Pacific Northwest
NATIONAL LABORATORY

PNNL-35126

Visualizing Uranium Crystallization from Melt: Experiment-Informed Phase Field Modeling and Machine Learning

September 2023

Matthew Athon
Nellie Ciesielski
Jordan Corbey
Shenyang Hu
Ethan King
Yulan Li
Jacqueline Royer
Panos Stinis
Robert G. Surbella III
Scott Swenson

U.S. DEPARTMENT OF ENERGY

Prepared for the U.S. Department of Energy
under Contract DE-AC05-76RL01830

DISCLAIMER

PACIFIC NORTHWEST NATIONAL LABORATORY
operated by
BATTELLE
for the
UNITED STATES DEPARTMENT OF ENERGY
under Contract DE-AC05-76RL01830

Printed in the United States of America

Available to DOE and DOE contractors from
the Office of Scientific and Technical Information,
P.O. Box 62, Oak Ridge, TN 37831-0062
www.osti.gov
ph: (865) 576-8401
fox: (865) 576-5728
email: reports@osti.gov

Available to the public from the National Technical Information Service
5301 Shawnee Rd., Alexandria, VA 22312
ph: (800) 553-NTIS (6847)
or (703) 605-6000
email: info@ntis.gov
Online ordering: http://www.ntis.gov

Visualizing Uranium Crystallization from Melt: Experiment-Informed Phase Field Modeling and Machine Learning

September 2023

Matthew Athon
Nellie Ciesielski
Jordan Corbey
Shenyang Hu
Ethan King
Yulan Li
Jacqueline Royer
Panos Stinis
Robert G. Surbella III
Scott Swenson

Prepared for
the U.S. Department of Energy
under Contract DE-AC05-76RL01830

Pacific Northwest National Laboratory
Richland, Washington 99354

Abstract

The focus of this project was to observe and simulate the solidification of uranium metal at the crystallographic level from its molten state. Melting experiments were conducted at two different scales to observe microstructural evolution using either a laboratory-scale induction furnace (hundreds of grams of metal) or a microscope heating stage (hundreds of milligrams of metal), respectively. Experimental parameters and characterization data were then used to inform a phase field model of gamma-U crystal growth as dendrites with or without secondary phase impurities in the form of uranium carbide particles. Finally, training datasets were generated by the phase field model as inputs to a neural network, developed with the aim of providing a faster, cheaper surrogate model for microstructural simulations within a given parameter space. Progress is reported herein for each of these task areas. Ultimately, 1) an optical microscope heating stage capability has been stood-up for uranium metal solidification studies, 2) a phase field model was advanced to simulate multiple uranium grains growing in the presence of carbide impurity particles and 3) a neural network was constructed and optimized to predict the microstructure features of individually growing uranium crystals.

Summary

Previous studies from our group describe the movement of carbide impurity particles in molten uranium during the casting process, using computational fluid modeling.[1, 2] Building on that work, the project reported here is interested in solidification of molten uranium, ultimately in the presence of impurities. To understand and simulate the crystallization process of gamma-U from melt, three interdependent tasks were undertaken:

1) Experimental characterization of uranium metal

   To experimentally characterize the microstructure of solidifying uranium (U), two types of melting experiments were conducted: 1) a 434 g block of previously characterized U metal was melted in an induction furnace and resolidified then analyzed by optical microscopy, scanning electron microscopy (SEM) and electron backscatter diffraction (EBSD) to understand the evolved grain shapes and sizes. 2) 100-200 mg pieces of copper (Cu) metal surrogate were melted individually using an optical microscope heating stage to observe crystal growth in real time and optimize parameters needed for U metal samples.

2) Training dataset development via phase field modeling

   Thermodynamic and kinetic properties of U, along with experimental melting parameters and microstructural characterization data, were used to inform a phase field model of U solidification under cooling. Within the time and budget of this project, the phase field model was advanced to include the evolution of multiple U grains with different orientations in a given region (areas representing tens of thousands of um^2) to consider the effect of existing uranium monocarbide impurity particles on U grain growth in a one-way coupling mode. Simulation results from varying the carbide particle distributions and cooling rates are presented herein. In addition, data sets for single U grain growth, in terms of phase field model parameters (or thermodynamic and kinetic properties of the system), were generated to train a neural network surrogate and eventually allow for faster, cheaper screening of input parameters and solution of phase-field equations, as described below, to support the more expensive, though higher fidelity phase field modeling capability.

3) Neural network construction and optimization

   A deep operator network (DeepONet) was adapted and optimized to explore the feasibility of using a machine learning (ML) surrogate to support faster, cheaper screening of input parameters to help guide the more expensive, higher fidelity phase field model toward targeted computations predicting evolving microstructure. Within the time and budget of this project, relative error in the network outputs, using phase field model results as ground truth, was minimized by implementing data augmentation, a feature expansion trunk, self-adaptive weights, a convolutional neural network for data compression, and parallelized computing. Machine learning outputs were also evaluated using relevant physics.

Acknowledgments

This research was supported by the National Security Mission Seed, under the Laboratory Directed Research and Development (LDRD) Program at Pacific Northwest National Laboratory (PNNL). The computational work was performed using PNNL Computing at Pacific Northwest National Laboratory. PNNL is a multi-program national laboratory operated for the U.S. Department of Energy (DOE) by Battelle Memorial Institute under Contract No. DE-AC05-76RL01830.

Acronyms and Abbreviations

Ar	Argon
CNN	Convolutional neural network
Cu	Copper
DeepONet	Deep operator network
EBSD	Electron back scatter diffraction
EDS	Energy dispersive spectroscopy
LDRD	Laboratory directed research and development
LSCM	Laser scanning confocal microscope/microscopy
ML	Machine learning
MSE	Mean squared error
O	Oxygen
PFM	Phase field model(ing)
PNNL	Pacific Northwest National Laboratory
SEM	Scanning electron microscopy
U	Uranium
VIM	Vacuum induction melting/melter
Zr	Zirconium

Contents

Figures

Tables

1.0 Project Rationale

Actinide metal processing involves multiple steps to arrive at the desired microstructure and purity of a final product. Arguably, the microstructure and purity of the solidified material from the final casting step in this process can impact the level of post-solidification processing effort that will be needed to meet required microstructural specifications in the final part. While mature technologies are implemented at industrial scales to optimize this process, fundamental scientific understanding of actinide crystallization from melts and kinetic interactions of the molten metal matrix with impurities is still lacking, especially as casting methods are modernized.[3-5] This project team initially set out to characterize and model the properties of carbon impurity in a given uranium (U) metal material undergoing solidification from its molten phase, as a first use case. The investigation of this nominally two-phase system inherently involves an understanding of the ideal single-phase system containing pure U, which was also addressed in this study.

As molten U cools and solidifies, it passes through three crystallographic phases: γ-U at 1135 °C, β-U at 775 °C and finally α-U after 668 °C,[6-8] implying that room temperature characterization of U grains may not provide representative data regarding their precipitated state. Historically, C has been one of the most common and difficult impurities to control in U metal, being linked to heterogeneity and reproducibility issues along with crack and void formation during processing of U metal and its alloys.[9-15] At equilibrium, accepted uranium (U) – carbon (C) phase diagrams[6, 8, 16, 17] suggest that C, present as an impurity between 1 and 70 at%, precipitates as insoluble uranium carbides before or concurrently with solidification of the U metal matrix. Therefore, fundamentally understanding the interplay between carbide impurity particles and growing U crystals is desirable to better control the evolving microstructure of the system and provide confidence regarding the behavior of impurity inclusions.

Direct chemical and crystallographic characterization of molten actinide systems is limited, due to their high temperatures, pyrophoricity and radioactivity. To provide a window of insight into this otherwise inaccessible step in U metal processing, this project team has pursued the construction of computational models of solidifying U metal at the microscale that include experimental validation where possible. The modeling method of choice is a phase field approach,[18-22] which has been useful for simulating crystal growth in other systems. However, the phase field computations require solution of partial differential equations that are computationally expensive, resulting in small physical domains that can be simulated in a reasonable time frame. For this reason, among others, we also explore the use of a machine learning neural networks to aid in the challenge areas that phase field modeling encounters during the simulation of metal grain growth from melt.

2.0 Experimental Characterization

2.1 Background

To our knowledge, precipitation and growth of discrete actinide metal crystals from melt have never been observed or imaged. Outside of radioactive systems, work, predominantly driven by the steel industry, has been done to optimize the use of high-temperature laser scanning confocal microscopy (LSCM) for the purposes of observing and controlling phase transformations in various materials, concurrent with impurity inclusion behavior.[23-25] This encouraged our team to explore high-temperature microscopy options for observing actinide solidification for fundamental studies and to support validation of developing phase field models of growing metal crystals. Additionally, our group at Pacific Northwest National Laboratory (PNNL) has research-scale actinide casting capabilities and electron microscopy tools that have been previously described [4, 5, 26, 27] and were brought to bear on this study.

2.2 Methods

2.2.1 Vacuum Induction Melting

Initially, this project team used existing laboratory-scale actinide casting capabilities[4, 26, 27] to study the change in microstructure for a known U metal sample at a given cooling rate after melting via a vacuum induction furnace. The pre-melted depleted U was purchased from Manufacturing Sciences Corporation with nominally 70 ppm C contaminant, as indicated by the manufacturer and confirmed by combustion analysis for C of six subsections ranging from 470 to 610 mg in mass and 12 to 65 ppm, ± 5% of the recorded value, of C using a LECO® C230 combustion analyzer. Electron back-scatter diffraction (EBSD) analysis was performed on a FEI Helios focused-ion beam scanning electron microscope (FIB/SEM) using an EDAX Hikari Super EBSD Camera. In-situ FIB polishing was employed to remove the oxide layer formed during samples transfer in air. EBSD maps were collected using a 1.5 μm step size (Figure 1), unless otherwise stated.

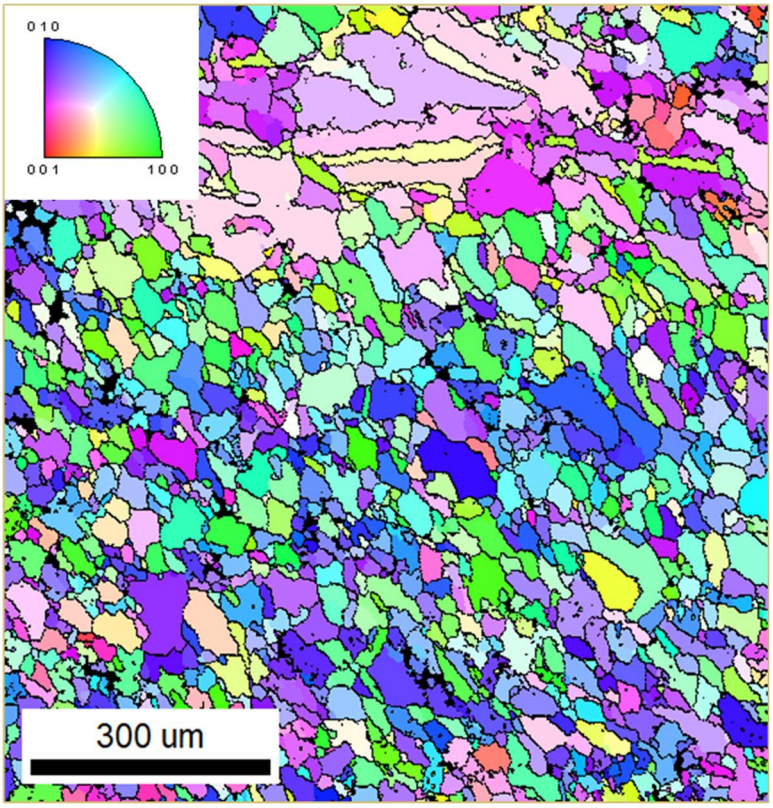


Figure 1. Representative orientation map of pre-melted α-U used in this study as received.

A 434 g block of this U was etched for 10 min in 8 M nitric acid before rinsing with deionized water followed by ethanol and towel drying (Figure 2), as previously described.[4] The U charge was then placed in a cylindrical magnesia-stabilized zirconia crucible (Zircoa, Inc.), which had been coated three times with yttria aerosol paint (ZYP Coatings, Inc.) prior to use, as previously described.[4] The loaded crucible was nested inside another larger zirconia crucible inside a quartz sleeve, and a jacketed thermocouple was inserted next to the U charge (Figure 2). The entire configuration was loaded into the coil of an Indutherm vacuum tilting casting machine (VTC 200 V Ti) operating at 15 kHz, and the crucible chamber was evacuated then cycled through three argon/vacuum washes before being placed under a dynamic high vacuum atmosphere.

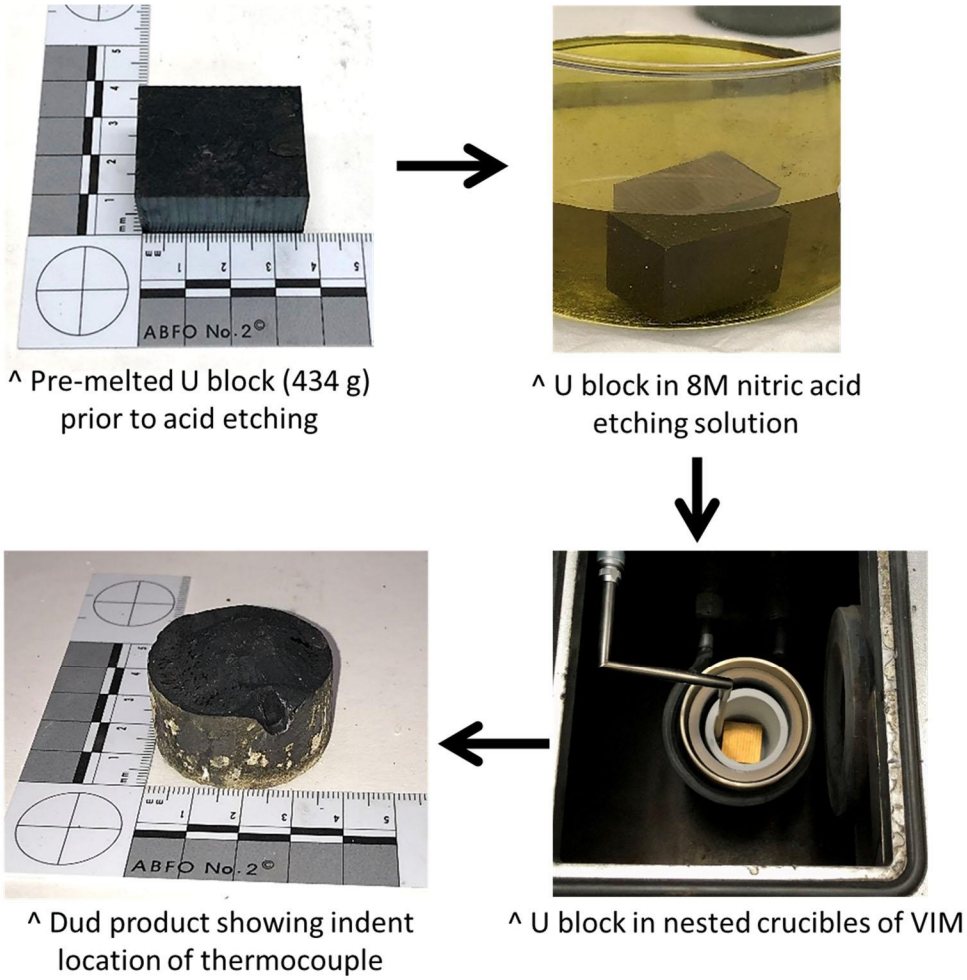


Figure 2. Photographic sequence of steps to prepare U metal for vacuum induction melting (VIM) and the resulting solidified ingot (bottom, left).

As previously described,[4] routine operations were followed to set the initial furnace power to 3 kW (time, t = 0 min). At t = 1:07, the power was increased to 6 kW. Within 3 min, the thermocouple read 1200 °C, and the U was molten. After 5 min, the charge had reached 1400 °C, and that temperature was held for 12 min, before the furnace power was turned off and the system was left to cool. After 1.7 h of cooling, the thermocouple read 100 °C, and the metal was left to cool overnight. The resulting ingot after melting the charge and cooling in-place can be seen in the bottom left of Figure 2.

The resulting ingot after in-place solidification was first sectioned using a Buehler Abrasimatic 300 abrasive cutter to obtain a vertical center strip from the cylindrical ingot, which was further sectioned using a Buehler IsoMet™ Low Speed Precision Cutter with a diamond-coated wafering blade to obtain two halves of the center strip for optical and electron microscopy along with subsections for combustion analysis, as described below. One half of the cut center strip was potted in epoxy and polished for microscopy, as previously described.[4] Optical microscopy (Figure 5) was carried out using a Keyence laser scanning confocal microscope (LSCM) model VK-X1050. SEM was performed using the FEI Helios focused-ion beam scanning electron microscope. Preliminary EBSD data was obtained for the post-melt ingot as described above at 100x or 350x magnification, respectively, with a 4 μm or 2 μm step size, respectively.

2.2.2 High-Temperature Microscopy

To validate our developing computational models and confirm the morphology of growing U crystals from melt, it was determined that high-temperature microscopy should be prioritized. While high-temperature SEM is desirable, it was unclear if it could have been achieved in a useful way for near-molten U on the timescale and budget of this project. Therefore, high-temperature optical spectroscopy was explored using a chiller-cooled HS1400V heating stage capable of achieving temperatures as high as 1400 °C with gas atmosphere control (Figure 3) and an EXAMET-5 series microscope from UNITRON®. A Jenoptik ProgRes GRYPHAX Arktur 8 mp digital camera and an Olympus SLMPLN long working-distance 20x objective were procured for this effort. Copper (Cu) metal was initially used as a surrogate for U, to test the heating stage and the video/image quality. The heating stage possesses a 16 mm diameter ceramic sample well (Figure 4), and several Cu samples between 100 and 200 mg were melted and resolidified under similar conditions to test the microscopy setup and optimize parameters for video capture of growing metal crystals. A small piece of zirconium (Zr, melting point = 1854 °C) was added to the sample holder for each run using a ceramic divider to segregate it from the molten Cu and react with potential residual oxygen (O) that may be present in the nominally inert sample environment.

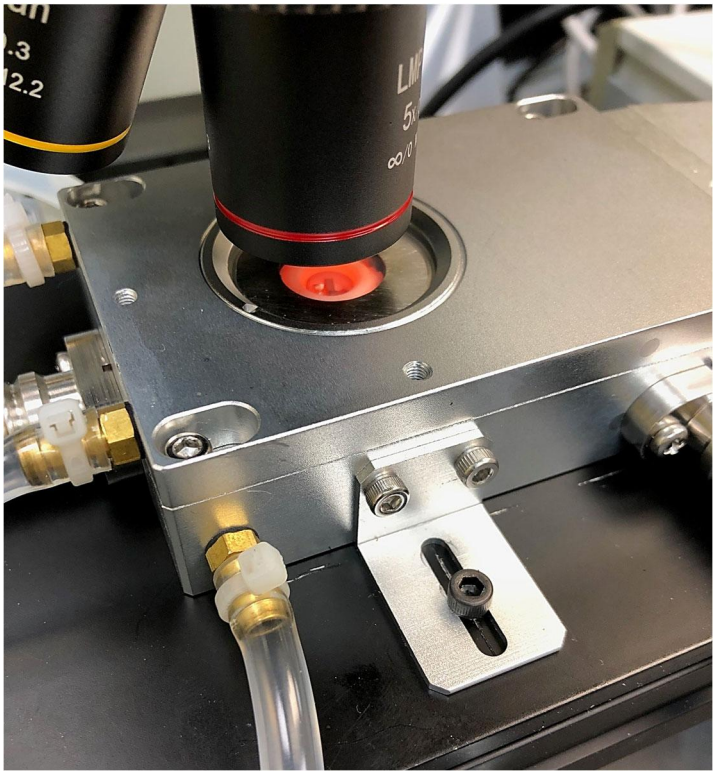


Figure 3. Heating stage used for optical microscopy experiments with molten metal.

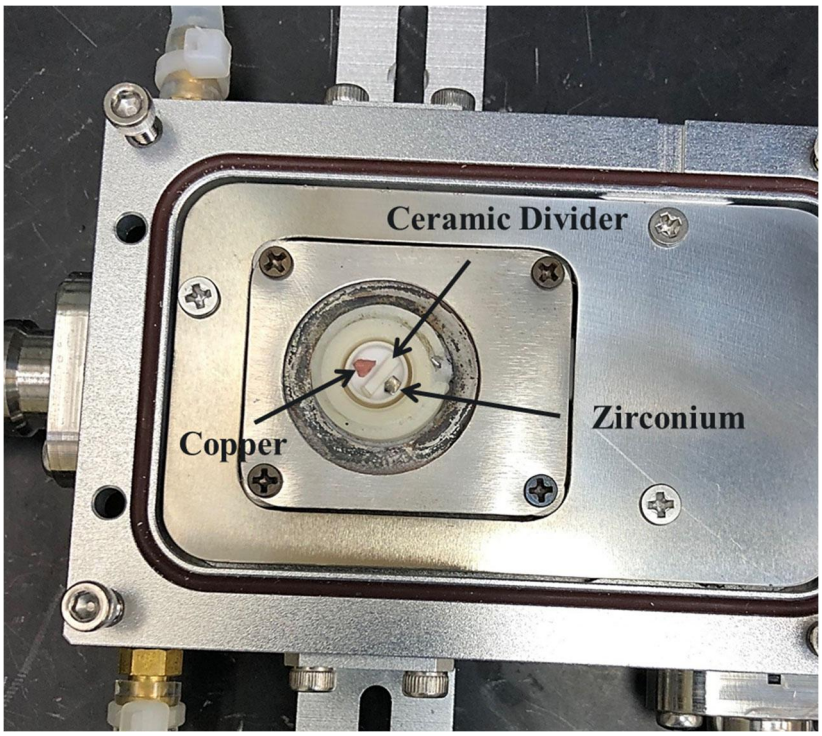


Figure 4. Photograph showing the opened heating stage and sample well containing 153 mg piece of pure Cu metal, a ceramic divider and a piece of Zr metal to act as an O getter.

For high-temperature microscopy experiments with molten Cu, the pre-melted Cu, fresh Zr and the sample crucible were baked in an oven at 100 °C overnight to remove surface water. A new quartz window was installed prior to sealing the sample configuration (Figure 4). Three vacuum and argon (Ar) wash cycles were conducted on the sample atmosphere using rough pumping and ultra-high purity Ar. The size and shape of the Cu charge dictated its contact with the heating element of the stage and the power supply set temperature at which it melted in the sample holder (ideal Cu melting point = 1084 °C). Each Cu sample melted at varying recorded temperatures, as described below in Table 1.

2.3 Results and Discussion

2.3.1 Post-melt U Ingot

As described above, the post-melt U ingot (~400 g) obtained by VIM was analyzed by LSCM to compare the relative grain structures before and after melting. A center strip was taken from the cylindrical ingot to capture the location of the thermocouple in the melt (Figure 5). The right edge of the left image in Figure 5 represents the vertical center line of the ingot, where a void can be seen at the bottom of the ingot. Pale sections near the top of the sample in the left image containing darker rhombic sections within are areas where ion polishing and imaging was conducting in the focused ion beam for SEM prior to LSCM. As expected, higher magnification of the post-melt ingot center strip (Figure 5, right) shows large irregular grains, when compared to the more refined grain structure of the pre-melt material (Figure 1), and commonly observed subgrain twinning attributed to crystallographic phase transitions in U metal as it cools.[3, 7]

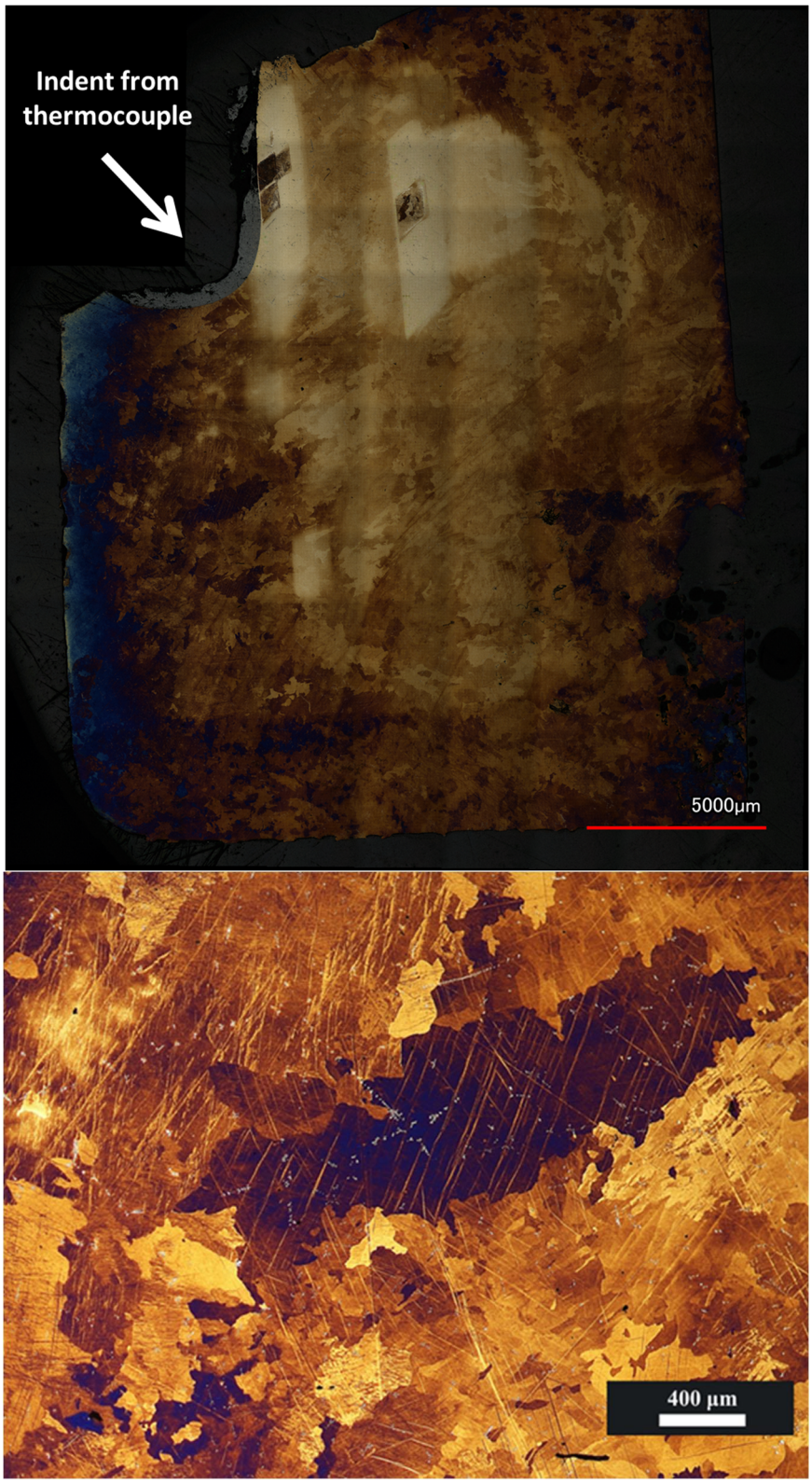


Figure 5. LSCM images of one half of the center strip sectioned from the post-melt U ingot melted by VIM with top-to-bottom montaging (top) or zoomed-in (bottom).

Carbon combustion analysis on six sub-samples of this post-melt ingot each between 300 and 800 mg in mass from areas near the top, middle and bottom of the vertical center line, respectively, gave consistent values between 187 and 228 ppm (± 5% of the recorded value) C, in no particular order. Examples of inclusions in the post-melt ingot obtained by SEM are shown in Figure 6 and are likely carbide-containing, as has been previously described.[4, 28] Additionally, some energy dispersive spectroscopy (EDS) (Figure 8) and EBSD (Figure 7) were performed near the area surrounding the thermocouple to show oxygen-rich columnar grains, similar to those previously published in relation to void areas in U.[4]

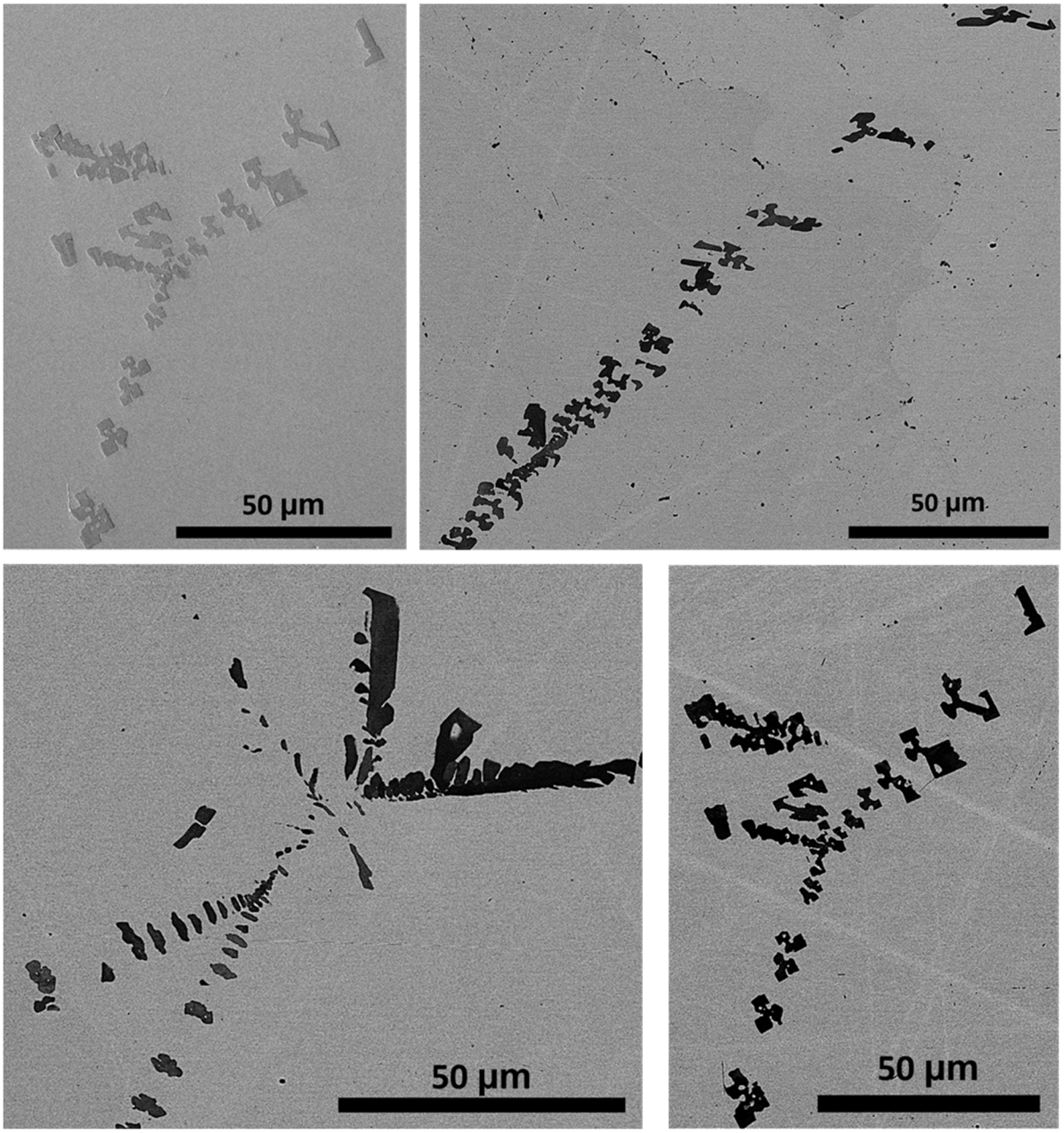


Figure 6. Secondary electron (top, left) and concentric backscatter images of examples of inclusions found in the post-melt U ingot by SEM.

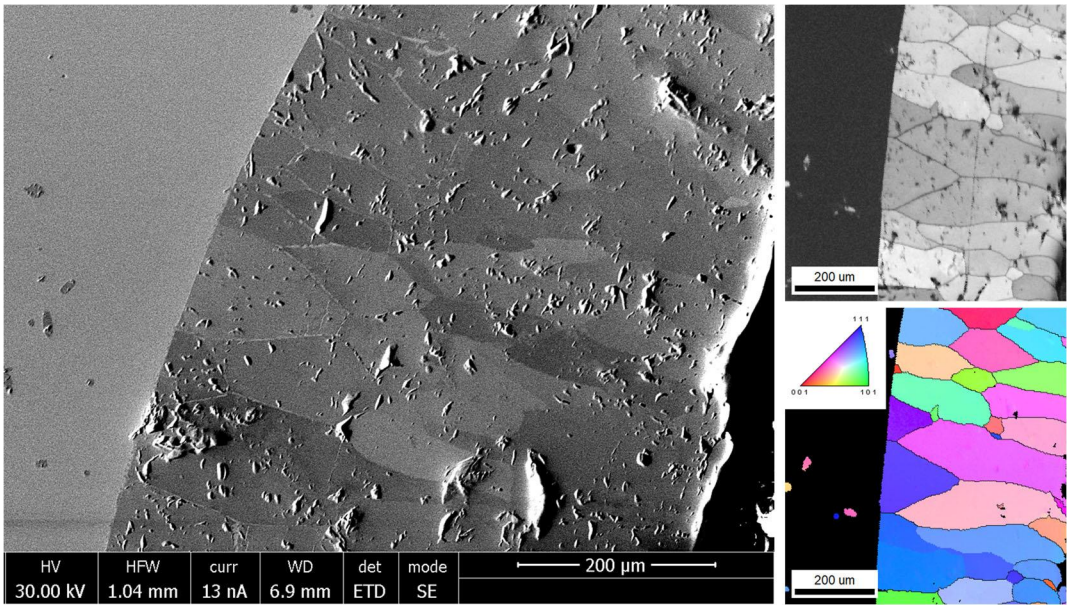


Figure 7. Secondary electron image (left) of columnar grains along the void where the thermocouple interfaced with the solidifying U metal and (right) an EBSD orientation map of that same region.

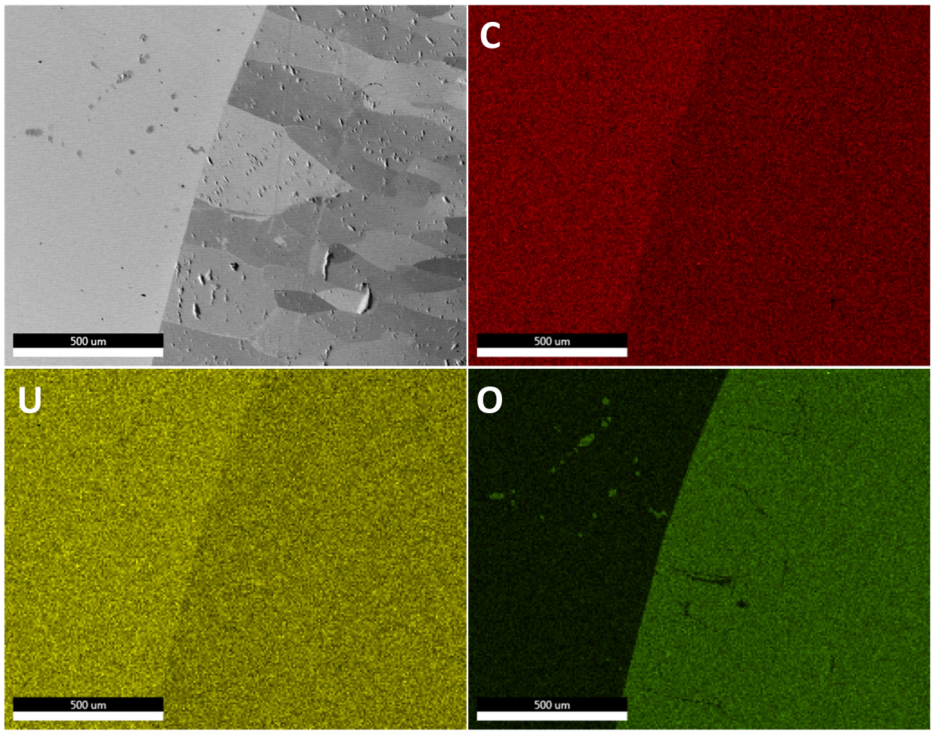


Figure 8. SEM image (top, left) of columnar grains surrounding the thermocouple interface area and associated elemental maps by EDS for C (red, top, right), U (yellow, bottom, left) and O (green, bottom, right). All scale bars represent 500 µm.

Historically, EBSD analysis of α-U has been particularly challenging, due in part to the typically large, irregular grains with prolific subgrain twinning in the initially solidified metal, which rapidly oxidizes. EBSD of the post-melt ingot from VIM was attempted, and it was found that obtaining quality data for this sample was more difficult than the pre-melted material with finer and more uniform strain-free grains. By EBSD, equivalent diameters measured for about 1000 grains from 3 different locations representing roughly 1.7 mm^2 in the pre-melt material equated to between 13 and 19 μm. Preliminary EBSD data for two locations in the post-melt ingot are shown in Figure 9 and Figure 10. Some areas of the maps could not be well-indexed, which is attributed to a large irregular grain structure, along with a high-volume of low-angle boundaries and rapid oxidation of the metal surface, as alluded to above. However, MapSweeper software has been procured by our team from Oxford Instruments for follow-on efforts in this area and has been shown to enhance EBSD data for rapidly oxidizing surfaces with low-angle boundaries, as in this case of α-U.

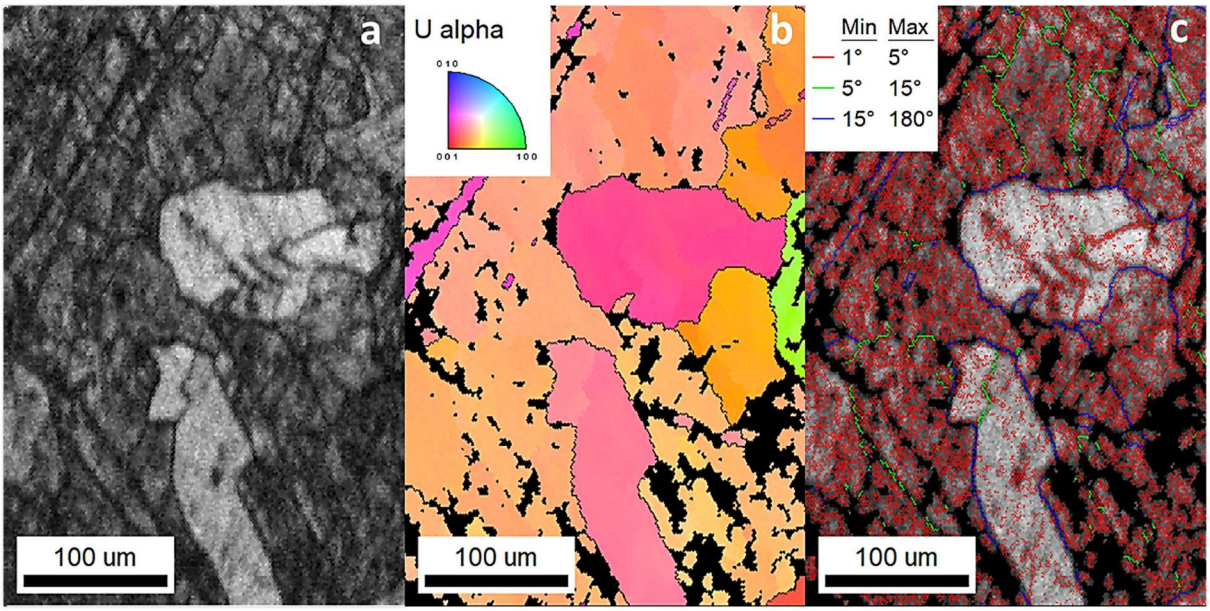


Figure 9. Preliminary maps representing EBSD image quality (a), grain orientation (b), and rotation angle (c) of α-U in the post-melt ingot (orthorhombic). The high area fraction of low-angle boundaries in c is indicative of a strained microstructure and/or poor surface polish.

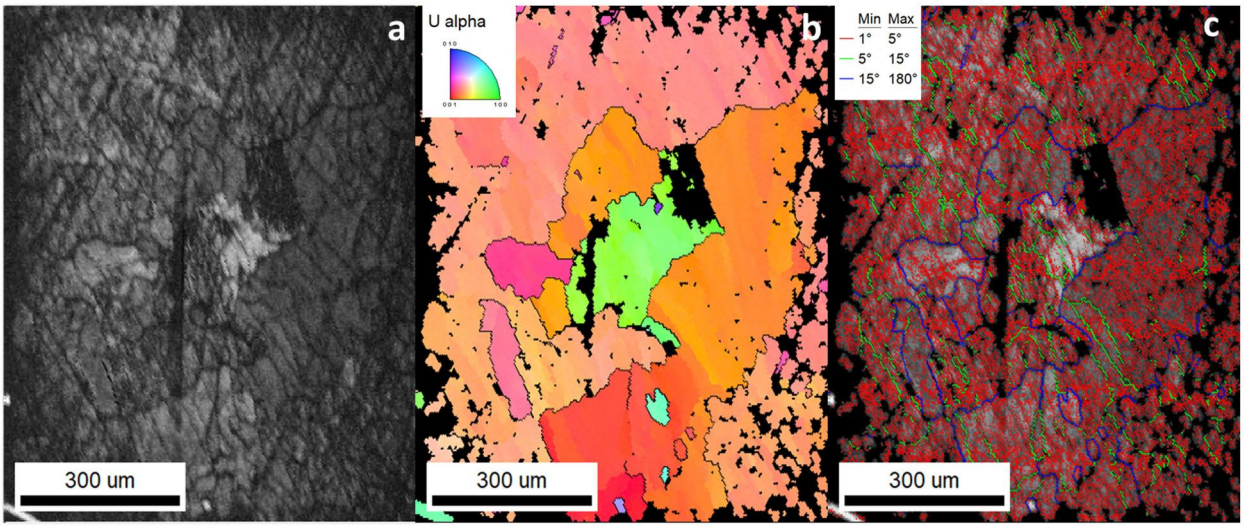


Figure 10. Preliminary maps representing EBSD image quality (a), grain orientation (b), and rotation angle (c) of α-U in the post-melt ingot (orthorhombic). The high area fraction of low-angle boundaries in c is indicative of a strained microstructure and/or poor surface polish.

2.3.2 High-Temperature Microscopy

High-temperature microscopy experiments with molten U will be conducted in the coming year under a different project. The current project focused on optimizing the parameters of the procured microscopy equipment using Cu as a surrogate for U. Six Cu melting/solidification experiments were conducted within the lifetime of this project, to familiarize the project team with imaging capability and optimized microscope/camera settings. Cu charges ranged from 100 to 200 mg. The size of the Cu charge seemed to effect the recorded melting temperature. The ideal melting temperature of Cu is 1084 °C. As detailed in Table 1, recorded temperatures at which Cu charges melted using the heating stage described above ranged between 1150 and 1205 °C, with ramp rates from room temperature between 15 and 100 °C/min. Recorded solidification temperatures ranged from 1095 to 1150 °C, with ramp rates from the recorded melting temperatures between 1 and 20 °C/min, except in one case, when the power supply was simply shut off, and the molten bead solidified instantly (Figure 12). A representative photograph of a post-melt Cu bead and an expended Zr piece can be seen in Figure 11. More work is needed to optimize user control of the heating environment for each metal sample, with an emphasis on the liquid to solid phase transition.

Table 1. Recorded melting/solidification temperatures and parameters for Cu surrogate melting experiments with Cu charges between 100 and 200 mg. The ideal melting/solidification temperature of Cu is 1084 °C.

Experiment Number	Ramp Rate to Melt (°C/min)	Recorded Melt Temperature (°C)	Ramp Rate to Solidify (°C/min)	Recorded Solidification Temperature (°C)
1	15	1160	20 from 1167 °C	1095
2	100	1150	10 from 1150 °C	1150
3	100	1150	20 from 1150 °C	1123
4	100	1170	20 from 1170 °C	1130
5	20	1205	1 from 1200 °C	1112
6	100	1187	Shut off heat at 1187 °C	~1150

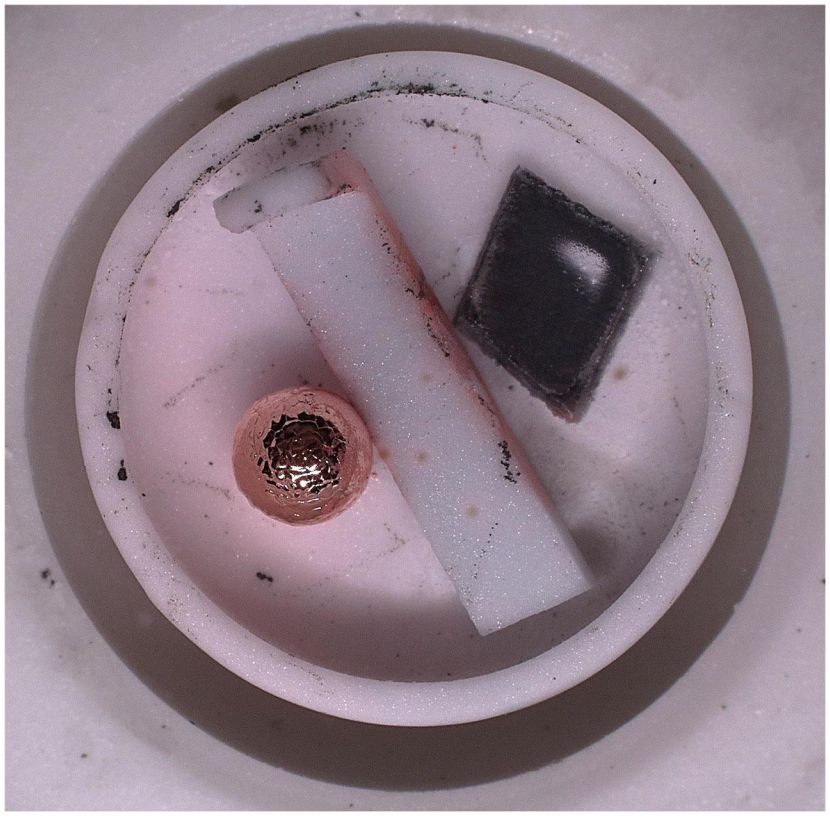


Figure 11. Representative photograph of a post-melt Cu bead (left) and an expended Zr piece with surface oxidation (right) separated by a ceramic divider in the heating stage sample well.

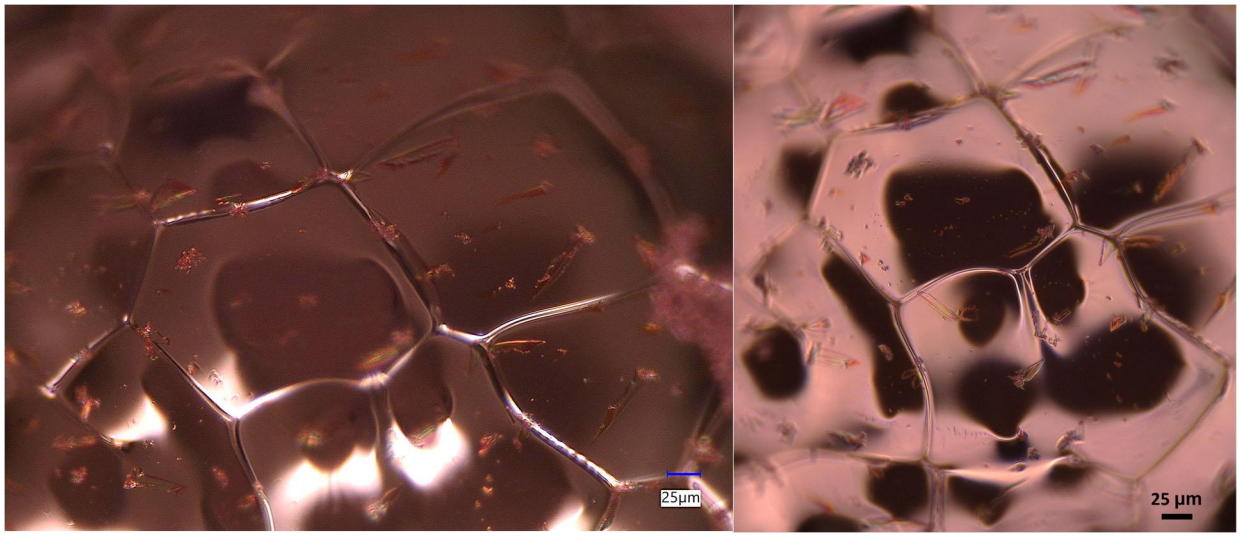


Figure 12. 10x magnified images, captured under different lighting, of the same shiny surface of a solidified Cu bead from experiment number 6.

Figure 12 above shows representative micrographs of the shiny, segmented surface of a solidified Cu bead observed in the heating stage. In this case, the power supply was simply shut off once the total recorded melt temperature of 1187 °C was achieved. However, a similar turtle shell grain structure could also be observed under conditions from Experiments 1-4. Metallurgical analysis and EBSD of these Cu samples has not yet been completed to confirm the grain boundaries; this will be pursued in the coming year. In one case, when the ramp rate to solidify was 1 °C/min, the entire surface of the solidified bead was smooth with no observed segmenting, making it difficult to know when the solidification process started and ended; the movement or arrest of minute likely oxide islands were used as indicators. Several videos of solidifying Cu, mostly at 10x and 20x magnification, have been archived to PNNL Records Management and may be viewed upon request to the authors of this report or PNNL Records Management. The ultimate hope for this high-temperature microscopy capability it to create a controlled environment that can mimic aspects of casting systems of interest, to observe morphological evolution of U crystals from melt to validate computational phase field models, which will be pursued in the coming years.

3.0 Phase Field Modeling

3.1 Background

Phase field modeling has been extensively used to simulate crystal growth [18-21] and, to a limited extent, in uranium metal.[29] However, in casting systems, it is still challenging to quantitatively predict the effects of operation conditions on microstructures in a product. One reason is because microstructure evolution is sensitive to thermodynamic and kinetic properties of the materials and the casting environment, which are often provided to computational models as measured values with large uncertainties, especially for actinide systems. Moreover, solving the governing partial differential equations (PDEs) of microstructure evolution is a computationally expensive process. Consequently, the simulations can only reach a relatively small physical domain in a reasonable timeframe. It is believed that machine learning (ML) may be able to aid in these challenge areas, as described in Section 4.0.

Despite the challenges mentioned above, phase field modeling is still the method of choice to computationally predict crystal growth in a two-phase system. Here, a quantitative phase field model of U solidification was advanced to systematically simulate the effect of thermodynamic and kinetics properties on U crystal dendrite growth. Dendrites have been shown to create microstructures similar to the irregular grains observed in α-U [30] and could exist in the melt environments of our laboratory systems. One focus of the high-temperature microscopy efforts described above will be to try and confirm this. In this section, the words “crystal” and “dendrite” are used interchangeable.

Melting parameters in the Experimental Characterization section above were taken into account in the advancement of this phase field model, however, practical simulation cooling rates in the model were typically much faster than those facilitated in the laboratory system, to accomplish routine phase field modeling computations within a reasonable timeframe. Previously published experimental observations of uranium carbide particle spatial distributions in the model U casting system [4] also informed some of the simulations described below of the effect of these impurity particles and cooling rates on grain structures, assuming that the particles are uranium monocarbide that form first in the U melt. Finally, simulated single-grain microstructure evolution data were also used to train a developing physics-informed ML approach, described later in this report.

3.2 Methods

3.2.1 Description of the Phase Field Model for Solidification

The phase field method is used to simulate the growth and evolution of an initial dendritic nucleus during solidification. An order parameter ϕ is used to describe the material phase, which takes a value of 0 in the solid phase and 1 in the liquid phase. In addition, the nondimensionalized temperature is denoted by u [18],

$$u = \frac{T - T_m}{T_m - T_\infty}, \tag{1}$$

where T is the local temperature. T_m is the melting temperature of the material. T_∞ is the environment temperature where solidification started. The free energy of the system is expressed as [20, 31]

$$\mathcal{F} = \int_V f_0 \left[\frac{1}{2}[W(\mathbf{n})]^2 |\nabla\phi|^2 + f_{chem}(\phi, u)\right] dV, \tag{2}$$

where f_0 is the energy density coefficient with units of J/m^3, and V is the volume of the simulated material. $[W(\mathbf{n})]^2$ is the gradient energy coefficient, $\boldsymbol{n} = \frac{\nabla\phi}{|\nabla\phi|}$ is the normal direction to the interface, and f_{chem} is the chemical free energy density and can be expressed with a double-well potential of a polynomial function as

$$f_{chem}(\phi, u) = \frac{1}{4}\phi^2(1-\phi)^2 + pu\phi^3\left(-\frac{1}{3} + \frac{1}{2}\phi - \frac{1}{5}\phi^2\right), \tag{3}$$

where $p = 30\epsilon\alpha\Omega$ and is a dimensionless coupling constant between the order parameter ϕ and the temperature u. $\Omega = \frac{T_m - T_\infty}{L/c_p}$, $\epsilon = \frac{\delta}{\omega}$, and $\alpha = \sqrt{2}\omega S_m L/\left(12 c_p \sigma_0\right)$ are all dimensionless quantities, while $S_m = L/T_m$, L, c_p, and δ stand for the melting entropy per unit volume [unit: J/(m^{3}K)], enthalpy/latent heat per unit volume [J/m^3], the specific heat per unit volume [J/(m^3·K)], and the interface thickness [m], respectively.[32] ω is a reference length. σ_0 is associated with the material interface energy [J/m^2]. The interface thickness and directional anisotropy are controlled by $W(\mathbf{n})$. In two dimensions,

$$W(\boldsymbol{n}) = \delta\tilde{\sigma}(\boldsymbol{n}),\ \tilde{\sigma}(\boldsymbol{n}) = \tilde{\sigma}(\theta) = 1 + \sigma_m \cos[m(\theta - \theta_0)],\ \tan(\theta) = \frac{n_y}{n_x} = \frac{\frac{\partial\phi}{\partial y}}{\frac{\partial\phi}{\partial x}}, \tag{4}$$

where m is a non-negative integer, and θ is the in-plane azimuthal angle. σ_m determines the anisotropy strength, and θ_0 is an offset azimuthal angle that specifies the misorientation of the crystalline lattice relative to the laboratory frame of reference (in this case, the x-axis in the simulation coordinate system (x,y)). The interfacial energy density is described by

$$\sigma(\boldsymbol{n}) = \sigma_0(1 + \sigma_m \cos[m(\theta - \theta_0)]) = \sigma_0\tilde{\sigma}(\theta). \tag{5}$$

The evolution of ϕ is governed by the Allen-Cahn equation

$$\tau(\boldsymbol{n})\frac{\partial\phi}{\partial t} = -\frac{\delta\mathcal{F}}{\delta\phi}, \tag{6}$$

where $\tau(\boldsymbol{n})$ is associated with interface mobility. The functional derivative in eq. (6) is given as [18, 33]

$$\begin{aligned}\epsilon^2\tau_0\tilde{\beta}\tilde{\sigma}\frac{\partial\phi}{\partial t} = {} & \phi(1-\phi)\left\{\left(\phi - \frac{1}{2}\right) + 30\epsilon\alpha\Omega u\phi(1-\phi)\right\} \\ & -\epsilon^2\frac{\partial}{\partial x}\left[\tilde{\sigma}\tilde{\sigma}'\frac{\partial\phi}{\partial y}\right] + \epsilon^2\frac{\partial}{\partial y}\left[\tilde{\sigma}\tilde{\sigma}'\frac{\partial\phi}{\partial x}\right] + \epsilon^2\nabla[\tilde{\sigma}^2\nabla\phi],\end{aligned} \tag{7}$$

where $\tilde{\beta}(\theta) = 1 + \beta_m \cos[m(\theta - \theta_0)]$, and $\tau_0 = S_m D\,\beta_0/\sigma_0$, with β_0 being a kinetic coefficient with units of [sK/m] and D being the material's thermal diffusivity [m^2/s]. β_m is a parameter determining the kinetic anisotropy strength.

The time evolution of u is governed by thermal diffusion and the release of latent heat at the interface during solidification

$$\frac{\partial u}{\partial t}+\frac{1}{\Omega}30\phi^2(1-\phi)^2\frac{\partial \phi}{\partial t}=\nabla^2 u+B(t), \tag{8}$$

with $B(t)$ being a cooling rate function.

Equations. (7) and (8) are normalized. The time t was normalized by $t_0=\omega^2/D$, and the length was normalized by the reference length ω. They are used to simulate a single dendrite growth evolution.

To simulate the growth of multiple dendrites, a set of order parameters ϕ_i $(i=1,2,\cdots,N)$ are employed with ϕ_i=0 being the dendrite i and ϕ_i=1 being liquid. The evolution equations of (7) and (8) can be rewritten as

$$\begin{aligned}\epsilon^2\tau_0\tilde{\beta}_i\tilde{\sigma}_i\frac{\partial \phi_i}{\partial t}=&\ \phi_i(1-\phi_i)\left\{\left(\phi_i-\frac{1}{2}\right)+30\epsilon\alpha\Omega u\phi_i(1-\phi_i)\right\}\\ &-\epsilon^2\frac{\partial}{\partial x}\left[\tilde{\sigma}_i\tilde{\sigma}_i'\frac{\partial \phi_i}{\partial y}\right]+\epsilon^2\frac{\partial}{\partial y}\left[\tilde{\sigma}_i\tilde{\sigma}_i'\frac{\partial \phi_i}{\partial x}\right]+\epsilon^2\nabla\left[\tilde{\sigma}_i^2\nabla\phi_i\right]\\ &+\lambda_i(1-\phi_i)\sum_{j=1,j\neq i}^{N}\left(1-\phi_j\right)^2,\quad i=1,2,\cdots,N\end{aligned} \tag{9}$$

$$\frac{\partial u}{\partial t}+\frac{30}{\Omega}\sum_{i=1}^{N}\left[\phi_i^2(1-\phi_i)^2\frac{\partial \phi_i}{\partial t}\right]=\nabla^2 u+B(t), \tag{10}$$

where $\tilde{\sigma}_i=1+\frac{1}{2}\sigma_m\cos[m(\theta-\theta_{0i})]$ and $\tilde{\beta}_i=1+\frac{1}{2}\beta_m\cos[m(\theta-\theta_{0i})]$. λ_i is a positive coefficient determining the interaction strength between the i^{th} dendrite and the others.

3.2.2 Thermodynamic and Kinetic Properties of U Solidification

Considering the solidification of U, Table 2 lists the material properties of U used in this study. Parameters specific to dendrite growth are given in Table 3 and were calculated using the material properties given in Table 2 along with simulation grid size.

Table 2. Material properties of U metal used in the solidification models.

Property	Value	units	ref
Melting point, T_m	1405	K	[34]
Density at STP, ρ	19.05	g/cm^3	[34]
Density at melting point, ρ_m	17.3	g/cm^3	[35]
Thermal conductivity, k	27.0	W/(mK)=J/(msK)	[34]
Specific heat, c_ρ	0.12 2.076	J/gK J/(cm^{3}K)	[34] Using $\rho_m = 17.3$ g/cm^3
Thermal diffusivity, D	0.130 1.30×10^{-5}	cm^2/s m^2/s	$D = \frac{k}{\rho_m c_\rho}$
Heat of fusion or latent heat or enthalpy of fusion, L	8.52 0.682	kJ/mol kJ/cm^3	[34]
Molecular weight	235.04	g/mol	
Molecular volume at STP	12.49	cm^3/mol	[36]
Melting entropy, $S_m = L/T_m$	4.854×10^5	J/m^{3}K	
Interfacial energy, $\sigma_{0i} = \sigma_0$	0.86	J/m^2	
Kinetic coefficient, $\beta_{0i} = \beta_0$	2.73	sK/m	

*Some U property values used in the numerical simulations, such as interfacial energy, have literature precedence [29] but have not been directly measured. In these cases, values were chosen that made the simulations reasonable after other measured parameters were set.

Table 3. Simulation parameters for dendrite growth.

Parameter	Value	units
T_∞	1224	K
$T_m - T_\infty$	181	K
$\Omega = c_p(T_m - T_\infty)/L$	0.55	N/A
Δx	0.08	μm
δ	0.08	μm
ω	16	μm
$\epsilon = \delta/\omega$	0.005	N/A
$\Delta x/\omega$	0.005	N/A
$\sigma_m\ (m = 4)$	0.3	N/A
$\beta_m\ (m = 4)$	0.3	N/A
$t_0 = \omega^2/D$	0.0000197	s
$\Delta t/t_0$	10^{-6}	N/A
$\alpha = \sqrt{2}\omega S_m L/(12 c_p \sigma_0)$	350.0	N/A
$\tau_0 = S_m D\, \beta_0/\sigma_0$	20.0	N/A
$B(t) = \dot{Q}_0$ *	-2.0	N/A
θ_0	0, 30, 45, 60	°
$\lambda_i = \lambda$	20.0	N/A

*$B(t) = \dot{Q}_0 = \frac{\partial u}{\partial t} = \frac{t_0 dT}{(T_m - T_\infty)dt}$, cooling rate is calculated with $\frac{dT}{dt} = \dot{Q}_0 \frac{(T_m - T_\infty)}{t_0}$.

3.3 Results and Discussion

3.3.1 Crystal Morphologies versus Model Parameters

This section illustrates how the simulated dendrite morphologies vary with the model parameters chosen. A simulation cell of 512 x 512 uniform grids is used. A single solid phase nucleus (i.e., N=1 in eq. (1)) with a radius of 5 grids is placed in the center of the simulation cell where $\phi = 0 \; and \; u = -1$ when $r = \sqrt{(x-x_0)^2 + (y-y_0)^2}$<5, and (x_0, y_0) is the nucleus center point coordinate. The offset azimuthal angle $\theta_0 = \frac{\pi}{4}$, and the cooling rate $\dot{Q}_0 = 0$ but are stated separately. Figure 13 displays the evolution of dendrite morphology and the corresponding temperature field with time. The model parameters used are listed in Table 3, and the white bar in Figure 13 (a1) shows the simulation size, used elsewhere in this report.

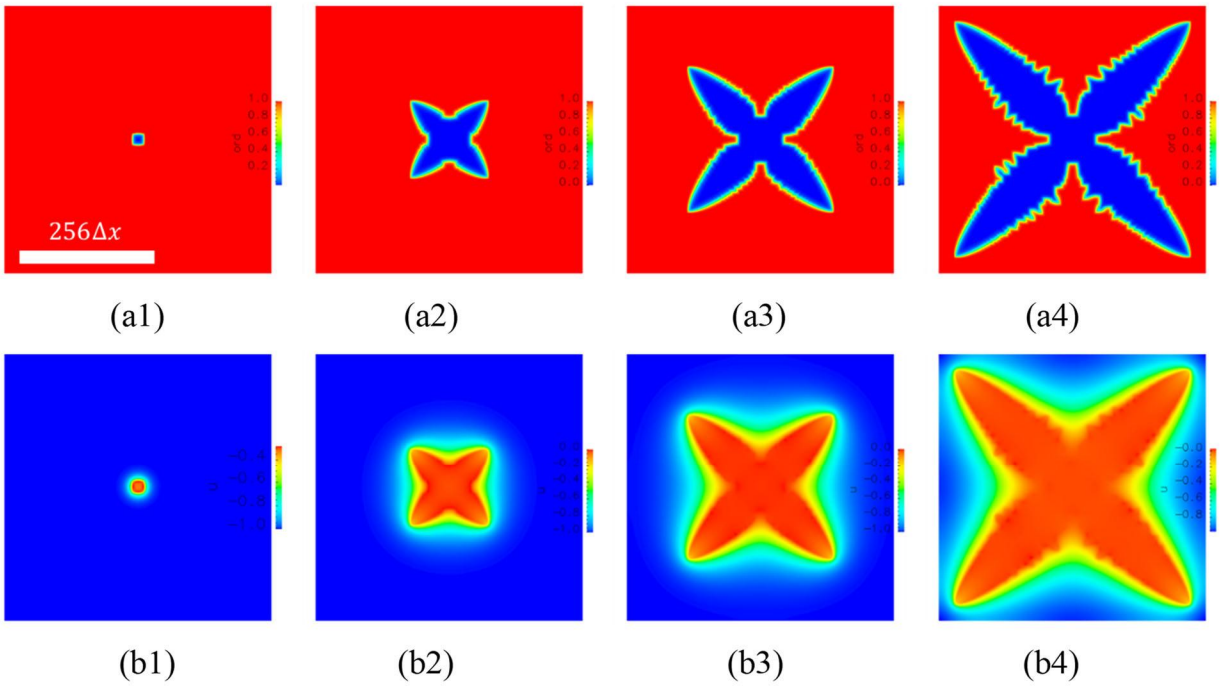


Figure 13. Simulated U crystal morphology evolution with time when $\sigma_4 = 0.3$ and $\beta_4 = 0.3$ (a1-a4). (b1-b4) give the corresponding temperature fields at time = $0.002t_0$, $0.03t_0$, $0.06t_0$, and $0.1t_0$, respectively.

In practice, σ_4 and β_4 can vary. Figure 14 displays U dendrite morphology evolution with time and the associated temperature field in the case when $\sigma_4 = 0.15$ and $\beta_4 = 0.3$. By fixing $\sigma_4 = 0.15$ and letting β_4 vary, Figure 15 shows the dendrite morphology changing with varying β_4. It is seen that not only the shape but also the orientation of the dendrite changes with varying β_4. Similarly, both the shape and orientation change with different α_4 by fixing β_4. Figure 16 displays the dendrite morphology change with varying α_4 when $\beta_4 = 0.24$.

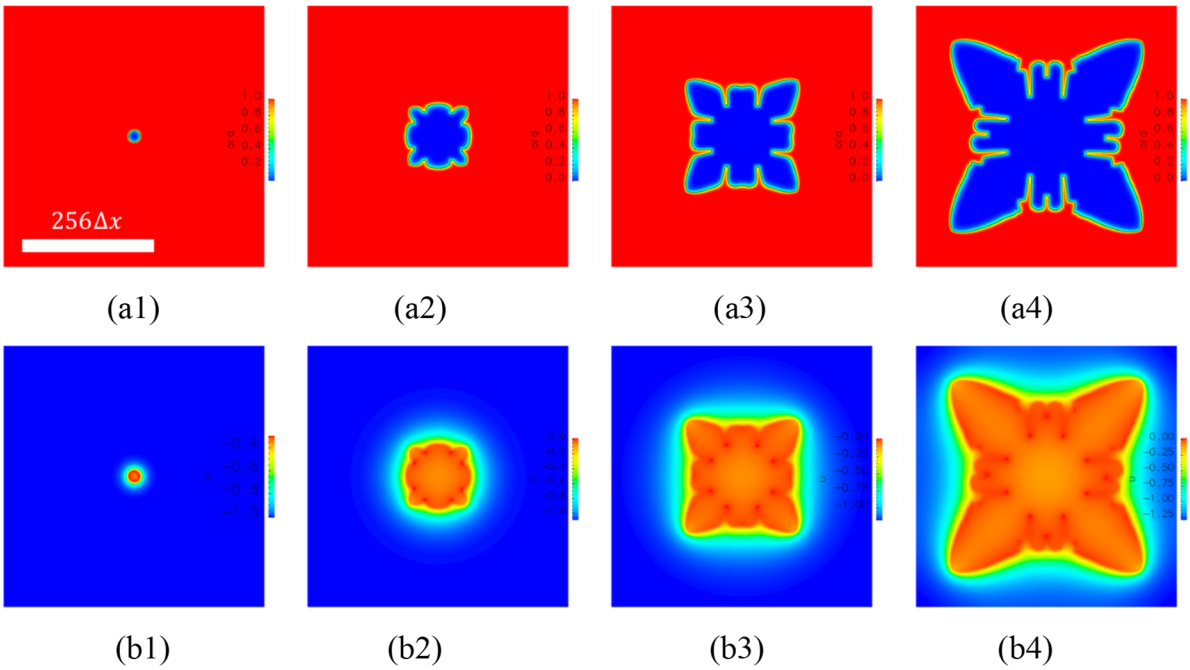


Figure 14. Simulated U crystal morphology evolution with time when $\sigma_4 = 0.15$ and $\beta_4 = 0.3$ (a1-a4). (b1-b4) give the corresponding temperature fields at time = $0.002t_0$, $0.03t_0$, $0.06t_0$, and $0.1t_0$, respectively.

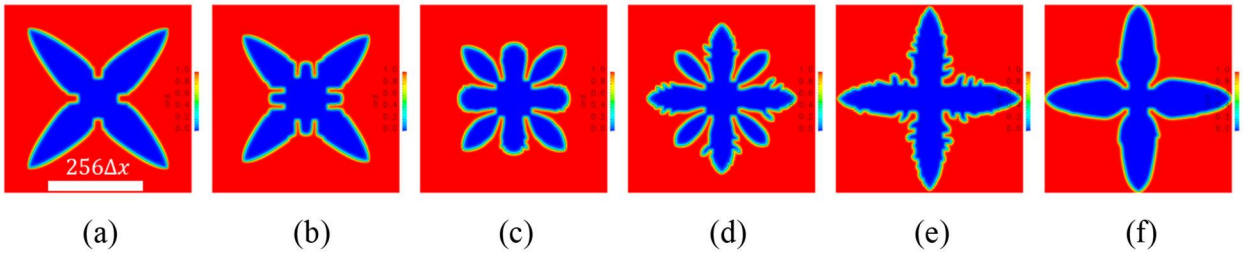


Figure 15. Simulated U crystal morphology variation, where red regions are liquid and blue regions are solid, with β_4 when $\sigma_4 = 0.15$. (a) $\beta_4 = 0.05$, (b) $\beta_4 = 0.24$, (c) $\beta_4 = 0.4$, (d) $\beta_4 = 0.5$, (e) $\beta_4 = 1.0$ and (f) $\beta_4 = 1.5$, respectively, at time = $0.1t_0$.

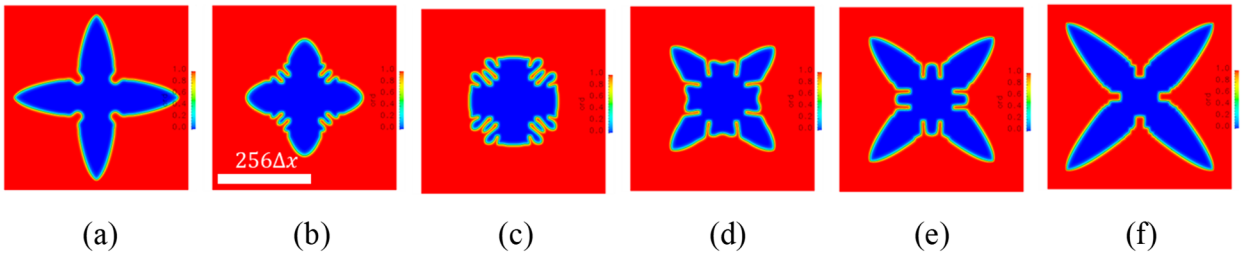
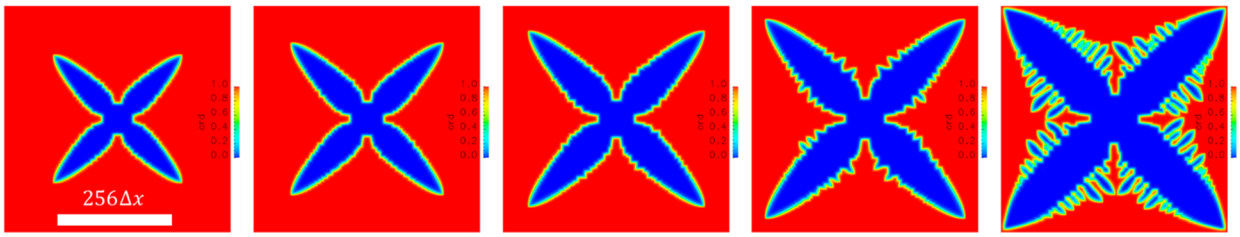


Figure 16. Simulated U crystal morphology variation with σ_4 when $\beta_4 = 0.24$. (a) $\alpha_4 = 0.01$, (b) $\alpha_4 = 0.075$, (c) $\alpha_4 = 0.1$, (d) $\alpha_4 = 0.13$, (e) $\alpha_4 = 0.15$ and (f) $\alpha_4 = 0.2$, respectively, at time = $0.1t_0$.

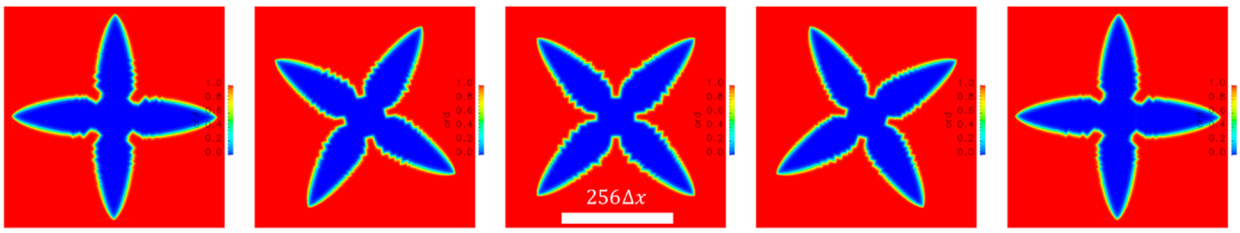
Parameter $\Omega = c_p(T_m - T_\infty)/L$ is determined by the solidification temperature T_∞. Higher environment temperature T_∞ gives a smaller Ω. The effect of varying Ω is simulated, and its effect can be seen from Figure 17. Obviously, larger Ω results in faster growth.



(a) (b) (c) (d) (e)

Figure 17. Morphology variation with Ω when $\alpha_4 = \beta_4 = 0.3$. (a) $\Omega = 0.40$, (b) $\Omega = 0.45$ (c) $\Omega = 0.50$, (d) $\Omega = 0.55$ and (e) $\Omega = 0.60$, respectively, at time = $0.1t_0$.

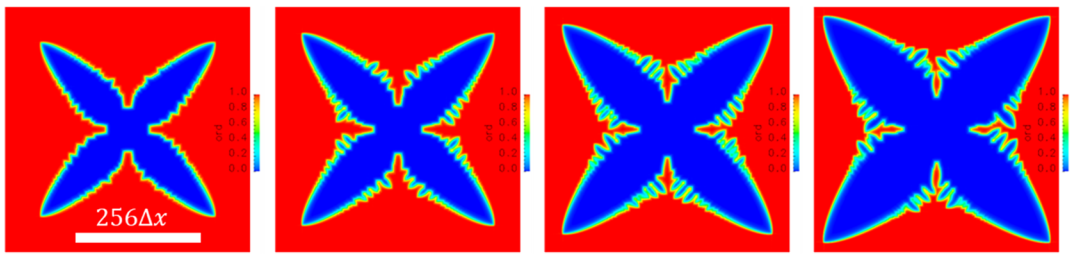
Figure 18 shows the dendrite orientation change with the given offset azimuthal angle θ_0. Figure 18 (a) and (e) are fundamentally identical as $\cos[4(\theta - \theta_0)]|_{\theta_0=0^\circ} = \cos[4(\theta - \theta_0)]|_{\theta_0=90^\circ}$.



(a) (b) (c) (d) (e)

Figure 18. Morphology variation with θ_0 when $\alpha_4 = \beta_4 = 0.3$ and $\Omega = 0.55$. (a) $\theta_0 = 0.0$, (b) $\theta_0 = 30°$, (c) $\theta_0 = 45°$, (d) $\theta_0 = 60°$ and (e) $\theta_0 = 90°$, respectively, at time = $0.08t_0$.

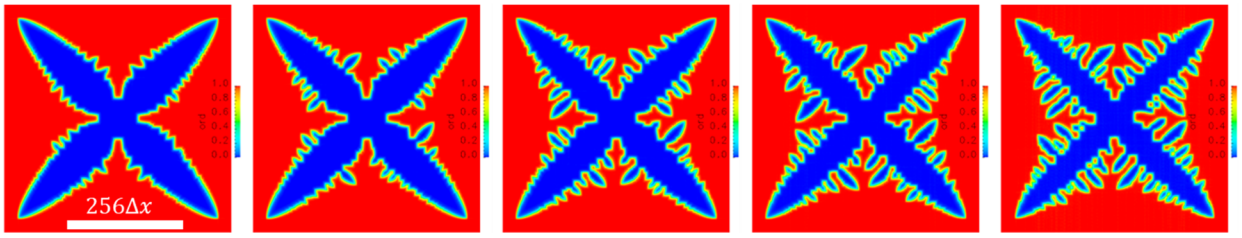
For the simulations shown above, $\dot{Q}_0 = 0.0$. The growth kinetics of a simulated U dendrite is enhanced by decreasing the environmental temperature, i.e., $\dot{Q}_0 < 0.0$. Figure 19 shows the effect of $\dot{Q}_0$ on the dendrite morphology when fixing $\alpha_4 = \beta_4 = 0.3$ and $\Omega = 0.55$, $\theta_0 = 45°$.



(a) (b) (c) (d)

Figure 19. Morphology variation with $\dot{Q}_0$, (a) $\dot{Q}_0 = 0.0$, (b) $\dot{Q}_0 = -2.0$, (c) $\dot{Q}_0 = -4.0$, (d) $\dot{Q}_0 = -6.0$.

Finally, for the simulations above, we did not consider thermal noise. We can describe thermal noise contributions to the simulation by adding a term on the right side of eq. (9) of $noise = p_{noise} num_{rand} \sqrt{\Delta t}(1-\phi_i)\phi_i$, $-0.5 < num_{rand} < 0.5$. Figure 20 illustrates the effect of the noise strength coefficient p_{noise} on the dendrite morphology. Obviously, thermal noise causes second arm formation. Large noise results in longer second arms.



(a) (b) (c) (d) (e)

Figure 20. Morphology variation with p_{noise}. (a) $p_{noise} = 0.0$, (b) $p_{noise} = 5.0$, (c) $p_{noise} = 15.0$, (d) $p_{noise} = 30.0$ and (e) $p_{noise} = 50.0$.

3.3.2 Effect of Impurity Particles and Multiple Crystals

Equation (10) can be modified to consider the effect of impurity particles in the melt. As mentioned in the Project Rationale section, if C is dissolved in liquid U, uranium carbide precipitates may form prior to solidification of the U matrix and affect the solid microstructure. Continuing with a dendritic growth model of U in the presence of secondary phase carbide particles, ϕ_1 is used to describe the pure U crystal, and ϕ_2 is used to represent all the existing impurity particles in the melt pool, but it does not evolve with time. ϕ_2=0, $u = -1$ inside the impurity particles and ϕ_2=1 outside the impurity particles. ϕ_2 varies continuously across 0 and 1. In eq. (9), a positive parameter λ_1=20 is used to describe the interaction between the growing U dendrite and the impurity particles. In eq. (10), a small random value is used to replace $\frac{\partial \phi_2}{\partial t}$, to describe the particle effect on the local temperature field.

Figure 21 compares U dendrite morphologies over time with (Figure 21 b1-b6) and without (Figure 21 a1-a6) the presence of uranium monocarbide impurity particles. For these simulations, a large simulation cell of 2048 x 2048 grids was used. All model parameters are the same as in Table 3, and $p_{noise} = 3.0$. It is seen that the presence of impurity particles affects early U dendrite arm formation, as shown in the expanded snapshots (b2') and (b4'). However, with dendrite growth, the effect becomes weaker over time. There is no obvious difference between (a6) and (b6).

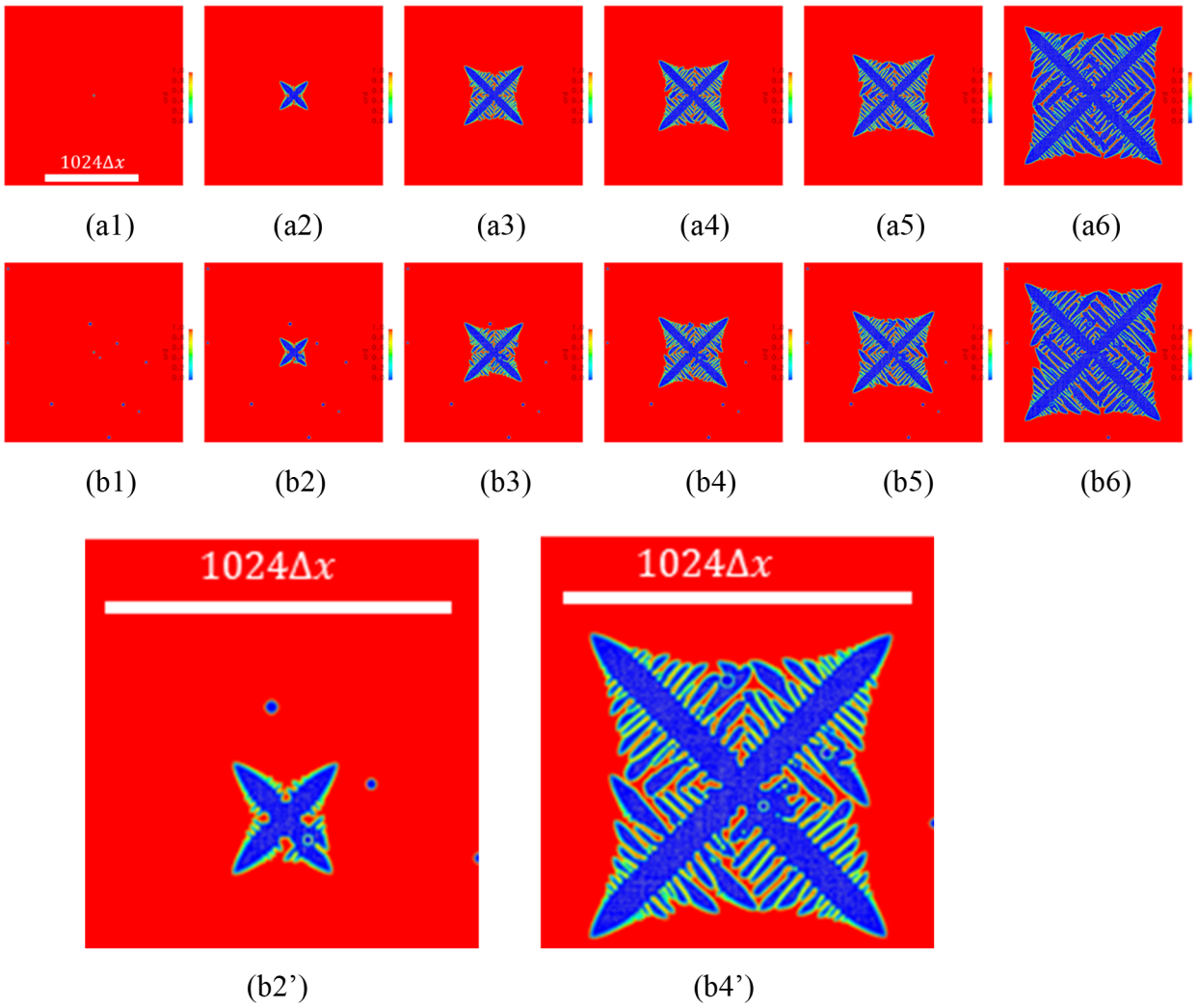


Figure 21. Simulated snapshots of U dendrite morphologies without impurity particles present in the region of interest (a1-a6) and with impurity particles present (b1-b6) at times of (1) $0.002t_0$, (2) $0.06t_0$, (3) $0.12t_0$, (4) $0.14t_0$, (5) $0.16t_0$, and (6) $0.25t_0$, respectively. (b2') and (b4') are the expanded views of (b2) and (b4), respectively. The displayed white bars give the simulated size. The blue solid circles in the molten red regions represent impurity particles.

We have demonstrated how the model parameters affect single dendrite morphology and dendrite growth kinetics. Next, we consider multiple dendrites and their interaction. The large simulation cell of 2048 x 2048 uniform grids was used for the following simulations. 4 solid phase nuclei (i.e., N=4 in eq. (9)) were initially used with radii of 5 grids. Inside each nucleus i, $\phi_i = 0, \phi_j = 1\ (j \neq i), u = -1,$ when $r = \sqrt{(x - x_{0i})^2 + (y - y_{0i})^2}$<5, where (x_{0i}, y_{0i}) is the center point coordinate of nucleus i. Elsewhere, $\phi_j = 1\ and\ u = 0\ (j = 1, \cdots, N)$. The offset azimuthal angles for each nuclei are as follows: $\theta_{01} = 0°, \theta_{02} = 45°, \theta_{03} = 30°\ and\ \theta_{04} = 60°$. The model parameters are the same as those listed in Table 3, except those stated separately.

Figure 22 shows U dendrite growth and grain formation with evolution time. In this simulation, $\dot{Q}_0 = -2.0$, and $p_{noise} = 3.0$. It is seen that before dendrites meet each other, they grow like single crystals. When they meet each other, each interface hinders growth, and final grain shapes depend on the initial nuclei locations and orientations.

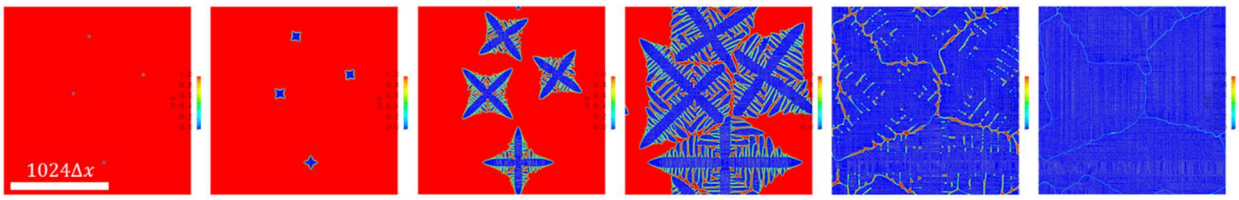


Figure 22. Snapshots of simulated U dendrite morphologies and grain formation with time when $\dot{Q}_0 = -2.0$. (a) $0.002t_0$, (b) $0.02t_0$, (c) $0.1t_0$, (d) $0.2t_0$, (e) $0.4t_0$, and (f) $0.59t_0$, respectively. The light blue lines of (f) are the grain boundaries.

Figure 23 gives the simulated U dendrite growth and grain formation with evolution time when $\dot{Q}_0 = -1.0$, a lower cooling rate than that shown in Figure 22, while other parameters are the same. It is seen that the dendrites grow much slower when $\dot{Q}_0 = -1.0$ than in the case of $\dot{Q}_0 = -2.0$, in particular, at the late stages. This is because the generated heat from dendrite growth cannot be released efficiently, due to the lower cooling rate, which hinders the further growth of dendrites. In practice, cooling rate also affects the U metal nucleation rate, though the presence of impurity particles likely also impacts U nucleation. Larger cooling rates cause denser dendrite nucleation. As a consequence, denser nucleation results in smaller grain size.

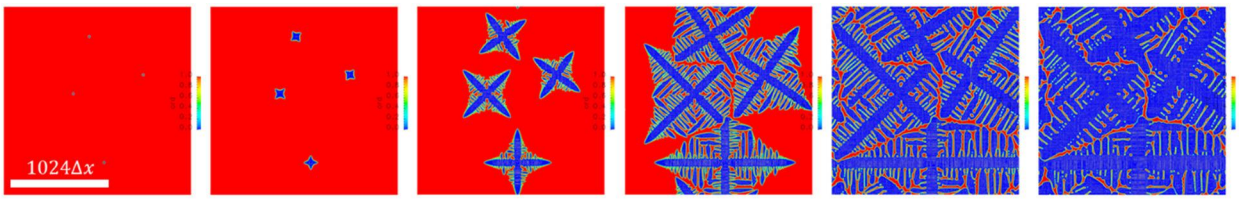


Figure 23. Snapshots of simulated U dendrite morphologies and grain formation with time when $\dot{Q}_0 = -1.0$. (a) $0.002t_0$, (b) $0.02t_0$, (c) $0.1t_0$, (d) $0.2t_0$, (e) $0.4t_0$, and (f) $0.59t_0$, respectively. The light blue lines are the grain boundaries.

Consider the case where there are preexisting impurity particles in the U liquid. We have considered 330 particles in the case shown in Figure 24. The parameters used in this simulation are the same as those used in Figure 22.

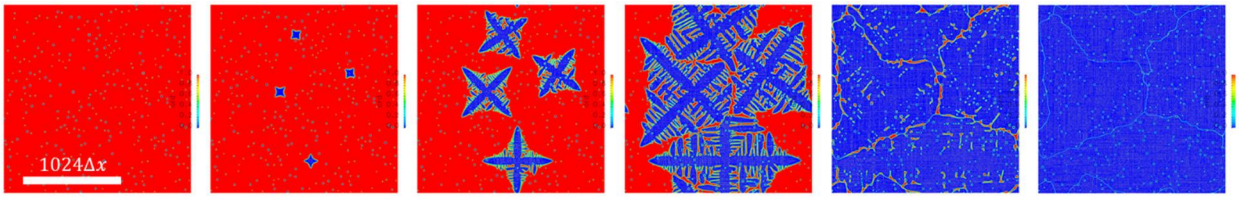


Figure 24. Snapshots of simulated U dendrite morphologies and grain formation with time when uranium monocarbide particles are present as the smallest blue spheres. (a) $0.002t_0$, (b) $0.02t_0$, (c) $0.1t_0$, (d) $0.2t_0$, (e) $0.4t_0$, and (f) $0.59t_0$, respectively. The light blue lines in (f) are the grain boundaries.

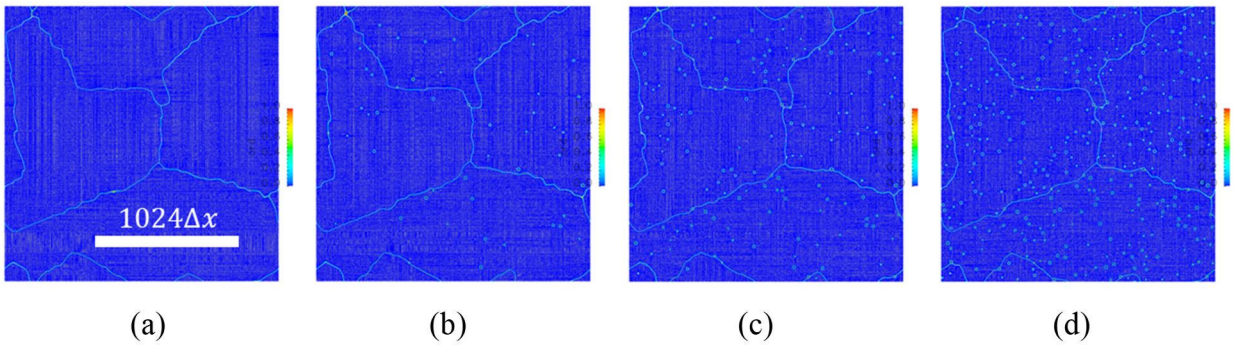


Figure 25. Snapshots of simulated U grain sizes and grain boundaries at $t = 0.59t_0$ in the presence of varying impurity particle distributions and concentrations. (a) no particles, (b) 50 particles, (c) 100 particles, (d) 330 particles, respectively. The light blue lines in the snapshots are grain boundaries.

More simulations were performed with varying distributions of existing particles, and grain morphologies at time $t = 0.59t_0$ are provided for these in Figure 25. The particles' locations and radii are randomly selected between 3 and 10 grids. It is seen that the presence of particles does not have a significant effect on the overall grain sizes and boundaries, under these simulation conditions, though locally, grain boundary curvatures near particles are slightly impacted.

The effect of the cooling rate $\dot{Q}_0$ was also investigated in the presence of impurity particles. Figure 26 displays snapshots of simulated U dendrite morphologies at $t = 0.3t_0$ versus cooling rate $\dot{Q}_0$. Obviously, the cooling rate significantly affects the dendrite growth kinetics.

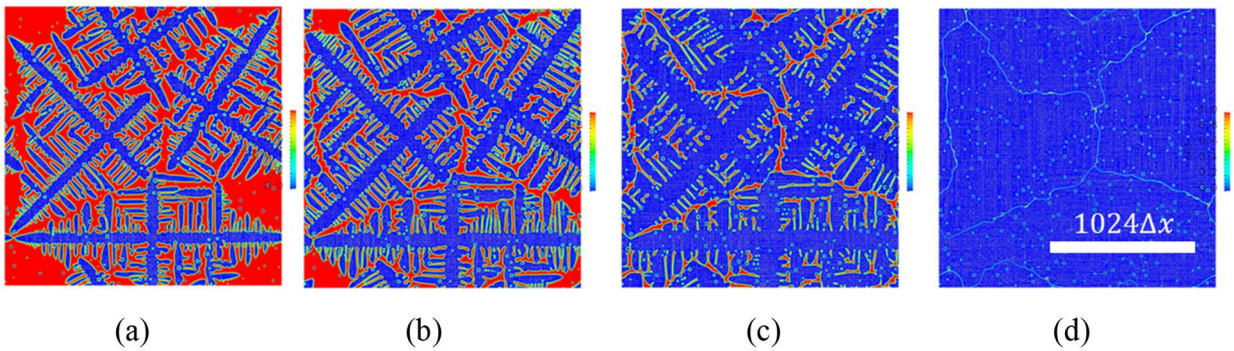


Figure 26. Snapshots of dendrite morphologies at $t = 0.3t_0$ versus cooling rate $\dot{Q}_0$: (a) $\dot{Q}_0 = -0.1$, (b) $\dot{Q}_0 = -1.0$, (c) $\dot{Q}_0 = -2.0$, and (d) $\dot{Q}_0 = -4.0$, respectively.

Finally, we have parallelized our computational code for larger simulation cells and more dendrite evolution. Figure 27 shows examples of such simulation outputs with a 4096 x 4096 grid simulation cell size and 8 dendrites in the presence of either 10 particles (Figure 27 a) or 300 particles (Figure 27 b). Again, it is seen that the presence of these particles has little influence on the final simulated grain structures, under these computational conditions.

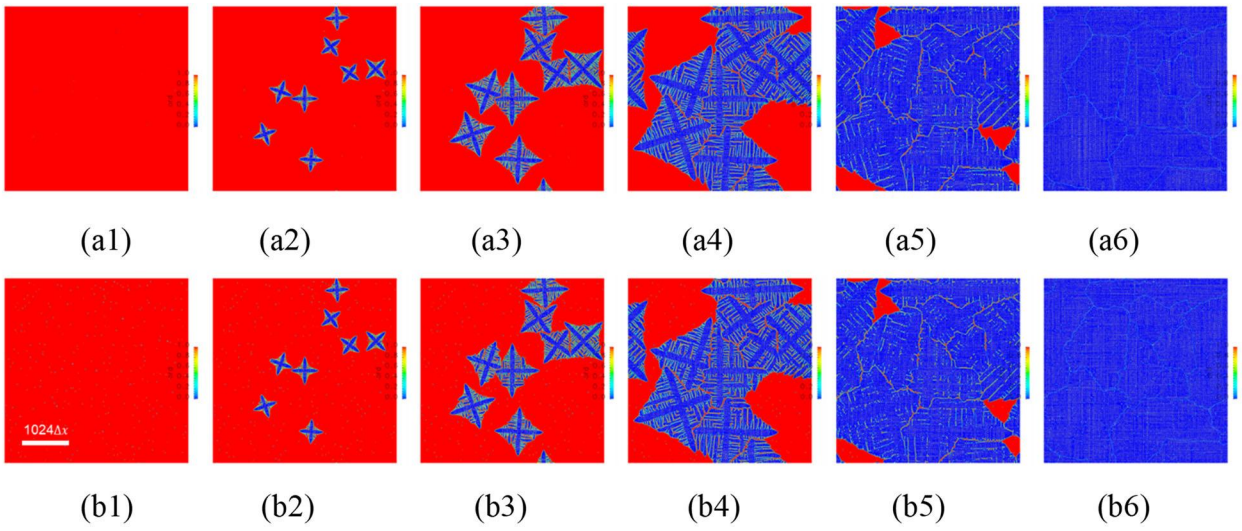


Figure 27. Snapshots of dendrite morphologies and grain formation with time when uranium monocarbide impurity particles are present: (a1, b1) $0.002t_0$, (a2, b2) $0.08t_0$, (a3, b3) $0.15t_0$, (a4, b4) $0.28t_0$, (a5, b5) $0.4t_0$, and (a6, b6) $0.59t_0$, respectively. The light blue lines in (a6, b6) are the grain boundaries. Case (a) contains 10 impurity particles and case (b) contains 300 particles.

In summary, within the time and budget of this project, we extended our previous phase field model of solidification in four aspects: 1) addition of thermodynamic properties of U from U phase diagram calculations; 2) interaction of multigrain growth with different grain orientations; 3) observation of the effect of existing impurity particles on dendrite grain growth; and 4) parallelization of the simulation code. With this extended model, we can simulate the effects of various thermodynamic and kinetic parameters on U crystal growth and morphology as well as develop data sets to train machine learning models.

Regarding the presence of impurity particles, the current modelling capability assumes that 1) uranium carbide impurity particles form in the molten U and 2) these particle don't evolve or move during solidification of the U. The first assumption is informed by accepted phase diagrams of the U-C system at relatively high carbon concentrations, as alluded to in the Project Rationale section. These high concentrations are achievable experimentally, for model validation purposes, though they may be higher than commonly expected in applicable U casting environments. These assumptions, however, greatly simplify the computational problem, at this stage.

In reality, uranium carbide may nucleate concurrently with the surrounding U. Given a starting concentration of C in the molten U, as γ-U grains grow, the C concentration in the melt is expected to increase, since the solubility of C in the solid γ-U crystals is likely relatively low. Therefore, carbide nucleation and growth may continue to occur during solidification. Adding complexity, the interaction between impurities particles and the γ-U solidification front in the real system may cause particles to move. Both the evolution and motion of these particles would affect the spatial

and size distributions of impurities in final grain structure, which will be addressed in future versions of this phase field model, as described below.

Finally, electron microscopy and EBSD of room temperature U from our laboratory casting system corroborates the grain disorientations commonly observed and attributed to crystallographic phase transitions that could also be important to simulate to gain a complete picture of microstructural evolution in U. Since phase field modeling is useful for simulating the relationship between microstructure and deformation, it could be ideal for modeling the phase transitions in cooling U, which accompanies large lattice mismatching resulting in local stress and unique martensitic structures.

Therefore, in future work, we expect to extend the current phase field model by adding more relevant physics: 1) the evolution and motion of carbide particles, which could help to improve understanding of particle behavior during casting; and 2) the crystallographic phase transitions, which take γ-U to α-U, and could help to link the crystalized grain structure to the one observed at room temperature.

4.0 Machine Learning

4.1 Background

In recent years, several approaches to solve various PDEs, such as those required for our phase field model above, with ML have emerged.[37-43] Additionally, there have been studies aimed at implementing ML techniques to understand solid transformations in the microstructure of U alloys using images.[44, 45] These previous studies set a precedence for application of ML toward the unique challenges encountered during efforts to describe and model actinide casting systems.

Our work uses powerful Deep Operator Networks (DeepONets) [39] to not just predict solutions but to learn an operator that can take in alternate parametric values. For this system, the variables of the solutions are (x, y, t), and the parameters that the DeepONet can choose from are listed in Table 3. For this study, a DeepONet was trained to create solutions for various values of $\beta_m, \dot{Q}_0$ and θ_0, defined by the phase field model, while the remaining parameters are held constant. Thus the learned neural operator should be able to generate solutions over an (x, y, t) tensor in the remaining (not yet numerically solved) parameter space of $\beta_m, \dot{Q}_0$ and θ_0 for the fixed values of the remaining parameters – that is, provided an initial condition and new values for $\beta_m, \dot{Q}_0$ and/or θ_0, the neural operator should generate reasonable accurate solutions. To change the training parameters, new phase field solutions over a desired range of parameters will need to be generated, and the model will need to be retrained (taking approximately 12 hours), but then, results over the remaining unsolved parameter space can be generated instantaneously. Unlike other attempts to predict microstructure evolution using neural operators,[46] this attempt applies architectural features that advantage a limited training set as much as possible and keep training times down, so the model can be quickly retrained to continue exploring parameter spaces.

4.2 Methods

A DeepONet is a machine learning algorithm that uses a specialized architecture to expand traditional deep learning's ability to learn nonlinear, continuous functionals (mappings from a space of functions into the space of real numbers) to a more powerful network capable of learning nonlinear mathematical operators (mappings from a space of functions into another space of functions). An example of a functional is the definite integral of a function, which maps every integrable function to a real number value. While neural networks have been able to learn functionals like these effectively for some time, DeepONets are a recent breakthrough in learning operators. For the work presented in this report, the operator being learned is the solution to a system of PDEs that are modified by altering the parameters of the systems. This capability supports phase field modeling by allowing modelers to quickly explore the parameter space, to identify key parameters that change the solution functions to satisfy desired properties.

The basic structure of the improved DeepONet architecture,[47] shown in Figure 28, indicates the key data structures required to train a DeepONet in orange boxes. The first required data structure is a branch input, which contains the initial conditions of the PDEs in the training set. To train the DeepONet, several PDE solutions must be provided, and the branch input structure contains the initial conditions of each solution in the training set. For this case study, the parameters $\Omega = (\beta_m, \dot{Q}_0, \theta_0)$ that are varied in the $v_1(\vec{m}; \Omega), v_2(\vec{m}; \Omega), \ldots v_n(\vec{m}; \Omega)$ functions are also included in the branch input, associated with their respective initial conditions. Because the DeepONet learns a mathematical operator, a second input - the trunk input - provides a mesh of sensors $\mathbb{M}$ to indicate the points at which the solution function should be evaluated. During training, the mesh must be consistent - all data in the training set must use the same mesh - but

there is no requirement of even spacing in the mesh. After the operator is learned, the DeepONet can provide solutions on novel meshes, regardless of the mesh on which the DeepONet was trained. Finally, the DeepONet uses full solutions from the phase field model in the validation step to determine how well it learned the operator. The solutions from the learned operator are generated at the points provided the trunk input, so - during training - the training set will align with the trunk.

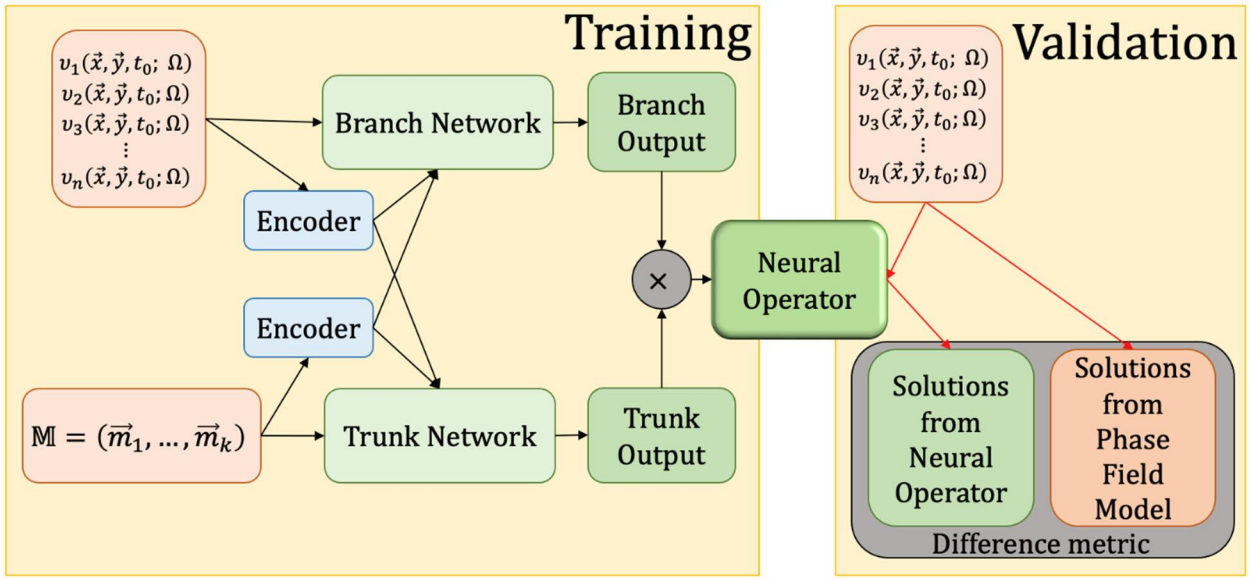


Figure 28. Basic structure of the improved DeepONet architecture.[47]

4.2.1 Data Preparation

Simulations of evolving U grains from melt were generated using phase field modeling, as described above, while changing three parameters (from Table 3):

1. a kinetic coefficient β_m with values ranging from 0 to 1.4 in increments of 0.1,
2. three cooling rate values $\dot{Q}_0$ indicating either system cooling (-3), system heating (3), or no change in the overall system temperature (0), and
3. rotations of the direction of the dendrite growth θ_0 at angles 0, 30, 45, and 60 degrees.

For β_m values in the range $[0.1, 1]$, a simulation was generated for each possible combination. For β_m values $\{0, 1.1, 1.2, 1.3, 1.4\}$, simulations were only generated with cooling rates $\{-3, 3\}$ and growth angles of $\{30, 45\}$.

Each simulation contained two 512 x 512 images at 40 time steps recorded at intervals of 0.002 seconds. The first image represents the solidification of a grain and is measured as $\varphi \in [0, 1]$, where 1 represents a liquid phase and 0 represent complete solidification, as described in the Phase Field Modeling section. The second image records a dimensionless temperature measurement $u \in [-1, 0]$. In an initial regime of insufficient data for ML early in the project, additional data sets were artificially generated by separating each simulation into two sets indexed by their time steps. The first set contained time steps representing $\{0.002, 0.006, 0.010, 0.014, \ldots\}$, and the second set contained time steps representing $\{0.004, 0.008, 0.012, 0.016, \ldots\}$. This method of structuring the data leans into a key strength of the DeepONet: the ability to interpolate data from an underlying grid structure. In this case, the trunk of the DeepONet can be trained at the

time steps represented in the training data, and the missing timesteps will be interpolated by the network when training is complete. This practice of separating the data by alternating time steps was maintained when more data became available, with the expectation that the training benefit would scale with the amount of training data available.

While traditional DeepONets require only the initial state provided as a branch structure, there were several limitations to this standard approach in early training regimes for this project. The key challenge was that, unlike traditional DeepONets, this model is attempting to predict a transient state of the growing crystal, and it was hypothesized that initial information about evolution would need to be provided. The second obstacle was that, before the implementation of a composite convolutional neural network (CNN)-DeepONet, the fine details of the early grain structure could be lost entirely in the process of creating coarse grained branch and target data. Finally, even if sufficient data was present in the early stages of grain evolution, some grains with different terminal morphologies initially look very similar and may provide little information for the DeepONet to use to distinguish simulations. Due to these limitations, initial regimes included the first three time steps of a sample (either time steps $\{0.002, 0.006, 0.010\}$ or $\{0.004, 0.008, 0.012\}$), to ensure the DeepONet was provided sufficient initial information.

A typical DeepONet trunk is constructed from Euclidean coordinates consisting of spatial and temporal dimensions. But for systems with periodic boundary conditions, researchers achieved stronger results by embedding structural information - in this case, periodicity - into the trunk structure.[48] For the 2D case, the Euclidean (x, y) coordinate values in the original trunk structure are replaced with square Fourier basis functions,

$$\{\cos n\omega x \cos m\omega y\,, \cos n\omega x \sin m\omega y\,, \sin n\omega x \cos m\omega y\,, \sin n\omega x \cos m\omega y\}_{m,n=0,1,\dots} \qquad (11)$$

where $\omega = \frac{2\pi}{P}$, and P is the period of the solution, which is the same in the x and y directions. In the current implementation, the basis functions are computed for $m, n \in \{0, 1\}$. Alternate basis formations are recommended in the original research [48] that also decrease the size of the feature expanded trunk for further computational efficiency. Finally, each grid point is associated with a corresponding time step, leading to a feature expanded trunk that naturally enforces periodic boundary conditions. Because a feature expansion trunk naturally enforces periodic boundary conditions, this model does not require additional components in the loss function to enforce the periodic boundary.

4.2.2 Model Structure

A primary objective for this model is to create accurate solutions that represent the transient states of grain evolution, while staying as computationally efficient as possible. Due to the complexity of the data being learned, it is theorized that the information gained while learning the transient states increases the accuracy of the final grain structure. Modeling transient states to replicate phase field models is not unprecedented for DeepONets, but the amount of data required calls for novel architectures to maintain the key desired benefit of computational efficiency. In a previous study,[46] a DeepONet is trained for 120,000 epochs on 1,600 high-fidelity solutions containing 80 time steps each. To retain as much information as possible while shrinking the size of the data to be provided to the DeepONet, the inputs are transformed into latent vectors by an autoencoder that is trained separately from the DeepONet.

The approach presented here is trained from a more limited data regime of 140 simulations with just 40 time steps each. As described in the Data Preparation section above, these simulations are separated to generate two sample sets from each simulation, both halving the size of the training samples and doubling the size of the training set. Since there are fewer data in each sample, the latent representations can be generated during the training of the DeepONet,

allowing the model to identify key features from the two 512×512 representations that improve its neural operator. For this approach, a two-layer CNN is applied to the φ and u branch input data, reducing the size of each frame from a 512×512 square to a 128×128 square and finally to a 64×64 square. Each frame in the branch input has its own channel. A 64×64 latent space was selected since satisfactorily representative solutions were generated in a reasonable training time of approximately 12 hours. The argument for using a composite model is that the CNN can update the decision process of how to encode the data based on how well the DeepONet performs. When the process of encoding the data is separated from the DeepONet, the encoder has less opportunity to support the training of the DeepONet.

After the two-layer CNN, the encoded output is provided to the modified DeepONet encoder as well as the branch network, as shown in Figure 28. The trunk mesh is also encoded and shared with the trunk network. The branch and trunk networks both contain 3 hidden layers, and the encoded branch and trunk inputs are inserted into each hidden layer, as described for the modified DeepONet.[47] The final layers are the branch and trunk outputs, which are combined with a dot product to create the learned neural operator. A final, physics informed requirement is imposed, to enforce that the solutions stay within the experimental domain. The order variable is defined such that it must take values in the interval [0,1], while the temperature variable remained within the interval (-1.5, 0.1), for all simulations. For an output x, a final layer can be applied to ensure the output is restricted to the interval $[x_a\ ,\ x_b]$, by applying a layer of the form

$$x_a + \frac{1}{2}(1 + \tanh(x))(x_b - x_a) \tag{12}$$

To aid in training, such a layer was applied to keep the order and temperature values within their respective intervals.

The difference metric applied to determine the accuracy of the neural operator solutions uses a mean squared error (MSE) across all the values in the training set,

$$L\big(\Upsilon(\vec{m};\Omega)\big) = \frac{1}{n}\sum_n \sum_x \sum_y \sum_t (\upsilon_n - \hat{\upsilon}_n)^2 \tag{13}$$

where $\Upsilon(\vec{m};\Omega)$ is the set of all coarsened phase field model solutions in the training batch, $\upsilon_n(\vec{m};\Omega)$ is a coarsened phase field model solution, $\hat{\upsilon}_n(\vec{m};\Omega)$ is the neural operator solution, and $n \in N$, where N is the set of all samples in the training batch, $\Upsilon(\vec{m};\Omega)$. The loss is computed as

$$\mathcal{L} = \lambda(w_1)L(\varphi) + \lambda(w_2)L(u) + \lambda(w_3)L(\varphi_{IC}) + \lambda(w_4)L(u_{IC}) \tag{14}$$

where $\varphi_{IC} = \varphi(\vec{x},\vec{y},t_0;\Omega)$ and $u = u(\vec{x},\vec{y},t_0;\Omega)$. The values $\{w_1, w_2, w_3, w_4\}$ are self-adaptive weights.[49] In modeling crystal growth and evolution, the most interesting details occur at the interface between the grain and the surrounding matrix, while the majority of the remaining space remains uniform. Incorporating self-adaptive weights into the loss function encourages the DeepONet to pay closer attention to the complicated interface. Finally, the λ functions are a mask for the self adaptive weights - in this case, a sigmoidal mask,

$$\lambda = \frac{1}{1+e^{-x}} \tag{15}$$

which ensures the self-adaptive weights stay positive.

4.2.2.1 Physics Residuals

The physics model of grain growth requires that the temperature and order surfaces evolve in time such that eq 7 and 8 above are always satisfied. These conditions can be captured by defining the following physics residuals:

$$F_\phi\big(v;(x,y,t)\big) = -\epsilon^2\tau_0\tilde{\beta}\tilde{\sigma}\frac{\partial\phi}{\partial t} + \phi(1-\phi)\left\{\left(\phi-\frac{1}{2}\right)+30\epsilon\alpha\Omega u\phi(1-\phi)\right\} - \epsilon^2\frac{\partial}{\partial x}\left[\tilde{\sigma}\tilde{\sigma}'\frac{\partial\phi}{\partial y}\right]\epsilon^2\frac{\partial}{\partial y}\left[\tilde{\sigma}\tilde{\sigma}'\frac{\partial\phi}{\partial x}\right] + \epsilon^2\nabla[\tilde{\sigma}^2\nabla\phi] \quad (16)$$

$$F_u\big(v;(x,y,t)\big) = \frac{\partial u}{\partial t} + \frac{1}{\Omega}30\phi^2(1-\phi)^2\frac{\partial\phi}{\partial t} - \nabla^2 u - B(t) \quad (17)$$

for a solution map $v\colon (x,y,t) \to \phi(x,y,t), u(x,y,t)$. Then for any v that is a valid solution of the model on the domain U, we must have for all $(x,y,t) \in U$,

$$F_u\big(v;(x,y,t)\big) = 0, \quad (18)$$

$$F_\phi\big(v;(x,y,t)\big) = 0. \quad (19)$$

A loss term can then be defined to capture any discrepancies in a candidate solution v and model solutions with

$$L_p\big(v(\vec{m},\Omega)\big) == \frac{1}{n}\sum_n\sum_x\sum_y\sum_t F_u\big(v;(x,y,t)\big)^2 + F_\varphi\big(v;(x,y,t)\big)^2 \quad (20)$$

such that the loss $\mathcal{L}$ can be expanded as

$$\mathcal{L} = \lambda(w_1)L(\varphi) + \lambda(w_2)L(u) + \lambda(w_3)L(\varphi_{IC}) + \lambda(w_4)L(u_{IC}) + \alpha L_p(v) \quad (21)$$

for $\alpha > 0$ a weighting parameter. As with the other loss terms, the weight assigned to the physics residuals is critical. Minimization of L_p only promotes v being a solution of eq 7 and 8, but of course only the particular solutions corresponding to given boundary and initial conditions are what is desired. Overweighting of L_p can promote learning trivial solutions that ignore initial conditions, while insufficient weight will of course not influence training.

Note that the expressions for the physics model (eq 7 and 8) contain higher order derivatives and their composition with nonlinear functions. This creates challenges for use of the physics residuals in training including computational cost, accuracy, and sensitivity. One potential avenue to circumvent these challenges is to instead consider the energy functional from which they are derived.

4.2.2.2 Energy Minimization

The governing equations for grain growth (eq 7 and 8) are derived following a variational principle. The temperature and order parameters solutions are those that minimize the system free energy $\mathcal{F}$ given by

$$\mathcal{F}(v) == \int_U f_0\left(\frac{1}{2}W(\vec{n})^2|\nabla\varphi|^2 + f_{chem}(\varphi,u)\right)dU. \quad (22)$$

Full details for this expression can be found in Section 3.2.1.

Computation of the energy functional does not require any of the higher order derivatives that evaluating the physics residuals F_u and F_ϕ does. Moreover, minimization of the functional can be directly incorporated into the training loss with

$$\mathcal{L} = \lambda(w_1)L(\varphi) + \lambda(w_2)L(u) + \lambda(w_3)L(\varphi_{IC}) + \lambda(w_4)L(u_{IC}) + \alpha\mathcal{F}(v) \quad (23)$$

so as to encourage learning solutions that minimize the free energy in accordance with model solutions. Again appropriate weighting of the energy term is crucial to balance its influence against fitting the initial and boundary conditions that define the solutions of interest.

4.2.3 Training Parameters

A brief overview of optimal training parameters is provided below:

1. Batch sizes: Training accuracy improved as batch sizes decreased, but this traded off with efficiency. Smaller batches means more batches have to be run and training takes longer. Batch sizes for this report were 8 samples, compared with the efficiency of using batch sizes of 12, 16, and 20.

2. Layer widths: Increasing the layer width eventually led to diminishing returns. The original model had layer widths of 40, but, when the CNN layer was added and target sizes were increased, the small width restricted how well the model trained. Training with layer widths of 60 and 80 nodes showed strong gains, but the improvements slowed down with 100 and 120 nodes. The use of 100 and 120 nodes also corresponded with longer training time. If too many nodes are used, it will become possible to overtrain. Due to the desire to keep the model easy to train, layer widths of 80 are recommended, but increasing layer size to 100 or 120 show no consequence to how well the model trains, just to how long it takes.

3. Activations: This model was generally trained with a hyperbolic tangent activation and briefly examined the use of rectified linear units (ReLU), which are not ideal for physics-informed cases. The hyperbolic tangent activation was much more effective. A similar study [46] used sine activations, which were not considered during this experimentation but may be considered in the future.

4. Epochs: Final trials were run at 30,000 epochs to ensure behavior was not continuing to improve after training ended. In cases where it looked like more improvement was possible, trials were run at 50,000 and 100,000 epochs to ensure that 30,000 was an acceptable place to stop. In many cases, training reaches a stable value around 10,000 or 20,000 epochs, but the model can continue to improve mildly when allowed to train longer. Training is recommended for minimally 10,000-15,000 epochs, but if there is time for 30,000 epochs, allowing the model to continue to train does not lead to overfitting.

5. Number of hidden layers: When training with two hidden layers, the results were significantly worse than the default of 3 layers, but 4 layers did not show enough improvement to account for the extended training time needed for 4 layers.

4.3 Results and Discussion

The final version of the composite CNN-DeepONet model achieves high accuracy results, with the base ten logarithm of the MSE generally less than -2 within 30,000 epochs of training. While the loss for training the composite CNN-DeepONet is computed with a mean square error, the effectiveness of the neural operator solutions are evaluated throughout this section, with the L^2 norm of the relative error at each time step,[46]

$$\mathcal{D}\big(\Upsilon(\vec{x},\vec{y},t;\Omega)\big) = \frac{\sum \ \sum_x \sum_y (v_n - \hat{v}_n)^2}{\sum_n \sum_x \sum_y (v_n)^2} \tag{24}$$

to emphasize the changes when different features are added to improve the efficiency with our limited data set. The full CNN-DeepONet, on average, achieves a final relative error less than half of a percent (0.5%). Figure 29 shows the individual relative error of samples in the training and testing sets, along with the average relative error in black. Particularly, in the training set, the majority of the samples stay under half a percent of relative error across all time steps, with just a few outliers raising the average. The test set was specifically curated to include difficult samples, so the split of well performing and poor performing solutions is more balanced. Overall, the order solutions start strong and deteriorate as time goes on, very poorly in certain cases. Due to less fine detail in the temperature solutions, these results have less error for longer and only rise as dramatically as the order errors at the very end of the evolution.

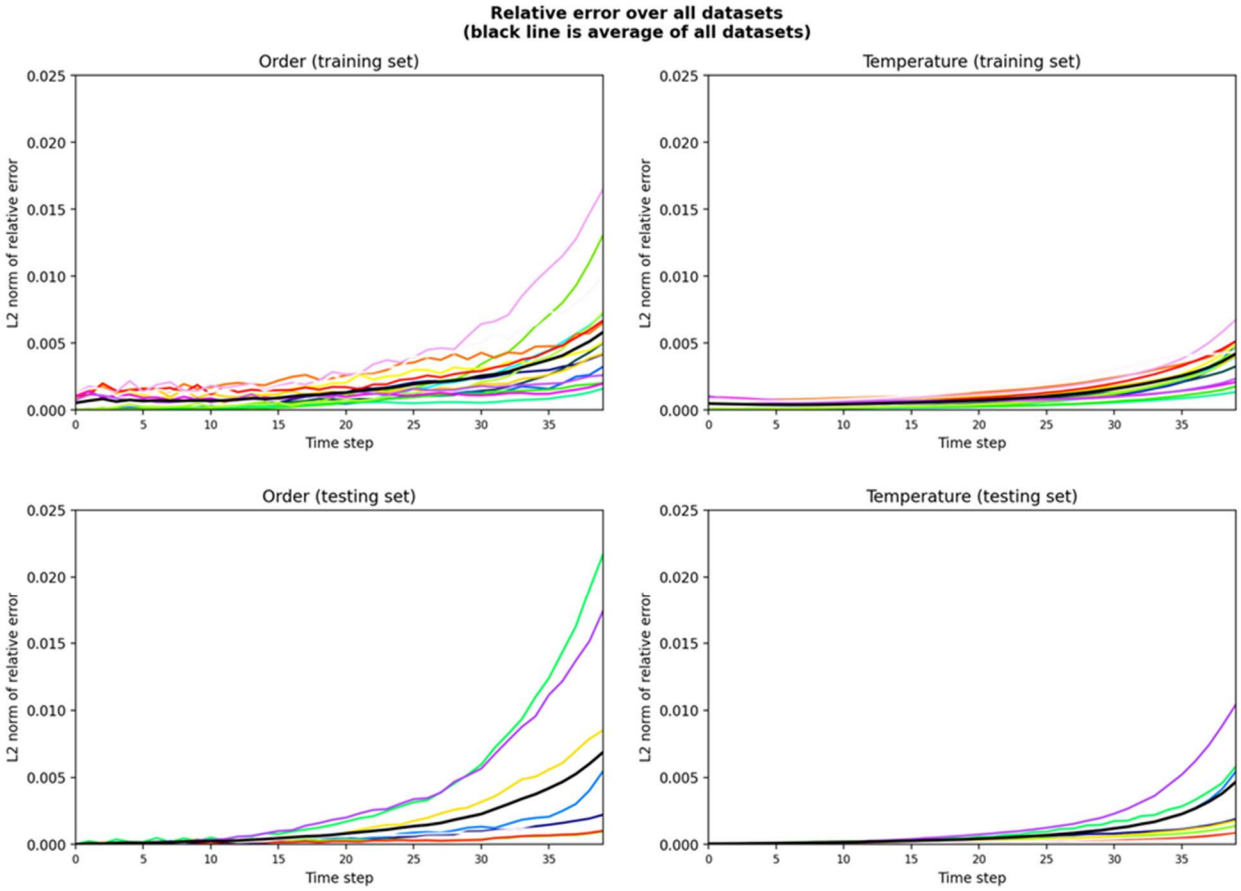


Figure 29. Sample relative error results from a training (top) and testing (bottom) set. The average across the set is the black line. The majority of the sample perform better than the average, which includes a strong bias from few outliers. The order solutions have finer details, which lead to more relative error than the temperature solutions. As expected, the errors compound as the grains evolve, leaving more error at the end of the solutions.

Figure 30 and Figure 31 show the test set outliers in relative error at the last time step. In general, the model struggles to train the irregular shapes that form on the nodes (top right Figure 31) and performs more efficiently on symmetrical shapes. Figure 31 also highlights poor performance in

the temperature solution where $\dot{Q}_0 < 0$. For the most part, the model is trained to expect important changes near the edges of the grain, and internal temperature changes in the cooling system add additional features for the model to learn.

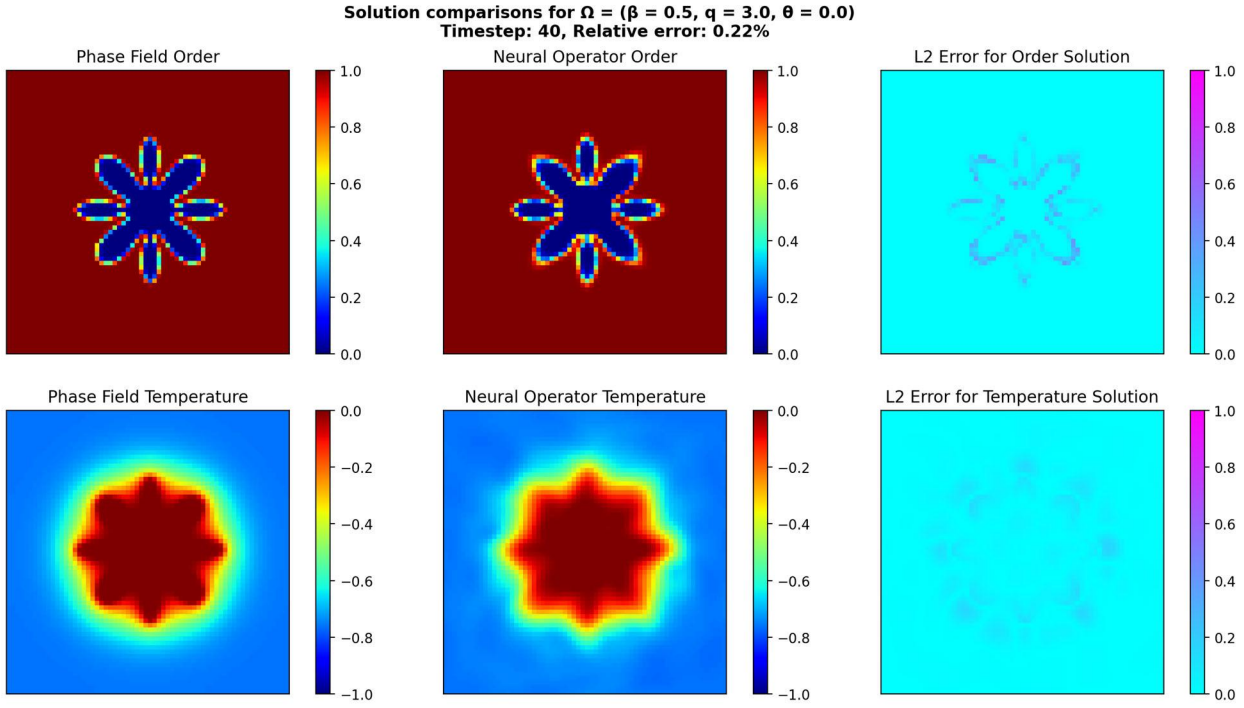


Figure 30. A snapshot solution of a highly accurate solution with only 0.22% relative error at the final timestep. Phase field targets are on the left for order and temperature parameters, while the network answer is in the center column, and the error in each solution is shown at the right. Solutions in the warming system regime, where temperatures stay uniform in the center of the grain, perform better than those in the cooling system regime. Also, highly symmetric solutions are easier for the model to solve. The majority of the error occurs on the interface in the order solution.

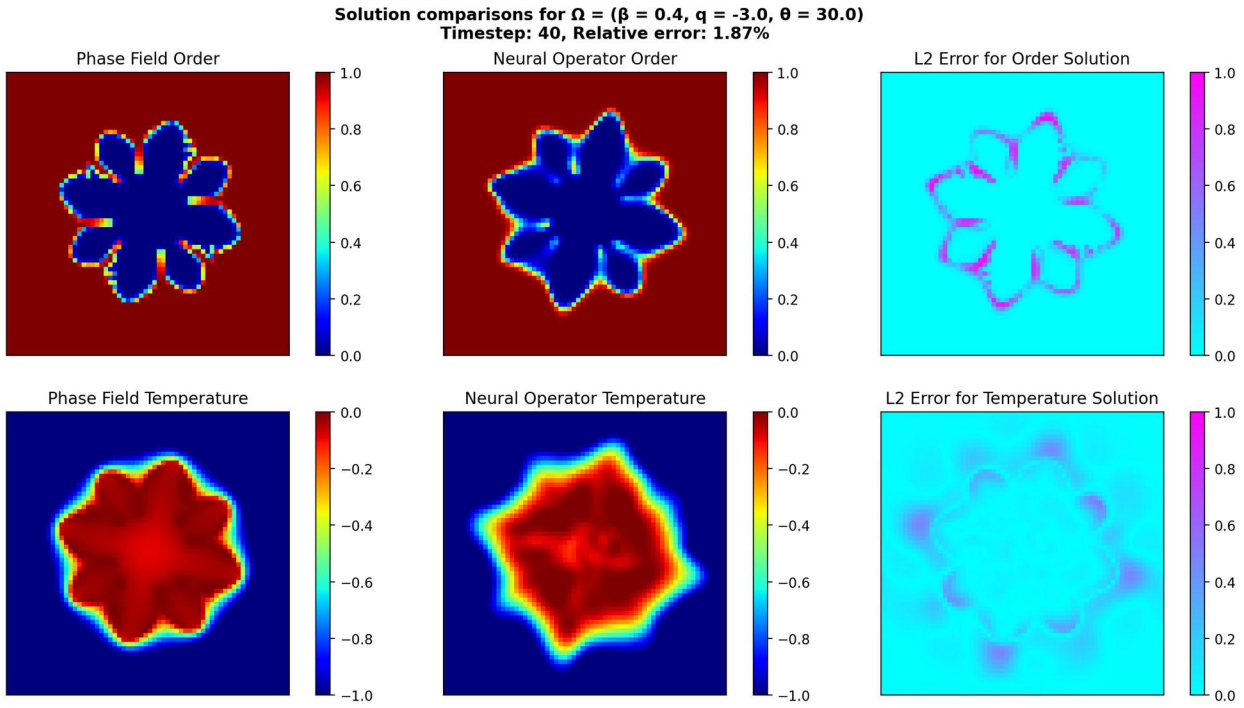


Figure 31. A snapshot solution of a solution with almost 2% relative error at the final timestep, showing the expected answer on the left, the network answer in the center column and the error between them on the right. Solutions in the cooling system regime add a layer of complexity due to the lack of uniformity inside the grain. This sample is also more difficult, due to the fine irregularities on the nodes observed in the top left corner. This solution shows much more error on the interface and increased inaccuracy in the temperature solutions.

To determine the efficacy of each model component added to support the limited information regime, the components were removed one at a time to compare the results. Starting with the separated simulations initially used to increase the size of the training set, Figure 32, direct comparisons reveal that both models reach similar loss and relative error values. When the training data are not separated, the convergence is slower but still achieves high accuracy within 30,000 epochs. Leaving this feature in allows the model to train in fewer epochs and is recommended. A second data preparation adaptation, the use of three time steps for the branch inputs instead of the normal DeepONet requirement of just one, decreases the learning capability of the CNN-DeepONet. Figure 33 shows both quick convergence and consistently lower MSE and relative error values. Upon observing this, the single time step branch input became the default for the full model which all other models are compared against. While this benefit was important before the addition of the CNN layer, the additional information appears to artificially inflate the loss and may distract the DeepONet from finding more important places to improve its neural operator. The final data preparation adaptation, the expanded feature trunk input, also yields noticeable improvement both to MSE and relative error values, as seen in Figure 34. Examination of sample solutions indicates that the inclusion of the expanded feature trunk encourages the background of the order solution to remain constant, as is expected for these solutions. On the boundaries alone, the neural operator solution achieves near machine precision accuracy when trained with the feature expansion trunk, which allows the omission of an extra term in the loss function to enforce the periodicity of the boundaries. Including fewer terms in the loss function increases training efficiency.

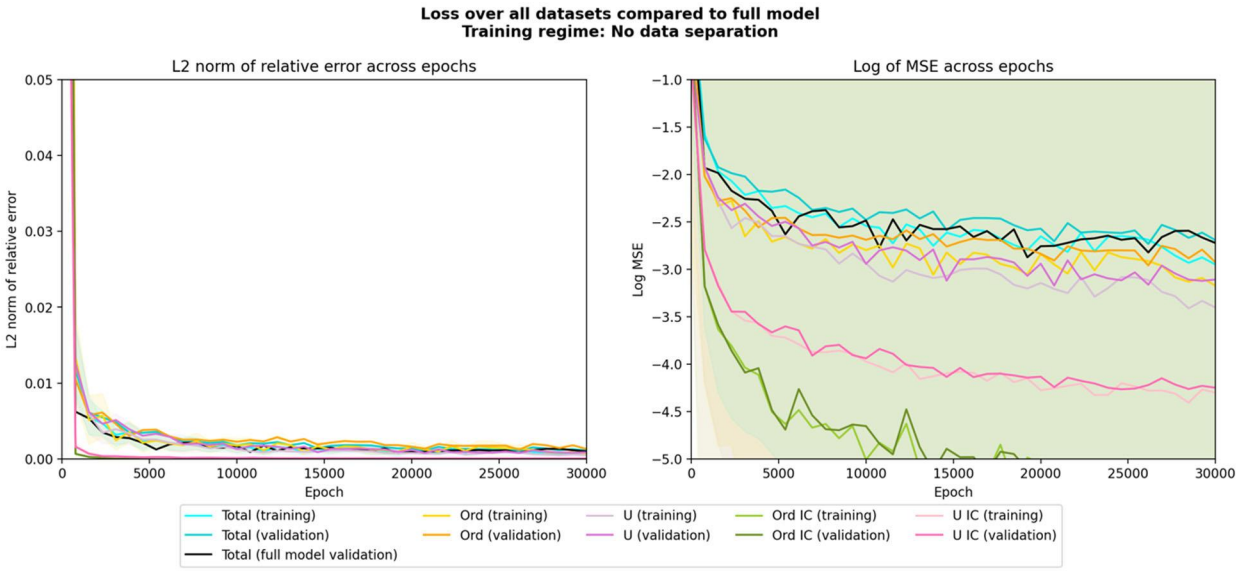


Figure 32. Loss of the model where input samples are not separated into two data sets. While convergence is slightly slower, similar accuracy is eventually achieved. Leaving this feature in allows a reduced training time.

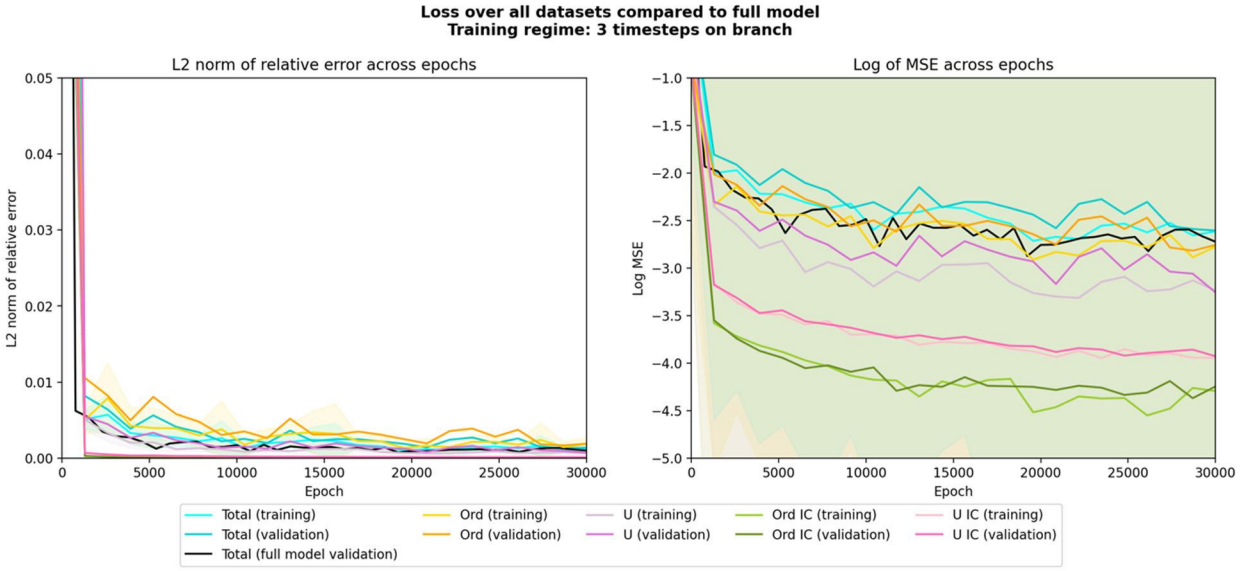


Figure 33. Loss of the model where input samples contain three time steps. This result shows significantly more difficulty in achieving high accuracy in matching the initial conditions of the order solution, which raises the overall error. Because similar errors are achieved when only one time step is included in the branch, there is no need to add additional data that can slow training.

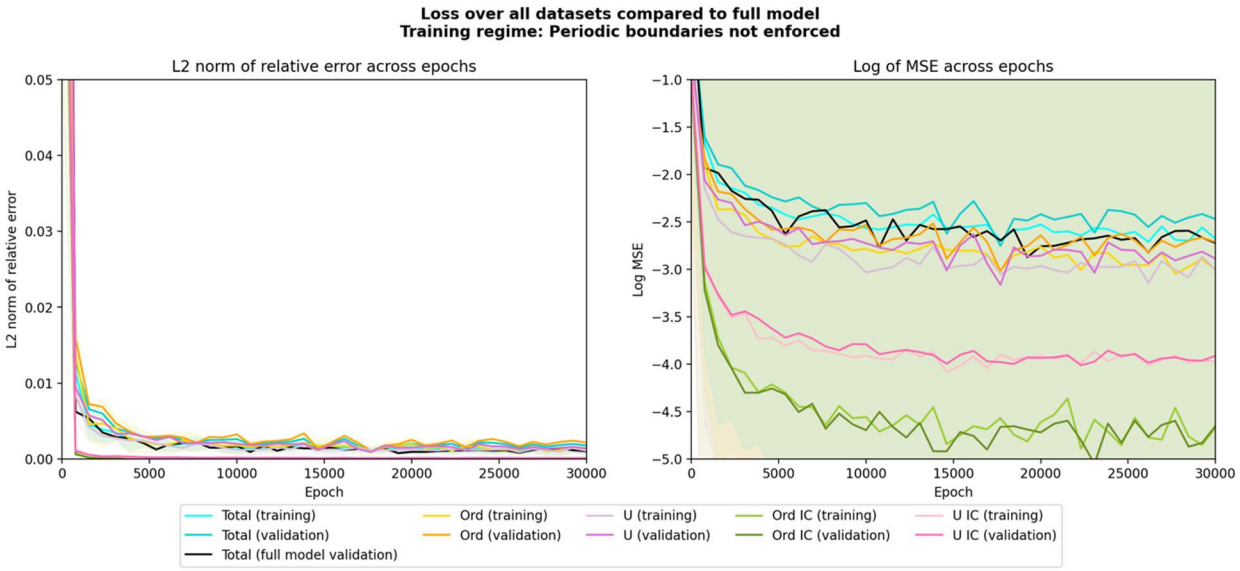


Figure 34. Loss of the model when periodic boundaries are not enforced. There is significant enough improvement with the Fourier basis trunk that this feature is shown to add accuracy to the model and is a strong addition. Trigonometric identities can be implemented to reduce the size of the trunk and increase the overall efficiency of the model.

The addition of the CNN that trains in tandem with the DeepONet both increases the speed with which the training converges and brings improved accuracy to the solutions (Figure 35), indicating that the coordinates of the solutions that are omitted from the targets are still useful for training. To fully consider the value of the CNN, this model was also trained with a modified learning rate decay schedule to improve convergence, but no results could compete with the value added by combining the CNN with the DeepONet. Efforts to use larger targets (128×128 grids) yielded similar results to our base model, Figure 36, but the training can take up to three times as longer. Because our neural network tool is designed to support - not replace – the phase field modeling capability, higher resolution solutions are not a priority for success, and the decrease in efficiency leaves this version with similar problems to fully accurate phase field modeling. Finally, it was observed that the inclusion of the self-adaptive weights, shown in Figure 37, only lead to minor improvements, mostly allowing for consistent stronger prediction of the order (φ) solution. This result is surprising, because the comparison model is constructed by setting $\lambda(w)$ to 1 in all cases, which cannot account for the fine detail at the interfaces. This may be because the mask function used for the weights eventually converges to 1, which aligns with the no weights case. Other masks were attempted, but similar results occurred in most cases. Snapshot inspections of individual cases (Figure 38) show they are slightly worse when weights are not included, but the changes are not significant.

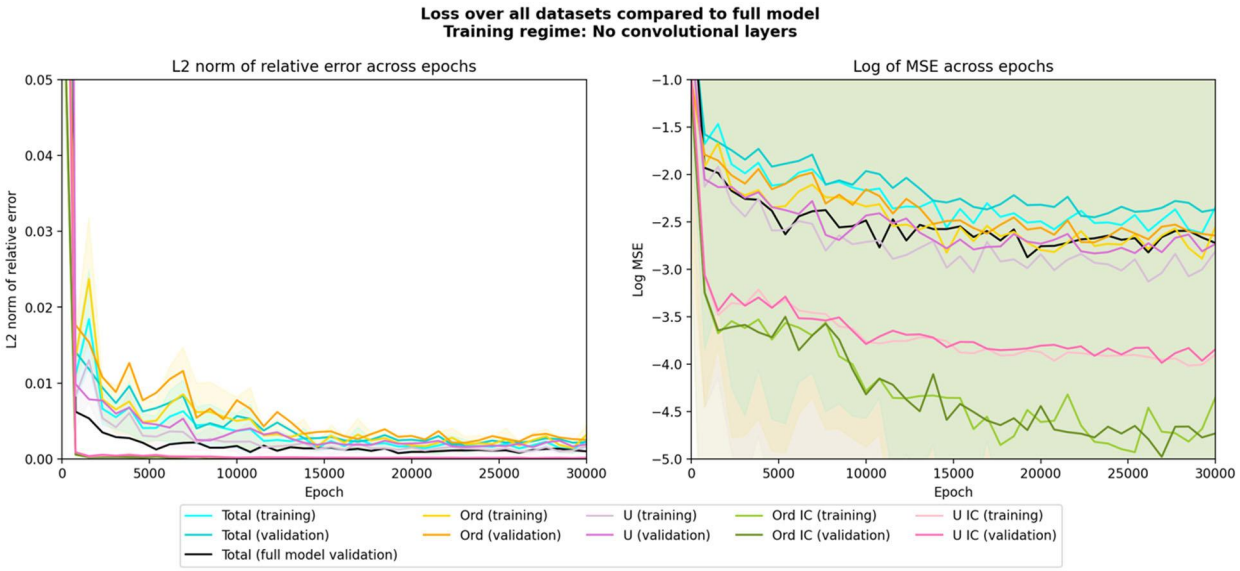


Figure 35. Loss of the model when the CNN layer is removed and the coarsened targets are used as inputs. This feature both simplifies data preparation and yields the highest increase in training accuracy.

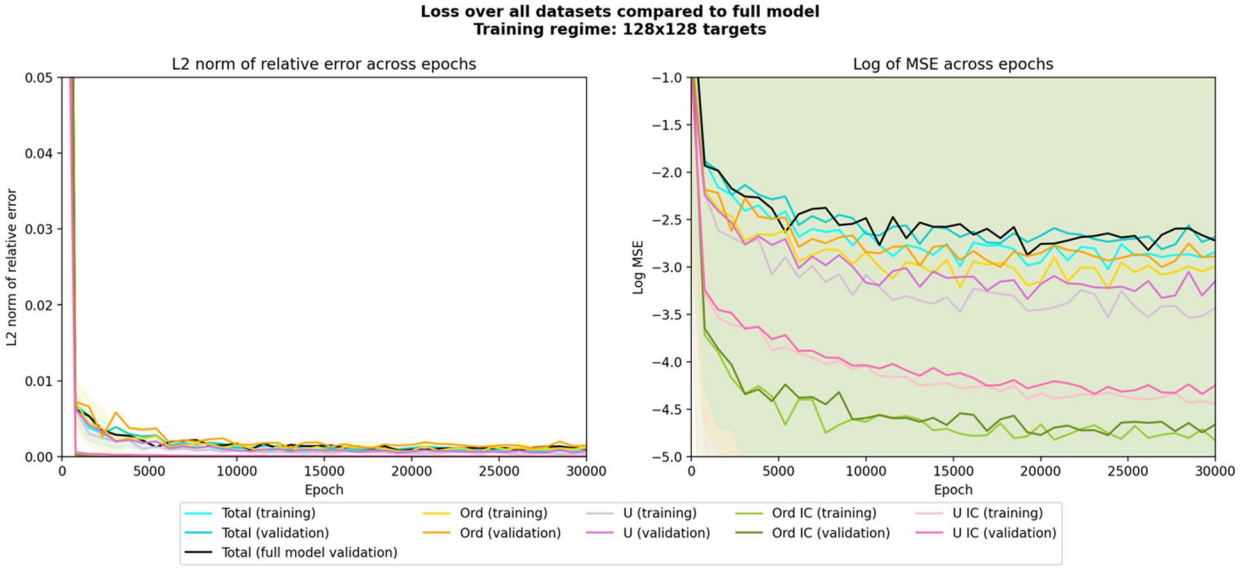


Figure 36. Loss of the model when higher resolution targets are used. While the accuracy remains similar, this change requires up to 3 fold more training time. If desired, this approach can be used but will slow down searches of the parameter space.

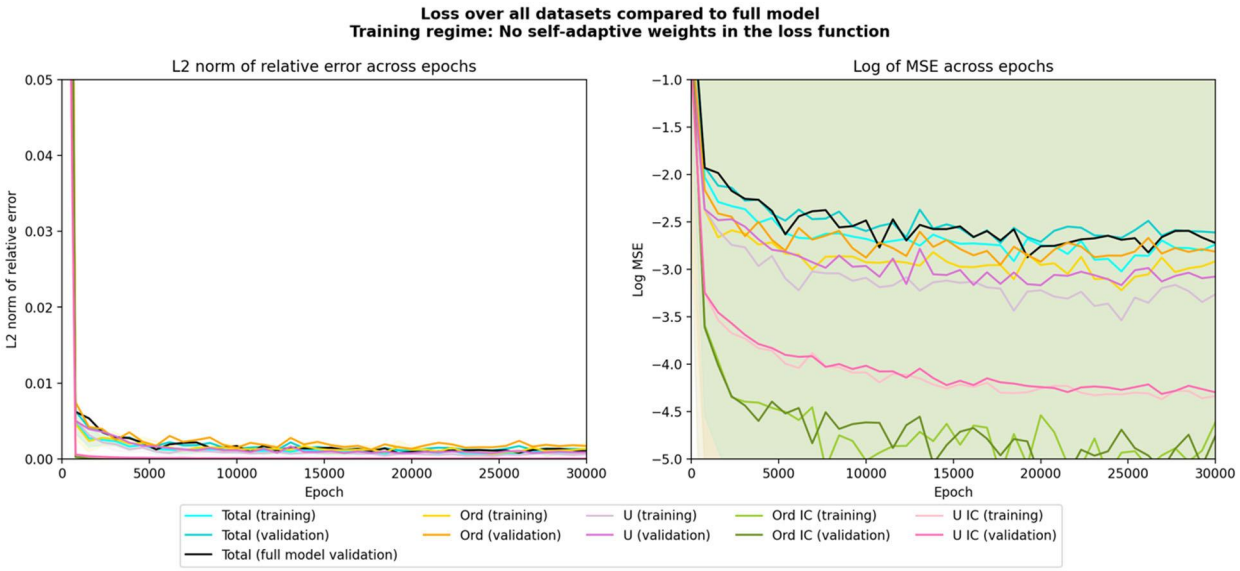


Figure 37. Loss of the model when self-adaptive weights are used. Overall losses here are similar to when the self-adaptive weights are not used at all.

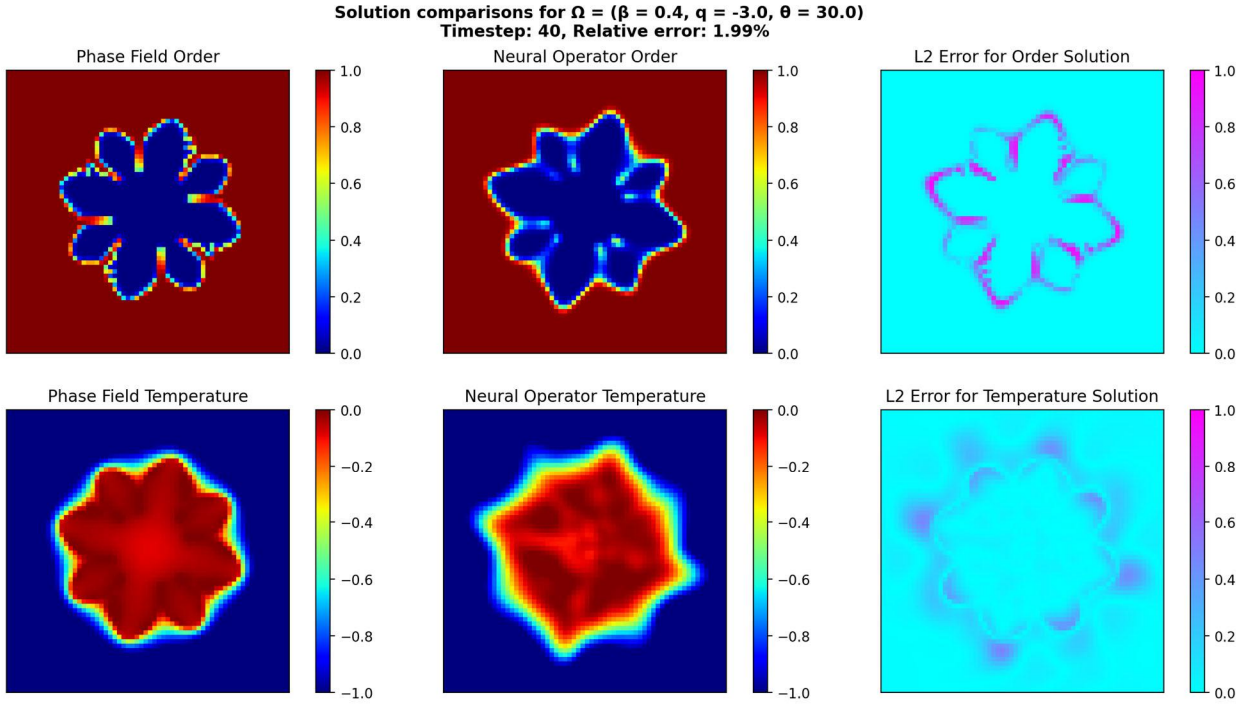


Figure 38. Snapshot solution of the same sample as Figure 31 but without the use of self-adaptive weights. Results appear to be uniformly worse, but only by a fraction of a percent.

4.3.1 Data Driven and Physics Informed Results

In applications such as this, with access to a model that corresponds to available data, an immediate question is: Can that model be used directly to improve training? The following section describes two methods that were tested for incorporating physics aspects of the phase field model into the training loss as a means to improve performance.

4.3.1.1 Numerical Considerations

Figure 39 and Figure 40 show plots of the output from a finite element simulation of grain growth over a small portion of the domain around an interface region. As can be seen in the figures, the value of the time derivative of the order variable varies by over three orders of magnitude across the narrow interface region but is at or near zero over much of the domain. This figure highlights both the large discrepancy in scale present in computations of the physics residual terms as well as the high degree of sensitivity to the order surface present. Both of which present challenges to effectively using the physics residuals in training.

For computation of the free energy functional Figure 41 shows a similar pattern to the physics residuals. The contribution to the integral is zero for much of the domain, except for the narrow interface region, where the energy expression varies over two orders of magnitude. Thus, while less expensive to compute, the free energy loss presents similar challenges as the physics residual loss.

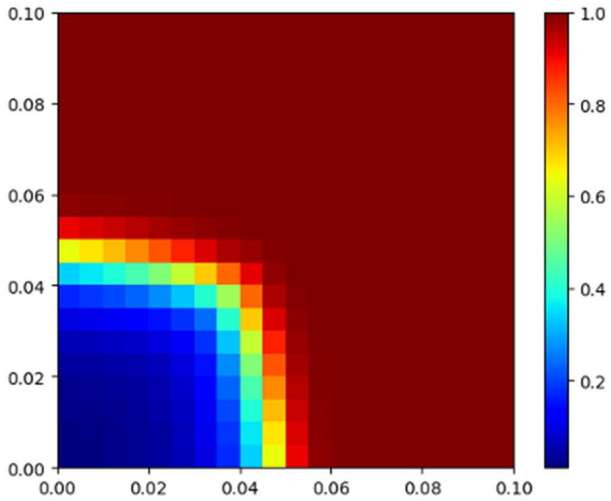


Figure 39. The order variable values in an interface region from an example finite element simulation.

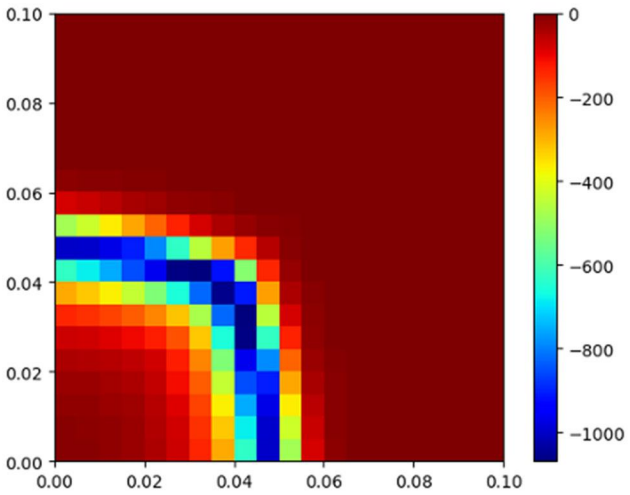


Figure 40. The corresponding partial derivatives of the order variable solution with respect to time.

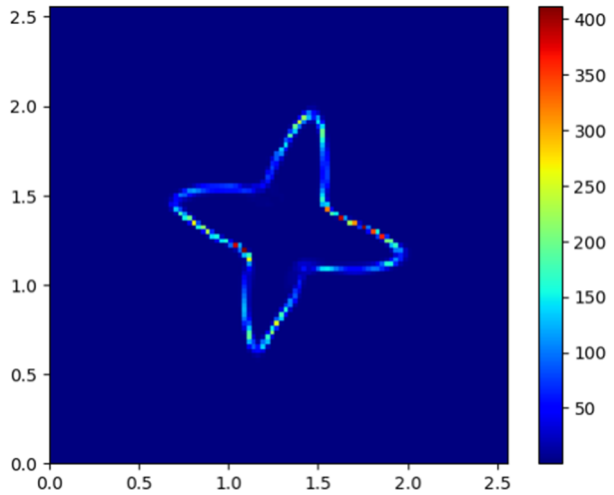


Figure 41. Plot showing evaluations of the free energy functional over the domain for an example simulation.

Computation of both the physics residual F_u and F_ϕ as well as the free energy functional $\mathcal{F}(v)$ require computing

$$\theta = \arctan\left(\frac{\partial\varphi}{\partial y} \Big/ \frac{\partial\varphi}{\partial x}\right), \tag{25}$$

which is only well defined in the interface regions. For those regions where the order variable is constant, θ is undefined. The following approximation was used for all computations, to ensure a value was defined on the whole domain:

$$\theta = \frac{\pi}{2}\left(1 - \text{sign}\left(\frac{\partial\varphi}{\partial x}\right)\right) + \frac{\pi}{2}\left(\text{sign}\left(\frac{\partial\varphi}{\partial x}\right) + 1\right)\left(1 - \text{sign}\left(\frac{\partial\varphi}{\partial y}\right)\right) + \text{sign}\left(\frac{\partial\varphi}{\partial x}\right)\arctan\left(\left.\frac{\partial\varphi}{\partial y}\right/\left(\left|\frac{\partial\varphi}{\partial x}\right| + 1\times 10^{-12}\right)\right) \quad (26)$$

The value of the integral $\mathcal{F}(v)$ (Equation 22) was approximated using Gauss-Legendre quadrature, with degree 100 at which the approximation was found to converge. Within training, use of the same evaluation points can quickly lead to overfitting, therefore, at each step, quadrature was performed over the grid

$$[x_{min} + \delta_x, x_{max} + \delta_x] \times [y_{min} + \delta_y, y_{max} + \delta_y] \quad (27)$$

for random shifts δ_x and δ_y. The shifts were uniformly distributed from $[-\varepsilon_d, \varepsilon_d]$, where each of the min and max values for x and y are ε_d from the boundary of the domain, such that the integral is always approximated over the interior of the domain. Note that the energy functional was zero in the region near the boundary for all simulations, and ε_d was chosen such that there was no change in the free energy approximation.

Training with both the physics residual and free energy losses resulted in significant increases to the computational time. To alleviate the computational burden, a two stage training approach was employed. First the DeepONet was trained just to minimize the loss $\mathcal{L}_s$ without the physics term to convergence. Then using the same test train split, the physics term was introduced, and the model was trained for additional epochs on the full loss.

To compute α values for weighting the physics loss terms, a relative computation was made with respect to the solutions trained on $\mathcal{L}_s$, where, for $\mathcal{L}_p$, the loss associated with the physics terms α was computed as

$$\alpha = \alpha_{rel}\frac{\mathcal{L}_s}{\mathcal{L}_p} \quad (28)$$

for α_{rel}, the target relative difference between the contribution of the physics term and the rest of the loss.

4.3.1.2 Physics-Informed Training Results

Training with the physics residual loss $\mathcal{L}$, as expanded in the Physics Residuals section above, resulted in worse performance in all experiments. Due to the computational cost, training experiments were performed on just three simulations. Introduction of the physics residuals into the loss destabilized convergence of the DeepONet and resulted in worse fits to the training data for all values of α_{rel}, which was reduced until there was no effect of the physics residuals on the loss. Training on the full training set with the physics residuals was estimated to take on the order of hundreds of hours of computational time, so no such experiments were performed.

Computational time for training with the free energy loss on the full train split was more reasonable, with ten thousand epochs requiring between two and three days. However again, α_{rel} was reduced until no effect on training was seen and produced worse fits of the model on both the train and test set for all values used.

5.0 Conclusions

Based on surrogate experiments with Cu, high-temperature optical microscopy is still considered a promising avenue to pursue to witness and capture real-time crystallization of U under a controlled environment. Studies are ongoing to optimize our microscope heating stage systems for U metal grain growth imaging and potential realization of in-situ impurity behavior. Enhanced EBSD will be key to quantitative grain analysis of α-U matrices and will be advanced, as described above, with future efforts. Our higher magnification optical configurations begin to meet the working scale of our phase field models; our objective remains to mimic solidification conditions of interest to actinide casting and phase field simulations then compare crystal morphologies to inform and validate the models.

Reported here is the advancement of a quantitative phase field model to simulate growth of multiple dendritic U crystals collocated with preexisting impurity particles comprised of uranium monocarbide. More work is needed in the actinide metallurgical community to refine and measured known and unknown thermophysical properties of U, respectively, under applicable conditions. Current model parameters draw from available thermodynamic and kinetic properties, which may have large uncertainties or are altogether unknown. With this phase field modeling platform, we systematically simulated the effect of varied model parameters on the evolution of U dendrites and generated a collection of microstructural data to develop and train a machine learning neural network for prediction of U solidification morphologies. Under our simulation conditions, the presence of impurity particles has a relatively insignificant effect on final grain structure when cooling rates are high but could cause dendrite arm branching and grain boundary pinning when the cooling rate is low and dendrites grow slower. As expected, cooling rate ($\dot{Q}_0$) has a significant effect on dendrite morphology and growth kinetics. Larger (-$\dot{Q}_0$) results in a higher nucleation rate and faster and more heterogeneous dendrite growth. The size and morphology of the final grains strongly depend on dendrite nucleation density and locations. As described above, the current modeling capability assumes that impurity particles are present at the moment of U solidification onset and remain fixed in their set locations as the simulation evolves. The next step in developing this model will be to allow for a two-way coupling interaction between the solidification front of the growing U crystal and movable impurity particles that may be impacted by the phase change boundary. Crystallographic transitions in the solid state should also be explored during cooling.

Finally, a deep operator network (DeepONet) was optimized to receive inputs from the phase field model and trained, with the aim of eventually providing surrogate support at a low cost. The existing work is well suited to further explore parameter spaces for the single grain phase field model equations, if there is time and interest in generating new training data sets to explore different parameters. Based on the current understanding of adapting DeepONets to perform well in a low data regime, the existing work could be extended and applied to the multi-grain simulations to generate neural operator solutions in a more applied framework for actinide solidification.

6.0 References

1. Abrecht, D.G., et al., *Modeling Induction Stirring and Particle Tracking in Molten Uranium*. 2021: United States. p. Medium: ED; Size: 44 p.
2. Singh, R.K., et al., *Computational studies of impurity migration during induction stirring of molten uranium.* Computational Materials Science, 2023. **229**: p. 112386.
3. *Uranium Processing and Properties*. 2013, New York: Springer.
4. Olszta, M.J., et al., *A Comparison of Carbon Impurities in Pre- and Post-Melt Uranium; Part 1: Scanning Electron Microscopy Analysis.* Journal of alloys and compounds, 2022. **925:** p. 166584.
5. Olszta, M.J., J.F. Corbey, and D.D. Reilly, *A Comparison of Carbon Impurities in Pre- and Post-Melt Uranium; Part 2: Scanning/Transmission Electron Microscopy Analysis.* Journal of Alloys and Compounds, 2022. **925**: p. 166583.
6. Grenthe, I., et al., *The Chemistry of the Actinide and Transactinide Elements: Uranium*. 4th ed. Vol. 1. 2010, The Netherlands: Springer.
7. Fairman, H.E. and A. Kelly, *Metallography and Microstructures of Uranium and Its Alloys*, in *Metallography and Microstructures*, G.F. Vander Voort, Editor. 2004, ASM International.
8. Berthinier, C., et al., *Experimental kinetic study of oxidation of uranium monocarbide powders under controlled oxygen partial pressures below 230°C.* Journal of Nuclear Materials, 2013. **432**(1): p. 505-519.
9. Stobbs, J.J. and I. Whittle, *The influence of impurities on the oxidation of uranium in carbon dioxide between 450° and 650° C.* Journal of Nuclear Materials, 1966. **19**(2): p. 160-168.
10. Zalkind, S., et al., *Influence of carbon on the tensile properties of U-0.1w%Cr.* Journal of Materials Science Letters, 2002. **21**(7): p. 551-553.
11. Davies, D.M. and J.W. Martin, *The effect of inclusions on the fracture of uranium.* Journal of Nuclear Materials, 1961. **3**(2): p. 156-161.
12. N. C. Jessen, J., *Melting and Casting Uranium and Uranium Alloys*. 1982, Oak Ridge Y-12 Plant: Oak Ridge, TN.
13. Chen, D., et al., *The influence of the carbon addition on the microstructure of a U-Nb alloy and the features of Nb2C inclusions.* Metall. Res. Technol., 2018. **115**(3): p. 309.
14. Ardell, A.J., *Precipitation hardening.* Metallurgical Transactions A, 1985. **16**(12): p. 2131-2165.
15. Olofson, C.T., G.E. Meyer, and A.L. Hoffmanner, *Processing and Applications of Depleted Uranium Alloy Products*. 1976, Battelle: Columbus, Ohio.
16. Wilkinson, W.D., *Uranium Process Metallurgy, vol. 1; Uranium Corrosion and Alloys, vol. 2*. Uranium Metallurgy. Vol. 1 & 2. 1962, New York: Intescience.
17. Mallett, M.W., A.F. Gerds, and H.R. Nelson, *The Uranium-Carbon System.* J. Electrochem. Soc., 1952. **99**: p. 197.
18. Börzsönyi, T., et al., *Dendrites Regularized by Spatially Homogeneous Time-Periodic Forcing.* Physical Review Letters, 1999. **83**(14): p. 2853-2856.
19. Chen, L.Q., *Phase-field models for microstructure evolution.* Annual Review of Materials Research, 2002. **32**: p. 113-140.
20. Karma, A. and W.J. Rappel, *Quantitative phase-field modeling of dendritic growth in two and three dimensions.* Physical Review E, 1998. **57**(4): p. 4323-4349.
21. Steinbach, I., *Phase-field models in materials science.* Modelling and Simulation in Materials Science and Engineering, 2009. **17**(7).
22. Takaki, T., et al., *Unexpected selection of growing dendrites by very-large-scale phase-field simulation.* Journal of Crystal Growth, 2013. **382**: p. 21-25.

23. Sohn, I. and R. Dippenaar, *In-Situ Observation of Crystallization and Growth in High-Temperature Melts Using the Confocal Laser Microscope.* Metallurgical and Materials Transactions B, 2016. **47**(4): p. 2083-2094.
24. Gao, X., et al., *Investigation on the formation mechanism of non-metallic inclusions in high-aluminum and titanium-alloyed Ni-based superalloy.* Vacuum, 2020. **177**: p. 109409.
25. Jung, S.S. and I. Sohn, *Crystallization Control for Remediation of an FetO-Rich CaO–SiO2–Al2O3–MgO EAF Waste Slag.* Environmental Science & Technology, 2014. **48**(3): p. 1886-1892.
26. LaHaye, N.L., et al., *Experimental Results for Carbon Monoxide Inoculation of Depleted Uranium*. 2020, Pacific Northwest National Laboratory: Richland, WA.
27. Surbella, R.G., et al., *On Direct Carbon-Monoxide Inoculation of Molten Uranium: A Potential Strategy for Developing Working Reference Materials*. 2020, Pacific Northwest National Laboratory: Richland, WA.
28. Athon, M.T., et al., *Quantitative morphological characterization of carbide inclusions in uranium metal.* Journal of Nuclear Materials, 2022. **558**: p. 153370.
29. Bair, J.L., et al., *Phase field model of uranium carbide solidification through a combined KKS and orientation field approach.* Journal of Physics: Condensed Matter, 2019. **31**(12): p. 125901.
30. Ofori-Opoku, N. and N. Provatas, *A quantitative multi-phase field model of polycrystalline alloy solidification.* Acta Materialia, 2010. **58**(6): p. 2155-2164.
31. Jokisaari, A.M., et al., *Phase field benchmark problems for dendritic growth and linear elasticity.* Computational Materials Science, 2018. **149**: p. 336-347.
32. Borzsonyi, T., A. Buka, and L. Kramer, *Effect of the anisotropic surface tension, crystallization kinetics, and heat diffusion on nonequilibrium growth of liquid crystals.* Physical Review E, 1998. **58**(5): p. 6236-6245.
33. Mcfadden, G.B., et al., *Phase-Field Models for Anisotropic Interfaces.* Physical Review E, 1993. **48**(3): p. 2016-2024.
34. *Uranium – Melting Point – Boiling Point*. Available from: https://www.nuclear-power.com/uranium-melting-point-boiling-point/.
35. *Uranium*. Available from: https://en.wikipedia.org/wiki/Uranium.
36. Singman, C.N., *Atomic volume and allotropy of the elements.* Journal of Chemical Education, 1984. **61**(2): p. 137.
37. Felsberger, L. and P.S. Koutsourelakis, *Physics-constrained, data-driven discovery of coarse-grained dynamics.* Communications in Computational Physics, 2019. **25**: p. 1259-1301.
38. Han, J., A. Jentzen, and W. E, *Solving high-dimensional partial differential equations using deep learning.* Proceedings of the National Academy of Sciences, 2018. **115**(34): p. 8505.
39. Lu, L., et al., *Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators.* Nature Machine Intelligence, 2021. **3**(3): p. 218-229.
40. Raissi, M., P. Perdikaris, and G.E. Karniadakis, *Numerical Gaussian Processes for Time-Dependent and Nonlinear Partial Differential Equations.* SIAM Journal on Scientific Computing, 2018. **40**(1): p. A172-A198.
41. Raissi, M., P. Perdikaris, and G.E. Karniadakis, *Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations.* Journal of Computational Physics, 2019. **378**: p. 686-707.
42. Sirignano, J. and K. Spiliopoulos, *DGM: A deep learning algorithm for solving partial differential equations.* Journal of Computational Physics, 2018. **375**: p. 1339-1364.
43. Yang, K., et al., *Self-supervised learning and prediction of microstructure evolution with convolutional recurrent neural networks.* Patterns, 2021. **2**(5): p. 100243.
44. Kautz, E., et al., *An image-driven machine learning approach to kinetic modeling of a discontinuous precipitation reaction.* Materials Characterization, 2020. **166**: p. 110379.

45. Ma, W., et al., *Image-driven discriminative and generative machine learning algorithms for establishing microstructure–processing relationships.* Journal of Applied Physics, 2020. **128**(13): p. 134901.
46. Oommen, V., et al., *Learning two-phase microstructure evolution using neural operators and autoencoder architectures.* npj Computational Materials, 2022. **8**(1): p. 190.
47. Wang, S., H. Wang, and P. Perdikaris, *Improved Architectures and Training Algorithms for Deep Operator Networks.* Journal of Scientific Computing, 2022. **92**(2): p. 35.
48. Lu, L., et al., *A comprehensive and fair comparison of two neural operators (with practical extensions) based on FAIR data.* Computer Methods in Applied Mechanics and Engineering, 2022. **393**: p. 114778.
49. McClenny, L. and U. Braga-Neto *Self-adaptive physics-informed neural networks using a soft attention mechanism*. 2022.

Pacific Northwest National Laboratory

902 Battelle Boulevard
P.O. Box 999
Richland, WA 99354

1-888-375-PNNL (7665)

www.pnnl.gov