales is crossed. An example appears in this work: the computational approach for statistical mechanics takes certain parameter values as input, and returns composition, order parameters, and the chemical potentials as output. However, the continuum thermodynamics view is of free energies, and therefore chemical potentials being outputs and compositions or order parameters as inputs. In an algorithmic setting, therefore, a continuum computation cannot "demand" chemical potentials at chosen composition or order parameter values. This inability to directly choose the values of the inputs as dictated by theory adds another level of complexity to the creation of a well-sampled dataset in higher dimensions.

Active machine learning approaches (active learning) can provide a solution to the need of sampling data in a high-dimensional space. Active learning algorithms are designed to query for additional data where they would be most useful [12]. In this work, we employ an error-based active learning routine in connection with an IDNN to sample chemical potential data for a material system with one composition and three order parameters as inputs. Embedded in the active learning routine is an iterative, boot-strapping approach that combines the input-output mapping property of neural networks with a linear (therefore invertible) relation between chemical potentials and auxiliary bias potentials. The resulting workflow also circumvents the difficulty of input-output relations alluded to above. With this constellation of innovations, we are able to compute DNN representations of the free energy density function, which is used in phase field simulations to model the growth of precipitates in a Ni-Al alloy.

The paper is organized as follows: Section 2 describes the atomistic and statistical mechanics methods used to obtain chemical potential data, using the Ni-Al system as an example. The IDNN is outlined in Section 3. The active learning workflow, a centerpiece of this communication, is described in Section 4. The phase field method is outlined in Section 5. Workflow and phase field results are presented in Section 6. Concluding remarks appear in Section 7.

2. Chemical potential data from atomic level models

As a model system we consider Ni-rich Ni-Al alloys, which exhibit interesting order-disorder phenomena on the face-centered cubic (fcc) crystal structure[13]. At dilute Al concentrations, Ni-Al alloys form an fcc solid solution characterized by disordered arrangements of Ni and Al over the sites of the fcc lattice. At

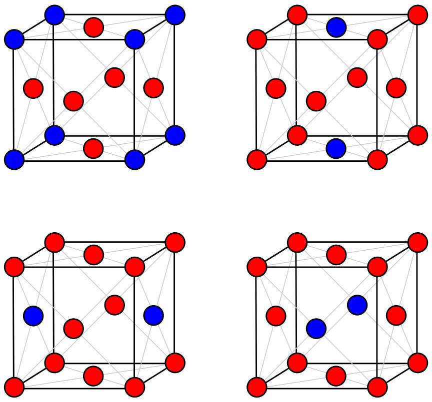


Figure 1: Four variants of the $L1_2$ crystal structure of $\mathrm{Ni_3Al}$. The Ni atoms are colored red, and the Al atoms are colored blue.

compositions around the $\mathrm{Ni_3Al}$ stoichiometry, the Ni and Al atoms prefer an ordered arrangement on fcc, adopting the $\mathrm{L1_2}$ ordering. The $\mathrm{L1_2}$ ordering has a lower translational symmetry than the underlying parent fcc lattice. While the primitive repeat unit of fcc consists of one site, that of the $\mathrm{L1_2}$ ordering has four sites. This results in four symmetrically equivalent translational variants of the $\mathrm{L1_2}$ ordering as illustrated in Figure 1. The translational variants can coexist and when they impinge on each other, they form an anti-phase boundary.

The thermodynamic properties of alloys that undergo order-disorder transformations can be calculated with statistical mechanics [14]. This requires a mathematical way of tracking the instantaneous arrangement of atoms over the sites of the parent crystal, which is realized by assigning an occupation variable σ_i to each lattice site i with $\sigma_i = \pm 1$ depending on whether the site is occupied by Ni or Al. The collection of all occupation variables forms the vector $\boldsymbol{\sigma} \in \mathbb{Z}^{n_{\mathrm{lat}}}$, where n_{lat} is the number of lattice sites. The energy of the crystal for any ordering $\boldsymbol{\sigma}$ can be expressed as a polynomial expansion of the occupation variables σ_i according to [15, 14]

$$E(\boldsymbol{\sigma}) = E_0 + \sum_i E_1^i \sigma_i + \sum_{i,j} E_2^{ij} \sigma_i \sigma_j + \sum_{i,j,k} E_3^{ijk} \sigma_i \sigma_j \sigma_k \ldots \tag{1}$$

where the successive sums on the right-hand side extend over all sites, i, all distinct pairs of sites i, j, all distinct triplets of sites i, j, k, etc., leading to the appellation of cluster expansion for this type of representation. The expansion coefficients, E_0, E_1^i, E_2^{ij}, etc. can be fit to a training set of energies for different configurations as calculated with a first-principles method such as density functional theory (DFT) [14]. The cluster expansion of Eq. 1 can be evaluated

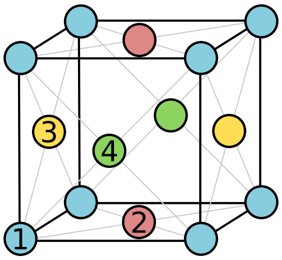


Figure 2: An $L1_2$ structure can be described by the composition of the four sublattice sites numbered here.

rapidly, making it ideally suited for Monte Carlo simulations to calculate thermodynamic averages. A cluster expansion Hamiltonian parameterized by Goiri and Van der Ven[13] was used to describe the effect of configurational ordering in the binary Ni-Al alloy.

The atoms of an alloy in thermal equilibrium constantly fluctuate from one arrangement to another. Nevertheless, the average degree of ordering remains constant in equilibrium. In this context, it is convenient to introduce thermodynamic long-range order parameters [16] that track the equilibrium degree of ordering. The degree with which Ni and Al adopt $\mathrm{L1}_2$ type ordering can be tracked with average sublattice concentration variables x_i, $i = 1, \dots, 4$, one for each of the four sublattices of the cubic unit cell of $\mathrm{L1}_2$ shown in Figure 2. Symmetry arguments then suggest the following linear combinations of the sublattice concentrations for the $\mathrm{L1}_2$ ordering [17, 16]:

$$\begin{aligned}
\eta_0 &= \frac{1}{4}\left(x_1 + x_2 + x_3 + x_4\right) \\
\eta_1 &= \frac{1}{4}\left(x_1 + x_2 - x_3 - x_4\right) \\
\eta_2 &= \frac{1}{4}\left(x_1 - x_2 - x_3 + x_4\right) \\
\eta_3 &= \frac{1}{4}\left(x_1 - x_2 + x_3 - x_4\right),
\end{aligned} \tag{2}$$

which can also be expressed using the transformation matrix $\boldsymbol{Q}$:

$$\boldsymbol{\eta} = \boldsymbol{Q}\boldsymbol{x} \tag{3}$$

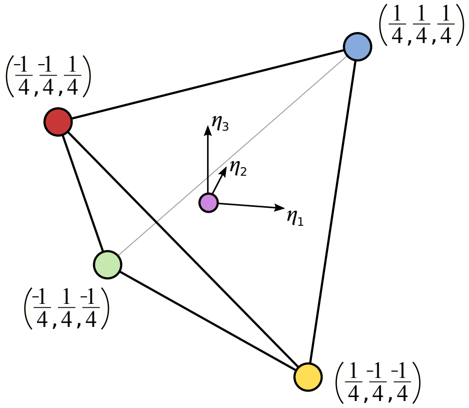


Figure 3: The four perfect $L1_2$ orderings lie on the vertices $(\frac{1}{4},\frac{1}{4},\frac{1}{4})$, $(-\frac{1}{4},-\frac{1}{4},\frac{1}{4})$, $(-\frac{1}{4},\frac{1}{4},-\frac{1}{4})$, $(\frac{1}{4},-\frac{1}{4},-\frac{1}{4})$ of the tetrahedron in the $(\eta_1.\eta_2,\eta_3)$ space The origin corresponding to a completely disordered state.

where

$$\boldsymbol{Q} = \frac{1}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \tag{4}$$

In this form, the first order parameter η_0 tracks the overall composition of the alloy. The three remaining order parameters, η_1, η_2 and η_3, measure the degree of long-range order that is commensurate with the periodicity of the $L1_2$ phase. They are equal to zero in the completely disordered alloy (since all sublattice concentrations are then equal to each other) and adopt non-zero values when the alloy exhibits average long-range order. Furthermore, the three order parameters are able to distinguish between the four translation variants of $L1_2$. This is illustrated in Figure 1, which shows that each translational variant of $L1_2$ (Figure 1) corresponds to a corner of a tetrahedron in the three dimensional η_1, η_2 and η_3 order-parameter space at a composition $\eta_0 = \frac{1}{4}$.

Simulating the microstructure evolution of a two-phase mixture of the disordered solid solution and the different translational variants of an ordered phase requires a free energy description, g, that is a function of composition (i.e. η_0) and order parameters (i.e. η_1, η_2 and η_3 for the $L1_2$ ordering). In the binary Ni-Al alloy, the free energy will have a minimum at the origin of the η_1, η_2 and η_3 space at compositions where the solid solution is stable. The energy landscape will also have four minima related by symmetry in the vicinity of the translational variants of $L1_2$ in η_1, η_2 and η_3 space at compositions close to the Ni_3Al stoichiometry. Since the free energy, $g(\eta_0,\eta_1,\eta_2,\eta_3)$, is a continuous curve, there will be regions in $\eta_0, \eta_1, \eta_2, \eta_3$ space where g has negative curvatures. In these regions the alloy is unstable with respect to ordering and/or composition fluctuations.

Each order parameter, η_i, has a conjugate 'chemical potential', μ_i, that can be derived from the free energy g according to $\mu_i = \partial g/\partial \eta_i$. In Monte Carlo approaches, it is easier to control μ_i than the order parameters η_i, since the latter are related to the thermodynamic averages of sublattice concentrations. A difficulty, however, emerges in regions where the free energy has negative curvatures. To access these regions, biased Monte-Carlo simulations[16] with additional bias parameters ϕ_i and κ_i, $i = 0, \dots, 3$ are used. The bias parameters are then the inputs to the Monte Carlo simulations, which return statistical averages of the order parameters $\langle \eta_i \rangle$, $i = 0, \dots, 3$. The bias parameters and statistical averages are related to the derivative of the free energy per atom, $g(\eta_0, \eta_1, \eta_2, \eta_3)$ through the following:

$$\mu_i := \frac{\partial g}{\partial \eta_i}\Big|_{\langle \boldsymbol{\eta} \rangle} = -2\phi_i(\langle \eta_i \rangle - \kappa_i), \qquad i = 0, \dots, 3 \tag{5}$$

The cluster expansions of Equation (1) and Monte Carlo statistical mechanics calculations were performed with the `CASM` code [18, 14, 19, 20]. The resulting statistical averages were used to calculate the free energy derivative data.

3. Integrable deep neural network

As explained above, the atomic models directly provide data as derivatives of the free energy. However, for reasons driven by physics-constrained modelling that were explained in the Introduction, we seek to represent the free energy itself in addition to its derivatives. For such purposes, we have previously introduced the notion of an integrable deep neural network (IDNN) [6]. IDNNs are trained to derivative data and can be analytically integrated to recover the antiderivative function (e.g. the free energy). We summarize their mathematical basis and construction here, and refer the reader to the original work [6] for details.

Mathematically, the IDNN is constructed by differentiating a standard deep neural network (DNN) by each of its inputs, x_k (see a schematic in Figure 4). The following equations described the structure of a standard DNN with n hidden layers, where $\boldsymbol{W}_\ell$, $\boldsymbol{b}_\ell$ are the weight matrix and bias vector of hidden layer ℓ, f is the activation function, a_ℓ and z_ℓ are intermediate vector values at

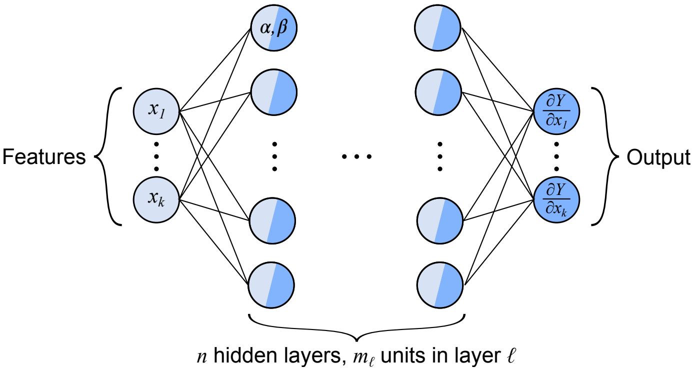


Figure 4: Schematic of an integrable deep neural network (DNN).

each layer, and Y is the DNN output:

$$\begin{aligned} \boldsymbol{z}_\ell &= \boldsymbol{b}_\ell + \boldsymbol{W}_\ell \boldsymbol{a}_{\ell-1} \\ \boldsymbol{a}_\ell &= f(\boldsymbol{z}_\ell) \\ Y &= \boldsymbol{b}_{n+1} + \boldsymbol{W}_{n+1} \boldsymbol{a}_n \end{aligned} \tag{6}$$

After differentiation, additional equations arise to describe the IDNN, which is represented by $\partial Y / \partial x_k$:

$$\begin{aligned} \frac{\partial \boldsymbol{a}_\ell}{\partial x_k} &= f'(\boldsymbol{z}_\ell) \odot \left(\boldsymbol{W}_\ell \frac{\partial \boldsymbol{a}_{\ell-1}}{\partial x_k} \right) \\ \frac{\partial Y}{\partial x_k} &= \boldsymbol{W}_{n+1} \frac{\partial \boldsymbol{a}_n}{\partial x_k} \end{aligned} \tag{7}$$

where the operator $\odot$ denotes element-wise multiplication. Note that both the activation function and its derivative are used in the IDNN. If the activation function is chosen to be the softplus function, $f(x) := \ln(1 + e^x)$, its derivative, $f'(x) = 1/(1 + e^{-x})$, is also a common activation function, namely the logistic function. Note that though the IDNN, $\partial Y / \partial x_k$, and its associated DNN, Y, have different structures, they share the same weights and biases. It is this fact that creates the derivative/integral relationship between the IDNN and DNN. Of relevance to implementation, the integration to obtain Y is available for no extra training.

Using modern deep learning libraries, an IDNN can simply be defined by constructing a standard DNN, then applying a gradient operator to the output. For a given set of inputs and derivative data $\{(\hat{\boldsymbol{x}}_\theta, \hat{\boldsymbol{y}}_\theta)\}$, the mean square error of the DNN gradient (i.e. the IDNN) and the chemical potential data is minimized over the space of weights and biases, as represented by the following:

$$\hat{\boldsymbol{W}}, \hat{\boldsymbol{b}} = \underset{\boldsymbol{W},\boldsymbol{b}}{\arg\min} \sum_{k=1}^{n} \mathrm{MSE}\left(\left.\frac{\partial \boldsymbol{Y}(\boldsymbol{x}, \boldsymbol{W}, \boldsymbol{b})}{\partial x_k}\right|_{\hat{\boldsymbol{x}}_\theta}, \hat{y}_{k_\theta} \right) \tag{8}$$

The resulting trained standard DNN gives the integrated DNN.

4. Active learning workflow

It is desirable to have a free energy derivative that is uniformly sampled in the space of order parameters for use in mesoscale models. However, Monte-Carlo techniques use the bias parameters ϕ_i and κ_i as input, with the order parameter values emerging as thermodynamic averages from the simulations. The bias parameters are related to the chemical potentials and order parameters through eq. 5. Typically, in biased Monte-Carlo simulations the bias curvature, ϕ_i is held constant, while κ_i values are varied.

Naive sampling of the κ_i parameters can lead to some regions in the order parameter space that are undersampled and others that are oversampled. The uniformity of sampling can be improved by creating and using a surrogate model, $\hat{\boldsymbol{\mu}}(\boldsymbol{\eta})$, to predict which values of κ_i will give uniform sampling in the $\boldsymbol{\eta}$ space. In our treatment, the surrogate $\hat{\boldsymbol{\mu}}(\boldsymbol{\eta})$ is an IDNN.

While all values of κ_i are physically valid, some values are more relevant than others. It is not initially apparent what the relevant range of κ_i values should be. However, physically valid values for each sublattice composition x_i lie in the range $[0, 1]$. Therefore, instead of using κ to define the domain of the search space, we sample from the sublattice composition space. We impose the additional constraint that $\eta_0 \leq 0.25$, since the Ni-Al system transitions from FCC to BCC above that point. Given that the PDEs defining the phase field model are written in terms of the composition and order parameters, we pose the problem in terms of $\boldsymbol{\eta}$.

For each iteration of the workflow, we perform a global sampling from the sublattice composition space using a Sobol$'$ sequence. We use a Sobol$'$ sequence because of its space-filling and noncollapsing properties [21, 22, 23]. The sublattice composition values are converted to order parameter values with Eq. (2). These are used as input to the surrogate model, which gives a prediction for the chemical potentials and, using Eq. (5), the associated κ_i bias parameters. With these κ_i values as input, the cluster expansions and Monte Carlo computations (within the `CASM` platform) return a set of composition and order parameter values, η_i, with their corresponding chemical potentials, μ_i, for $i = 0, \ldots 3$.

Once the dataset is updated, the IDNN is trained using all of the chemical potential data. After training is complete for the current iteration, the active learning component of the workflow takes place. The pointwise training error is evaluated for the IDNN using only the data points from the most recent global sampling. The data points are sorted according to error. The N data points giving the highest error are used to identify areas that would benefit from additional data. Additionally, the appearance of energy wells in the surface are of interest, since they correspond with the material phases. These energy wells are identified by evaluating the Hessian of the free energy surface for sampled points and selecting points with a positive definite Hessian and a low gradient norm. Random points near these data with either high error or within an energy well are used to define a local sampling of order parameter values. As before, the IDNN as the surrogate model and associated equations provide κ_i values that become input to `CASM`, resulting in an updated dataset and concluding the iteration.

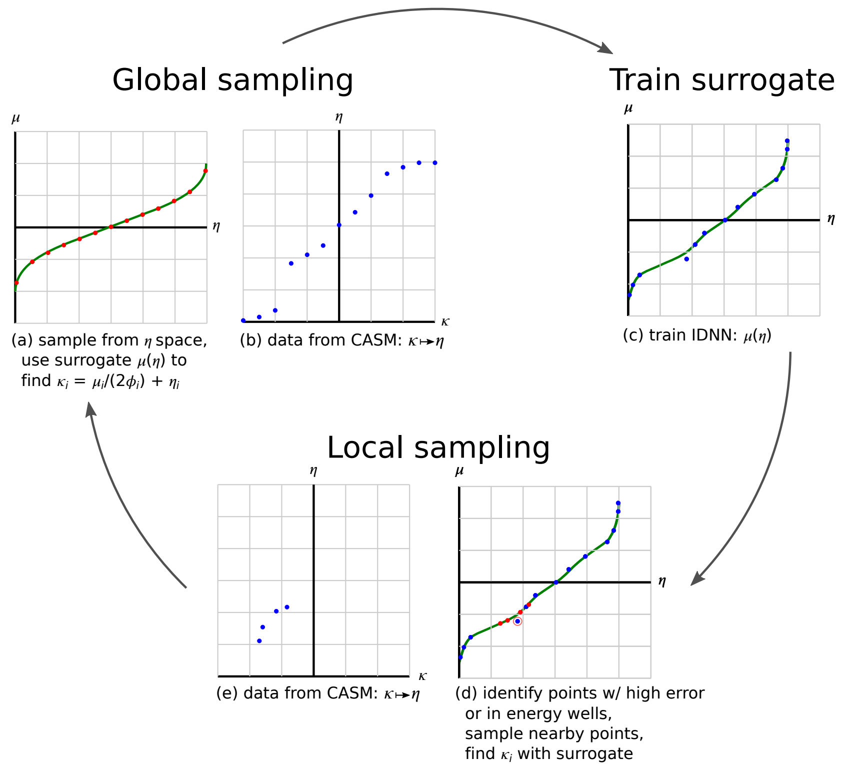
For the first iteration of the workflow, there are no data to use to create an IDNN surrogate model for the chemical potential. We instead use the equations for the chemical potentials of an ideal solution, which are the partial derivatives of the ideal solution free energy with respect to the order parameters. With the free eneergy and chemical potentials expressed in terms of the more transparent sublattice compositions, the relations are:

$$\tilde{g}(\boldsymbol{x}) = \frac{k_B T}{4} \sum_{i=1}^{4} \left(x_i \log x_i + (1 - x_i)\log(1 - x_i)\right) \tag{9}$$

$$\tilde{\mu}_{i-1}(\boldsymbol{x}) := \frac{k_B T}{4} \sum_{j=1}^{4} \log\left(\frac{x_j}{1 - x_j}\right) Q^{-1}_{ji} \qquad i = 1, .., 4 \tag{10}$$

where k_B is the Boltzmann constant and T is the temperature. Figure 4 and algorithm 1 summarize the full workflow.

Algorithm 1. *Active learning of free energy*



(a) sample from η space, use surrogate $\mu(\eta)$ to find $\kappa_i = \mu_i/(2\phi_i) + \eta_i$

(b) data from CASM: $\kappa \mapsto \eta$

(c) train IDNN: $\mu(\eta)$

(d) identify points w/ high error or in energy wells, sample nearby points, find κ_i with surrogate

(e) data from CASM: $\kappa \mapsto \eta$

Figure 5: Schematic of the active learning workflow, described with hypothetical 1D data.

```
Initialize $k = 1$, $\mathcal{D} = \emptyset$, $\hat{\boldsymbol{\mu}}_0(\boldsymbol{\eta}) = \tilde{\boldsymbol{\mu}}(\boldsymbol{Q}^{-1}\boldsymbol{\eta})$.  Iterate over the following:
    1. Global sampling:
        (a) Select sample points in the sublattice composition space:
                $\{\boldsymbol{x} \in (0,1) \times \cdots \times (0,1) \mid \eta_0 \leq 0.25\}$
        (b) Evaluate the corresponding bias parameter values:
                $\kappa_i = \frac{1}{2\phi_i}\hat{\mu}_i(\boldsymbol{Qx}) + \sum_j Q_{ij}x_j$
        (c) Use the $\boldsymbol{\kappa}$ values as input to CASM to compute the order parameter values, $\boldsymbol{\eta}$ and chemical potential values, $\boldsymbol{\mu}$.
            Resulting values form data set $\mathcal{G}_k = \{(\boldsymbol{\eta}, \boldsymbol{\mu})\}$.
        (d) Update $\mathcal{D} := \mathcal{D} \cup \mathcal{G}_k$.
    2. Train IDNN surrogate model $\hat{\boldsymbol{\mu}}_k(\boldsymbol{\eta})$ to the data set $\mathcal{D}$, initialized from $\hat{\boldsymbol{\mu}}_{k-1}(\boldsymbol{\eta})$ when $k > 2$.
        (a) Break if $||\hat{\boldsymbol{\mu}}_k(\boldsymbol{\eta}) - \hat{\boldsymbol{\mu}}_{k-1}(\boldsymbol{\eta})||_2 <$ tol,
            for $\boldsymbol{\eta}$ sampled using a Sobol$'$ sequence.
    3. Local (error-based) sampling:
        (a) Identify points in $\mathcal{G}_k$ that give highest IDNN error.
        (b) Identify points with a positive definite Hessian and low gradient norm.
        (c) Submit nearby points to CASM; results form data set $\mathcal{L}_k = \{(\boldsymbol{\eta}, \boldsymbol{\mu})\}$.
        (d) Update $\mathcal{D} := \mathcal{D} \cup \mathcal{L}_k$.
        (e) k = k + 1
```

4.1. FCC symmetry

Due to the symmetry of the FCC crystal structure in the Ni-Al system, the free energy density should be invariant to permutations of η_1, η_2, η_3 and changes in the sign of any two of the order parameters η_1, η_2, and η_3 [17]. To impose this invariance, we express the free energy density as a function of the following invariants:

$$\begin{aligned} h_1(\eta_1, \eta_2, \eta_3) &= \eta_1\eta_2\eta_3 \\ h_2(\eta_1, \eta_2, \eta_3) &= \eta_1^2 + \eta_2^2 + \eta_3^2 \\ h_3(\eta_1, \eta_2, \eta_3) &= \eta_1^2\eta_2^2 + \eta_2^2\eta_3^2 + \eta_3^2\eta_1^2 \end{aligned} \tag{11}$$

Thus, the proper symmetry is perfectly enforced by setting $g(\eta_0, \eta_1, \eta_2, \eta_3) := \hat{g}(\eta_0, h_1, h_2, h_3)$.

5. Phase field formulation

We used the analytically integrated free energy DNN in phase field computations. The phase field model was based on the coupled Cahn-Hilliard and Allen-Cahn equations [1, 2].

The order parameter η_0 being equal to the homogeneous composition, is a conserved value. The remaining order parameters η_1, η_2, η_3 are nonconserved variables. Given the homogeneous free energy density $g(\boldsymbol{\eta})$ as a function of order parameters, we define the total free energy as the following:

$$\Pi[\boldsymbol{\eta}] = \int\limits_{\Omega} \left[g(\boldsymbol{\eta}) + \sum_{i=0}^{3} \frac{1}{2} \chi_i |\nabla \eta_i|^2 \right] \mathrm{d}V \tag{12}$$

The corresponding chemical potentials are given by the variational derivatives of the total free energy, namely $\mu_i := \delta \Pi / \delta \eta_i$. Using standard variational methods results in the following equations for the chemical potentials:

$$\mu_i = \frac{\partial g}{\partial \eta_i} - \chi_i \nabla^2 \eta, \qquad i = 0, \dots, 3 \tag{13}$$

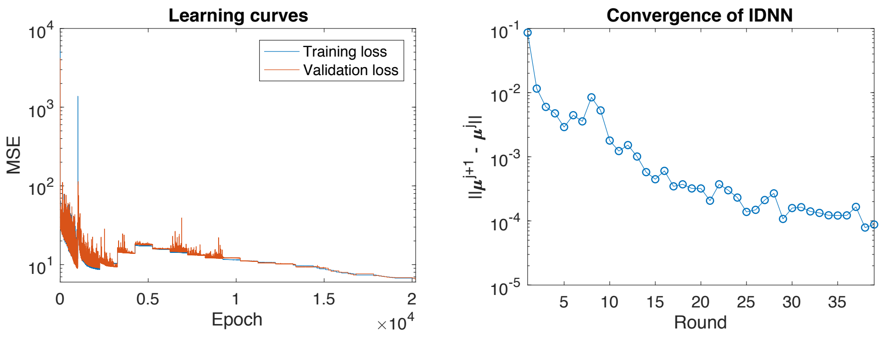
The phase field model consists of the Cahn-Hilliard and Allen-Cahn equations, given by the following, respectively:

$$\frac{\partial \eta_0}{\partial t} = -\nabla \cdot \boldsymbol{J} \tag{14}$$

$$\frac{\partial \eta_i}{\partial t} = -L \mu_{\eta_i}, \qquad i = 1, 2, 3 \tag{15}$$

The Cahn-Hilliard equation is in conservation form, with the flux defined as $\boldsymbol{J} := -M \nabla \mu_0$. It models the overall composition of the system through η_0, while conserving mass. The Allen-Cahn equation models the time evolution of the long-range ordering of the system through the non-conserved order parameters η_i, $i = 1, 2, 3$. The two equations are coupled through the chemical potentials being derived from the same free energy. Periodic boundary conditions were applied.

The weak form of the equations for the case with a uniform mobility and periodic boundary conditions, as solved by the IGA formulation [24, 25, 5], takes



(a) (b)

Figure 6: (a) Learning curves for the IDNN training over the active learning workflow. Periodic jumps in the loss occur at the beginning of each new round of the workflow. (b) Convergence of the IDNN over rounds of the active learning workflow. The IDNN is evaluated after each round using the same set of points from a Sobol′ sequence to find the predicted chemical potentials. In this case, the Sobol′ sequence is used to approximate integration [21]. The norm of the difference in the chemical potential evaluations from one round to the next are plotted.

the following form:

$$0 = \int_\Omega \left[w \frac{\partial \eta_0}{\partial t} + M \left(\nabla w \cdot \nabla \frac{\partial g}{\partial \eta_0} + \chi_0 \nabla^2 w \nabla^2 \eta_0 \right) \right] \mathrm{d}V \tag{16}$$

$$0 = \int_\Omega \left[w \frac{\partial \eta_i}{\partial t} + L \left(w \frac{\partial g}{\partial \eta_i} + \chi_i \nabla w \cdot \nabla \eta_i \right) \right] \mathrm{d}V, \qquad i = 1, 2, 3 \tag{17}$$

6. Implementation and results

The workflow was run on the ConFlux high performance computing cluster at the University of Michigan, with the `CASM` Monte Carlo runs taking place on the CPU nodes and training of the IDNN, implemented with Keras and Tensorflow, utilizing GPUs. Between two and three thousand new points were calculated with each global sampling, and up to 2,800 new data points were added with each local sampling. Over 136,000 data points had been sampled by the end of the 39th iteration of the workflow in Algorithm 1. The values of the chemical potentials were scaled by 100 to improve the IDNN fit.

Since the data after the first global sampling were still quite sparse, a hyperparameter search was performed only after the second global sampling. The IDNN in the first iteration of the workflow was set to have two hidden layers with 20 units each and a learning rate of 0.2. The hyperparameter search was performed by comparing 30 IDNN architectures and learning rates. Learning rate values were randomly chosen log-uniformly from the domain $[0.005, 0.5]$, and the units per layer were chosen uniformly from the domain $[20, 500]$. We kept the number of hidden layers low for two reasons. First, the IDNN will be evaluated at every quadrature point in the phase field simulation, so it is beneficial to have a small network to reduce computation time. Second, while the combination of softplus and sigmoid activation functions works well with the IDNN structure, sigmoid activation functions suffer from the vanishing gradient pathology during training if they are very deep [26, 27]. Thus, all IDNNs were set with either two or three hidden layers. Each of the 30 IDNNs was trained for 250 epochs, and the IDNN with the lowest validation loss was chosen. With this approach, an initial learning rate of 0.199 and two hidden layers with a width of 158 units were selected. For all subsequent iterations of the workflow, the architecture of the IDNN was kept fixed, and the training of the weights and biases resumed at each new workflow iteration without reinitialization.

The learning curves for the full workflow are shown in Figure 6a. The IDNN was trained for 1000 epochs in each workflow iteration using the `AdagradOptimizer`. Training was terminated early for a workflow iteration if there was no decrease in the validation loss for 150 consecutive epochs. A learning rate decay of 0.8 was multiplied at each new iteration of the workflow. Additionally, the learning rate was temporarily reduced by half whenever the validation loss plateaued for 100 epochs, then reset at the beginning of the next workflow iteration. Periodic jumps in the loss occur at the beginning of each new round of the workflow, as new data are added to the set. For the first five rounds, the search space is slightly expanded to oversample the edges of the physical domain and resolve the data as the chemical potentials diverge according to the ideal solution equation (10). This is reflected in the upward jump in the loss between the first five rounds.

Convergence of the IDNN was monitored by computing the 2-norm of the difference in predicted chemical potential values from one round to the next, evaluated at points determined by a Sobol$'$ sequence. The workflow converged with a tolerance of 1×10^{-4} within 39 iterations, as seen in Figure 6b.

The evolution of the IDNN is presented in Figure 7 by plotting a slice of the predicted chemical potential μ_0 as a function of η_0 and η_1, with $\eta_1 = \eta_2 = \eta_3$. Significant changes are seen in the first few iterations of the workflow, with evident convergence in the later iterations. A slice of the final, analytically integrated free energy DNN, referenced to pure Ni and the perfectly ordered $L1_2$, is shown in Figure 8, again with $\eta_1 = \eta_2 = \eta_3$. An energy well is seen at about $\eta_0 = 0.23$, corresponding to the γ' Ni-Al precipitates for the $L1_2$ variant

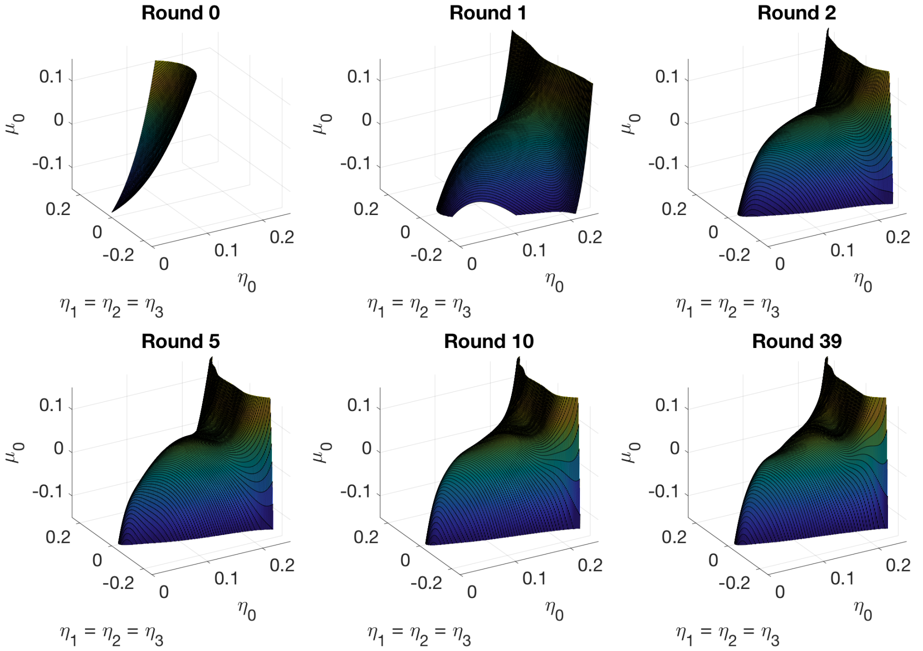


Figure 7: Evolution of the IDNN representing the chemical potential for a two-dimensional subspace, over iterations of the active learning workflow.

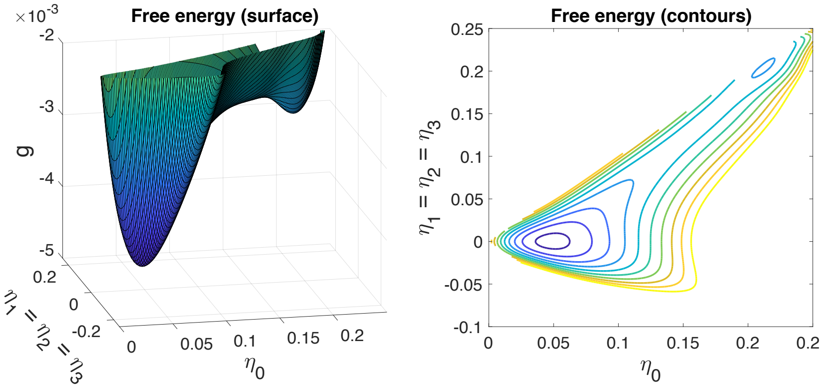
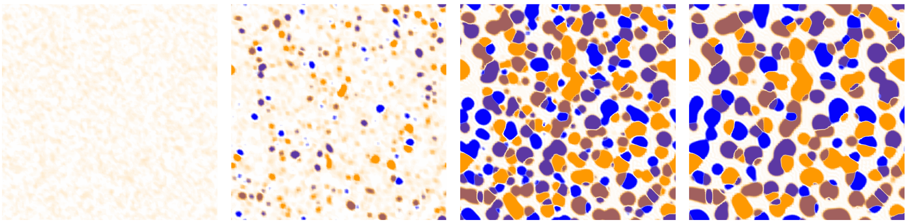


Figure 8: Surface and contour plots for a 2D subspace ($\eta_1 = \eta_2 = \eta_3$) of the converged DNN representation of the free energy density.



(a) Time step 0 (b) Time step 30 (c) Time step 50 (d) Time step 70

Figure 9: Evolution of the Ni_3Al precipitates, with four $L1_2$ variants shown as blue, orange, brown, and purple.

with all positive valued order parameters. A well near $\eta_0 = 0.05$ represents the γ solid solution phase. A few spurious, shallow wells exist in the DNN surface, but they do not seem to negatively affect the resulting precipitate formation in the phase field results (see Figure 9).

The phase field equations were solved numerically using isogeometric analysis (IGA) [24]. The simulation was performed using the `mechanoChemIGA` code[1], which is based on the `PetIGA` [28] and `PETSc` [29, 30, 31] libraries, and run on the XSEDE Comet HPC cluster [32]. Initial conditions were random about $\eta_0 = 0.12$ and $\eta_i = 0$, $i = 1, 2, 3$. Adaptive time stepping was used. Results are plotted in Figure 9. The expected development of precipitates with multiple $L1_2$ variants are shown, demonstrating that the free energy DNN has captured the appropriate physics. The blue, orange, brown, and purple regions represent the four $L1_2$ variants that develop in the Ni-Al precipitates at about $\eta_0 = 0.23$, separated by anti-phase boundaries. The tan background shows the γ phase solid solution. The precipitates and anti-phase domains are seen to grow and coarsen over time, while the antiphase boundaries become flatter. Since elasticity has not been included in this simulation, the faceting that occurs in experiments is not observed in the simulation.

7. Conclusions

In this work, we have presented an active learning workflow to improve sampling of chemical potential data while simultaneously constructing a deep neural network (DNN) representation of the free energy. The application of active learning, in which the machine learning method identifies regions of the

---

[1] Code available at github.com/mechanoChem/mechanoChemIGA

data space where more data are needed and also drives the sampling of the high-dimensional space in those regions.

Using an integrable deep neural network (IDNN) to train to the chemical potential provides an analytically integrated free energy density DNN. This integrability is critical in mechano-chemical coupling, wherein the stresses are defined as derivatives of the free energy density with respect to strains. However, even in the absence of coupling to elasticity, it is essential to maintain consistency of the free energy/chemical potential representation. In this context, naïvely training to derivative data without enforcing consistency by ensuring a unique antiderivative (up to constants) will manifest as unphysical results: The chemical potentials will not reflect the appropriate physics inherent in their being derivatives of a single free energy density function.

To demonstrate that the resulting free energy DNN accurately reflects the physics of the Ni-Al system, we performed phase field simulations using the free energy DNN as input. The phase field results show the creation, growth, and coarsening of Ni_3Al precipitates with anti-phase boundaries. Building on our previous work, these results continue to demonstrate the effectiveness of machine learning methods in addressing challenges in computational physics.

8. Acknowledgements

We gratefully acknowledge the support of Toyota Research Institute, Award #849910, "Computational framework for data-driven, predictive, multi-scale and multi-physics modeling of battery materials". This work has also been supported in part by National Science Foundation DMREF grant #1729166, "Integrated Framework for Design of Alloy-Oxide Structures". Simulations in this work were performed using the Extreme Science and Engineering Discovery Environment (XSEDE) Comet at the San Diego Supercomputer Center through allocation TG-DMR180072. XSEDE is supported by National Science Foundation grant number ACI-1548562. Computing resources also were provided in part by the NSF via grant 1531752 MRI: Acquisition of Conflux, A Novel Platform for Data-Driven Computational Physics (Tech. Monitor: Ed Walker).

References

[1] J. W. Cahn and J. E. Hilliard. Free energy of a nonuniform system. i interfacial energy. *The Journal of Chemical Physics*, 28:258–267, 1958.

[2] S. M. Allen and J. W. Cahn. A microscopic theory for antiphase boundary motion and its application to antiphase boundary coarsening. *Acta Metallurgica*, 27:1085–1091, 1979.

[3] Nikolas Provatas and Ken Elder. *Phase-field methods in materials science and engineering.* John Wiley & Sons, 2011.

[4] Jerrold E Marsden and Thomas JR Hughes. *Mathematical foundations of elasticity.* Courier Corporation, 1994.

[5] Gregory H. Teichert, N.S. Harsha Gunda, Shiva Rudraraju, Anirudh Raju Natarajan, Brian Puchala, Krishna Garikipati, and Anton Van der Ven. A comparison of Redlich-Kister polynomial and cubic spline representations of the chemical potential in phase field computations. *Computational Materials Science*, 128:127 – 139, 2017.

[6] G.H. Teichert, A.R. Natarajan, A. Van der Ven, and K. Garikipati. Machine learning materials physics: Integrable deep neural networks enable scale bridging by learning free energy functions. *Computer Methods in Applied Mechanics and Engineering*, 353:201 – 216, 2019.

[7] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[8] Yong Xu, Jun Du, Li-Rong Dai, and Chin-Hui Lee. A regression approach to speech enhancement based on deep neural networks. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 23(1):7–19, January 2015.

[9] Anirudh Raju Natarajan and Anton Van der Ven. Machine-learning the configurational energy of multicomponent crystalline solids. *npj Computational Materials*, 4(1), 2018.

[10] G. Teichert and K. Garikipati. Machine learning materials physics: Surrogate optimization and multi-fidelity algorithms predict precipitate morphology in an alternative to phase field dynamics. *Computer Methods in Applied Mechanics and Engineering*, 344:666–693, 2019.

[11] Evgeny Putin, Polina Mamoshina, Alexander Aliper, Mikhail Korzinkin, Alexey Moskalev, Alexey Kolosov, Alexander Ostrovskiy, Charles Cantor, Jan Vijg, and Alex Zhavoronkov. Deep biomarkers of human aging:

Application of deep neural networks to biomarker development. *Aging*, 8(5):1021–1033, 05 2016.

[12] Burr Settles. *Active learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2012.

[13] Jon Gabriel Goiri and Anton Van der Ven. Phase and structural stability in Ni-Al systems from first principles. *Physical Review B*, 94(9), September 2016.

[14] Anton Van der Ven, John C. Thomas, Brian Puchala, and Anirudh Raju Natarajan. First-principles statistical mechanics of mult-component crystals. *Annual Review of Materials Research*, 48:27–55, 2018.

[15] J. M. Sanchez, F. Ducastelle, and D. Gratias. Generalized cluster description of multicomponent systems. *Physica A*, 128:334–350, 1984.

[16] Anirudh Raju Natarajan, John C. Thomas, Brian Puchala, and Anton Van der Ven. Symmetry-adapted order parameters and free energies for solids undergoing order-disorder phase transitions. *Phys. Rev. B*, 96:134204, Oct 2017.

[17] J. W. Braun, R. J. annd Cahn, G. B. McFadden, and A. A. Wheeler. Anisotropy of interfaces in an ordered alloy: a multiple–order–parameter model. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 355, 1997.

[18] `https://github.com/prisms-center/CASMcode`. CASM: A Clusters Approach to Statistical Mechanics, 2018.

[19] John C. Thomas and Anton Van der Ven. Finite-temperature properties of strongly anharmonic and mechanically unstable crystal phases from first principles. *Physical Review B*, 88(21):214111–214111, December 2013.

[20] B. Puchala and A. Van der Ven. Thermodynamics of the Zr-O system from first-principles calculations. *Phys. Rev. B*, 88:094108, Sep 2013.

[21] I.M. Sobol'. On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Computational Mathematics and Mathematical Physics*, 7(4):86 – 112, 1967.

[22] Paul Bratley and Bennett L. Fox. Algorithm 659: Implementing sobol's quasirandom sequence generator. *ACM Trans. Math. Softw.*, 14(1):88–100, March 1988.

[23] M.A. Bessa, R. Bostanabad, Z. Liu, A. Hu, Daniel W. Apley, C. Brinson, W. Chen, and Wing Kam Liu. A framework for data-driven analysis of materials under uncertainty: Countering the curse of dimensionality. *Computer Methods in Applied Mechanics and Engineering*, 320:633 – 667, 2017.

[24] J. Cottrell, T. J. R. Hughes, and Y. Bazilevs. *Isogeometric Analysis: Toward Integration of CAD and FEA.* Wiley, Chichester, 2009.

[25] S. Rudraraju, A. Van der Ven, and K. Garikipati. Mechano-chemical spinodal decomposition: A phenomenological theory of phase transformations in multi-component crystalline solids. *Nature Computational Materials*, 2:16012, 2016.

[26] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In Yee Whye Teh and Mike Titterington, editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 249–256, Chia Laguna Resort, Sardinia, Italy, 13–15 May 2010. PMLR.

[27] Hao Zheng, Zhanlei Yang, Wenju Liu, Jizhong Liang, and Yanpeng Li. Improving deep neural networks using softplus units. In *2015 International Joint Conference on Neural Networks (IJCNN)*, pages 1–4, July 2015.

[28] L. Dalcin, N. Collier, P. Vignal, A.M.A. Côrtes, and V.M. Calo. Petiga: A framework for high-performance isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 308:151–181, 2016.

[29] Satish Balay, Shrirang Abhyankar, Mark F. Adams, Jed Brown, Peter Brune, Kris Buschelman, Lisandro Dalcin, Alp Dener, Victor Eijkhout, William D. Gropp, Dinesh Kaushik, Matthew G. Knepley, Dave A. May, Lois Curfman McInnes, Richard Tran Mills, Todd Munson, Karl Rupp, Patrick Sanan, Barry F. Smith, Stefano Zampini, Hong Zhang, and Hong Zhang. PETSc Web page. `http://www.mcs.anl.gov/petsc`, 2018.

[30] Satish Balay, Shrirang Abhyankar, Mark F. Adams, Jed Brown, Peter Brune, Kris Buschelman, Lisandro Dalcin, Alp Dener, Victor Eijkhout, William D. Gropp, Dinesh Kaushik, Matthew G. Knepley, Dave A. May, Lois Curfman McInnes, Richard Tran Mills, Todd Munson, Karl Rupp, Patrick Sanan, Barry F. Smith, Stefano Zampini, Hong Zhang, and Hong Zhang. PETSc users manual. Technical Report ANL-95/11 - Revision 3.10, Argonne National Laboratory, 2018.

[31] Satish Balay, William D. Gropp, Lois Curfman McInnes, and Barry F. Smith. Efficient management of parallelism in object oriented numerical software libraries. In E. Arge, A. M. Bruaset, and H. P. Langtangen, editors, *Modern Software Tools in Scientific Computing*, pages 163–202. Birkhäuser Press, 1997.

[32] J. Towns, T. Cockerill, M. Dahan, I. Foster, K. Gaither, A. Grimshaw, V. Hazlewood, S. Lathrop, D. Lifka, G. D. Peterson, R. Roskies, J. R. Scott, and N. Wilkins-Diehr. Xsede: Accelerating scientific discovery. *Computing in Science & Engineering*, 16(5):62–74, Sept.-Oct. 2014.