UC Berkeley
UC Berkeley Electronic Theses and Dissertations

Title
Uncertainty Quantification and a Model Reduction Procedure of a Laser Fusion Additive Manufacturing Simulation

Permalink
https://escholarship.org/uc/item/2v09h4k4

Author
Castrillon, Nicolas

Publication Date
2022

Peer reviewed|Thesis/dissertation

eScholarship.org Powered by the California Digital Library
University of California

Uncertainty Quantification and a Model Reduction Procedure of a Laser Fusion Additive Manufacturing Simulation

by

Nicolas Castrillon

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering - Mechanical Engineering

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Tarek I. Zohdi, Chair
Professor Shawn Shadden
Professor James Sethian

Fall 2022

Uncertainty Quantification and a Model Reduction Procedure of a Laser Fusion Additive Manufacturing Simulation

Copyright 2022
by
Nicolas Castrillon

Abstract

Uncertainty Quantification and a Model Reduction Procedure of a Laser Fusion Additive Manufacturing Simulation

by

Nicolas Castrillon

Doctor of Philosophy in Engineering - Mechanical Engineering

University of California, Berkeley

Professor Tarek I. Zohdi, Chair

Additive Manufacturing (AM) is an umbrella term to describe processes that build a final part by fusing together raw material layer-by-layer according to a digital 3D CAD model design. This emerging but promising suite of technologies now have the capability of generating builds in the factory floor due to recent advances in precision, process repeatability, and a vast material library, which includes thermoplastics, metals, ceramics, and composites. AM now allows for the production of highly customizable parts with complex geometries in production-ready material grades. The use of numerical methods of AM processes in industry is an economical approach for optimizing technologies efficiently and in an environmentally conscious way.

A coupled Discrete Element–Finite Difference (DE-FD) model is leveraged for solving the thermal field of 316L stainless steel in a Selective Laser Sintering (SLS) manufacturing process. The DE method models the powder particles as discrete, lumped capacitance bodies resting on a solid substrate, which is discretized in space in a uniform grid of voxels. An additional heat-flux condition enforces the coupling between both domains. Data from experiments in literature serve to validate the present model by measuring the material melt pool shape from single-track simulations.

Two practical data-driven techniques are then presented for bridging the numerical simulation into an industrial setting. (1) A statistical Uncertainty Quantification (UQ) study of the material melt pool shape due to random powder layer thickness variations is explored. A voxel-wise powder depth heuristic modulates the local amount of laser energy that is able to penetrate through the powder layer down to the substrate. (2) A Reduced Order Model (ROM) is built via a Proper Orthogonal Decomposition (POD) of the thermal field data from a single run. The POD procedure is able to compress the data from the full simulation from 588,060 degrees of freedom to about 6 degrees of freedom with a 99.8% data recovery upon reconstruction with significant speed-up in the time integration of the simulation.

To Adriana María, Lucía María, and our next Baby, who is already on the way.

Contents

Contents **ii**

List of Figures **iv**

List of Tables **viii**

1 Introduction **1**
1.1 Overview of Numerical Modeling in Engineering 1
1.2 Quantifying Uncertainty and Process Variation 2
1.3 Surrogate Modeling and Reduced Order Modeling 3
1.4 Organization of the Present Work 4

2 Background and Theory **5**
2.1 An AM Application: Selective Laser Sintering 5

3 Numerical Problem **12**
3.1 Invoking Continuum Mechanics 12
3.2 A Thermal Model for SLS 16
3.3 Model Validation and Verification 32
3.4 Parameter Studies 37

4 Quantifying Uncertainty and Variability **47**
4.1 Thermal Modeling and Uncertainty Quantification of Tool for Automated Garment Assembly 47
4.2 Sensitivity Computation of Integral Quantities with Moment Derivatives . . 65
4.3 Variability of Targeted Material Thermal Responses to Laser-Induced Heating in AM 84
4.4 UQ in the SLS Process 105

5 Reducing the Model **114**
5.1 A Proper Orthogonal Decomposition 114
5.2 Applying a POD to the SLS Thermal Data 115

6 Conclusions **130**

Bibliography **133**

List of Figures

2.1 Schematic of an AM system [38]. 6
2.2 Components of an SLS system [37]. 7
2.3 SLS process parameters by [135]. 8
2.4 In-process foil curling due to material temperature differentials between layers [92]. 9

3.1 **Left:** A representation of the 2D Gaussian-shaped laser beam and the radial distance d to an arbitrary particle. **Right:** The depth distance $\hat{z}$, taken from the top of the powder layer. 20
3.2 Example of an iteration of Valera's 2D particle deposition algorithm [143]. Particles in the advancing front `F` are in black; **(a)** a particle size is sampled from a distribution; **(b)** a particle in `F` is chosen at random as s_0 with its neighborhood s_v, but there is no feasible locations for a new particle; **(c)** a new s_0 is chosen and the previous is removed from `F`; **(d)** a particle s_1 is evaluated inside the neighborhood `V` and a new particle s_2 is added tangent to both s_0 and s_1 and to `F`. 22
3.3 Control volume of an arbitrary particle's thermal contributions for the discrete element method (DEM). 25
3.4 Contact surface area A_{lm} between two intersecting particles. The radius of A_{lm} is h and the distance $\|\boldsymbol{x}_m - \boldsymbol{x}_l\|$ is d in the diagram. This diagram was adapted from [37]. 26
3.5 Central differences with a 7-point stencil to approximate Fourier's Law for heat conduction; adapted from [25]. 27
3.6 Sample arrangements of spherical SS 316L powder from **(a)** a micrograph by Rombouts et. al [111] and **(b)** the advancing-front algorithm in this SLS model. Examples are illustrative and do not possess the same particle size distributions. 33
3.7 Radius size distribution after particle deposition sampled from a monomodal distribution of 13.5 µm and a FWHM range of 5 µm. The orange line represents the PDF of the sampled distribution and the green line is the CDF of the distribution. 34
3.8 Comparison of frontal cross sections of the melt pool via **(a)** a micrograph from [61] of a frozen melt pool and **(b)** a temperature field output from the SLS simulation (grayscale color is melted material). Both arose from a laser power setting of 200 W with a scanning speed of 2 m/s and a monomodal particle radius distribution of 13.5 µm and a FWHM range of 5 µm. 35

3.9 Screenshots for a top and a sectioned side view of the temperature field at $\frac{1}{9}, \frac{3}{9}, \frac{5}{9}$, and $\frac{7}{9}$ of the final simulation time. 36
3.10 Sampled particle size distributions (left y-axis) and their probability and cumulative density function plots (right y-axis) for the parameter study on μ and FWHM. The first row pertains to the validation case, the second column varies the mean size, and the third row varies the spread (FWHM). 43
3.11 Particle arrangements according to the parameter study on the mean particle size μ and the FWHM range. The particles are colorized based on their radii. 44
3.12 Top and isometric views of the melt pool, including cross sections at $y = 0$ µm and $x = -40$ µm. Grayscale material is above 1700 K, the melting temperature for stainless steel 316L. 45
3.13 Variation over laser power. Top and isometric views of the melt pool, including cross sections at $y = 0$ µm and $x = -40$ µm. Grayscale material is above 1700 K, the melting temperature for stainless steel 316L. 46

4.1 Generic tool profile. 50
4.2 Thermal field along the working edge, highlighted, is the key QoI for the bending process. 54
4.3 Schematic of example "pad" heat source. 58
4.4 Schematic of example "wire" heat source. 58
4.5 Example steady state thermal fields. Top to bottom: copper with pads, copper with wire, aluminum with pads, aluminum with wire. 59
4.6 Example transient and steady thermal fields along working edge. Top to bottom: copper with pads, copper with wire, aluminum with pads, aluminum with wire. 60
4.7 Input distributions of the design parameters. A normal distribution is assumed for the environmental variables and a uniform distribution is assumed for the geometrical variables. Clockwise from top left: h, ϵ, left pad width, right pad width. 63
4.8 Output distribution of the temperature range at the working edge (histogram). . 63
4.9 Output distribution of the temperature range at the working edge (probability density). 64
4.10 Pairwise plots for the design parameters and the temperature range at the working edge. The diagonals showcase the continuous probability density curve for each. The off-diagonals show relationships between each pair. The bottom row shows the contributions of each design parameter to the distribution of the QoI. 65
4.11 Comparison of direct sensitivity computation with the moment sensitivity approach. **(a)** Design optimization programs must compute sensitivity for each objective function and for each design space, leading to combinatorial complexity in incorporating new elements of either side. **(b)** Moment derivatives allow a unified interface to a variety of objective functions. 67
4.12 Illustration of a finite element discretization which does not conform to the geometric domain 71

4.13 Illustration of the voxel structural compliance sensitivity example. Red regions signify greater magnitude and blue less magnitude. **(a)** Voxelized model of a bracket with structural boundary conditions. **(b)** Compliance sensitivity field visualized over a voxel geometry. 79
4.14 Scenario setup for the bracket hole sizing example. The red surface is loaded with a compressive 100 MPa pressure load and the blue surface is restrained. The yellow surface represents the hole whose radius is the design parameter. . . 81
4.15 Illustration of the fandisk example, with the velocity field illustrated as vectors on the surface. The top (red) surface is loaded with 100 MPa downwards pressure and the bottom surface (blue) is restrained. 81
4.16 Scenario setup for the FGM wing example. **(a)** Density field visualized over the wing surface mesh. **(b)** The close (blue) end is restrained and the far (red) end is loaded with a transverse 1000 kN force. 82
4.17 Illustration of the thermal compliance sensitivity example. **(a)** Voxelized model of a bracket with thermal boundary conditions. **(b)** Thermal sensitivity field visualized over a voxel geometry. Red regions signify greater magnitude and blue less magnitude. 83
4.18 Illustration of the mixed design space structural compliance sensitivity example. **(a)** Scenario setup for a mixed assembly. The blue surface is restrained and the red surface has a torque applied on the red surface of 100N$\cdot$ m^2 about the Z axis. **(b)** Structural sensitivity field visualized over the voxel geometry. Red regions signify greater magnitude and blue less magnitude. 83
4.19 An idealized powder particle domain deposited on a substrate material and irradiated by laser energy. 86
4.20 Cylindrical Domain: Sensitivity of final temperatures due to laser irradiation perturbations for a lumped mass for cases (1) without thermal contributions (no cooling), (2) with all thermal contributions, and (3)-(5) with isolated thermal contributions. 100
4.21 An idealization of an embedded particle in a powder pack. 102
4.22 Spherical Domain: Sensitivity of final temperatures due to laser irradiation perturbations for a lumped mass with all thermal contributions. 103
4.23 Sampled particle size distributions for the temperature field variations study. The cases are summarized in Table 4.11. 109
4.24 Local powder depths across a spatial domain measuring 801×101 in the x and y directions, respectively, and their corresponding voxel-wise distribution. 110
4.25 Voxel-wise maximum temperature and minimum temperature across time steps in the steady state stage. The top row is for Case 1 and the bottom for Case 2. The far-right plots show the regions of high temperature variation via the difference between the maximum temperature field and the minimum temperature field. . 111

4.26 Traces of the maximum and minimum temperature along a line with a fixed x value, and through the peak temperature within the melt pool. The horizontal gap between the maximum and minimum lines represent actual variations in the melt pool width. 112
4.27 An idealization of an embedded particle in a powder pack. 113

5.1 A visualization of the spatial covariance matrix $\boldsymbol{C}$. Only a $2,500 \times 2,500$ portion is shown as computing the full $n_x \times n_x$ matrix is computationally prohibitive. . 117
5.2 A visualization of the temporal covariance matrix $\boldsymbol{C}_s$, built with the transient and the steady state stages of the SLS simulation. Time step indices up to 901 are included. 118
5.3 Sample screenshots for a top and a sectioned side view of the temperature field at **(a)** 100 and **(b)** 500 time steps. 119
5.4 A visualization of the temporal covariance matrix $\boldsymbol{C}_s$, built with the steady state stage only of the SLS simulation. Note that time step indices 300 to 901 are included. 120
5.5 (Top) Sorted eigenvalues of $\boldsymbol{C}$ associated with the eigenvectors (spatial modes) in $\boldsymbol{\Phi}$. (Bottom) Same plot showing the region with the most significant eigenvalues. 124
5.6 (Top) Reconstruction error e_R by incrementally employing spatial modes. (Bottom) Same plot showing the region with the most significant spatial modes. The dashed horizontal lines highlight the additional modes necessary to decrease e_R from 0.20% to 0.15%. 125
5.7 Absolute error e_A^n distribution at the plane where z-index = 60 of the SLS thermal simulation domain and at an arbitrary time step. Blue voxels pertain to an underestimation of the temperature and red areas pertain to an overestimation of the temperature 127
5.8 Absolute error e_A^n distribution at various x-index planes of the SLS thermal simulation domain and at an arbitrary time step. Blue voxels pertain to an underestimation of the temperature and red areas pertain to an overestimation of the temperature data. 128

List of Tables

3.1 Summary of the simulation parameters used for all thermal simulations. 32
3.2 Summary of the experimental cases for the particle size distribution parameter study. The mean particle size μ and the FWHM range are varied across a low, medium, and high setting, while the laser power and scan speeds are fixed. . . . 37
3.3 Summary of the experimental cases for the laser power parameter study. The laser power P is varied across a low, medium, and high setting, while the particle size distribution parameters (μ and FWHM) are based on Case 1. 40
3.4 Melt pool shape measures in µm for all parameter studies. The depth is taken in the z direction, the width in the y direction, and the length in the x direction. Ranges on the length for Cases 5 and 7 correspond to disjoint material melt pools. 41

4.1 Example of a sampling design from the translational propagation Latin hypercube algorithm by [144] with 4 data points. Therefore, each input variable is discretized into 4 levels: low, mid-low, mid-high, and high. 55
4.2 Thermal properties 56
4.3 Dimensions for numerical examples in centimeters. 57
4.4 Sample output from the design sensitivity utility via central differences and $\delta = 10^{-4}$. Included is the query design point, the temperature range at the working edge (QoI) for the query design point, and the sensitivity of the QoI. 61
4.5 Sample output from the design space characterization utility. The design space is 4-dimensional and 4 design points were output from a TPLH sampling design. Included in the data is are the design points and the temperature range at the working edge (QoI). 62
4.6 Comparison of moment derivatives vs. finite differences for the hole sizing example. 80
4.7 Comparison of moment derivatives vs. finite differences for the fandisk example. 80
4.8 Comparison of moment derivatives vs. finite differences for shape sensitivity. . . 81
4.9 Cylindrical Domain: Thermal variation as a function of laser input power with perturbations. The peak normalized standard deviation occurs when some sample have reached their steady state temperature and others have not, at approximately 0.05 W. 101

4.10 Spherical Domain: Thermal variation as a function of laser input power with perturbations. The peak normalized standard deviation occurs when some samples have reached their steady state temperature and others have not, at approximately 0.11 W. 104
4.11 Summary of experiments for quantifying the effects of local depth variations across the domain on the temperature contributions by the laser beam. 108

Acknowledgments

I am deeply indebted to my doctoral advisor Professor Tarek I. Zohdi for his invaluable mentorship and for granting me the freedom to pursue this work, for his continuous support through engaging projects with industry partners, and for providing the tools necessary for computational research (laptop, software, and a personal work area). His never-ending curiosity and focus on modeling manufacturing processes was a constant inspiration. Additionally, many thanks for empowering me to teach graduate students as a Graduate Student Instructor in MEC201.

Special thanks to my industrial partners, which included BASF SE, Siemens AG, the Lawrence Livermore National Laboratory (LLNL), and Intact Solutions Inc., for providing interesting projects and the freedom to focus on my personal interests.

This doctoral endeavor would not have been possible without Youngsoo Choi at the Reduced Order Model group at LLNL, for his mentorship and expertise on all things reduced order models and the proper orthogonal decomposition approach.

Many thanks to my fellow research laboratory members, past and present, as they provided much emotional and academic support through research and coursework alike. Most importantly, for their friendship.

Most notably, for her inexhaustible support, and whose name should be included in my diploma, I thank my wife Adriana María. She shared in all the emotions I felt and provided me with an immovable anchor to the real world. Thanks to my daughter Lucía María for her inherent ability to change my perspective every day and for her continuous joy about the world. I would also like to thank my parents Lina María and Andrés and my brothers Felipe and Juan Diego for cheering me on, and on, until the end. I could not have made it without you!

Chapter 1

Introduction

1.1 Overview of Numerical Modeling in Engineering

Predictive sciences pertain to the application of computational models to simulate physical phenomena with the aim of studying their behavior. These models are useful tools for gaining additional insights beyond running physical experiments due to their faithful approximation of the physics from solving systems differential equations. These equations are discretized across the domain, yielding behaviors and results everywhere in the domain. These findings are typically impossible to capture in a physical setting without the right sensors and instrumentation. Generating observations from the computational model deepens the understanding of the physical phenomena, leading to confirmations of untested theoretical propositions, leading to new theorems, or the generation of new propositions for further studies. The cost of running computational models is vastly lower than the cost of running physical experiments, which might require materials and generate waste. However, the detriment to computational approaches hinges on their extreme complexity to simulate events that span fractions of a second, but with an algorithmic runtime in the order of minutes, days, or even months. Thus, numerical techniques must be leveraged to minimize the computational time cost, such as a coarser discretization, a less stringent allowance of the solution accuracy, a simplification of the material geometry, adaptive methods, or the use of symmetry. Once those numerical techniques are exhausted, surrogate models and reduced order models can act as alternate representations of the full model by minimizing the number of degrees of freedom, often with a loss of information. Additionally, if the resources are available, computational approaches are excellent at exploring a design space in the form of uncertainty quantification, sensitivity analysis, and design optimization via multiple executions of the numerical simulation in succession with varying input parameter settings.

In some regimes and related manufacturing processes a particle-mixture may be fluid-like (as opposed to solid-like) and saturated with an interstitial fluid or embedded within a fluid medium. In such cases, the fluid dynamics interaction with the particles can play a major role on the system behavior. This can also involve the evolution of heat. A spatio-temporal

discretization can be used to solve coupled fluid-particle systems, by employing Finite Element, Finite Difference, or Finite Volume methods coupled to Discrete Element Methods. There has been ample research that is suited to describe the powder bed dynamics of an SLS process. Oñate et al [99, 101, 100] give an extension of Lagrangian particle finite element methods (PFEM) for fluid interactions at solid-fluid and solid-solid interfaces in soil mechanics, Carbonell et al [16] describes the adaptation of PFEM to model tool wear for tunneling applications through soil particles, Labra and Oñate [69] employ a finite element mesh to develop an algorithm to efficiently generate high-density packing of spherical particles, Rojek et al [110] and Rojek [109] present discrete element models for contacting spherical particles, Avci and Wriggers [4] use a coupled DEM-FEM for particle-fluid interactions, Leonardi et al [71] present a Lattice-Boltzmann adaptation for simulating non-Newtonian rheology, free surfaces, and moving boundaries, and Bolintineanu et al [13] review methods for handling simulations of the dynamics of colloidal suspensions at the mesoscale. All of these studies present techniques to simulate the dynamics of a complex powder particle deposition phase and the rheology of polymer melt flow in an SLS process. Furthermore, work by Salloum et al [117] represents advances in the optimization of computation cost by providing an efficient data compression/decompression method for storing and transferring large data sets.

1.2 Quantifying Uncertainty and Process Variation

Working with the selective laser sintering (SLS) and other similar additive manufacturing technologies is a complex endeavor. This manufacturing process uses a laser to impart heat upon a powdered material in order to melt it and fuse it to previously-melted regions of the powder bed. Since the process works at the micro-scale and the laser generates steep thermal gradients across the material, then managing the extent of material that does undergo a phase change is challenging. The study of uncertainty quantification (UQ) attempts to identify and explain the sources of variation of a system, and in this case the temperature field throughout a material domain.

UQ is the science of identifying sources of variation and understanding how likely a certain outcome or behavior of a system is based on these input variations. Indeed, the quality of the UQ result will depend heavily on the quality of the characterized distributions of variation of the inputs. Sources of uncertainty can be classified into aleatoric and epistemic. Aleatoric uncertainty are variations due to phenomena that is either difficult to measure and predict. For instance, gusts of wind affecting the direction of flight of a arrow is an aleatoric source of uncertainty. Epistemic uncertainty generates variations that are available, but are chosen not to be included in the system. The simplification of a computational model to minimize the algorithmic runtime removes physical effects by choice. Thus, epistemic uncertainty includes inaccuracies of a numerical model. The variational effects from uncertain inputs are measured on an output quantity of interest (QoI), which is chosen to be a proxy for the measuring behavior or outcome of the system. For example, the maximum stress in a body can be used as a QoI to summarize the overall level of loading on the part. Castrillon, Rock,

& Zohdi [17] leverages a maximum nodal temperature at the edge of a heated blade for automated garment assembly. This QoI is an indicator for the possibility of causing thermal degradation damage on the garment if the heater blade is too hot, or not creating a crisp bend at the fold of the garment if the blade is too hot. This study will be showcased in Section 4.1.

The application of UQ techniques for additive manufacturing (AM) processes is still in its nascent form. Early applications of UQ on AM includes [58] from Lawrence Livermore National Laboratory, which ran experiments across a design space to capture the size of the material melt pool shape due to variations in the laser scan speed and the power setting of the laser. Another application include [79], which measure model uncertainty from assumptions, unknown simulation parameters, numerical approximations, and measurement error in calibration data for an SLS process. A third example showcases a model that predicts the entire temperature history of the material during a single track of the laser in SLS [91].

1.3 Surrogate Modeling and Reduced Order Modeling

As mentioned earlier in this chapter, computational costs can me greatly minimized by the use of surrogate models and reduced order models (ROMs). Typically, surrogate models represent a data-driven model that mimics the physics of the full model via a mathematical relation. Ample amounts of full-order data are employed to build the surrogate model, and afterwards the trained model is an accurate, low cost alternative to the full model. Surrogate models are also called meta-models and they have been applied for estimating the levels of toxic chemicals in humans and pollutants in the atmosphere using stochastic response surface methods [54], for approximating response surfaces using Kriging interpolation and multivariate adaptive regression splines in a Rosenbrock function [39], for design optimization of combustion engine pistons using multivariate tensor-product basis functions [20], and for solving the 2D Burgers' equations with a high Reynolds number via a non-linear manifold projection using artificial neural networks (ANNs) [70].

On the other hand, a ROM does follow the physics of the problem, but the representation of the system is defined by a linear algebra transformation onto optimal bases that maximize the explanation of the data as more dimensions are employed. This projection onto a new basis allows for the removal of the least significant directions, thus reducing the order of the model. These reduced models are also less costly and are accurate since less computations are carried out across a fewer number of dimensions. A survey of ROM techniques can be fund in [35], where the proper orthogonal decomposition (POD) and factor analysis are discussed as linear dimension reduction methods are discussed, as well as higher-order methods such as the independent component analysis.

It is possible to leverage the advantages of both surrogate models and ROMs. Recently, surrogate modeling has been dominated by machine learning (ML) techniques, most notably ANNs, but these approaches are data-hungry in that a large amount of full order data is required to train these models, their results tend to not be physically permissible, and

they must be retrained if extrapolation is desired on a new unsampled simulation case [59]. Coupling ML techniques with models based on fundamental principles is often referred to as physics-informed ML or physics-aware ML. These models have been widely disseminated across multiple disciplines such as Earth system science [107], signal processing [113, 14], turbulence modeling [12, 89], novel material design [105, 123], hydrology [154], and quantum processes in chemical reactions [125]. In the realm of engineering, physics-informed ML methods have been applied to solve the heat equation PDE by employing ANN features as the convective coefficients in advanced manufacturing [157], the study of defect formation for commonly-used alloys in additive manufacturing [27], the prediction of porosity of a 3D-printed part in a laser powder bed fusion manufacturing process [77], the prediction of grain structure in a material microstructure in a directed energy deposition manufacturing process [60], and the prediction of metal temperature and melt pool fluid dynamics in AM [156]. Additional physics-informed ML techniques are surveyed in [152].

1.4 Organization of the Present Work

This dissertation is presented as follows. Chapter 2 provides background information about the selective laser sintering process, an typical AM process employed in industry, and the physical phenomena that govern the printing of parts. Next, Chapter 3 dives deeper into the formulation of the numerical problem that solves the balance of thermal energy PDE, stemming from the theory of solid mechanics. The thermal model is verified against an physical experimental result and then a model capability study is performed on material and laser-specific input parameters. Output temperature data is then used in Chapter 4 to analyze the variation and sensitivity on the shape and size of the geometry of the material that is processed by the laser. In parallel, a ROM is built in Chapter 5 using a POD approach and the data reconstruction error is visualized and analyzed. Lastly, 6 provides concluding remarks on the present work. These remarks summarize the findings on the shape of processed material from the capability study and its geometrical variation from the uncertianty quantification study in Chapter 4.

Chapter 2

Background and Theory

2.1 An AM Application: Selective Laser Sintering

One of the primary uses of lasers in manufacturing is to induce a prescribed temperature field within a target material. Specifically, a subset of scenarios of key interest within Additive Manufacturing (AM) processes involve depositing dry powder-based materials onto a surface and then to heat the targeted material with a laser in order to sinter or melt the target material. Householder [48] pioneered selective laser sintering (SLS) and melting (SLM) in 1979 and then Deckard & Beaman [23] followed with their work in the mid-1980s. Since then, significant advances have been achieved in the metal-based AM industry as reviewed by Smith et al [132]. Experimentally, in the realm of materials and microstructure characterization, Voisin et al [145] studied the detrimental effects of defects and porosity on as-built tensile properties of titanium alloys. Numerically, Wang et al [149] presented an integrated Discrete Element Method-Computational Fluid Dynamics (DEM-CFD) framework for simulating the deposition of dry powder particles and subsequent phase changes by SLM and electron beam selective melting (EBSM) processes[1]. Additional models and simulation methods for capturing the SLS/SLM processes are presented by Zohdi [158] and [162]. Selective laser sintering and melting processes are not restricted to metallic powders, however, as it also applies to polymeric [67] and ceramic [10] powder materials, as highlighted by Kinstlinger et al [63].

Heating of materials by way of laser-based processes is attractive in manufacturing due to the precision that it offers, with a nominal focus diameter not larger than 500 nm. Lasers offer a highly controllable method to deliver the necessary amount of localized energy to process powdered material due to its collimated (parallel) and monochromatic energy, and to its ability to pulse at a prescribed frequency by beam chopping or by modulating the voltage.

[1]Additional techniques used in a generalized AM process can include (a) ink jet printing, consists of a pressurized, a heated, and/or a vibrating small orifice to project small ink droplets for deposition on a substrate, and dried by UV light or other means, (b) aerosol jetting, which directs streams of atomized, high-velocity particles towards a substrate, and (c) electron beam melting (EBM), which bonds powdered material under a high vacuum layer-by-layer via an electron beam.

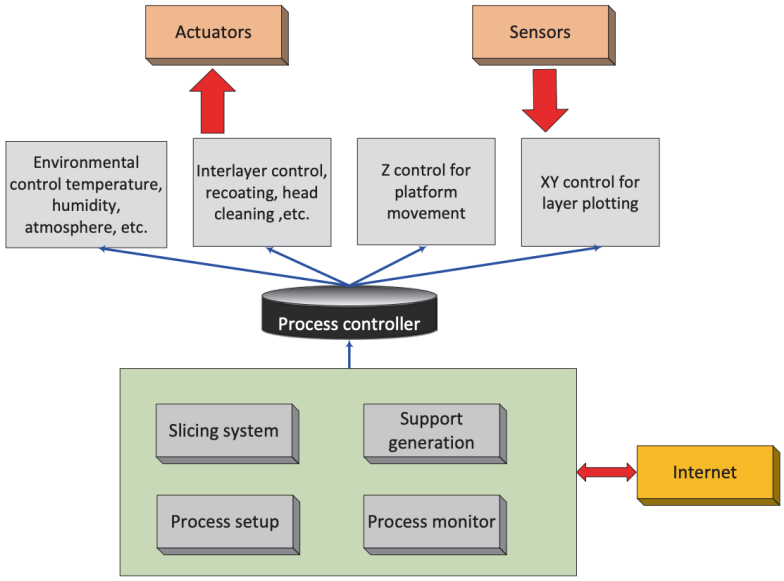


Figure 2.1: Schematic of an AM system [38].

Some typical lasers used are yttrium aluminum garnet (YAG) for processing ceramics and metals and carbon dioxide (CO_2) for polymeric materials. Industrial lasers range in power, approximately from 1 to 10,000 Watts for polymeric materials. However, if not properly controlled, laser procedures can cause excessive variability in the quality of the as-built parts as well as thermal damage to the material and the substrate. These concerns have motivated the present work, in which we build a thermal spatio-temporal numerical model to simulate temperature changes of a powdered metallic material due to input power from a laser beam, based on the SLS process. This model will allow us to quantify the sensitivity of the material temperature when the input laser power and other material specifications are perturbed.

Overview of the SLS Process

An SLS printer is made up of an arrangement of components that must work in unison in order to carryout the same set of actions hundreds or thousands of times. To achieve this, the assembly must be well designed and accompanied by a battery of thermal, optical, and positional sensors to feed information into a controller. Figure 2.1 shows a general schematic of the systems that make up an AM machine from *Additive Manufacturing Technologies* by Gibson et. al [38]. Seminal work by Fish et. al [33] in 2015 describes the design and construction of the Laser Additive Manufacturing Pilot System (LAMPS), which monitors and controls real-time the SLS printing process with the goal of understanding physical phenomena and guiding the improvement of future equipment.

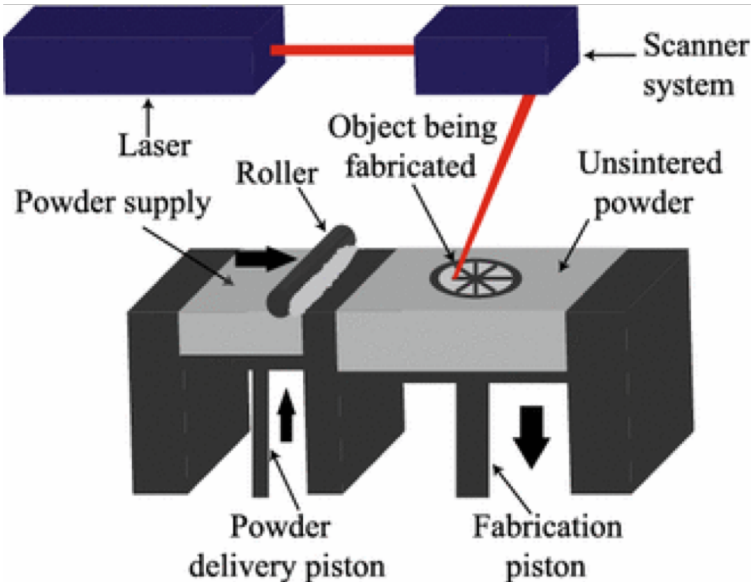


Figure 2.2: Components of an SLS system [37].

Furthermore, the printer must have a central processing unit that receives instructions for scanning the pattern of movement of the laser over the powder bed. These instructions normally are passed from an external computer that has processed a CAD geometry, slicing it into layers and translating each layer into a pattern of a tool path. This tool path is then further processed into simple motion instructions in the form of g-code, which specifies a direction and duration of motion in that direction. Completing a set of g-code instructions end up melting and consolidating the material in a layer, and building all layers will create a three-dimensional part.

In general, a typical SLS printer has a bed full of powdered material and a smaller supply bed with virgin powder within the build chamber, as shown in Figure 2.2. The build chamber is normally filled with nitrogen gas or other inert gas to prevent oxidation of the material as it heats up. Each of these collections of powder have been preheated to a controlled elevated temperature. A roller, wiper blade, or a spreader will transfer this virgin powder onto the powder bed, covering the material by a thin layer that is typically 20 to 100 nm deep. This is when preheating the virgin material is critical: the previous layer would otherwise be thermally shocked.

Once the new layer is laid then the control allows a few seconds for the temperature to equalize across the powder bed volume. Then, the laser is activated at a set power, passing through a series of lenses and mirrors, and ultimately irradiating the top surface of the the powder bed. Based on the g-code instructions, servomotors controlling the position and angle to the set of mirrors and lenses direct the laser beam to move at a prescribed direction and speed. This irradiation will cause the powder particles to fuse together and bond upon solidification after the laser beam moves away.

After scanning the entire 2D geometry of the layer, the laser shuts off. Lastly, the powder bed piston is lowered by a distance equal to the thickness of a single layer and a new powder layer is deposited before laser processing resumes. The process repeats for all layers until the final part is fabricated. This layer-by-layer build pattern is what characterizes SLS as an AM process.

The SLS printing process is not complete until the part is extracted from the powder bed and the build chamber to undergo post-processing procedures. These procedures ensure that the part is fit for its intended function. The three steps in post-production include part extraction, media blasting, and used powder recovery. However, before these processes can commence, the built part must be allowed to cool down at a controlled rated to minimize large thermal gradients within volume of the part which can cause significant internal stresses, a culprit of gross warping of the part and local deformations. These effects are additional reasons why a careful design of the thermal control of the SLS process is crucial.

Beyond the consideration of cosmetic or geometrical defects, the notion of the material porosity across the part volume is a major contributor to degraded mechanical material properties compared to the properties of a fully dense part, controlling for the material selection. Porosity in the material will yield lower densities and also provide sites fracture cracks to develop and grow. The goal of a well-designed SLS process must employ printing parameters that work together to generate as close as possible to a fully-dense part. The SLS process is controlled by the printing parameters shown in Figure 2.3 by Sun et al. [135], which are grouped by the aspect that they control.

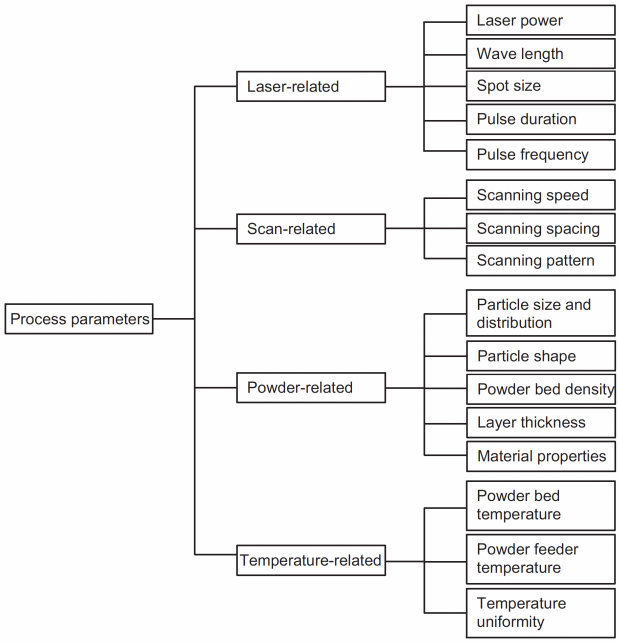


Figure 2.3: SLS process parameters by [135].

In-process Flaws and Defects

Figure 2.3 provides evidence that the SLS process is a complex procedure with complex inputs and systems. Calibrating and controlling these inputs initially and in time during printing in the lab or the manufacturing shop continues to be an art. Thus, human monitoring is normally a critical component for minimizing in-process flaws and defects.

The build chamber is equipped by overhead lamps that provide radiative heat to the top 5-10 mm of the powder bed to reach and maintain a temperature. These lamps provide control to the SLS machine operator if foil curling (in polymers) or local warping is observed when the powder bed is too cool and is drawing too much energy from the thin, warmer, and newly-printed layer. Foil curling is also possible if the temperature of the virgin material is too low. The temperature differential will shock the underlying, warmer layer, causing severe localized residual stresses and visible deformations such as the print in Figure 2.4.



Figure 2.4: In-process foil curling due to material temperature differentials between layers [92].

Conversely, if the powder bed temperature is too high, particles will begin to become mushy and will start to superficially fuse and sinter. Caking will begin to happen at the top layers, especially in polymeric materials, and this has the potential to age and degrade the powder and influence its recyclability. Usually, to minimize waste, used powder from a previous print is mixed with virgin powder. To curb the effect of caking the operator will have to manually reduce the temperature set point of the top surface of the powder bed.

Gross part warping during and after printing can also occur if the bottom layers cool too quickly compared to the hotter top layers. Thus, the walls of the powder bed normally have resistance heaters embedded in the top 10 cm to minimize heat loss across the walls. These heaters will keep the newly-printed layers warm and equalize any unwanted thermal gradients before the layers are allowed to slowly cool as the print progresses and they lower below the heating zone. If the material does not recrystallize in a homogeneous fashion across all regions of the part then warping will occur. Importantly, there is another embedded heater at the base of the powder bed to ensure that the earliest layers do not cool down too severely during processing, especially if the part is tall in the print direction.

Physical Phenomena

Maintaining the shape of the melt pool constant along the path of the laser beam is of importance so that the geometric specifications of the printed part are as close as possible to the intended shape and resolution. The dynamics melt pool must also be controlled to yield a maximal density of the material so that the actual bulk material properties approximate the theoretical material properties well. To reach a constant melt pool with high density, the processing parameters must be chosen carefully, and, in turn, the contribution of each parameter must be studied within a relevant material scale. Thus, the physics of the SLS process must be described at a part scale, a feature scale, and a powder/micro-scale.

Powder Scale

The sudden and targeted addition of heat via an energy-dense infrared beam is the originator of a highly dynamic chain of effects in the micro-scale, the feature scale, and the part-scale. This input of energy will heat up the powdered material, ensuing a phase change and thermal conduction across neighboring particles and underlying solidified material from the previous layer. A melt pool of melted or sintered material then begins to form, creating flow and surface tension forces. High temperature gradients between the melt and the surrounding material also create pressure differentials that attempt to push the melt away from the laser beam. Combined with the surface tension imbalances this outward flow is called the Marangoni convection effect, a phenomenon that is also prevalent in tungsten inert gas (TIG) welding and other similar processes. The Marangoni effect is so significant that the outward flow will create a forward molten wave that pinches off and ejects hot spatter.

A denudation zone is additionally formed around the laser beam, characterized by a depression void of material, which is a major contributor of porosity as the melt pool flows back into the denudation zone, trapping gas before cooling and solidifying. Furthermore, the laser energy will also create flow currents in the surrounding nitrogen, generating spatter as material particles are picked up and sent as projectiles far away from the melt pool. This spatter can create unintended localized melting away from the laser beam. All of these dynamical processes are thoroughly described by Khairallah et al. [62] and further modeled using the smooth particle hydrodynamics (SPH) method by Russell et al. [114]. As the laser beam moves on and the trailing melt pool begins to cool and solidify. Residual stresses form and the local microstructure of the printed material is set, finalizing the actual bulk material properties of the part.

Feature Scale

The feature scale of the 3D print is a collection of laser tracks or paths that together build features in a part, such as walls, holes, overhangs, planar regions, among others. The laser scan pattern must be such that it can approximate highly dense features. The strategies to achieve this goal is two-fold: first, the spacing between neighboring tracks must have overlap to ensure intralayer bonding and, secondly, the depth of the melt pool must reach into the

underlying layer to ensure interlayer bonding. Ideally, the intersection of neighboring 2D melt pool cross sections should ensure that no unmelted powder remains in between tracks.

The overlap between processed material tracks means that material is constantly being cycled thermally and creating complex and highly localized thermal histories. As a melt pool cools it contracts and work hardens the local material, altering the crystallography of a solidified metallic material. Liu & DuPont [78] mathematically correlate the grain formation and the geometry of a melt pool cross section for several single-crystal alloys and Montgomery et al. [90] map a processing parameter space based on experimental grain growth results for Inconel 625. Thus, the microstructure can vary greatly in space, even if a pattern may be discernible across a set of neighboring tracks. This variation of microstructure variation within a part feature is a clear indicator that residual stresses and even microcracking may exist.

Part Scale

The success of a part in its intended function will be dictated by its actual bulk material properties. A part with a uniform distribution of micro-scale and feature scale material structures will perform according to these bulk properties. However, any local deficiencies in the structure will be exaggerated and will cause the part to fail prematurely. The bulk material properties will me mostly affected by feature scale defects since these are geometrically large enough to cause localized failures, such as crack development and propagation. A void, for example, at the micro-scale might only be large enough to affect the local volumetric microstructure, but not large enough to affect a bulk material property. Clearly, if the processing parameters are not chosen carefully and voids or material degradation in the micro-scale is typical, then this situation will affect the bulk material properties.

Chapter 3

Numerical Problem

3.1 Invoking Continuum Mechanics

Continuum mechanics is a framework for capturing macroscopic behaviors as materials undergo changes. This work only deals with the quantification of the thermal field, as the measure of temperature, but amongst other fields that can be modeled include displacements via kinematic quantities. In general, continuum mechanics allow for the evolution in time of these fields using balance principles and material constitutive equations to close the system of equations. The latter are tailored to capture the behavior of materials as governed by the physical world.

In the vast majority cases, these systems of differential equations cannot be solved analytically, and so numerical methods are required for the task. This engenders the need for the careful selection and efficient execution of the numerical methods, a research field known as computational mechanics.

This chapter is organized as follows...

Kinematic Quantities

Consider the three-dimensional Euclidean space $\mathcal{E}^3$ with an orthonormal basis $\{\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3\}$ and origin O. This space is equipped with the dot product $(\boldsymbol{u} \cdot \boldsymbol{v})$ to compute the Euclidean norm $(\|\boldsymbol{v}\|_2)$ for the measure of distances across this space. This allows for the definition of vectors and tensors and their cross products $(\boldsymbol{u} \times \boldsymbol{v})$ and tensor products $(\boldsymbol{u} \otimes \boldsymbol{v})$. Furthermore, we can also define relevant tensor calculus for arbitrary vectors $\boldsymbol{u}$ and $\boldsymbol{v}$, second-order tensors $\boldsymbol{F}$ and $\boldsymbol{G}$, and scalar α in Einstein notation as

$$\begin{aligned} &\text{Divergence of a vector:} && \nabla \cdot \boldsymbol{v} = v_{i,i} = \alpha \\ &\text{Divergence of a tensor:} && \nabla \cdot \boldsymbol{F} = F_{ij,j} = \boldsymbol{u} \\ &\text{Gradient of a vector:} && \nabla \boldsymbol{v} = v_{i,j} = \boldsymbol{G}. \end{aligned}$$

Consider now a body $\mathscr{B}$ that is a collection of any number of particles with positive mass. This body occupies a region $\mathcal{R} \in \mathcal{E}^3$. Now, suppose that we take an arbitrary material region $\mathcal{P}$ with smooth boundary $\partial\mathcal{P}$ inside of $\mathcal{R}$. Convention dictates that tensors used to capture quantities of the region $\mathcal{P}$ in the reference (initial) configuration are to be written with a capital letter and the same quantity for a current configuration at a later time t with a lower case letter. Thus, we are able to map the referential location $\boldsymbol{X}$ of a point in $\mathcal{P}$ as $\boldsymbol{x}$ for any motion as

$$\boldsymbol{x} = \boldsymbol{\chi}(\boldsymbol{X}, t) \tag{3.1}$$

$$\boldsymbol{X} = \boldsymbol{\chi}^{-1}(\boldsymbol{x}, t), \tag{3.2}$$

since the motion $\boldsymbol{\chi}$ is invertible. Now, we can define the displacement vector $\boldsymbol{u}$ as the difference in position of the material point as it deforms from $\boldsymbol{X}$ to $\boldsymbol{x}$ within the orthonormal basis:

$$\boldsymbol{u} = \boldsymbol{x} - \boldsymbol{X}. \tag{3.3}$$

Note that this work does not solve the deformation field, meaning that $\boldsymbol{u} = 0$, and thus, the current configuration is always coincident to the reference configuration of the region $\mathcal{P}$:

$$\boldsymbol{x} = \boldsymbol{X}. \tag{3.4}$$

Throughout this dissertation, we will discuss all vector and tensor quantities in the referential configuration.

Balance Laws

Based on the behavior of physical matter, the filed of mechanics postulates the existence of the conservation of mass, linear momentum, and angular momentum. Also, the first law of thermodynamics governs the conservation of energy. These balance laws always apply in order for a material process to be physically permissible. These laws are now briefly discussed in the reference configuration.

The Conservation of Mass

Body $\mathscr{B}$ undergoing a deformation will do so in such a way that its mass will remain constant throughout all time. From the perspective of a collection of regions $\mathcal{P}$, each with volume dv, the conservation of mass is expressed as the rate of change of the integral of the density ρ for all material points must equal zero.

$$\frac{d}{dt} m(\mathscr{B}) = \frac{d}{dt} \int_{\mathcal{P}} \rho \, dv = 0. \tag{3.5}$$

We can use Reynold's transport theorem and the product rule for the material time derivative as

$$\int_{\mathcal{P}} \left(\dot{\rho} + \rho \nabla \cdot \boldsymbol{v}\right)\, dv = 0. \tag{3.6}$$

The conservation of mass must hold at every point, then it follows from the localization theorem that

$$\dot{\rho} + \rho \nabla \cdot \boldsymbol{v} = 0. \tag{3.7}$$

In the special case with no deformations ($\boldsymbol{v} = \boldsymbol{0}$) we find that the time derivative of the density is zero:

$$\dot{\rho} = 0, \tag{3.8}$$

which means that the density will be the same as the referential density ρ_0 at all times.

The Conservation of Linear Momentum

The conservation of linear momentum stems for Newton's second law, which balances the change of momentum of a particle is in accordance with the sum of external forces on the particle. In mathematical terms, this means that

$$\frac{d}{dt}\int_{\mathcal{P}} \rho \boldsymbol{v}\, dv = \int_{\partial\mathcal{P}} \boldsymbol{t}_{(\boldsymbol{n})}\, da + \int_{\mathcal{P}} \rho \boldsymbol{b}\, dv, \tag{3.9}$$

where we balance the change of momentum ($\rho\boldsymbol{v}$) with the sum of external tractions $\left(\boldsymbol{t}_{(\boldsymbol{n})}\right)$ acting normal to the surface of the particle plus the sum of all body forces ($\rho\boldsymbol{b}$), which include gravity. In simple terms, Equation 3.9 has the mass-weighted acceleration on the left-hand side and the sum of forces on the right-hand side, rewritten as:

$$\int_{\mathcal{P}} \rho \boldsymbol{a}\, dv = \int_{\partial\mathcal{P}} \boldsymbol{t}_{(\boldsymbol{n})}\, da + \int_{\mathcal{P}} \rho \boldsymbol{b}\, dv. \tag{3.10}$$

Based on the localization theorem for a single particle, and knowing that the surface tractions are related to the Cauchy stress tensor $\boldsymbol{T}$ via Cauchy's theorem $\boldsymbol{t}_{(\boldsymbol{n})} = \boldsymbol{T}\boldsymbol{n}$, the local balance of linear momentum becomes

$$\rho \boldsymbol{a} = \nabla \cdot \boldsymbol{T} + \rho \boldsymbol{b}. \tag{3.11}$$

The Cauchy stress $\boldsymbol{T}$ is the stress within the material in response to its deformation. This quantity is the stress used for gauging failure criteria, amongst other practical engineering applications.

In the special case where the particle is not acted upon by any external forces ($\boldsymbol{t} = \boldsymbol{0}$ and $\boldsymbol{b} = \boldsymbol{0}$), then the particle is stationary and not deforming. We can simplify the balance of linear momentum as

$$\rho \boldsymbol{a} = \nabla \cdot \boldsymbol{T} + \rho \boldsymbol{b} = \boldsymbol{0}. \tag{3.12}$$

The Conservation of Angular Momentum

The conservation of angular momentum balances the change of angular momentum of a particle with the applied moments on the particle. For a collection of particles, the angular momentum around a reference location is computed based on the moment arm $\boldsymbol{x}$ to each particle. Mathematically,

$$\frac{d}{dt}\int_{\mathcal{P}} \boldsymbol{x} \times \rho \boldsymbol{v}\, dv = \int_{\mathcal{P}} \boldsymbol{x} \times \rho \boldsymbol{a}\, dv = \int_{\partial \mathcal{P}} \boldsymbol{x} \times \boldsymbol{t}_{(\boldsymbol{n})}\, da + \int_{\mathcal{P}} \boldsymbol{x} \times \rho \boldsymbol{b}\, dv, \tag{3.13}$$

and based on the localization theorem for a single particle and applying Cauchy's theorem,

$$\boldsymbol{x} \times \rho \boldsymbol{a} = \nabla \cdot (\boldsymbol{x} \times \boldsymbol{T}) + \boldsymbol{x} \times \rho \boldsymbol{b}, \tag{3.14}$$

Factoring out the moment arm $\boldsymbol{x}$, employing the product rule, and rearranging terms, we have

$$\boldsymbol{x} \times (\boldsymbol{a} - \nabla \cdot \boldsymbol{T} - \rho \boldsymbol{b}) = \nabla \cdot \boldsymbol{x} \times \boldsymbol{T} \tag{3.15}$$

Due to the balance of linear momentum, we arrive at

$$\nabla \cdot \boldsymbol{x} \times \boldsymbol{T} = \boldsymbol{0} \tag{3.16}$$

$$\boldsymbol{I} \times \boldsymbol{T} = \boldsymbol{0}, \tag{3.17}$$

where $\boldsymbol{I}$ is the identity tensor. Lastly, using index notation we can show that the Cauchy stress tensor must have symmetry in order for Equation 3.17 to hold:

$$\boldsymbol{T} = \boldsymbol{T}^T \tag{3.18}$$

The Principle of Energy Balance

The balance of energy includes the balance of mechanical energy and the balance of thermal energy. Balancing the mechanical energy stems from the localized balance of linear momentum in Equation 3.11, dotted on both sides by the velocity vector $\boldsymbol{v}$. Then, integrating across $\mathcal{P}$ and using conservation of mass and the divergence theorem, we have the integral statement of the balance of mechanical energy:

$$\frac{d}{dt}\int_{\mathcal{P}} \frac{1}{2}\rho \boldsymbol{v} \cdot \boldsymbol{v}\, dv = \int_{\mathcal{P}} \rho \boldsymbol{b} \cdot \boldsymbol{v}\, dv + \int_{\partial \mathcal{P}} \boldsymbol{t}_{(\boldsymbol{n})} \cdot \boldsymbol{v}\, da - \int_{\mathcal{P}} \boldsymbol{T} \cdot \boldsymbol{D}\, dv. \tag{3.19}$$

Here, $\boldsymbol{D}$ is the rate-of-deformation tensor, and the term $\boldsymbol{T} \cdot \boldsymbol{D}$ is the stress power, which is the rate of energy available to do work by the internal stress. In short, the left hand side is the rate of change of the kinetic energy, and the right hand side is the work done by the external forces acting within or normal to the boundary of the region.

The balance of thermal energy, also known as the First Law of Thermodynamics when applied to thermodynamic processes, models the relationship between the change of the total

kinetic and the specific internal energy w stored in a system with the inputs and outputs of energy through its boundaries.

Before providing the balance of energy of region $\mathcal{P} \subseteq \mathcal{R}$, we must define the scalar field for energy generated or supplied per unit mass $\ell = \ell(\boldsymbol{x}, t)$, typically via radiation. Also, define the scalar field for the energy flux per unit mass through boundary $\partial\mathcal{P}$ with outward unit normal $\boldsymbol{n}$ as $s = s(\boldsymbol{x}, t; \boldsymbol{n}) = s_{(\boldsymbol{n})}(\boldsymbol{x}, t)$. Now, the balance of energy is stated as

$$\frac{d}{dt}\int_{\mathcal{P}} \left(\frac{1}{2}\rho\boldsymbol{v}\cdot\boldsymbol{v} + \rho w\right)\, dv = \int_{\mathcal{P}} \rho\boldsymbol{b}\cdot\boldsymbol{v}\, dv + \int_{\partial\mathcal{P}} \boldsymbol{t}_{(\boldsymbol{n})}\cdot\boldsymbol{v}\, da + \int_{\mathcal{P}} \rho\ell\, dv - \int_{\partial\mathcal{P}} s_{(\boldsymbol{n})}\, da. \tag{3.20}$$

Rearranging terms and referencing the balance of mechanical energy in Equation 3.19, we arrive at the balance of thermal energy:

$$\frac{d}{dt}\int_{\mathcal{P}} \rho w\, dv = \int_{\mathcal{P}} \rho\dot{w}\, dv = \int_{\mathcal{P}} \boldsymbol{T}\cdot\boldsymbol{D}\, dv + \int_{\mathcal{P}} \rho\ell\, dv - \int_{\partial\mathcal{P}} s_{(\boldsymbol{n})}\, da, \tag{3.21}$$

where $\dot{w}$ is the rate of change of the specific internal energy of the system.

Since $s_{(\boldsymbol{n})}$ is the heat flux int he outward normal directions, we can model this quantity via a heat flux vector $\boldsymbol{q}$ with relation $s_{(\boldsymbol{n})} = \boldsymbol{q}\cdot\boldsymbol{n}$. Thus, via the divergence theorem, Equation 3.21 becomes

$$\frac{d}{dt}\int_{\mathcal{P}} \rho w\, dv = \int_{\mathcal{P}} \rho\dot{w}\, dv = \int_{\mathcal{P}} \boldsymbol{T}\cdot\boldsymbol{D}\, dv + \int_{\mathcal{P}} \rho\ell\, dv - \int_{\mathcal{P}} \nabla\cdot\boldsymbol{q}\, dv, \tag{3.22}$$

and its local form:

$$\rho\dot{w} = \boldsymbol{T}\cdot\boldsymbol{D} + \rho\ell - \nabla\cdot\boldsymbol{q}. \tag{3.23}$$

In the special case where we assume no deformations, then the rate of deformation is zero ($\boldsymbol{D} = \boldsymbol{0}$). This gives us the localized balance of thermal energy for a body at rest:

$$\rho\dot{w} = \rho\ell - \nabla\cdot\boldsymbol{q}. \tag{3.24}$$

3.2 A Thermal Model for SLS

There is a strong need for modeling additive manufacturing (AM) processes due to the complex interaction between multiple physical phenomena. These phenomena make AM processes difficult to calibrate and setup in the field, albeit for research in the laboratory or industrial fabrication on the production floor. Selective Laser Sintering (SLS) is governed by more than 100 process parameters and their individual effects on the process and the final part have not been fully studied with sufficient experimental data and mathematical models. Processing parameters pertain to the laser settings, powder bed properties, and the print

strategy. For example, unknown effects might arise from choosing certain combinations of laser beam power and wavelength, laser scan speed, the scan pattern, material properties, and powder particle sizes and packing, which interact to impart a certain localized electromagnetic energy per second that effect phase changes and highly dynamic material flows. These phenomena create dynamics in the melted material due to thermal gradients. Furthermore, the thermal controls of the SLS printer must be tuned so that the material cools in a predetermined rate to minimize residual stresses in the material and warping due to thermal differentials between different regions of the printed part. In a macro scale, process parameters can greatly affect the extend of bonding between neighboring layers of the printing process, thus dictating bulk mechanical properties such as the yield strength, the stiffness, and the elasticity of the printed part. Taking great care to tune the processing parameters in a successful combination is paramount in order to have a printed part that must perform within tight tolerance limits via material properties that must be achieved reliably.

Building an experimental apparatus and measuring most of these complex effects accurately with limited sensing devices pose practical difficulties. Therefore, numerical simulations are powerful tools to generate data and to study the AM process in hand with a considerable reduction of spent resources.

We must systematically build a numerical model that solves for the thermal field for a generic single laser track of the SLS process. Capturing the evolution of temperature in time and space will provide great insight about the dynamic nature of the geometry of the material melt pool. Quantifying this shape is an indicator of how much newly-melted material has fused with re-melted material from adjacent tracks and the underlying layer. Advantages include further insight about the capability of the process to create certain microstructures and the capability to print certain part features.

Let us now accommodate the balance of thermal energy without deformations to the simulation of the processing of a single laser track from a layer of powder particles previously deposited on top of a solid substrate. The substrate is a stand-in for solid material from an underlying layer. We modify Equation 3.24 with the following practical assumptions on the material and the technology of the SLS printer:

- The specific internal energy w of region $\mathcal{P}$ is stored solely as thermal energy and it is linearly dependent on the heat capacity C of the material $w \approx C(T - T_0)$, where T_0 is based on a baseline temperature and C has units of J/(kg·K).

- The heat input/generation term models the volumetric absorbed laser heat $\hat{\ell}$ by the material in W/m^3. It suffices to say at this point that the heat intensity depends on space and time, thus $\hat{\ell}(\boldsymbol{x}, t) = \rho\ell(\boldsymbol{x}, t)$. More information about the heat source will be provided later in Section 3.2.

- The heat flux is modeled using Fourier's Law of heat conduction, and the phenomena of heat convection and radiation to the surroundings is also to be considered. These effects will be described in greater detail in Section 3.2. Fourier's Law assumes internal conduction as $\boldsymbol{q} = -K\nabla T$, where K is the thermal conductivity in W/(m·K).

Thus, ignoring convection and radiation, the governing equation becomes

$$\rho C\dot{T} = \nabla \cdot (K\nabla T) + \hat{\ell}(\boldsymbol{x}, t). \tag{3.25}$$

Heat Source Model

Lasers used for the SLS process are typically pulsed at a certain frequency, delivering power in relatively short bursts of uniform energy. Even though continuous lasers are less common, for simplicity, this work will assume a continuous delivery of power.

The laser energy is defined by a total power P delivered from the laser. Neglecting any energy lost through the assembly of mirrors and lenses, P is the power in Watts that reaches the powdered material from overhead. Spatially, this total power is distributed across a small area da via a 2D intensity function $I_r(d)$, where d is the radial distance from the center of the laser beam to a certain circular perimeter:

$$\hat{P}(d, t) = P\,(t)\, I_r\,(d)\, da \text{ with } \int_0^\infty I_r\,(d)\, da = 1 \tag{3.26}$$

We maintain the dependence of P on time since the laser beam is in motion with scanning velocity $\boldsymbol{v}_\ell$. Furthermore, typical of SLS lasers, the laser beam energy distribution is described by a Gaussian shape as

$$I_r(d) = e^{-2\frac{d^2}{\omega^2}}, \tag{3.27}$$

where ω is the radial distance where the intensity falls below the peak intensity I_0 at the center of the laser beam by a factor of e^{-2}. The peak intensity in $\mathrm{W/m^2}$ is calculated as

$$I_0 = \frac{2P(t)}{\pi\omega^2}. \tag{3.28}$$

We now have a description of the local laser energy that reaches the top surface of the powder layer, and how it is distributed across the circular beam diameter with respect to a peak intensity at the center of the focused beam. The laser energy does penetrate the powder layer to impinge on particles below the surface, reaching the solid substrate below and attenuating through the material. To describe these processes, we use an empirical relation named the Beer-Lambert equation to describe the irradiation of powder particles under the top surface of the powder bed. This is a volumetric approach that is popular due to its simplicity. As opposed to volumetric approaches, ray tracing and Maxwell's equations offer increasingly more complex, but accurate methods to model the delivery of laser energy.

Remark. *In ray tracing, the laser beam is modelled as collimated rays in the shape of planar pulses that impinge on triangulated surfaces of the powdered layer and of the substrates and each ray's position is integrated in time. This triangulation procedure is expensive, but it allows for the computation of angles of incident, diffraction, and reflection for each ray. The*

amount of energy that is absorbed by the particle is based on the material property. Again, a computationally expensive computation. The rays are randomly generated inside of each planar pulse and they are released at a set frequency. The energy associated with each ray is a function of the amount of rays per wave and the number of pulses per second. Ray tracing is not an empirical method as it is based on Maxwell's equations.

We can compute the laser energy as a function of depth through a powder layer using a model akin to the Beer-Lambert equation as

$$I_z = e^{-\beta z}, \tag{3.29}$$

where z is the depth of the material point, measured from the top surface of the powder bed, and β is the optical extinction coefficient, as originally described by Gusarov et al. [41, 42], which is a non-dimensional parameter that models the internal reflections of the laser rays though a metallic powder layer, as opposed to a solid material. The optical extinction coefficient is a function of the volume fraction density ν of the layer and the diameter of the powder particles D:

$$\beta = \frac{3(1-\nu)}{2\nu D}. \tag{3.30}$$

In short, Equation 3.30 summarizes the results of computational experiments of rays penetrating through a powder layers with varying densities and particle sizes.

Lastly, we must acknowledge that the material will only absorb a fraction of the incident laser energy, governed by the material's absorptivity ($0 \leq \alpha \leq 1$) based on the wavelength emitted by the laser. This material property is non-dimensional. We assume a homogeneous value across the material since the laser's wavelength λ should not change during a printing procedure and experimental data on its dependence on temperature is difficult to analyze and explain. Therefore, $\hat{\alpha}(T, \lambda) \approx \alpha \; \forall \; T, \lambda$ in this study.

We are ready to compute the local laser power absorbed in $\mathrm{W/m^2}$ through the boundary of a material point that is a radial distance d away from the centerline of the laser beam and is at a distance of $\hat{z}$ below the top surface of the powder bed:

$$I(d, \hat{z}) = \alpha \; I_0 \; I_r \; I_z = \alpha \; I_0 \; e^{-2\frac{d^2}{\omega^2}} \; e^{-\beta \hat{z}}. \tag{3.31}$$

Figure 3.1 shows the radial distance d and the depth distance $\hat{z}$. Note that both exponential terms are always negative since their arguments represent positives quantities (beam radius, particle diameter, distance, and volume fraction), and so they always scale down the peak intensity I_0 quantity. In other words, mathematically, at a particular moment in time and at a particular material point, the incident laser energy exponentially decays away from the center line of the laser beam and below the top surface of the powder layer.

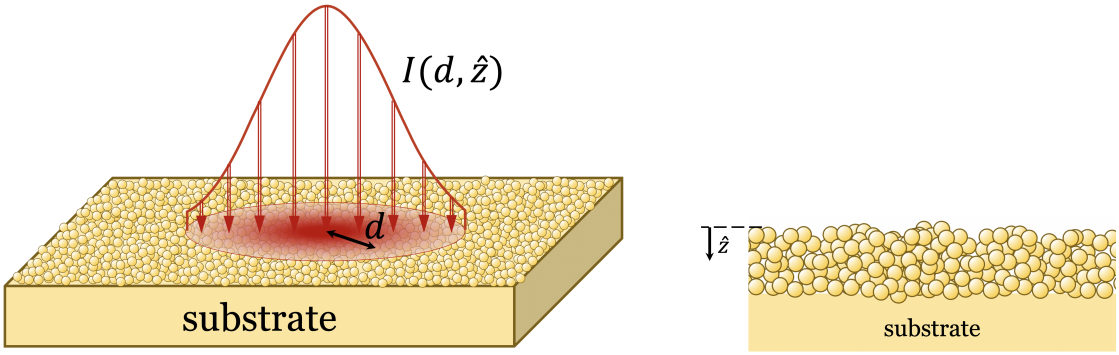


Figure 3.1: **Left:** A representation of the 2D Gaussian-shaped laser beam and the radial distance d to an arbitrary particle. **Right:** The depth distance $\hat{z}$, taken from the top of the powder layer.

Particle Deposition

The subject of particle deposition is critical to the study of an SLS process. As mentioned in Section 2.1, new layers are deposited onto the powder bed by transferring new powdered material using either a roller, wiper blade, or a spreader mechanism. The end result of this process is a collection of powder particles that have each settled into a final position by tumbling, rolling, bouncing, etc. We can say that the final arrangement of particles is unique and is impossible to recreate since in reality particle sizes fall within a distribution, not an exact value, and each particle will have geometries with unique imperfections. These non-deterministic properties of the raw material prevent a theoretical close packing of the particles throughout the volume of the powder layer.

There are several approaches to model a random arrangement of particles, each with diminishing complexity. Off course, the most accurate results would be one where the deposition tool is rendered and triangulated and the particles are set in motion by the contact forces from the tool. This approach would require a complex particle dynamics model complete with particle-particle contact forces, particle-wall contact forces, friction, environmental forces such as drag, and gravity. A second approach is a simple drop from a small height, after which the particles are allowed ample time to settle into a layer. Again, a particle dynamics model would have to be created and integrated in time.

A third approach is the one used in this work and it is the simplest, and yet, it still yields realistic volume fraction results for feasible particle arrangements. The method that will be explained in short obviates the need for simulating the dynamics of a set of particles: it solely requires an algorithm that is able to generate a random distribution of particles, just as the more complex methods do, but the particles are generated in their final resting positions. This random particle generation method uses an advancing front packing algorithm inspired by Valera et al. [143], which solely based on geometric calculations making it an efficient

approach. The number of generated particles is linear with respect to runtime.

Before describing the algorithm used in this work, it is notable to mention that the SLS system studied here has particle sizes and layer depths of very similar length scales. This allowed for a simplification of the arrangement of the perfectly spherical particles where all particles rest on the solid substrate and their diameter describes the layer height. Another assumption is that no particles lie entirely or partially in the shadow of other larger particles. Therefore, when seen from above the powder particles appear as 2D disks, with tangencies between particles, and filling the area of the top surface of the substrate. Therefore, we can use Valera's algorithm applied to 2D disks. After the algorithm finishes, the disks are turned into spheres and their positions projected to rest on the substrate's surface. We will use the terms *particle* and *disk* interchangeably throughout this section.

Remark. *This section will use symbols and letters similar to ones used in other sections. Please treat these symbols in isolation to avoid any confusion.*

The input of the algorithm is a distribution of particle radii from which to sample and create a new particle at each iteration. In our case the distribution is Gaussian with the tails chopped at a predetermined maximum and minimum radius limits, r_{max} and r_{min}, respectively. The output of the algorithm is a list of (x, y) center positions for all generated particles.

The spirit of the algorithm is such that if you have a root disk s_0 and another s_1 in its vicinity, or neighborhood, then we can find two possible locations to place a third disk s_2 with (s_0, s_2) and (s_1, s_2) tangencies. Note that we do not need for s_0 and s_1 to be mutually tangent, as long as the neighborhood of s_0 is sufficiently tight. Finally, one of the two positions is chosen at random to place the third disk. These two positions are identified via triangle geometry since we know the three distances between all centers and the coordinates of s_0 and s_1. Let s_0 be at position $s_0 = (x_0, y_0)$ and with radius r_0, and let s_1 be at position $s_1 = (x_1, y_1)$ and with radius r_1. The new particle has radius r_2. To find both solutions of its final positions, we perform the following geometric computation with $L_{01} = r_0 + r_1$, $L_{02} = r_0 + r_2$, and $L_{12} = r_1 + r_2$:

$$\begin{aligned} \theta &= \arctan 2(y_1 - y_0, x_1 - x_0) \\ \gamma &= \arccos\left(\frac{L_{01}^2 + L_{02}^2 - L_{12}^2}{2L_{01}L_{02}}\right) \\ s_2 &= (x_0 + L_{02}\cos(\theta \pm \gamma), y_0 + L_{02}\sin(\theta \pm \gamma))\,. \end{aligned} \tag{3.32}$$

First, an empty list `F` is initialized for storing particle IDs or center coordinates in the front. In this case, the particle coordinates are the identifiers since we are not interested in the order in which they are generated. The front generally describes particles at the perimeter of the set of particles for the latest iteration, and which are candidates for attaching the next particle. Then, two mutually tangent particles are generated with sampled radii and both are added to the front `F`.

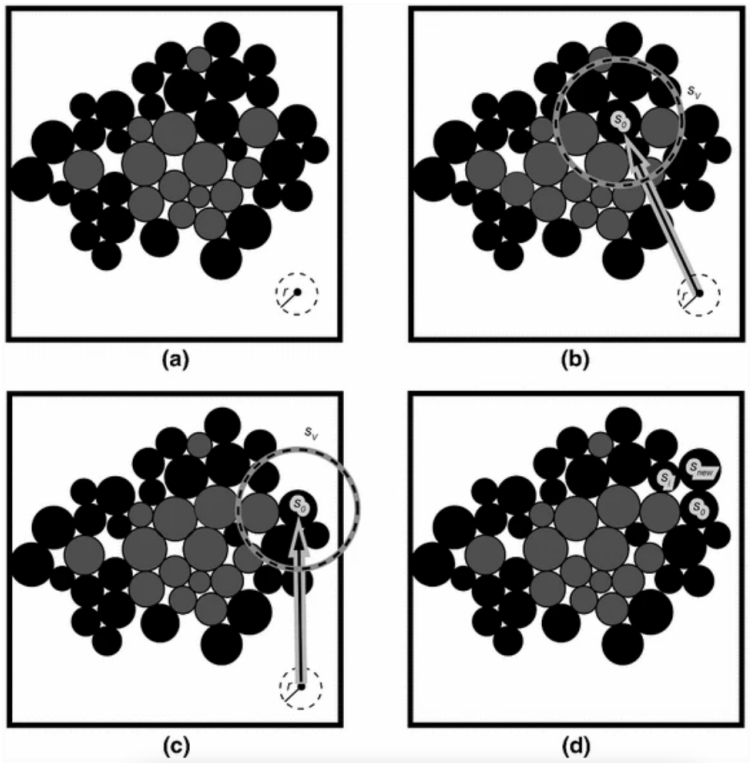


Figure 3.2: Example of an iteration of Valera's 2D particle deposition algorithm [143]. Particles in the advancing front `F` are in black; **(a)** a particle size is sampled from a distribution; **(b)** a particle in `F` is chosen at random as s_0 with its neighborhood s_v, but there is no feasible locations for a new particle; **(c)** a new s_0 is chosen and the previous is removed from `F`; **(d)** a particle s_1 is evaluated inside the neighborhood `V` and a new particle s_2 is added tangent to both s_0 and s_1 and to `F`.

The iterations can now begin, in which a new particle is always generated and placed before moving on to the next iteration. A new particle is generated by sampling a radius measure from the Gaussian distribution. To place it, one particle in `F` is randomly selected as the root particle s_0 and its neighborhood `V` is populated. `V` is defined as all particles with centers within a distance d_V from s_0, and it is a geometric tool to test whether s_0 has sufficient space around it to place the new particle. Here, $d_V = r_0 + r' + 2r_2$, where r_0 is the radius of s_0 and r' is the radius of a neighbor particle. Each particle in the neighborhood `V` of s_0 is tested based on overlaps with existing particles and with boundaries of the domain. Once a neighbor of s_0 is found to have at least one feasible position for s_2, then the algorithm stops searching through `V` and the new particle is placed in one of the two possible positions. Lastly, s_2 is added to `F`.

In the case where a particle in `F` is initially chosen as s_0, but its neighborhood is already filled with particles (no neighbors yield solutions for p_2 using Equation 3.32), then this

particle is not in the advancing front and so it is removed from `F`. The algorithm restarts the iteration with a different randomly-selected s_0 from `F`. Figure 3.2 from [143] illustrates the algorithmic order of one iteration, where the black disks are currently in `F`, a chosen s_0 is evaluated and removed from the front, and finally s_2 is placed after identifying a viable s_0 and with neighbor s_1.

The algorithm is guaranteed to finalize when `F` is empty. Simply, this means that all sides of the advancing front have either encountered a 2D domain boundary, filling the area. Mathematically, this means that all particles have been tested for geometrical solutions for the position of new particles, and none was found.

There is still one final post-processing step to perform. In order to return to a volumetric 3D domain, all (x, y) positions of the disks must be projected down to the top surface of the substrate, each by a distance equal to its radius. In this way the disks become spheres with a center at position (x, y, r_i), $i = 1, 2, ..., N_p$, if we assume the datum plane for the z-axis to be the top surface of the substrate, and we let N_p be the number of particles generated.

A Discrete Element–Finite Difference Method

To model the SLS process thermally, we must acknowledge that the domain incorporates two systems that are different in the geometry that describes them and the physical phenomena that governs them. These systems are the powder particles, modeled as perfect spheres and deposited as explained earlier in Section 3.2, and the solid substrate beneath the powder layer. We must also handle both systems simultaneously since they occur in parallel, and so their interactions must also be captured.

The work by Ganeriwala & Zohdi [37] provides a solid foundation for a numerical solver approach that captures the temperatures of the coupled particle-substrate domain. Here, we explain how their numerical model was adapted for studying the design space of an SLS process.

A Thermal Model for the Particles

The discrete element method (DEM) is used to capture the thermal interactions between the particles. This method firstly assumes that the Biot number is low since the size of the particles is sufficiently small that the internal conduction equalizes all thermal gradients inside of the particle. Thus, we assume that all particles have a homogeneous temperature and material properties, as shown in Figure 3.3. Also, since the particles are all exposed to the ambient nitrogen gas inside of the build chamber, we additionally assume these effects across the entire population of particles.

We modify the balance of energy in Equation 3.21 for the special case without deformations and the volumetric absorbed laser heat term:

$$\int_{\mathcal{P}} \rho C \dot{T} \, dv = -\int_{\partial\mathcal{P}} \boldsymbol{q} \cdot \boldsymbol{n} \, da + \int_{\mathcal{P}} \hat{\ell} \, dv, \tag{3.33}$$

Algorithm 1 A 2D advancing front algorithm for particle deposition
Inputs: Boundary limits of domain and a Gaussian distribution of particle sizes with minimum and maximum limits.
Output: A list of particle coordinates and sizes `P`.

1: Initialize two tangent disks $\{s_0, s_1\}$
2: Initialize particles `P` $= [s_0, s_1]$
3: Initialize advancing front `F` $= [s_0, s_1]$
4: **while** `F` is not empty **do**
5: Sample new particle radius
6: $s_0 \leftarrow$ random particle in `F`
7: Identify all particles in `V`, neighborhood of s_0
8: **for** each neighbor particle in a shuffled instance of `V` **do**
9: $s_1 \leftarrow$ neighbor particle
10: $s_2 \leftarrow$ solutions for candidate positions from Equation 3.32
11: **for** each position from a shuffled instance of s_2 **do**
12: Check for any overlaps with particles in `V`
13: Check for any overlaps with the boundaries
14: **if** there is any overlaps **then**
15: Remove s_0 from `F`
16: Return to Step 4
17: **else**
18: $s_2 \leftarrow$ position of new particle
19: **end if**
20: **end for**
21: **end for**
22: `P` $\leftarrow$ `P` $\cup\, s_2$
23: `F` $\leftarrow$ `F` $\cup\, s_2$
24: **end while**
25: Return `P`

and we express the local heat flux term as conduction between particles, and convection and radiation with the ambient inside the build chamber as

$$(-\boldsymbol{q}\cdot\boldsymbol{n})_l = \sum_{m=1}^{N_c} K\frac{T_m - T_l}{\|\boldsymbol{x}_m - \boldsymbol{x}_l\|}A_{lm} + h\left(T_l - T_\infty\right)A_l + \epsilon\sigma\left(T_l^4 - T_\infty^4\right)A_l. \tag{3.34}$$

Note that each term has units of heat, W. Equation 3.34 accounts for the total heat flux for particle l, its heat conduction with N_c contacted particles, each with subscript m, across contact area A_{lm}. Heat convection is modeled against a build chamber temperature T_∞, with a homogeneous heat convection coefficient h in W/(m^2·K), through an exposed particle surface area A_l. Radiation from the particle to the build chamber involves the dimensionless black-body emissivity of the material ϵ, and the Stefan-Boltzmann constant $\sigma = 5.67 \times 10^{-8}$

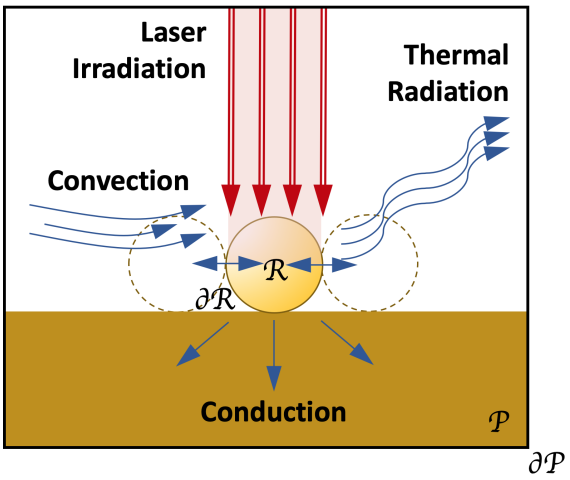


Figure 3.3: Control volume of an arbitrary particle's thermal contributions for the discrete element method (DEM).

in $\mathrm{W}/(\mathrm{m}^2{\cdot}\mathrm{K}^4)$. The exposed surface area to the build chamber is the projected area of top hemisphere of each particle: $A_l = \pi r_l^2$ and the contact surface area for heat conduction between particles:

$$A_{lm} = \pi h^2 = \pi \left(r_l \sin \left(\arccos \left(\frac{r_m^2 - r_l^2 - \|\boldsymbol{x}_m - \boldsymbol{x}_l\|^2}{-2 r_l \|\boldsymbol{x}_m - \boldsymbol{x}_l\|} \right) \right) \right)^2, \tag{3.35}$$

where we reference the triangle geometry based on Figure 3.4 and the law of cosines is used to derive the expression. However, due to the 2D particle generation scheme explained in Section 3.2 the contact area measure is minimal, which yields particle-particle conduction behaviors that are valid since this heat transfer phenomenon is insignificant compared to the short time scale of high heat input by the laser.

The heat input term in Equation 3.33 now describes the laser energy rate that is absorbed by all particles. For a single particle, however, the laser intensity $(\mathrm{W}/\mathrm{m}^2)$ in Equation 3.31 must be scaled by the particle's projected surface area A_i, such that

$$\hat{\ell} = I(d, z) A_l = \alpha \; I_0 \; e^{-2\frac{d^2}{\omega^2}} \; e^{-\beta z} \; A_l. \tag{3.36}$$

The semi-discretized heat balance of each particle is then,

$$\begin{aligned} \rho C \dot{T}_l = &\sum_{m=1}^{N_c} K \frac{T_m - T_l}{\|\boldsymbol{x}_m - \boldsymbol{x}_i\|} A_{lm} + h \left(T_l - T_\infty \right) A_l + \epsilon\sigma \left(T_l^4 - T_\infty^4 \right) A_l \\ &+ \alpha \; I_0 \; e^{-2\frac{d^2}{\omega^2}} \; e^{-\beta z} \; A_l, \end{aligned} \tag{3.37}$$

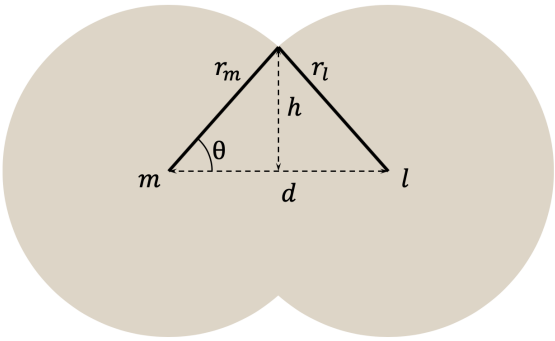


Figure 3.4: Contact surface area A_{lm} between two intersecting particles. The radius of A_{lm} is h and the distance $\|\boldsymbol{x}_m - \boldsymbol{x}_l\|$ is d in the diagram. This diagram was adapted from [37].

A Thermal Model for the Substrate

A Finite Difference (FD) Model will now be formulated for capturing the thermal field of the solid material, underlying substrate. This thermal data sheds light about the extend of remelted material that would ultimately fuse with melted powder material, thus fusing the new layer to the previously processed layer underneath. In this study, the substrate is discretized into voxels (a.k.a. "3D pixels"), which are cubes of homogeneous material described by the location of their centers. The centers form a rectilinear grid for the entire substrate domain. This discretization of the domain allows us to handle the flux of heat across its square boundaries in all 3 directions.

Knowing that the powder layer shields the substrate from being exposed to the ambient gas in the build chamber, then we assume that there are no convective and radiative cooling effects at the substrate. Therefore, the material at each voxel experiences a momentary heating effect from the laser and a prolonged cooling, and in a few brief instances heating, effect from voxel-voxel heat conduction. The local heat balance in Equation 3.25 for the case without deformations is used as a starting point.

Since we are employing the FD method, we discretize in space the derivatives in the heat flux term with a 7-point central differences stencil in Figure 3.5 and half-steps $\frac{\Delta x}{2}$ with the following generic expressions. Assuming a 1D system ($\boldsymbol{x} = x$), the derivative of the temperature T at location x is approximated as:

$$\frac{dT}{dx} \approx \frac{T\left(x + \frac{\Delta x}{2}\right) - T\left(x - \frac{\Delta x}{2}\right)}{\Delta x}, \tag{3.38}$$

and scaled by the heat conduction coefficient K, the expression simply becomes

$$K\frac{dT}{dx} \approx K\ \frac{T\left(x + \frac{\Delta x}{2}\right) - T\left(x - \frac{\Delta x}{2}\right)}{\Delta x}. \tag{3.39}$$

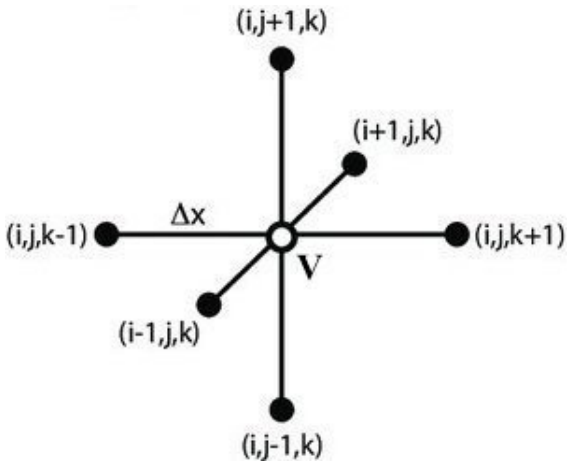


Figure 3.5: Central differences with a 7-point stencil to approximate Fourier's Law for heat conduction; adapted from [25].

The derivative of Equation 3.39, again with half steps is, thus,

$$\frac{d}{dx}\left(K\frac{dT}{dx}\right) \approx K\ \frac{\frac{dT\left(x+\frac{\Delta x}{2}\right)}{dx} - \frac{dT\left(x-\frac{\Delta x}{2}\right)}{dx}}{\Delta x}, \tag{3.40}$$

which simplifies algebraically to

$$\frac{d}{dx}\left(K\frac{dT}{dx}\right) \approx K\ \frac{T\left(x+\Delta x\right) - 2T(x) + T\left(x-\Delta x\right)}{\Delta x^2}. \tag{3.41}$$

Note that we do not need to compute the information at the half steps. This approximation makes it clear why half steps are necessary: these allow us to evaluate the second derivatives via central differences with the information at the neighboring voxels. Using this approach allows us to compute the flux at all non-boundary voxels with the FD scheme in Equation 3.41. In shorthand notation, the heat FD scheme in 1D for heat conduction at the i^{th} voxel becomes

$$\frac{d}{dx}\left(K\frac{dT_i}{dx}\right) \approx K\ \frac{T_{i+1} - 2T_i + T_{i-1}}{\Delta x^2}. \tag{3.42}$$

Extending the scheme to 3 dimensions, $\boldsymbol{x} = (x_1, x_2, x_3)$, the heat flux term in Equation 3.25 is approximated at the voxel with indices (i, j, k) as

$$\begin{aligned}\nabla \cdot (K\nabla T_{ijk}) &\approx \frac{d}{dx_1}\left(K\frac{dT_{ijk}}{dx_1}\right) + \frac{d}{dx_2}\left(K\frac{dT_{ijk}}{dx_2}\right) + \frac{d}{dx_3}\left(K\frac{dT_{ijk}}{dx_3}\right) \\ &= \frac{K}{\Delta x^2}\left(T_{i+1,j,k} + T_{i-1,j,k} + T_{i,j+1,k} + T_{i,j-1,k} + T_{i,j,k+1} + T_{i,j,k-1} - 6T_{ijk}\right),\end{aligned} \tag{3.43}$$

and since we are spatially discretizing the domain equally in all 3 directions, then $\Delta x_1 = \Delta x_2 = \Delta x_3 = \Delta x$.

Now, similarly to the heat input term for a particle, we can express the heat absorbed by each voxel using the Beer-Lambert equation for the attenuation of energy through a material. This expression explains how as the radiative energy travels through an infinitesimal region of material, the energy is either absorbed or reflected by the material. The portion that is reflected will continue on to the next region of material. In other words, each voxel will volumetrically absorb a portion of the incident laser energy and the balance will be incident on the voxel beneath. The Beer-Lambert equation is used as a multiplier for modifying the intensity at the surface as a function of depth z into fully dense material. The Beer-Lambert law, based on an intensity at the surface $\hat{I}$, is calculated as

$$I_{BL}(z) = a\ \hat{I}\ e^{-\int_0^z a\ dz} = a\ \hat{I}\ e^{-az}, \tag{3.44}$$

where a is a material property for the loss of laser flux through a dense material with units of 1/m. Thus, in general, a voxel at a depth of $z = \hat{z}$ will have the following volumetric power absorbed in W/m^3:

$$I(d, \hat{z}) = \alpha\ a\ I_0\ e^{-2\frac{d^2}{\omega^2}}\ e^{-a\hat{z}}. \tag{3.45}$$

However, the penetration of laser energy through a solid metal is typically in the order of a few nm. Therefore, we model the incident flux as affecting only the voxels at the top surface of the substrate and assume that no laser energy reaches the second layer of voxels. The total heat in W that reaches the top surface of the substrate is equal to the laser power that is able to penetrate the full depth of the local powder layer L_p, modeled by Equation 3.29 as

$$\hat{\ell} = I(d, L_p)A_v = \alpha\ I_0\ e^{-2\frac{d^2}{\omega^2}}\ e^{-\beta L_p}\ A_v, \tag{3.46}$$

where A_v is the surface area of the top face of one voxel.

Finally, we can gather Equations 3.43 and 3.46 into Equation 3.25 to assemble a local heat balance of a voxel. This expression becomes a semi-discretization in space at voxel T_{ijk}:

$$\begin{aligned}\rho C\dot{T}_{ijk} = &\frac{K}{\Delta x^2}\left(T_{i+1,j,k} + T_{i-1,j,k} + T_{i,j+1,k} + T_{i,j-1,k} + T_{i,j,k+1} + T_{i,j,k-1} - 6T_{ijk}\right) \\ &+ \alpha\ I_0\ e^{-2\frac{d^2}{\omega^2}}\ e^{-\beta L_p}\ A_v.\end{aligned} \tag{3.47}$$

Coupling the Particle and the Substrate Domains

Indeed, particles will exchange heat with the substrate below, and vice-versa. Thermally coupling both the particle system and the voxel system is achieved by an additional particle-voxel heat conduction term. This interaction is modeled by first identifying the (x_1, x_2) position of each particle center in reference to the FD voxel grid of the substrate in $\mathscr{O}(N_p)$ time, where N_p is the number of particles. This downward projection of the particle center

will be enclosed by a square that is defined by the four centers of the nearest voxels. Finding the enclosing voxels is achieved simply by flooring the quotient of the particle coordinates and Δx to get the minimal corner, say with indices (i, j), then its neighbors $(i+1, j)$,and $(i, j+1)$, and finally the opposite corner $(i+1, j+1)$. We can now make an assumption that the particle will engage in heat conduction with each these four voxels, and that the heat flux acts upon one fourth of the area of the top face.

Similar to the discrete element particle-particle heat conduction term, the particle-voxel heat conduction due to particle l is modeled at a voxel with indices (i, j, n_z) as

$$(-\boldsymbol{q} \cdot \boldsymbol{n})_{ijn_z} = K \frac{T_l - T_{ijn_z}}{\frac{1}{2}\Delta z} \frac{1}{4} A_{ijn_z} := Q, \tag{3.48}$$

where n_z is the number of voxels in the z direction, the length scale of heat conduction is half of the voxel height $\frac{1}{2}\Delta z$, and the heat flux enters the voxel through one fourth of its top surface A_{ijn_z}. In a general sense, let us define this coupling heat flux as Q. Finally, to preserve the total energy, the particle-voxel heat conduction due to these voxels at particle l is, thus, a summation of the heat transferred to these four voxels.

Time Integration Scheme

We use a first-order forward Euler time integration scheme to approximate the rate of change of temperature $\dot{T}$. The scheme is derived from a truncated Taylor series expansion at T_i^{n+1}:

$$\begin{aligned} T_i^{n+1} &= T_i^n + \Delta t \frac{dT_i^n}{dt} + \mathscr{O}\left(\Delta t^2\right) \\ \Rightarrow \quad \frac{dT_i^n}{dt} &= \frac{T_i^{n+1} - T_i^n}{\Delta t} + \mathscr{O}\left(\Delta t^2\right), \end{aligned} \tag{3.49}$$

where the superscript ^{n} is the time step index and Δt is the time step size. This scheme incurs a local truncation error at each time step with magnitude $\mathscr{O}(\Delta t^2)$. Additionally, due to its explicit nature, the scheme is simple to implement, incurring a low computational overhead, but its drawback is that it is numerically stable for a set of time step and spatial steps size combinations. More on the numerical stability condition later in this section. Special caution must be taken when employing explicit numerical schemes, as opposed to the implicit schemes. Implicit schemes, such as the backwards Euler scheme, derive the unknown temperature at the new new time step by using the unknown derivative at the new time step. Thus, iterative fixed point methods such as the root-finding Newton-Raphson method must be employed at each time step, adding computational complexity to the time integration scheme. An advantage of these methods include unconditional numerical stability (meaning no restrictions on the step size). The explicit forward Euler scheme will be used throughout this thesis.

This explicit scheme lets us solve for the temperature of the l^{th} particle at the next time step as a function of the current temperature at T_l and its particles in contact. The left-hand-side of the heat balance is approximated as

$$\rho C \dot{T}_l = \rho C \frac{dT_l^n}{dt} \approx \rho C \frac{T_l^{n+1} - T_l^n}{\Delta t}, \tag{3.50}$$

and, similarly, for the temperature at a voxel with indices (i, j, k):

$$\rho C \dot{T}_{ijk} = \rho C \frac{dT_{ijk}^n}{dt} \approx \rho C \frac{T_{ijk}^{n+1} - T_{ijk}^n}{\Delta t}. \tag{3.51}$$

We can now fully discretize the thermal balance for particles by substituting Equation 3.50 and the particle-voxel coupling term into 3.37,

$$\begin{aligned} T_l^{n+1} = T_l^n + \frac{\Delta t}{\rho C} \Bigg[& \sum_{m=1}^{N_c} K \frac{T_m^n - T_l^n}{\|\boldsymbol{x}_m - \boldsymbol{x}_l\|} A_{lm} + h \left(T_l^n - T_\infty\right) A_l + \epsilon\sigma \left(\left(T_l^n\right)^4 - T_\infty^4\right) A_l \\ & + \alpha \; I_0 \; e^{-2\frac{d^2}{\omega^2}} \; e^{-\beta z} \; A_l + \sum_{v=1}^{4} Q_v \Bigg], \end{aligned} \tag{3.52}$$

where Q_v is the heat exchanged with each of the four voxels beneath the particle. The discretized thermal balance for voxels is built by substituting Equation 3.51 and the particle-voxel coupling term in Equation 3.48 into Equation 3.47.

$$\begin{aligned} T_{ijk}^{n+1} = T_{ijk}^n + \frac{\Delta t}{\rho C} \Bigg[& \frac{K}{\Delta x^2} \left(T_{i+1,j,k}^n + T_{i-1,j,k}^n + T_{i,j+1,k}^n + T_{i,j-1,k}^n + T_{i,j,k+1}^n + T_{i,j,k-1}^n - 6T_{ijk}^n\right) \\ & + \alpha \; I_0 \; e^{-2\frac{d^2}{\omega^2}} \; e^{-\beta L_p} \; A_v + \frac{2K}{\Delta z} \left(T_l - T_{ijn_z}\right) \frac{1}{4} A_{ijn_z} \Bigg]. \end{aligned} \tag{3.53}$$

The expressions describe an integration scheme for the thermal field for the particles and for the substrate, respectively.

The question on the numerical stability of these scheme arises. An unstable scheme is characterized by incorrect, oscillating, and divergent numerical results. This numerical phenomena becomes a reality if any of the terms of the scheme, when all terms are algebraically distributed, flips its sign at each time step. We highlight the fact that the quantities $\frac{\Delta t}{\rho C}$, $\frac{K}{\Delta x^2}$, h, and $\epsilon\sigma$ are all strictly positive. Also, we can see that the coefficient for T_l^n and the coefficient for T_{ijk}^n do have the potential to be negative, which would flip the sign of their respective terms at each time step. All other terms are strictly positive. Thus, the numerical schemes must be set up so that it ensures positive coefficients for T_l^n and for T_{ijk}^n. This stability condition is regarded as the Courant–Friedrichs–Lewy (CFL) condition.

The coefficient in question for the particle scheme must be positive semidefinite, meaning that

$$\begin{aligned} 0 &\leq 1 - N_c \frac{K}{\rho C} \frac{\Delta t}{\|\boldsymbol{x}_m - \boldsymbol{x}_l\|} + h \\ \Rightarrow \quad \frac{\Delta t}{\|\boldsymbol{x}_m - \boldsymbol{x}_l\|} &\leq \frac{\rho C\,(1+h)}{N_c K}, \end{aligned} \tag{3.54}$$

and so the CFL condition of this scheme constraints us with ratios of time step and particle length scale sizes as functions of the thermal dynamics of the particle system. Likewise, the coefficient for the T^n_{ijk} term in the voxel time integration scheme must also be positive semidefinite to preserve numerical stability. The CFL condition for the voxel system is formulated as

$$\begin{aligned} 0 &\leq 1 - 6 \frac{K}{\rho C} \frac{\Delta t}{\Delta x^2} \\ \Rightarrow \quad \frac{\Delta t}{\Delta x^2} &\leq \frac{\rho C}{6K}. \end{aligned} \tag{3.55}$$

Here, we note that the numerical stability requirement on the time step is based on the square of the heat conduction length scale between voxels. As will be explained later the length scale of the voxels Δx is shorter than the length scale of the particles $\|\boldsymbol{x}_m - \boldsymbol{x}_l\|$, and since the Δx is squared, then the thermal dynamics of the voxels will dominate the selection of the time step size for ensuring numerical stability.

Boundary Conditions of the Substrate

The numerical scheme in Equation 3.53 is relevant for integrating the thermal field at all internal voxels. The boundaries of the domain will coincide with material that is sufficiently away from the laser beam energy so that the solution is not eschewed away from the numerical solution if the domain was infinitely large. Therefore, we assume that the substrate domain has Dirichlet boundary conditions at all side walls and at the bottom surface, and that the top wall is insulated, with the exception of voxels that are engaged in heat conduction with overhead particles.

To maintain generality and assuming that the build direction is in the $+z$ direction, the walls will have a constant temperature

$$T(0, y, z) = T(L_x, y, z) = T(x, 0, z) = T(x, L_y, z) = T_{wall}, \tag{3.56}$$

the bottom surface

$$T(x, y, 0) = T_{bottom}, \tag{3.57}$$

and the top has a prescribed flux

$$\frac{\partial T(x, y, L_z)}{\partial z} = Q_{top}. \tag{3.58}$$

In this work, (L_x, L_y, L_z) are the respective lengths of the domain in each Cartesian direction. The Neumann boundary condition is approximated with a backwards FD stencil that is second order accurate for first derivatives:

$$K\frac{\partial T(x,y,L_z)}{\partial z} \approx \frac{K}{\Delta z}\left(\frac{3}{2}T_{i,j,n_z} - 2T_{i,j,n_z-1} + \frac{1}{2}T_{i,j,n_z-2}\right) = Q_{top}. \tag{3.59}$$

3.3 Model Validation and Verification

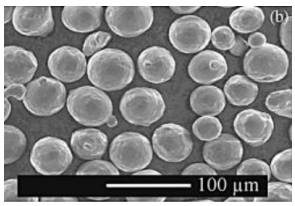
As explained by Ling [75], the validation and verification of computational models is a process to determine the extent of its agreement with physical phenomena in the real world. It is important to consider that this extent of agreement must be defined against an intent or application. It may be that the computational model is built on top of simplifying assumptions that might not agree with reality, but that its solution does provide an acceptable fidelity. Gauging the validity of a model can be done with qualitative or with quantitative methods. Qualitative methods allow for a visual comparison between the results in question and accepted results, whereas quantitative results will numerically compare the results by using metric or heuristic. The source of accepted results may stem either from experimental results or from results from peer-reviewed computational models, typically with higher-fidelity than the model in question.

Common quantitative validation techniques involve mean-based methods [120, 26], hypothesis testing methods [96, 122], area metrics [106], and reliability-based metrics [32, 31]. A review by Ling of some of these methods can be found in [75].

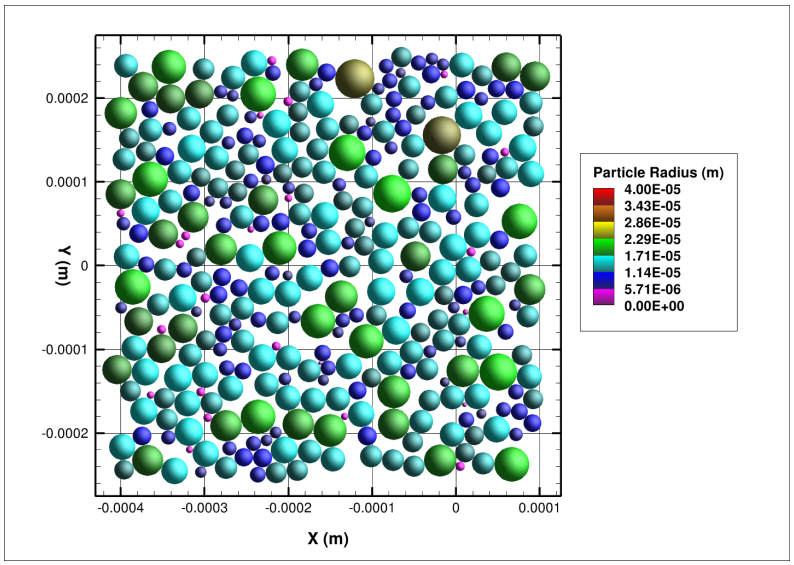
Table 3.1: Summary of the simulation parameters used for all thermal simulations.

Property	**Symbol**	**Value**	**Units**
Convection coefficient	h	40	$\mathrm{W/(m^2{\cdot}K)}$
Material emissivity	ϵ	0.33	
Stefan Boltzmann constant	σ	5.670373×10^{-8}	$\mathrm{W/(m^2{\cdot}K^4)}$
Material absorptivity	α	0.33	
Specific heat	C	815	$\mathrm{J/(kg{\cdot}K)}$
Material density	ρ	7,740	$\mathrm{kg/m^3}$
Thermal conductivity	K	32.4	$\mathrm{W/(m{\cdot}K)}$
Evaporation temperature	T_v	3,130	K
Melting temperature	T_m	1,700	K
Environment temperature	T_∞	363	K
Material initial temperature	T_0	363	K
Laser power	P	100 – 400	W
Laser spot size	ω	54	µm
Laser scan speed	v_ℓ	2.0	m/s

To validate the coupled DEM-FD thermal numerical model the thermal field is to be simulated for the material after a single pass of a laser beam and the domain properties as shown in table 3.1. Similar to [37], the results will be compared qualitatively with experiments conducted by Khairallah & Anderson [61]. This work is used as benchmark because of the experimental domain that this thesis targets: a thin powder layer over a solid substrate is processed with a Gaussian-shaped laser beam. In order to have a direct comparison, the material of choice is 316L stainless steel (316L SS) with a monomodal particle radius size distribution of 13.5 µm and a full width at half maximum (FWHM) range of 5 µm. When the particles were deposited with the advancing-front algorithm on a substrate area measuring $L_x = 4,000$ µm by $L_y = 500$ µm, such a distribution of particle sizes yielded 2,488 particles with a layer depth of about 27 µm. To add intuition, a reference micrograph of powdered 316L SS by [111] is shown in Figure 3.6, along with an output using Valera's advancing-front algorithm for comparison. Also, see Figure 3.7 for more details on the sampled size distribution. The volume fraction achieved was about 46.4%, which approximates 42% in [61] well.



(a)



(b)

Figure 3.6: Sample arrangements of spherical SS 316L powder from **(a)** a micrograph by Rombouts et. al [111] and **(b)** the advancing-front algorithm in this SLS model. Examples are illustrative and do not possess the same particle size distributions.

The laser power for the experimental setup in [61] was set at 200 W with a scanning speed of 2 m/s in the +x direction along the mid-plane $y = \frac{L_y}{2} = 0.25$ mm. The center of the laser beam is initially positioned inside of the domain at distance equal to $\frac{L_x}{10}$ from the boundary condition at the plane $x = 0$. This allows for a simulation window that follows the

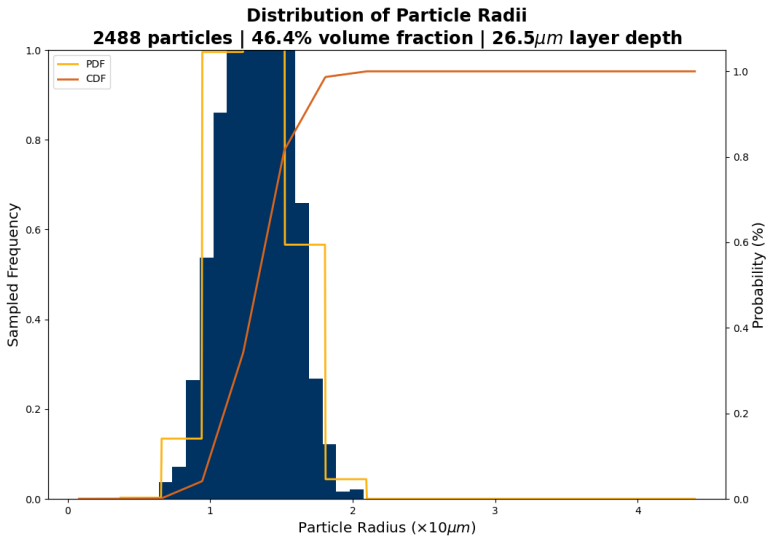


Figure 3.7: Radius size distribution after particle deposition sampled from a monomodal distribution of 13.5 µm and a FWHM range of 5 µm. The orange line represents the PDF of the sampled distribution and the green line is the CDF of the distribution.

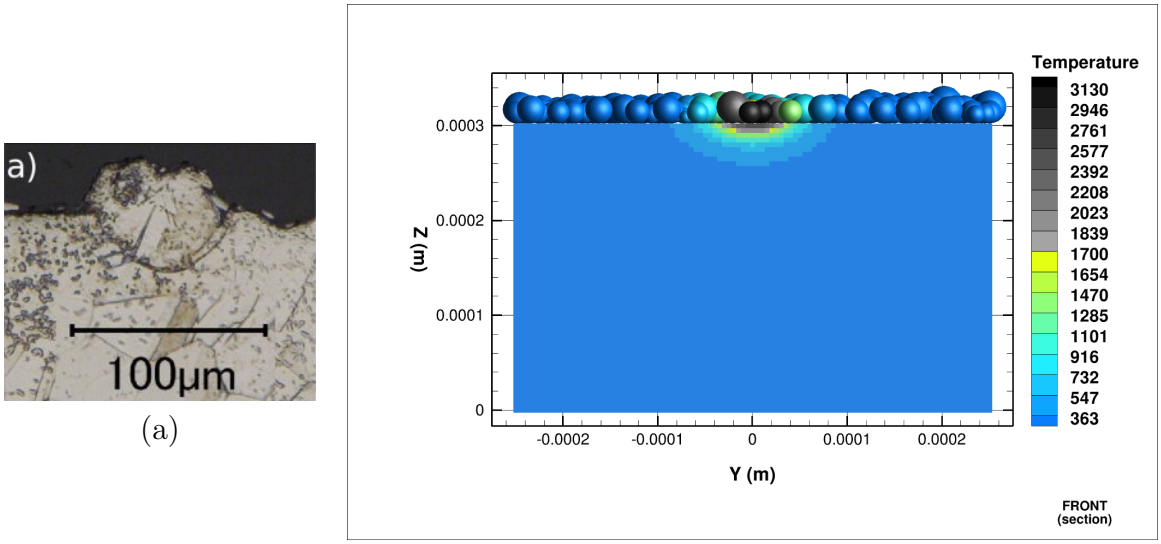
laser beam center and only solves for the temperature inside of the window. The window is built with the following boundaries:

- $\frac{1}{10}L_x = 400$ µm in the –x direction behind the laser beam
- $\frac{1}{40}L_x = 100$ µm in the +x direction ahead of the laser beam
- Full $L_y = 500$ µm substrate domain "width"
- Full $L_z = 300$ µm substrate domain "height"

Note that the laser center is always offset from the center for the simulation window. This is simply due to the fact that the melt pool shape always lags the laser spot and so the region that separates the laser center and the unprocessed powder ahead is very narrow. It is assumed that the laser will always irradiate the powder directly from overhead, thus eliminating any beam shape deformations from directing the laser beam at an incident angle.

The domain of the substrate was discretized in space with $\Delta x = \Delta y = \Delta z = 5$ µm. Trials were carried out and this spacing yielded an optimal setup for the tightest domain size before adverse effects were significant from boundary conditions and further refinements did not provide significant gains in accuracy. This accounts for a full-domain discretization of 801 voxels in the x direction, 101 voxels in the y direction, and 61 voxels in the z direction. The simulation window handled 101 voxels in the x direction, for a total of 622,261 voxels and 588,060 degrees of freedom (including the $z = L_z$ plane). To account for the CFL condition,

a maximum Δt size was computed for the substrate and the number of time steps n_t was adjusted to capture the traverse of the laser across the full domain based on the laser speed. The time step size was $\Delta t = 0.811$ µs and $n_t = 2{,}101$, including the initial condition, for a total simulated time of about 1.704 ms.



(a)

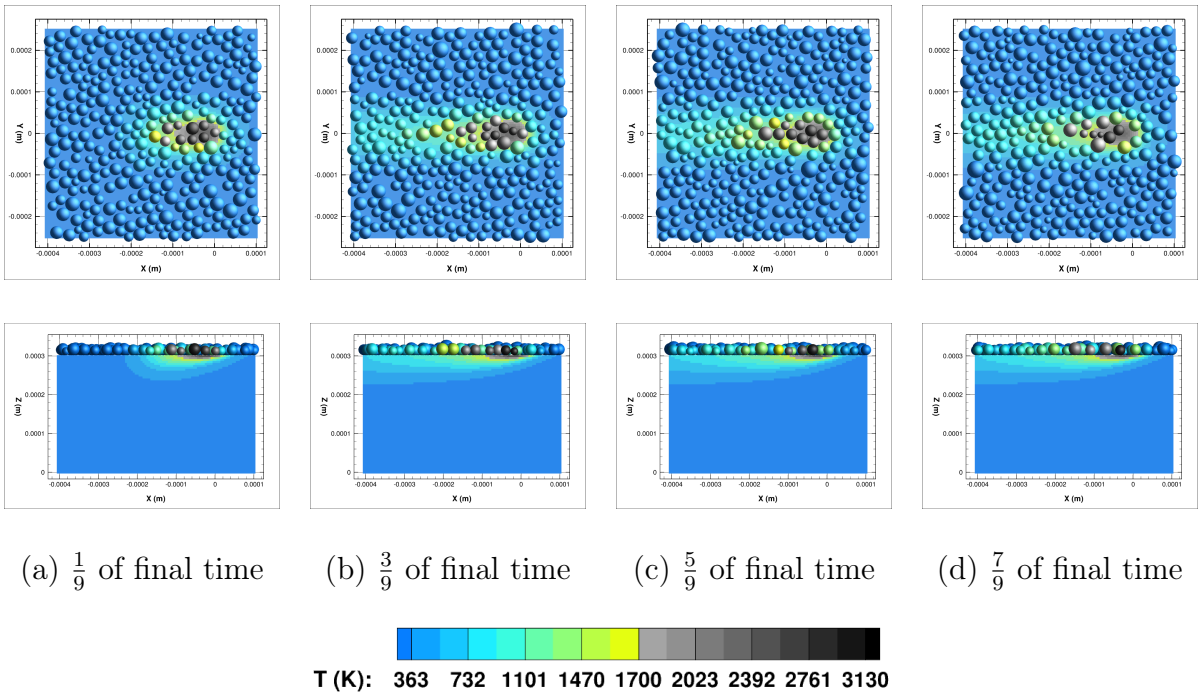
(b)

Figure 3.8: Comparison of frontal cross sections of the melt pool via **(a)** a micrograph from [61] of a frozen melt pool and **(b)** a temperature field output from the SLS simulation (grayscale color is melted material). Both arose from a laser power setting of 200 W with a scanning speed of 2 m/s and a monomodal particle radius distribution of 13.5 µm and a FWHM range of 5 µm.

The micrograph shown in Figure 3.8a shows a melt pool size with a depth and width of 30 and 72-75 µm, respectively. The height of the melt pool above the plane of the substrate is 26 µm, but this dimension is difficult to simulate in the numerical model since it omits the fusion and flow of the melt and the highly dynamic Marangoni surface tension effect. However, the mean powdered layer height of 27 µm in the simulation is measured based on the height of the particles, or, in other words, the mean particle diameter of the Gaussian size distribution. The experimental run by [61] is more akin to the selective laser melting (SLM) process and not the SLS process, thus approximating what would be the processed layer height with the unprocessed powder layer is adequate.

Figure 3.8b shows the extent of the melt pool within the grayscale color measuring 10 µm deep and 65 µm in width. To reconcile the discrepancy in the depth dimension with the experimental run, we must remember that the present SLS model includes simplifications on the allowance for the material melt to flow and for the Marangoni surface tension dynamics to take effect. Thus, we are removing the surface tension forces that push the material away

from the laser beam, creating a depressed denudation zone, and the gravitational forces that allow for the material to flow back into the depression after the laser beam has moved away. Additionally, the model does not evaporate material away that has surpassed 3,130 K. Thus, we are not capturing the advective and conductive mechanisms that would otherwise yield a deeper melt pool. However, we do notice that the width does approximate the experiment well by less than 10%.



(a) $\frac{1}{9}$ of final time (b) $\frac{3}{9}$ of final time (c) $\frac{5}{9}$ of final time (d) $\frac{7}{9}$ of final time

Figure 3.9: Screenshots for a top and a sectioned side view of the temperature field at $\frac{1}{9}, \frac{3}{9}, \frac{5}{9}$, and $\frac{7}{9}$ of the final simulation time.

Figure 3.9 shows a progression of screenshots that provide a top and a sectioned side view at four different time fractions of the validation run simulation, represented by each column. Notice that the screenshot at $\frac{1}{9}$ of the final time is still during the transient portion of the simulation since the initial laser beam spot has not reached the left wall at $x = -40$ µm and the melt pool is still developing. The steady state portion does not begin until just before $\frac{3}{9}$ of the simulation time, as demonstrated by material that has changed in temperature all the way to the left wall.

The thermal coupled DE-FD simulation was run in series using a single core in a MacBook Pro laptop computer, which took about 7 minutes and 15 seconds to run without writing any data to the hard drive. The simulation takes approximately 1 hour and 10 minutes when the temperature field data is saved onto the hard drive. Khairallah & Anderson [61] make use of ALE3D [86], a parallel multi-physics code, to resolve a thermo-mechanical system which took

about 100,000 CPU hours even with techniques to relax the CFL condition, such as artificially scaling the density of the material to alter the dynamics of the system favorably. This model is then a feasibility tool for scientists, process designers, and technicians to quickly generate thermal data for directional purposes. Subsequent computational experiments with higher fidelity simulations can be more targeted. Additionally, this coupled DE-FD simulation is excellent for gaining insights about a process parameter design space or generating ample thermal data for performing data-driven studies.

3.4 Parameter Studies

It is interesting to explore the capability of the DE-FD thermal model by varying controllable input parameters of the SLS process. Most importantly, the effects of the chosen powder particle Gaussian size distribution is crucial since these results translate into material specifications when sourcing raw material for experimental and production builds. Thus, we explore the effects of varying the mean particle size and the full width at half maximum (FWHM) of the Gaussian distribution. The FWHM has units of meters and is related to the standard deviation σ of particle sizes through the following relation:

$$\sigma = \frac{2\sqrt{2\ln 2}}{\text{FWHM}} \approx \frac{2.3548}{\text{FWHM}}. \tag{3.60}$$

Thus, the FWHM range corresponds to about 2.3548 standard deviations and it bounds about 76% of the area under the Gaussian curve, regardless of the mean particle size.

Table 3.2: Summary of the experimental cases for the particle size distribution parameter study. The mean particle size μ and the FWHM range are varied across a low, medium, and high setting, while the laser power and scan speeds are fixed.

	Mean Size	**FWHM**	**Laser Power** (W)	**Laser Speed** (m/s)
Case 1	medium	medium	200	2.0
Case 2	low	medium	200	2.0
Case 3	high	medium	200	2.0
Case 4	medium	low	200	2.0
Case 5	medium	high	200	2.0

The following parameter study is based on a laser power of 200 W, a setting within the range of 100 – 400 W, which is typically employed by experimental researchers such as [58, 61, 62] to name a few. Lasers can even be set at around 50 W for purely sintering operations, relatively smaller mean particle sizes, or shallow powder layer depths. Also, the scanning speed of the laser is fixed for the simulations so that the volumetric heat per second that is delivered by the laser beam stays constant throughout the experiments.

Following the studies on the particle size distribution, the laser power setting will also be varied to highlight the effects on the temperature field, with focus on the dimensions of the melt pool shape. We will explore cases where the laser power is set at 100 W and 400 W, the minimum and the maximum extremes, respectively, for the laser in the Concept Laser M2 printer employed by Kamath [58] in their experimental single-track runs.

Variations on the Particle Size Distribution

Three levels for the mean particle size μ and the FWHM are defined as low, medium, and high:

$$(\mu_l, \mu_m, \mu_h) = (10, 13.5, 30)\,\mu\text{m} \tag{3.61}$$

$$(\text{FWHM}_l, \text{FWHM}_m, \text{FWHM}_h) = (2, 10, 20)\,\mu\text{m} \tag{3.62}$$

A total of 5 runs were performed: one keeping both μ and the FWHM range at their medium settings, two by varying μ at its high and low settings and keeping the FWHM range at its medium setting, and finally two by varying the FWHM range at its high and low settings and keeping μ at its medium setting. Table 3.2 summarizes the parameter cases in this study and Figure 3.10 represents the sampled particle size distributions and their probability (PDF) and cumulative (CDF) density function plots. The first row in the figure pertains to the validation in Section 3.3, or Case 1 here, the second column varies the mean size for Cases 2 and 3, and the third row varies the spread (FWHM) of the sizes for Cases 4 and 5. Taking Case 1 as the point of comparison, Cases 2 and 3 vary μ, which simply shifts the sampled distribution to the low and high means, as expected. It is interesting to note that the random particle generator explained in Algorithm 1 yielded about 4,700 particles in Case 2 since its distribution is shifted to the smaller particle sizes and about 500 in Case 3 with larger particles. This behavior is arises since both cases fill the same bounding box. Cases 4 and 5 represent sampled distributions centered at μ_m with high homogeneity of particle sizes and a diverse range of sizes, respectively. The CDF plots (red line) are evidence of this difference in the spread of the sampled distributions due to its steepness.

Although these cases sampled vastly differing sizes, they are still centered around the same mean size and this led to the generation of very similar particle counts. Case 4 sampled 2,373 particles and Case 5 just two more. This result is attributed to the packing arrangement tendency for the smaller particles to fit in the interstitial spaces between larger particles, which is true for Case 5, over the same surface area where the same number of mean-sized particles would fit, which approximates the sampled distribution of Case 4. Additional runs of Cases 4 and 5 led to 2,352 and 2,380, respectively, both within 1% of the other results in Figure 3.10.

The particle generation algorithm requires the definition of a maximum and a minimum particle radius to eliminate the chance of size outliers that are not physically permissible with extremely large radii or negative radii. These outliers have the potential to disrupt the geometrical assumptions of the algorithm, which would generate errors when attempting

to find pair-wise tangencies. Thus, all simulations are run with a minimum particle radius of 1 µm and 40 µm, which correspond to the positions of 0.1 and 4 on the horizontal axes in Figure 3.10. Cases 2 and 5 show that the sampled distributions have been clipped at the minimum radius and, less severely, Case 3 at the maximum radius. When sampling the Gaussian distribution, these clipping procedures forced the algorithm to re-sample a different size and this led to a wider distribution away from a normal distribution.

To fully appreciate the arrangements of sampled particles in all 5 cases a visualization is in order. Figure 3.11 shows a representative portion of the full domain for each case. The view is directly from overhead, providing a demonstration of some of the characteristics yielded by the advancing-front algorithm, such as the randomness of the packing arrangement of the particles, the size of the interstitial spaces between the particles, and the extent at which the particles fill the space. Also, the colorization represents the radius of each particle, highlighting particle size ranges with discrete colors and differentiation within each color with a slight, continuous hue change from darker for smaller radii to light for larger radii. Variations between the sampled size distributions from Figure 3.10 become evident.

Case 1 is centered at 13.5 µm with a distribution that fits within the radius limits, and indeed, we see a spread of about 5 colors, mainly light blue-colored particles, followed by green and dark blue, and a significantly lesser amount of pink and yellow. Case 2 shows a plurality of dark blue particles, followed by its adjacent colors light blue and pink, and a few green particles. We notice also that Case 3 has a similar spread of color representations, but toward the opposite end of the spectrum, dominated by orange and yellow and some green and red. The higher particle count of Case 3 over Case 2 becomes evident, but the volume fraction of the powder layer occupied by the material is about 1.9% higher in Case 2 (48.5%) compared to Case 3 (46.6%). This similarity in volume fractions is attributed to the fact that the packing density is generally the same if we were to zoom-in on Case 2 or zoom-out on Case 3, assuming an infinite domain, because the FWHM of the Gaussian size distribution from which the particle radii were sampled was kept equal. The volume fractions for size distributions with means shifted towards the larger sizes will yield slightly lower volume fractions since we do have a bounded domain. Thus, adding larger particles between the advancing front and the bounds becomes more challenging and so packing density is locally reduced.

Now, varying the spread of the size distribution around the same mean (Cases 4 and 5) yield particle arrangements dominated by the light-blue color, but we do see that Case 4 pertains to a narrow FWHM range in that almost all particle radii fall within the light-blue color (11.4 - 17.1 µm) and Case 5 to a wide FWHM range with the inclusion of all color hues. Indeed, Figure 3.10e shows a sampled distribution that spans most of the range. Comparing particle volume fractions, Case 4 achieved a measure of 44.9%, the lowest across all cases, owing to the deficiency of smaller particles that are able to fill the interstitial spaces between larger particles. Conversely, Case 5 does contain small particles which increase the packing density to increase the volume fraction to 47.9%.

Cases 1 – 3 showed high volume fractions. This is attributed to the FWHM range which, compared to the mean size for the distribution, makes small particles available to increase

the packing density and it excludes grossly large particles that may potentially crowd out new particles during the advancing front algorithm. Thus, there is an optimal FWHM range away from the low and high extremes that maximizes the volume fraction.

Figure 3.12 shows the numerical thermal field outputs from the SLS model for each of the 5 cases presented thus far. A representative melt pool is shown for each case in multiple views. Taking Figure 3.12a, for example, the first row shows the entire domain in a top view and an isometric view, whereas the bottom row shows a longitudinal cross section at the mid-plane of the y-axis, which coincides with the path of the laser, labeled as a "side" view and a transverse cross section at about the $x = -40$ µm labeled as a "front" view. Special attention will be paid to the cross section views since they show critical dimensions of the melt pool: the depth in the -z direction, the width in the y direction, and the length in the x direction. These length measures are critical to characterize since they are heuristics for achieving a dense part build. The depth of the melt pool gauges how much did the melted particle material fuse with the re-melted solid layer below, the width dictates the hatch spacing between adjacent tracks within the same layer, and the length can be used as an input for building a material temperature history for simulating the final crystal growth of the microstructure during material re-solidification.

Remark. *In this work, however, we are only interested in generating temperature data of the substrate material, and so this section focuses on these melt pool shape measurements as points of comparison between the test cases to fully illustrate the capabilities and results of the SLS model.*

Variations on the Laser Power Setting

Table 3.3: Summary of the experimental cases for the laser power parameter study. The laser power P is varied across a low, medium, and high setting, while the particle size distribution parameters (μ and FWHM) are based on Case 1.

	Mean Size	**FWHM**	**Laser Power** (W)	**Laser Speed** (m/s)
Case 1	medium	medium	200	2.0
Case 6	medium	medium	100	2.0
Case 7	medium	medium	400	2.0

Additionally, Figure 3.13 depicts the same temperature field information for variations on the laser power, P_l at 100 W (Case 6) and P_h at 400 W (Case 7). The medium setting is taken as 200 W with the same particle size mean and FWHM range as in Case 1. Table 3.3 summarizes these settings.

The melt pool shape results for all 7 parameter study cases are shown in Table 3.4. These results are heuristics for the volumetric heat that is able to penetrate the powder layer and the heat conducted from each particle into the substrate. Recall Gusarov's [41]

optical extinction coefficient β for a metal powder layer, which is present in the laser term in the discretized thermal governing equation. This extinction coefficient is a function of the particle diameter and the volume fraction of the layer, so a densely packed layer and larger particles will prevent the laser heat from penetrating down to the substrate. Furthermore, the deeper the material point inside the layer, the more extinction of the heat that occurs.

Table 3.4: Melt pool shape measures in µm for all parameter studies. The depth is taken in the z direction, the width in the y direction, and the length in the x direction. Ranges on the length for Cases 5 and 7 correspond to disjoint material melt pools.

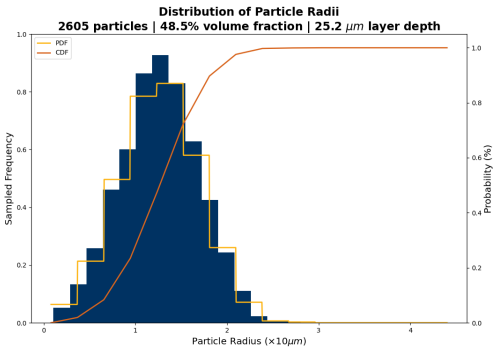
	Variation	**Depth** (µm)	**Width** (µm)	**Length** (µm)
Case 1	-	15	60	135
Case 2	μ_l	15	75	190
Case 3	μ_h	10	30	50
Case 4	FWHM_l	15	60	155
Case 5	FWHM_h	15	65	110 – 125
Case 6	P_l	-	-	-
Case 7	P_h	25	85	295 – 320

We notice with Case 2 that when the particles are smaller than in Case 1, then the layer is generally shallower and the β coefficient is such that the layer attenuates less laser heat, leading to a melt pool that is wider and longer by 25% and 41%, respectively. The converse situation occurs with Case 3, where larger particles yield a less-penetrating, narrower and a shorter melt pool by 50%, 50%, and 63%. These results show a strong correlation between the melt pool shape and the particle shapes, all else equal. This is not exactly the case for the Cases 4 and 5, where the FWHM range was varied. We note that the depth and width of the melt pools were not impacted as compared to Case 1, but we do see a higher concentration of heat within the melt pool in Case 4 and a lower concentration of heat in Case 5. This led to 15% lengthening of the melt pool for a more homogeneous particle size distribution in Case 4 and a shortening of the melt pool by 19% with a broader spread of particle size in Case 5. These results highlight a milder sensitivity of the heat that penetrates the powder layer to variations of the spread of the particle sizes. Cases 2 – 5 highlight the importance for the engineers and scientists realizing SLS builds to specify raw material that closely abides to a tighter specification on the 50-percentile value (mean) of the powder particle sizes with a less restrictive tolerance around the mean.

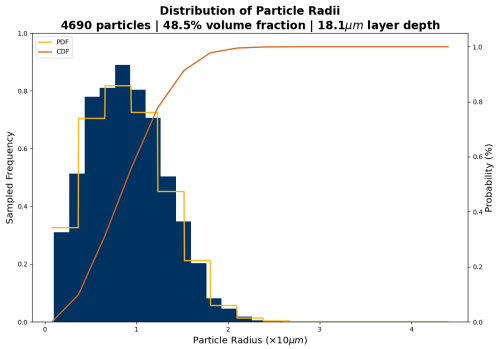
Cases 6 and 7 varied the laser power setting between 100 and 400 W. This parameter governs the intensity of the laser heat throughout the Gaussian beam shape, and thus it is directly related to the amount of heat that penetrates the powder layer into the substrate, all else equal. Indeed, decreasing the laser power to 100 W did not provide enough heat to penetrate into the substrate for melting the material. This is to be expected since polymeric powders are sintered between 50 – 100 W. Case 7 represented a typical maximum setting for

lasers used in SLS process. This high heat intensity yielded a melt pool that was 67% deeper, 42% wider, and more than twice the length, compared to Case 1. We also see that between Figures 3.12 and 3.13, that Case 7 at 400 W imparts enough heat that melted particles trail the laser position by as much as 320 µm, compared to the 160 µm for Cases 1, 4, and 5, which had the same μ and FWHM ranges. This effect provides additional heat into the substrate behind the melt pool, which further promotes its elongation. This parameter study highlights the need for understanding well the interactions between the laser beam energy, such as its wavelength and its power range, with the powdered material to be processed so that adequate laser equipment can be employed.

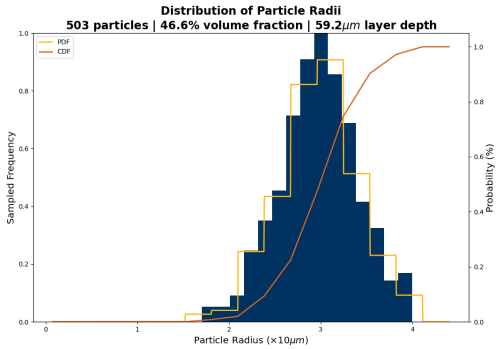
To conclude the melt pool overall size is governed by the mean of the particle size distribution, which is entirely due the resulting effective powder layer depth globally. A particle size distribution centered around relatively small particles will yield a larger melt pool, whereas a larger mean particle size will yield a smaller melt pool size. Furthermore, a stronger laser is able to increase the size of the melt pool, thus, if a particular width or depth is desired then the laser power setting and the mean particle size must be considered together.



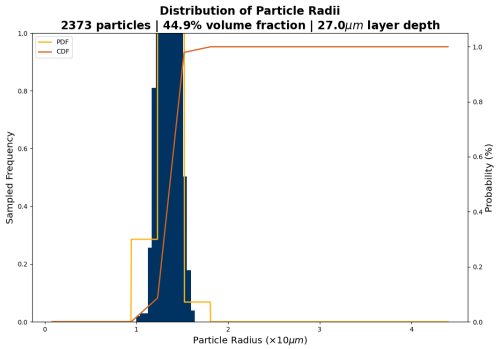
(a) Case 1: μ_m, FWHM_m



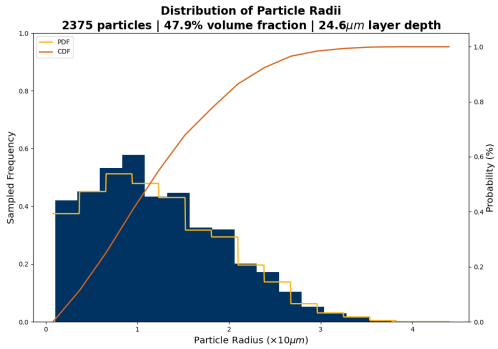
(b) Case 2: μ_l, FWHM_m



(c) Case 3: μ_h, FWHM_m

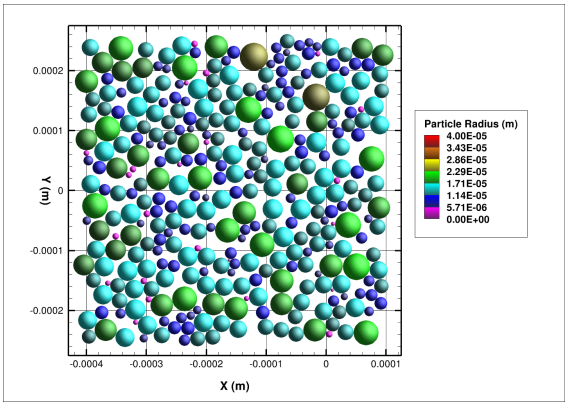


(d) Case 4: μ_m, FWHM_l

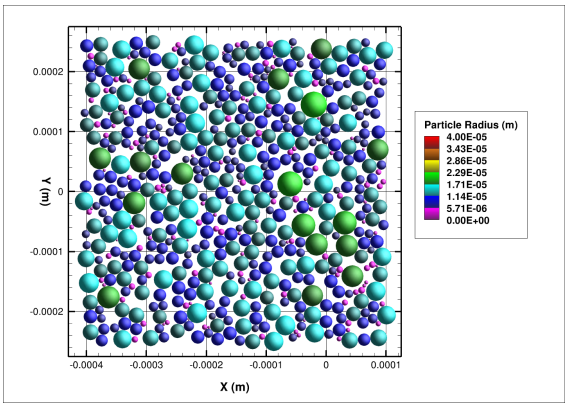


(e) Case 5: μ_m, FWHM_h

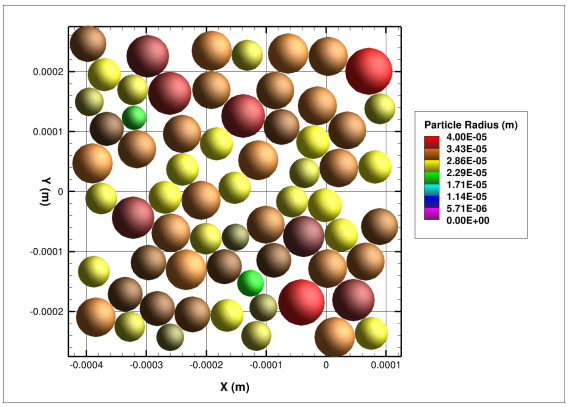
Figure 3.10: Sampled particle size distributions (left y-axis) and their probability and cumulative density function plots (right y-axis) for the parameter study on μ and FWHM. The first row pertains to the validation case, the second column varies the mean size, and the third row varies the spread (FWHM).



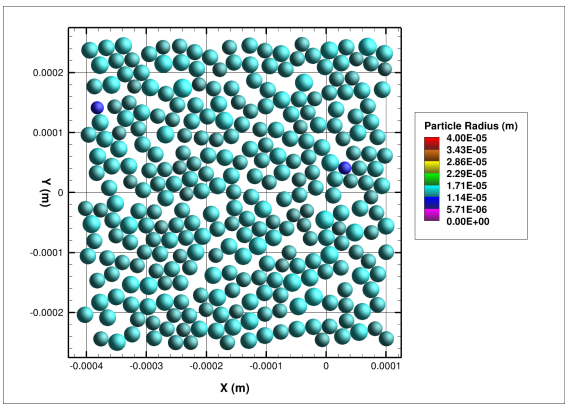
(a) Case 1: μ_m, FWHM_m



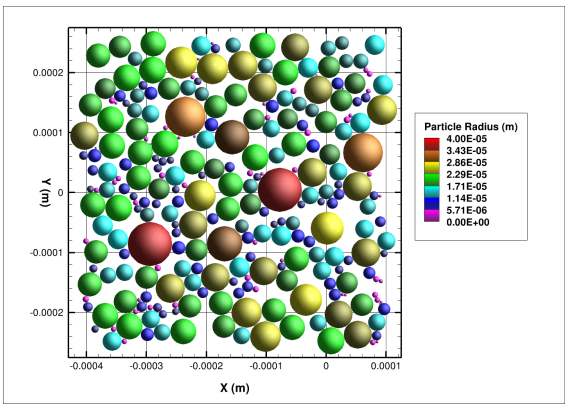
(b) Case 2: μ_l, FWHM_m



(c) Case 3: μ_h, FWHM_m

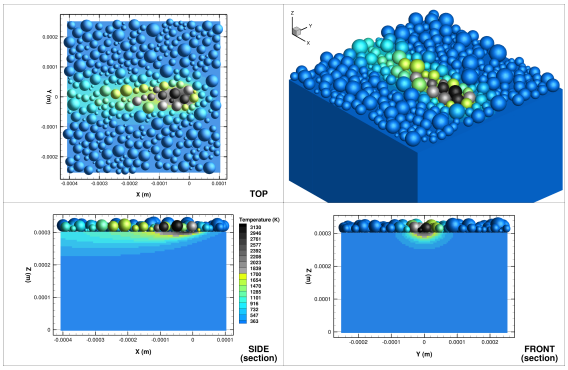


(d) Case 4: μ_m, FWHM_l

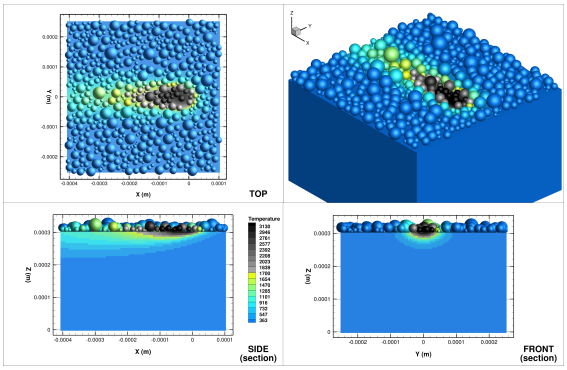


(e) Case 5: μ_m, FWHM_h

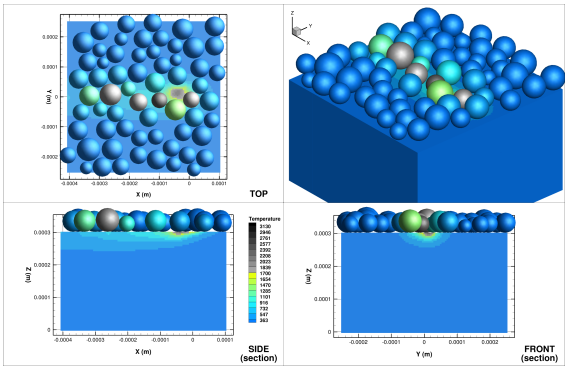
Figure 3.11: Particle arrangements according to the parameter study on the mean particle size μ and the FWHM range. The particles are colorized based on their radii.



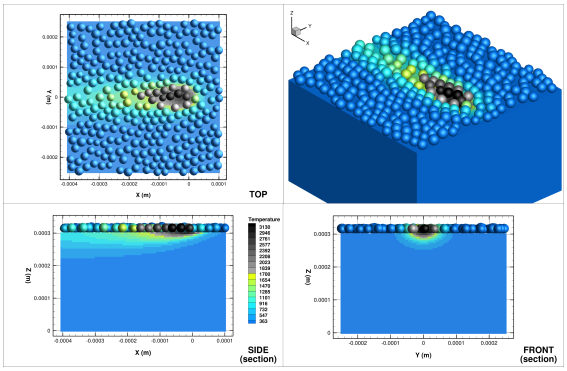
(a) Case 1: μ_m, FWHM_m



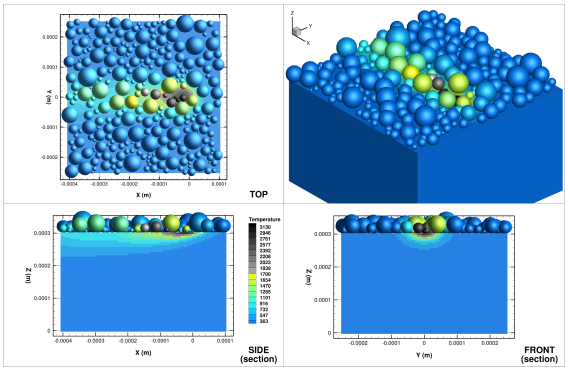
(b) Case 2: μ_l, FWHM_m



(c) Case 3: μ_h, FWHM_m

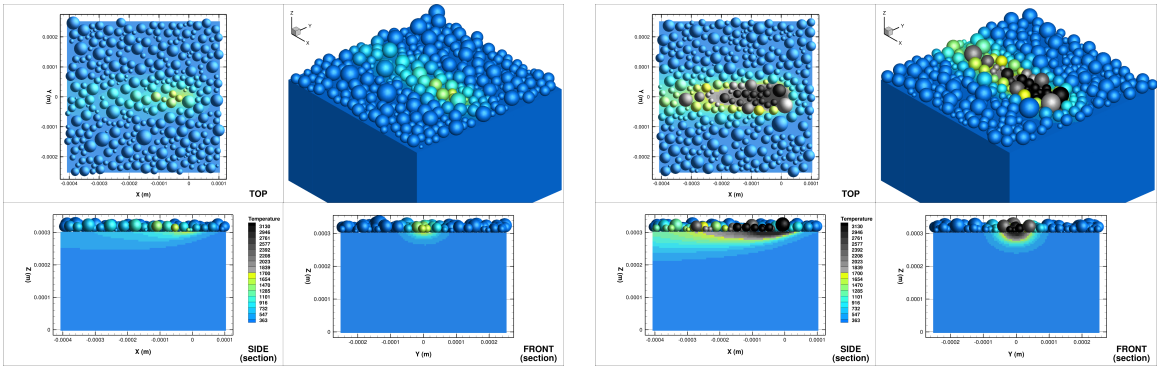


(d) Case 4: μ_m, FWHM_l



(e) Case 5: μ_m, FWHM_h

Figure 3.12: Top and isometric views of the melt pool, including cross sections at $y = 0$ µm and $x = -40$ µm. Grayscale material is above 1700 K, the melting temperature for stainless steel 316L.



a) Case 6: Laser power at 100 W (b) Case 7: Laser power at 400 W

Figure 3.13: Variation over laser power. Top and isometric views of the melt pool, including cross sections at $y = 0$ µm and $x = -40$ µm. Grayscale material is above 1700 K, the melting temperature for stainless steel 316L.

Chapter 4

Quantifying Uncertainty and Variability

In this chapter, we will delve deeper into the SLS process to get more intuition about the factors and inputs that may or may not affect its performance. To accomplish this, we will focus on the study of quantifying uncertainty (UQ) and variability during the processing of material for AM. In general, UQ is defined as a statistics-based mathematical process that attempts to characterize the variations in either a physical or a computational system to determine the likelihood of a certain outcome, or performance, based on the system's unknown properties or unclear inputs.

Examples of UQ studies in manufacturing and engineering analysis by the author of this thesis and other collaborators will be presented in this chapter before we are prepared to study the sources of uncertainty of the SLS process presented in Chapter 3.

4.1 Thermal Modeling and Uncertainty Quantification of Tool for Automated Garment Assembly

UQ in Manufacturing

As a general application of UQ in manufacturing, this section will showcase an example [17] that models the thermal field of a metallic blade that is used as a tool to fold textiles in an automated garment assembly cell. The study makes use of a thermal steady state FEM model and the field data is used to perform various UQ techniques to analyze the performance of the system. Based on a defined QoI, these UQ techniques are threefold:

- The sensitivity of the QoI with respect to the input parameters,
- The characterization of the design space, and
- The propagation of uncertainty through the numerical solver.

Current State of the Garments Industry

The garment industry is a collection of supply chains, primarily comprised of fiber production, yarn and fabric production, and finally the assembly of consumer products.

While recent technological advances have enabled robots to manufacture high-value goods like car parts and electronics, the garment industry is still largely reliant on manual labor. Some assembly steps, such as attaching buttons and zippers, have been automated; these tasks are performed by single-purpose machines which are inherently more costly and more difficult to change between tasks compared to human workers. Little significant automation currently exists for the labor-intensive cutting and sewing steps required for garment assembly. With cheap labor readily available and international supply chains historically dependable, there has been little incentive to innovate [95, 73]. However, there have been several recent initiatives put forth by non-governmental organizations, researchers, and other stakeholder organizations to calculate and determine local living wages for garment workers [124], generating pressures on clothing brands and factories to identify other cost-effective fabrication processes. For this reason and in the spirit of the Industrial Revolution 4.0, the development of automation technologies for cut-and-sew operations is a promising avenue for modernizing garment assembly supply chains.

Challenges with Automation

On the technical side, handling limp textiles is more challenging for robots than handling rigid objects. Unlike a rigid body, a piece of fabric may be folded, wrinkled, or otherwise distorted, presenting challenges for both the mechanics of grasping and for computer vision algorithms. This makes manipulating textiles a challenge for robotic systems [83].

One method to address this challenge is to stiffen textiles during assembly and reverse the process once finished [165]. A novel process may be used to treat textiles with a thermoplastic laminating layer. The laminating material can be removed with hot water to produce a conventional final garment. Stiffening the fabric sheets in this manner allows mature robotic technologies from other industries to be used in the garment industry, but requires the development of new manufacturing processes and tools to work with the novel textile-polymer sheets.

To introduce permanent bends easily and repeatedly along straight lines, this work will consider a process that uses heat to soften the thermoplastic binder, then applies mechanical force to bend the sheet along the softened line. Once the bend cools, the sheet solidifies and can be manipulated as a rigid body for additional assembly steps. One potential workflow using this idea involves supplying small robotic arms with a constantly heated metal attachment. To remain comparable to a human worker, the robotic arms must be general-purpose units capable of picking up, sewing, and bending a variety of work pieces.

The design of the bending tool was inspired by a "blade" tool used in robotic origami folding by researchers at Carnegie Mellon University [5] with the addition of a heating

element. If this blade tool is hotter than the working temperature of the stiffening agent, it will greatly reduce the effort required to create permanent bends.

Variability and Uncertainty in Manufacturing

The study of UQ can be traced back to Sir Ronald Fisher's seminal paper, The Design of Experiments [34] published in 1935, where he discussed the principles of experimental design for the agricultural industry. Since then the study of variability and uncertainty has been widely used in wide-ranging applications such as the Antarctic ice sheet melting model [15], computational mechanics models [74], human toxicity dose estimation models [54], among others. Sacks et. al [116] provide a window into the state of the art of UQ in the late 1980s and Abdar et. al [1] gives a more contemporary review of the role of UQ in deep learning applications for optimization and decision making.

Several UQ frameworks have been constructed for performance evaluation of a wide array of manufacturing processes. These frameworks have shown merit in that they showcase how different sources of uncertainty in complex models can interact with each other, even in the total or partial absence of experimental data. Typically, Bayesian approaches estimate the network of uncertainty sources through a manufacturing system and then Monte Carlo approaches serve to calibrate the system for maximizing the accuracy of the UQ framework. In subtractive manufacturing, UQ has been performed for fatigue life predictions of welded structures [129] by studying variations in weld shape, residual stress, loading conditions and microstructure, as well as data uncertainty from measurement and numerical errors. UQ has also been used in milling stability region estimation in the parameter space [72] and in a deep drawing process [51] based on variations of material properties and friction forces inherent in the drawing operation. More recently, UQ frameworks have provided excellent approaches for studying highly complex methods such as selective electron beam melting (SEBM) [150]. Additionally, Hu and Mahadevan [50] provide a survey of current work in UQ for additive manufacturing and immediate needs. Lastly, Nannapaneni and Mahadevan [94] provide a systematic framework that propagate epistemic or aleatoric source of uncertainty in any manufacturing process application, showcasing examples in an injection molding process and a welding operation.

This work develops a thermal finite element model of the proposed garment assembly tool. This model is then used to assess the sensitivity of tool designs to environmental conditions such as the ambient temperature, and design specifications such as the blade dimensions or the input power. In section 4.1, we provide more information about the construction of the blade tool, the formulation of a governing equation to solve for the thermal field, and the theory and uses of uncertainty quantification. Then, in section 4.1 we apply the governing equation to a Finite Element Method (FEM) solver and how we use the thermal field solution to analyze the blade design for calculating sensitivities, characterizing the design space, and propagating uncertainties through the solver. We then show blade designs and a series of UQ examples in section 4.1 before we provide our concluding remarks.

The Garment Bending Tool

Tool Design

The tool considered in this work is inspired by an origami folding robot developed by Balkcom et al.[5] composed of an arm equipped with a long blade and a table top with a long opening. A sheet of paper is placed over the table opening and the robot arm guides the blade downwards, wiping the sheet of paper into the long opening. Then, a clamp mounted on the under side of the table closes the table opening, thus fixing the fold on the sheet of paper. Finally, the clamp retracts and the folded paper releases.

This work will focus exclusively on a part similar to the blade of the origami folding device, which directly contacts the sheet of paper. A simple framework for designs of our blade is shown in Figure 4.1 and can be described by six dimensions: the thickness of the sheet metal used (not shown in figure), three lengths and two heights, each denoted L_1, L_2, L_3 and H_1, H_2 respectively. This geometry allows enough flexibility for a wide range of designs and minimizing the number design parameters. Future work could broaden the scope of possible designs by using a more complex geometry.

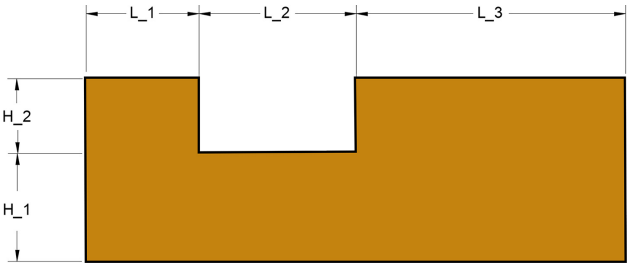


Figure 4.1: Generic tool profile.

Governing Equations

This work requires determining the thermal field resulting from a given heat source, surface losses, and internal conduction. Analysis is simplified using following assumptions:

- The thermal properties of the tool are not temperature-dependent in the regime of interest.
- Thermal expansion and other deformations are negligible.
- Thermal radiation may be approximated as black body radiation.
- The environment can be treated as an infinite heat sink, e.g., the ambient temperature is not coupled to the tool.
- Interactions between surfaces can be neglected, e.g., radiation from one area of the domain is not absorbed at another.

The governing equation for this thermal problem is a formulation of the first law of thermodynamics, written as:

$$\underbrace{\rho c \dot{T}}_{\text{thermal power}} = \underbrace{\nabla \cdot (k \nabla T)}_{\text{conduction}} + \underbrace{q}_{\text{source}} - \underbrace{l}_{\text{losses}} \tag{4.1}$$

The flux terms q and l require surface integrals, approximated for a given element e using Gaussian integration as:

$$\boldsymbol{S}^e = \int_{\delta\Omega e} \boldsymbol{\phi}^T \boldsymbol{\phi} dA = \sum_{p=1}^{g} \sum_{q=1}^{g} \boldsymbol{\phi}^T \boldsymbol{\phi} J w_p w_q \tag{4.2}$$

This model assumes that both convection and radiation contribute to the balance of energy. The thermal loss $\boldsymbol{l}$ is thus given by:

$$\boldsymbol{l} = \boldsymbol{c} + \boldsymbol{r} \tag{4.3}$$

Here, the vectors $\boldsymbol{c}$ and $\boldsymbol{r}$ represent surface area integrals of the domain.

$$\boldsymbol{c} = \underbrace{h\boldsymbol{S}}_{\boldsymbol{C}}(\boldsymbol{T} - \boldsymbol{T}_a), \qquad \boldsymbol{r} = \underbrace{\sigma \boldsymbol{S}}_{\boldsymbol{R}}(\boldsymbol{T}^4 - \boldsymbol{T}_a^4) \tag{4.4}$$

Where σ is the Boltzmann constant and h is the convective loss coefficient. The shorthand $\boldsymbol{C}$ and $\boldsymbol{R}$ is used for these surface integral matrices for convection and radiation losses, respectively, in further derivations.

The convective loss term enforces a Robin boundary condition [65] since it creates a linear relationship between the value and the gradient of the thermal field at some boundary locations. The radiation term is similar but enforces a nonlinear relationship.

Uncertainty Quantification

The practice of uncertainty quantification (UQ) attempts to measure the sources of variation that affect the output of experimental systems and computational models. In manufacturing, UQ is used to study the contributions of aleatoric uncertainties such as the seemingly random effects from external, environmental effects or the local variations in the microstructure of the material that is being processed. These sources of variation can be modeled via Monte Carlo (MC) [80] or polynomial chaos methods [139], the former typically acting as the benchmark to the latter. Uncertainty can also be epistemic, dealing with errors due to the lack of knowledge or measurements.

UQ methods are normally used to solve forward uncertainty propagation problems. In these scenarios, previously-measured or estimated distributions of the inputs are propagated through a numerical solver to estimate the expected variations of the output. This topic has been studied extensively, for example in the application of structural mechanics [133]. An MC simulation is a simple and effective method for propagating the uncertainties since a set of input test points can be generated by randomly sampling each input variable's

distribution. With enough input points, the variation of the output becomes evident and it can be quantified.

Another powerful use of UQ is to compute the sensitivity of the output with respect to each input individually. Perturbing any input within a small range, via a central difference scheme, for example, is a common strategy to locally approximate the derivative. However, the true sensitivity is defined in the limit as the perturbation range is reduced to zero, but it can also be computed analytically, if attainable, through direct differentiation of the governing equation. A high sensitivity to the perturbed input will yield relatively large changes to the output, and, inversely, a low sensitivity describes a relatively small to no change of the output value. Understanding these sensitivities can aid in design optimization [136, 3, 141].

Throughout this section, the terms *sensitivity* and *derivatives* are used interchangeably.

When performing UQ for fitness of the design, an important output parameter, commonly referred as a quantity of interest (QoI), must be identified. In a typical application, a numerical simulation is used to produce a measure of a field, from which a chosen QoI can be calculated through post-processing. The QoI serves as a proxy for the effectiveness of a combination of input parameter values. Typical QoIs in structural simulations include maximum stress [47] and structural compliance [153], both calculated form the mechanical field of a deforming body.

This work presents a set of utilities for performing UQ for the design of the robotic bending attachment shown in Figure 4.1. Our approach applies MC sampling and propagates uncertainties through a thermal FEM solver, computes the sensitivity of the QoI, and defines a hypersurface for design space characterization. The QoI selected in this study summarizes the temperature profile at the blade tool working edge, identified in Figure 4.2 to aid us in predicting an acceptable or unacceptable fold procedure on the garment.

Methods

Thermal FEM Solver

Using a standard FEM formulation with thermal stiffness matrix $\boldsymbol{K}$ and mass matrix $\boldsymbol{M}$, equation 4.1 may be written as:

$$\dot{\boldsymbol{T}} = \boldsymbol{M}^{-1}(\boldsymbol{K}\boldsymbol{T} - \boldsymbol{C}(\boldsymbol{T} - \boldsymbol{T}_a) - \boldsymbol{R}(\boldsymbol{T}^4 - \boldsymbol{T}_a^4) + \boldsymbol{q}) \tag{4.5}$$

where $\boldsymbol{T}$ is a vector of temperature values for all nodes and $\boldsymbol{T}_a$ stores the ambient temperature. Using Forward Euler time integration, the discretized matrix-vector equation used to update the nodal temperatures:

$$\boldsymbol{T}^{n+1} = \boldsymbol{T}^n + \Delta t\, \dot{\boldsymbol{T}}. \tag{4.6}$$

Steady State Solver

While transient temperature profiles, the response to perturbations, and properties such as the time required to reach operating temperatures all require time integration, the most important aspects of a tool's performance can be directly extracted from the temperature field at steady state. To resolve steady state thermal profiles, an efficient approximation method is necessary.

The steady state temperature may be defined by setting $\dot{\boldsymbol{T}} = 0$ in equation 4.5. This system is nonlinear and lends itself to an iterative solution scheme. To solve this system, Newton's method may be used to successively solve linearized approximations of the true system until a satisfactory numerical tolerance is achieved.

The inverse mass matrix $\boldsymbol{M}^{-1}$ from equation 4.6 may be factored out of non-trivial solutions:

$$\dot{\boldsymbol{T}} = 0 \rightarrow \boldsymbol{K}\boldsymbol{T} - \boldsymbol{C}(\boldsymbol{T} - \boldsymbol{T}_a) - \boldsymbol{R}(\boldsymbol{T}^4 - \boldsymbol{T}_a^4) + \boldsymbol{q} = 0 \tag{4.7}$$

where $\boldsymbol{T}_a$ is the ambient temperature.

Update terms are determined to refine the solution:

$$\boldsymbol{T}_{i+1} = \boldsymbol{T}_i + \boldsymbol{\Delta T}_i. \tag{4.8}$$

Here, i represents the number of iterations. Substituting this update rule into equation 4.7 yields:

$$\boldsymbol{K}(\boldsymbol{T}_i + \Delta\boldsymbol{T}_i) - \boldsymbol{C}(\boldsymbol{T}_i + \Delta\boldsymbol{T}_i - \boldsymbol{T}_a) - \boldsymbol{R}((\boldsymbol{T}_i + \Delta\boldsymbol{T}_i)^4 - \boldsymbol{T}_a^4) + \boldsymbol{q} = 0 \tag{4.9}$$

A first-order Taylor expansion of the system about $\boldsymbol{T}_i$ with offset size $\Delta\boldsymbol{T}_i$ may be used to linearize the system:

$$(\boldsymbol{T}_i + \Delta\boldsymbol{T}_i)^4 \approx \boldsymbol{T}_i^4 + (4\boldsymbol{T}_i^3)\Delta\boldsymbol{T}_i \tag{4.10}$$

Substituting the approximation from equation 4.10 into equation 4.9 and rearranging yields:

$$\underbrace{\Big(\boldsymbol{C} - \boldsymbol{K} + 4\boldsymbol{R}\boldsymbol{T}_i^3\Big)}_{\boldsymbol{A}}\underbrace{\Delta\boldsymbol{T}_i}_{\boldsymbol{x}} = \underbrace{\Big(\boldsymbol{K} - \boldsymbol{C}\Big)\boldsymbol{T}_i - \boldsymbol{R}\boldsymbol{T}_i^4 + \Big(\boldsymbol{R}\boldsymbol{T}_a^4 + \boldsymbol{q} + \boldsymbol{C}\boldsymbol{T}_a\Big)}_{\boldsymbol{b}} \tag{4.11}$$

Using this formulation, the steady state temperature may be iteratively determined to arbitrary precision.

For the remaining statistical analysis, the thermal solver may be treated as a black box that generates the nodal temperature array $\boldsymbol{T}(t)$ over time. Illustrative results are shown in figure 4.5.

Thermal Field Uncertainty Quantification

In addition to directly improving key performance metrics, it is also desirable to ensure that the performance of the blade tool is predictable and agnostic to environmental disturbances or imprecise tool construction. First, sensitivity data can inform the tool designers about

design options that yield increasingly robust designs. Resiliency of the output thermal field to perturbations of a certain input parameter translates to a viable blade design. Second, parameter studies give insight about the design space and the output's expectation.

The Quantity of Interest The design of a thermal blade tool requires defining its geometry, as well as construction material and a heat source, in hopes of contributing to a desired performance. The metric for the desired performance is what is called a quantity of interest (QoI). Typically, QoI data is a single scalar value for a given design point, and the output of a post-processing step on the field solution from the numerical solution.

The primary QoI selected for this study is the temperature range along the working edge of the blade at steady state. This region of interest is shown in figure 4.2.

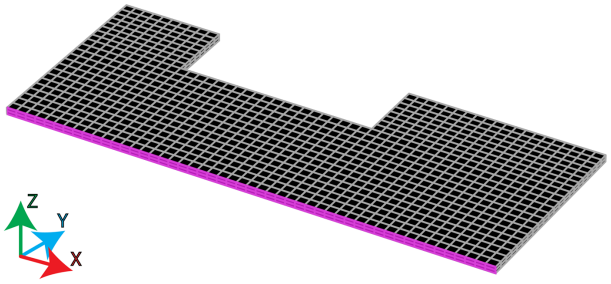


Figure 4.2: Thermal field along the working edge, highlighted, is the key QoI for the bending process.

Minimizing this QoI is a proxy for a consistent folding operation on the garment work piece. A wide temperature range at this edge will create uneven softening and result in unpredictable bending characteristics. Accordingly, successful tool designs will minimize temperature ranges in this area.

There are a myriad of QoI metrics that we can analyze with our UQ utility. An example would be the time the blade requires to reach operating temperatures. We estimate that a blade with rough outer dimensions of $0.2 \times 0.35m$ can take up to 20 minutes to reach its steady state temperature. Long wait times will reduce productivity on the manufacturing floor, driving the need for designs that minimize this particular QoI.

Data Generation The structure of the output UQ data is a 2-dimensional array with labeled columns and rows, sized $m \times (n+1)$, where m is the number of queried design points and n is the number of input parameters and the extra column stores QoI data. We can then analyze the design of the blade tool in three ways: sensitivity computations, design space characterization, and the propagation of uncertainty through the solver.

The aim of sensitivity computations is to approximate the local gradient of the QoI with respect to the design parameters. A typical method is using central differences. This type of analysis runs on a single query point, returning its QoI value and its partial derivatives. Explicitly, the user-specified query point $P(v_1, v_2, ..., v_n)$ is perturbed independently on each

Table 4.1: Example of a sampling design from the translational propagation Latin hypercube algorithm by [144] with 4 data points. Therefore, each input variable is discretized into 4 levels: low, mid-low, mid-high, and high.

	Conv. Coeff.	**Reflectivity**	**Base Height**	**Left Pad Width**
Design point 1	low	med-low	med-high	low
Design point 2	med-high	med-high	high	med-low
Design point 3	high	low	low	med-high
Design point 4	med-low	high	med-low	high

variable via a user-specified resolution, δ. The perturbation of design variable v_j is computed by:

$$P_{v_j+} = P\left(v_1, v_2, ..., (1+\delta)v_j, ..., v_n\right)$$
$$P_{v_j-} = P\left(v_1, v_2, ..., (1-\delta)v_j, ..., v_n\right).$$

The QoI value is calculated by $\mathcal{Q}^+ = f\left(P_{v_j+}\right)$ and $\mathcal{Q}^- = f\left(P_{v_j-}\right)$, where function f represents the thermal FEM solver. The sensitivity due to v_j is:

$$S_j = \frac{d\mathcal{Q}}{dv_j} \approx \frac{\mathcal{Q}^+ - \mathcal{Q}^-}{P_{v_j+} - P_{v_j-}}. \tag{4.12}$$

The MC simulation can also be leveraged to explore the design space, either across the entire space within its boundaries or in a particular region of interest. The design space may be defined with an orthogonal basis, where each dimension corresponds to a design parameter. In this study, a predetermined number of query points are generated from a translational propagation Latin hypercube (TPLH) sampling design algorithm by [144]. This algorithm ensures a fast, high-quality sampling design that is modular depending on the chosen number of query points. Table 4.1 showcases a sample symbolic design with 4 design points, and thus 4 levels are queried per variable (low, mid-low, mid-high, and high). In the spirit of Latin hypercube sampling, note that each level is represented once for each input variable.

Lastly, we can propagate the distributions of the inputs through the solver to discover a resulting output distribution of the QoI. Typically, the distributions of the inputs have been previously measured experimentally or they may be estimates. The design points can then be randomly sampled from these input distributions until a pre-determined number of samples has been reached, or until the QoI distribution does not change, outside a tolerance, with subsequent samples.

Aided by the definition of sensitivity analysis by [118], we can see that this type of analysis attempts to explain the individual contributions of each design parameter to the output distribution. In other words, the major contributors to the uncertainty of the QoI can be identified, know as factor prioritization, and the least influential parameters can be removed from the set of design inputs, know as factor fixing.

Table 4.2: Thermal properties

Property	Aluminum	Copper
$c\ [J/(kg \cdot K)]$	920.	375
$\rho\ [kg/m^3]$	2720	8960
$k\ [W/(m \cdot K)]$	237	400.

As a best practice, experimental data should calibrate estimates of the input distributions, e.g. with Bayesian inference methods. However, this is beyond the scope of this study. Our computational thermal tool for garment fabrication is agnostic to the distributions of the inputs and so it is the user's task to identify the most accurate data to employ.

Results

In this section, numerical thermal field results are shown for a characteristic thin, rectilinear tool with a slot geometry subjected to varied thermal loads. The resulting thermal field data will be queried for the range of steady state temperatures at the working edge (the QoI). We will showcase the usefulness of this data: calculate local sensitivities, map out the design space, and quantify the uncertainty of the QoI via MC propagation.

The meshing and simulation tool is able to simulate and visualize the performance of varied blade designs. For moderate resolutions like those shown below, the entire meshing, numerical solution with equation 4.11, and plotting procedure takes up to 4 seconds per design on a modern laptop.

Numerical Example

The thermal simulation model takes selected geometric and thermal design parameters and solves for the resulting thermal field at steady state and over time starting with given initial conditions.

Example results with two different heat sources and construction materials are shown in Figures 4.3 and 4.4 to demonstrate the inputs for quantifying the uncertainty of the thermal field. Four possible designs are considered, using either aluminum or copper as a construction material and either a thin heated wire through the middle of the tool or a pair of rectangular thermoelectric pads on the sides. In all designs, 200 Watts of thermal power is applied as a flux boundary condition uniformly over the defined source geometry.

The thermal properties of aluminum and copper are assumed to be constant with respect to temperature and are given by Table 4.2.

The following dimensions were used for all example designs:

- $L_1 = 7.5\ cm$
- $L_2 = 15\ cm$

Table 4.3: Dimensions for numerical examples in centimeters.

$\boldsymbol{i}$	$\boldsymbol{x}_{min,i}$	$\boldsymbol{x}_{max,i}$	$\boldsymbol{y}_{min,i}$	$\boldsymbol{y}_{max,i}$	$\boldsymbol{z}_i$
1	1.875	5.625	2.500	12.500	0.500
2	31.875	35.625	2.500	12.500	0.500
3	0.000	37.500	6.167	7.167	0.000
4	0.000	37.500	6.167	7.167	0.500

- $L_3 = 15\ cm$
- $H_1 = 10\ cm$
- $H_2 = 5\ cm$
- $thickness = 5\ mm$

Two different heat source geometries will be considered: rectangular pads on the front side of the blade or a narrow strip wrapped all the way around the blade to imitate a coil of wire, referred to as the pad design and wire design. These source regions will be idealized as constant within rectangular regions orthogonal to the $\hat{\boldsymbol{z}}$ direction, described by upper and lower bounds in the $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$ directions and an exact coordinate $\hat{\boldsymbol{z}}$.

For a specified power p_i, a source term q_i is modeled with the following form:

$$q_i(x,y,z) = \begin{cases} \frac{p_i}{\alpha_i} & (x_{max,i} \geq x \geq x_{min,i}) \ \& \\ & (y_{max,i} \geq y \geq y_{min,i}) \ \& \\ & (z = z_i) \\ 0 & \text{otherwise} \end{cases} \tag{4.13}$$

Where α_i is the area of a given source region:

$$\alpha_i = (x_{max,i} - x_{min,i}) \times (y_{max,i} - y_{min,i}).$$

Note that numerical integration of step functions using Gaussian quadrature is not exact since it is not possible to exactly fit a general step function with a polynomial. Any piece-wise source term, regardless of form, may result in integration errors. The impact of discontinuities may be reduced by refining the mesh.

Both the pad and wire designs will have two source regions, defined by

$$q_{pad} = q_1 + q_2 \text{ and} \tag{4.14}$$

$$q_{wire} = q_3 + q_4. \tag{4.15}$$

The pad geometries are defined by the values shown in table 4.3.

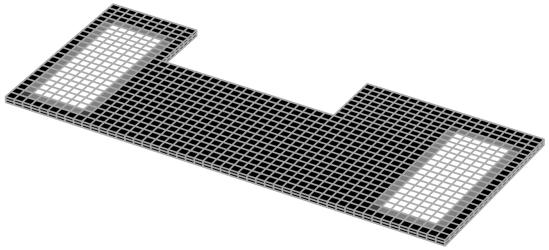


Figure 4.3: Schematic of example "pad" heat source.

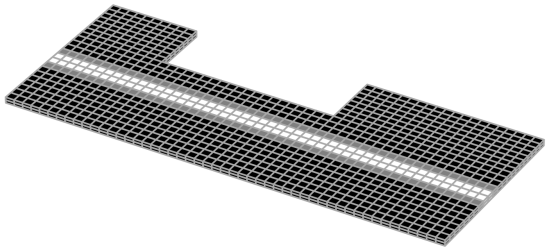


Figure 4.4: Schematic of example "wire" heat source.

All example source regions impart 100 W of thermal power for a total of 200 W of thermal power per design. The source regions with these parameters are shown in figures 4.3 and 4.4. Note that the second source region for the wire design is on the bottom side of the blade, directly below the visible region. Elements colored white are entirely within the source region, elements colored black are entirely outside, and elements colored gray are on the border.

It can be seen both qualitatively in figure 4.5 and quantitatively in figure 4.6 that the rectangular sources concentrate heat and create a significant deviation in temperature. The wire sources keep a much lower peak temperature since the source region has a much larger perimeter for heat to dissipate outward through. Additionally, since the wire source region is closer to the working edge on average, it produces a more uniform profile as reflected in the greatly reduced gap between the minimum and maximum edge temperatures shown in figure 4.6.

The thermal profiles at steady state with copper or aluminum construction are very similar with the exception of higher edge temperatures with copper when combined with pad heat sources since the higher conductivity of copper spreads the hot spot under the source toward the edge more effectively. Since neither material is clearly superior for the quantities of interest in this study, considerations like weight and cost should drive the selection of materials, not the thermal profile along the front edge.

While inspection and comparison of possible designs is one possible use of the thermal model of the bending tool, using sensitivity analysis to compare designs automatically can

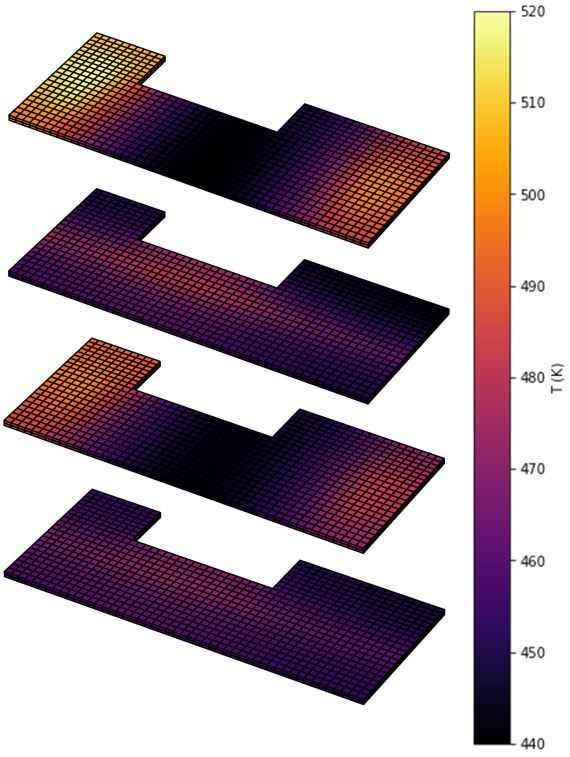


Figure 4.5: Example steady state thermal fields. Top to bottom: copper with pads, copper with wire, aluminum with pads, aluminum with wire.

offer greater insight into the effectiveness of different design choices.

Quantifying the Uncertainty

Now, here we present sample UQ data structures and results for the utilities introduced in Section 4.1: sensitivity calculations for a query point, characterizations of design space, and propagation of the input uncertainties. For all results that follow, the temperature range QoI is calculated as:

$$\mathcal{Q} = max\ \tilde{T} - min\ \tilde{T}, \tag{4.16}$$

where $\tilde{T} \in \{T(x, y, z) \mid y = 0\}$ for a blade tool oriented as in Figure 4.2.

The results that follow pertain to a copper tool heated via two thermal pads mounted on the areas outlined by the white grid cells in Figure 4.3.

Sensitivity Calculations Here, we compute the gradient of $\mathcal{Q}$ with respect to each design parameter as shown in equation 4.12. Table 4.4 demonstrates the data structure of the output array: the query point, the temperature range value, and the sensitivity values. This data pertains to a relatively large perturbation, $\delta = 10^{-4}$, but the sensitivities can be computed with any positive resolution.

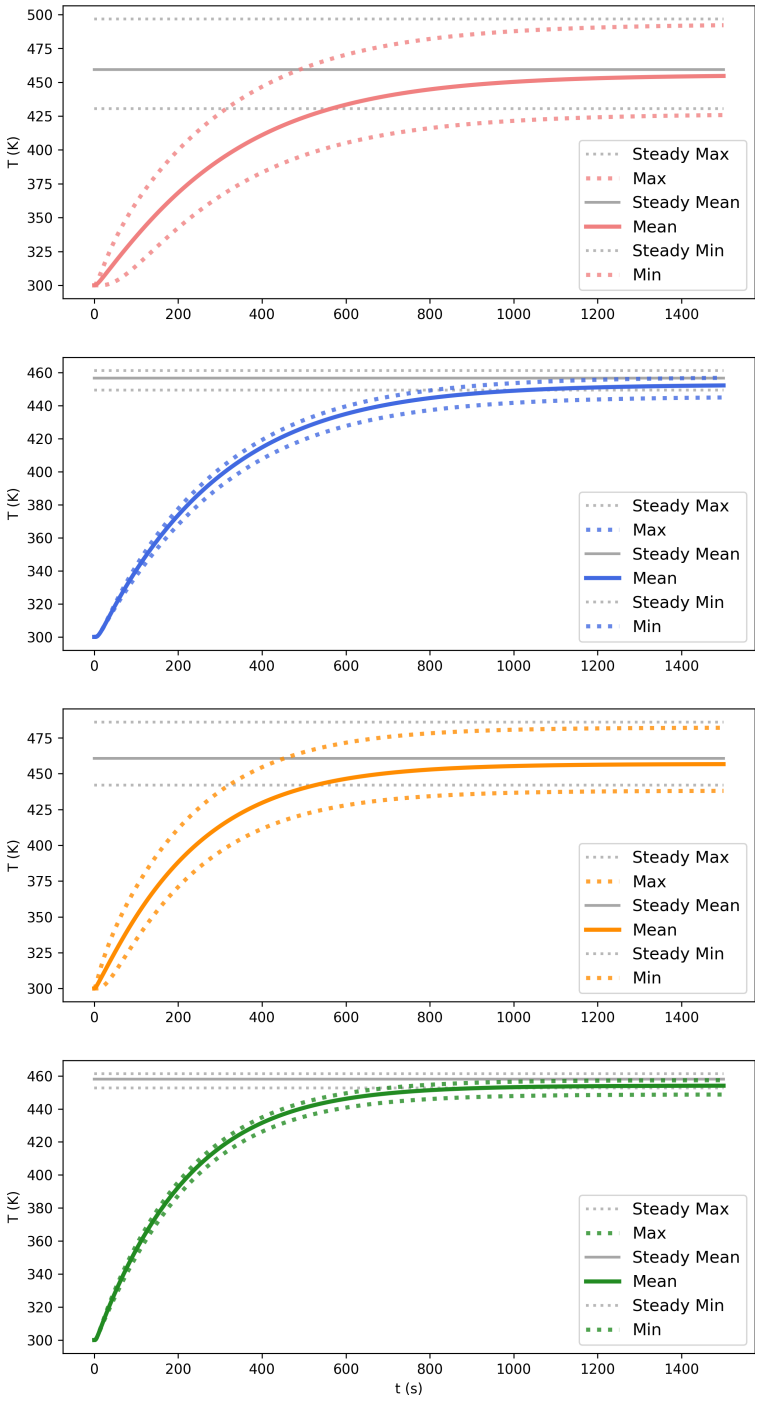


Figure 4.6: Example transient and steady thermal fields along working edge. Top to bottom: copper with pads, copper with wire, aluminum with pads, aluminum with wire.

Table 4.4: Sample output from the design sensitivity utility via central differences and $\delta = 10^{-4}$. Included is the query design point, the temperature range at the working edge (QoI) for the query design point, and the sensitivity of the QoI.

Query Point				**Temp. Range**	**Temp. Range Sensitivity**			
h	ϵ	H_1	L_1	$\mathcal{Q}$	$d\mathcal{Q}/dh$	$d\mathcal{Q}/d\epsilon$	$d\mathcal{Q}/dH_1$	$d\mathcal{Q}/dL_1$
$[W/(m^2 \cdot K)]$		$[m]$	$[m]$	$[K]$	$[(K^2 \cdot m^2)/W]$	$[K]$	$[K/m]$	$[K/m]$
40	0.7	0.1	0.11	15.59	-0.086	-1.030	-141.0	45.2

At the chosen resolution, sensitivity results suggest that all design variables have an inverse relation to the temperature range at the working edge. These results confirm our intuition. The environmental parameters will tend to quell hot and cold spots, thus reducing local temperature spikes. Low values of the geometric parameters enlarge the size of the slot and this tends to constrain and guide the flow of heat towards a particular region of the working edge, thus producing warmer spots.

Finding actual deviations of the QoI $\Delta\mathcal{Q}$ from a given design parameter perturbation, e.g. perturbations of the material thermal reflectivity, can be accomplished by

$$\Delta\mathcal{Q} = \frac{d\mathcal{Q}}{d\epsilon}\Delta\epsilon, \tag{4.17}$$

where $\Delta\epsilon = \epsilon - (1-\delta)\epsilon = \delta\epsilon$ is the perturbation amount. This computation also normalizes all sensitivities to the units of the QoI for directly comparing the sensitivity of $\mathcal{Q}$ due to each design variable.

These sensitivity results are the cornerstone for gradient-based optimization routines such as the gradient descent or the conjugate gradient methods.

Characterizing the Design Space This utility outputs data for building a response hypersurface of the design space, with its local value $\mathcal{Q}_i$ at each design point $P_i(h, \epsilon, H_1, L_1)$. From a predetermined axis-aligned, 4-dimensional bounding cube and a chosen number of design points, we sample the space using the fast, optimal TPLH design. The data in Table 4.5 are results of a 4-point sampling in our 4-dimensional design space following the design example in Table 4.1. The design parameters spanned the following bounds:

- $h = [0.0, 100.0]\ \frac{W}{m^2 \cdot K}$
- $\epsilon = [0.1, 1.0]$
- $H_1 = [0.050, 0.150]m$
- $L_1 = [0.050, 0.160]m$

Of course, to effectively capture a useful hypersurface in a 4-dimensional space, the TPLH sampling design must incorporate a considerably higher quantity of design points than shown in Table 4.5.

Table 4.5: Sample output from the design space characterization utility. The design space is 4-dimensional and 4 design points were output from a TPLH sampling design. Included in the data is are the design points and the temperature range at the working edge (QoI).

	Design Parameters				**Temp. Range**
	h $[W/(m^2 \cdot K)]$	ϵ	H_1 $[m]$	L_1 $[m]$	$\mathcal{Q}$ $[K]$
Design point 1	0.0	0.4	0.117	0.050	22.23
Design point 2	66.7	0.7	0.150	0.087	6.52
Design point 3	100.0	0.1	0.050	0.123	34.17
Design point 4	33.3	1.0	0.083	0.160	7.92

Propagating the Uncertainties of the Inputs We now assume that we can estimate each design parameter distribution based on typical laboratory settings. Our UQ utility is capable of sampling any distribution in standard Python packages in addition to any user-built density function, making it versatile for using corrected design parameter distributions.

Here, we model the case where the blade tool is made of copper and there is an environmental control in the laboratory. Hence, the convection coefficient and the material reflectivity are only subject to random variations, best modelled by a normal distribution with a standard deviation smaller than 10% of the mean. In the case of the geometrical parameters, we can say that the user is interested in exploring all possible dimensions with equal interest and so H_1 and L_1 are assigned uniform distributions, bounded by physical restrictions imposed by the thermal pad size. In summary:

- $h \sim \mathcal{N}(40, 4.0^2)$
- $\epsilon \sim \mathcal{N}(0.7, 0.1^2)$
- $H_1 \sim \mathcal{U}(0.06, 0.14)$
- $L_1 \sim \mathcal{U}(0.06, 0.165)$

The output data structure is identical to Table 4.5. We have decided to sample $1,000$ points, each taking in average of 3.67 s to generate a QoI value on a 2.3 GHz quad-core Intel Core i7 processor. Figure 4.7 has frequency plots for the input parameter values that were sampled.

The MC simulation propagated the input distributions through the thermal FEM solver. The output distribution for the temperature range at the working edge QoI is shown as a frequency plot and a kernel density estimation plot, Figures 4.8 and 4.9, respectively. Note that the QoI distribution has grossly deviated from the input distributions: it neither exhibits a uniform nor a normal distribution. Also, since the output distribution is purely empirical, every set of data from different batches of 1,000 sample points results in a slightly different

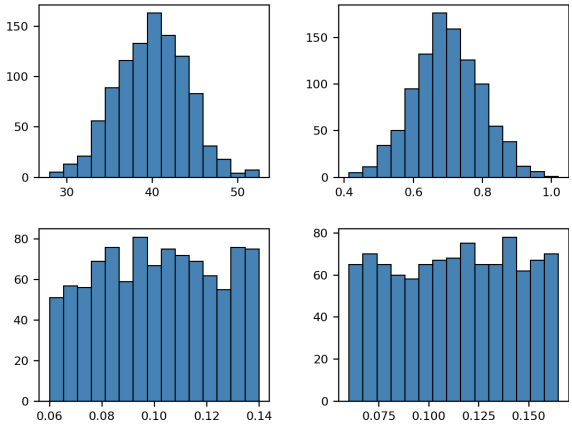


Figure 4.7: Input distributions of the design parameters. A normal distribution is assumed for the environmental variables and a uniform distribution is assumed for the geometrical variables. Clockwise from top left: h, ϵ, left pad width, right pad width.

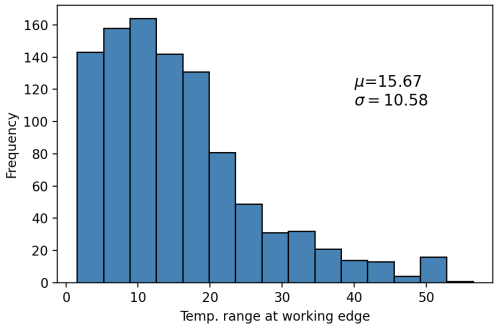


Figure 4.8: Output distribution of the temperature range at the working edge (histogram).

QoI distribution, but the shape will always exhibit a positive third moment (skewness) measure.

Pairwise plots as the ones in Figure 4.10 provide further insight about the QoI uncertainty. The plots on the off diagonals show the 2-dimensional scatter, highlighting hidden relationships between all input-input or input-QoI pairs. We can see that on the bottom line of scatter plots, for all input variables, the greatest density of points is manifested towards the lower bound of the temperature range QoI. In other words, closer to the $\mathcal{Q} = 0$ K versus the 60 K vicinity. This pattern in the scatter plots is the major contributor to a positive skewness of the output distribution.

Furthermore, we can also observe from these scatter plots that varying the environmental variables do not contribute to large changes in the temperature range QoI. However, the

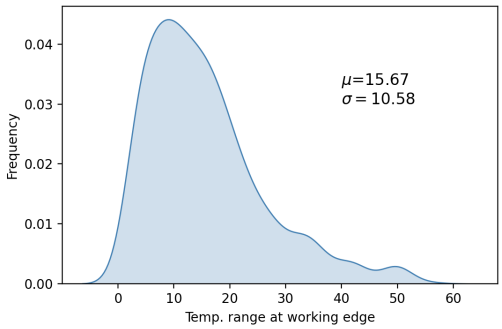


Figure 4.9: Output distribution of the temperature range at the working edge (probability density).

dimensions of the blade do contribute inversely to changes in the expected value of the QoI. For instance, a base height H_1 dimension closer to the higher bound will contribute to a lower expected temperature range QoI value than a shorter base dimension.

Propagating the input uncertainties through the solver is a powerful utility to set the expectations for the performance of the system, in this case the blade tool temperatures. Let's say that for a particular folding procedure application a temperature range at the working edge of more than 30 K is unacceptable. Therefore, the blade tool designers must make a decision whether the output distribution represented in Figures 4.8 and 4.9 represents a risk of undesirable performance of the tool.

To conclude this example application of UQ, we determined that any system can be evaluated with respect to a heuristic value, or QoI. The working edge temperature was a proxy for gauging the performance of the blade. To design the blade the user of the UQ computational tool is able to identify regions in the design space to target or to keep away and they will also possess a better understanding about how to adjust the design parameters to move across the design space in the desired direction and towards a desired destination.

The example in the following section will focus on a novel method for computing sensitivities for the application of engineering analysis and design optimization.

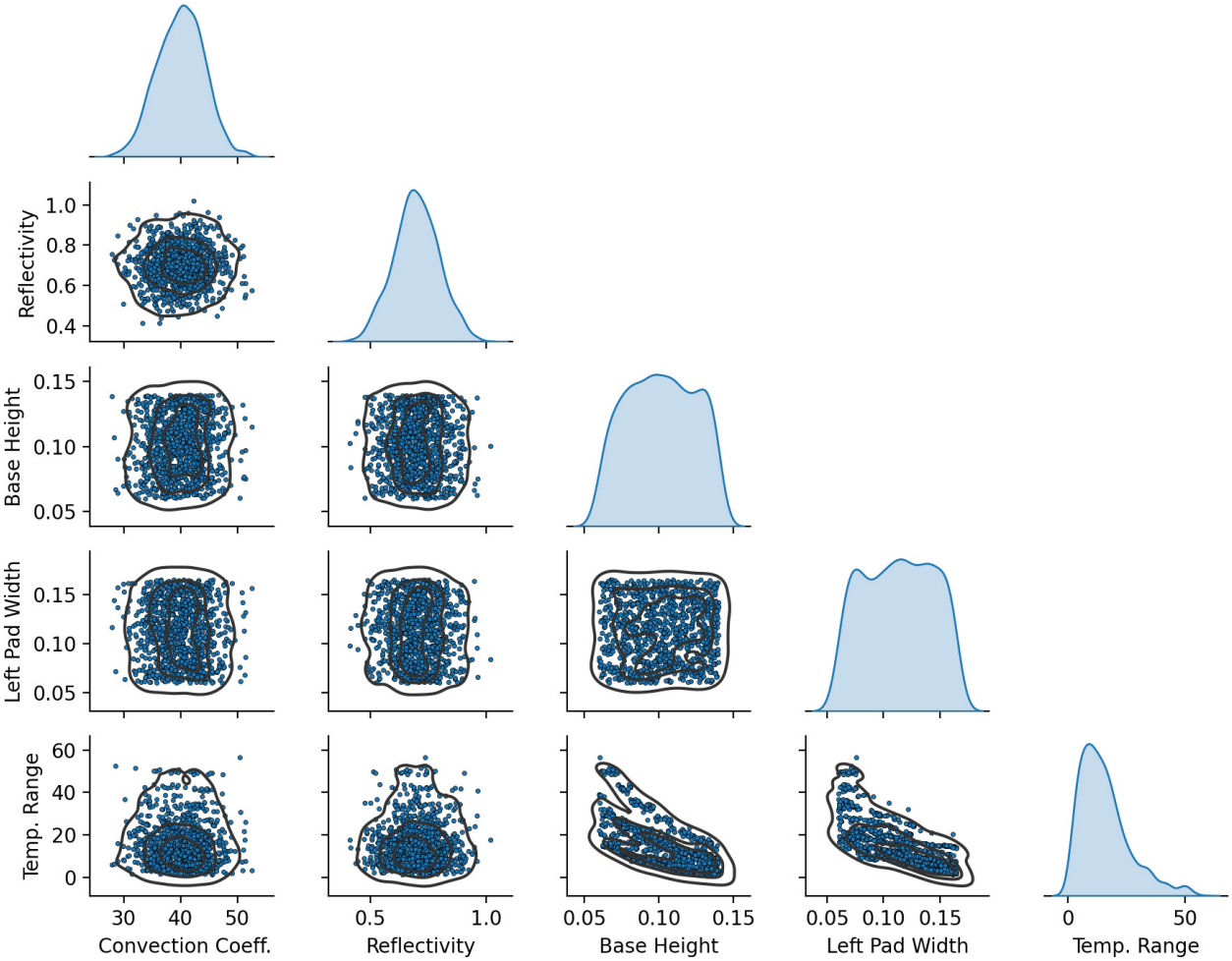


Figure 4.10: Pairwise plots for the design parameters and the temperature range at the working edge. The diagonals showcase the continuous probability density curve for each. The off-diagonals show relationships between each pair. The bottom row shows the contributions of each design parameter to the distribution of the QoI.

4.2 Sensitivity Computation of Integral Quantities with Moment Derivatives

The following UQ study[1] proposes a method for computing sensitivity that leverages derivatives of integral properties, in the form of shape moments, to incorporate diverse geometric

[1]This research was funded by the Defense Advanced Research Projects Agency (DARPA) and developed in partnership with Intact Solutions Inc. while the author of this dissertation was serving as a Summer Research Intern. The first author served as the PI and the third author provided application examples throughout the study. The company owns all intellectual property stemming from this work.

representations and design objective functions into a single system. By separating the concerns of geometric design space representation from the evaluation of physics and sensitivity computations, this approach can drastically reduce the development cost of novel design optimization programs. With this example, demonstrate the advantages of the proposed architecture on design spaces inspired by topology optimization, shape optimization, and material field design.

Introduction

Motivation

Sensitivity computation is a necessary component of many algorithms in PDE-constrained optimization [30], inverse problems (e.g. parameter estimation) [97], and error estimation [9]. Furthermore sensitivity data is useful in its own right; for a full survey of the uses and formulations of sensitivity we refer the reader to [141]. Sensitivity data is often used within an optimization loop, and thus can be a performance-critical bottleneck in software implementations. Furthermore, each design space generally requires a specially-formulated numerical routine to compute sensitivities.

Every sensitivity quantity is defined as the derivative of an objective function with respect to a design parameter. The objective depends on the application and optimization goal, while the design parameter depends on the design space. This paper deals with objective functions that arise from PDEs which are solved numerically using methods such as the finite element method. Examples of such objectives are structural compliance, average temperature and structural vibration frequency[43, 19].

Many useful objective functions are integral quantities, both for physical reasons and because it is often preferable to formulate objectives as integrals in order to preserve differentiability, such as in the case of approximating max stress as the L^p norm of stress[47]. Therefore when computing the derivatives of such objectives, it is necessary to integrate the objective function and then differentiate that integral quantity. Many approaches symbolically derive simple expressions for sensitivity, especially using the adjoint formulation. However, there is no general formulation that applies to every design space and objective function. This means that for every new design space or objective function, new routines must be written to calculate the sensitivity (Figure 4.11a). This hampers development of new design optimization programs and constrains the possibilities of the research field.

Contribution

This paper proposes a novel method for computing sensitivities of integral quantities which gives strong guarantees on accuracy, as well as provides a uniform interface to a variety of geometric representations. This technique builds on moment-based interoperable analysis[137] and introduces *moment derivatives* as the common interface for sensitivity computation, as illustrated in the context of adjoint method for computing sensitivity in Figure 4.11b.

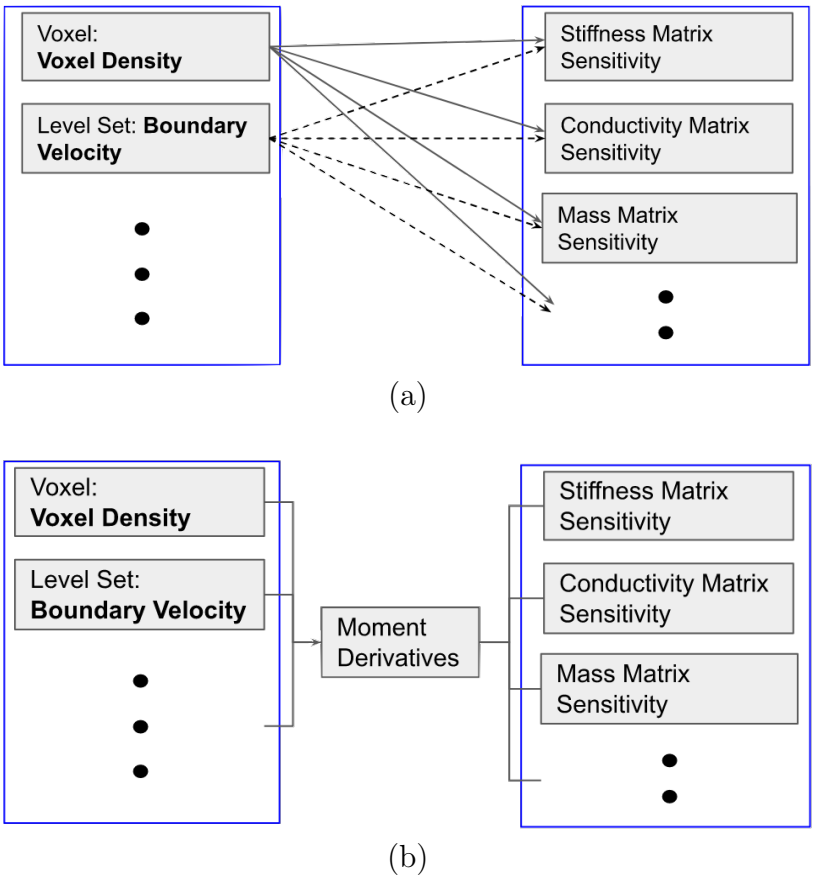


Figure 4.11: Comparison of direct sensitivity computation with the moment sensitivity approach. **(a)** Design optimization programs must compute sensitivity for each objective function and for each design space, leading to combinatorial complexity in incorporating new elements of either side. **(b)** Moment derivatives allow a unified interface to a variety of objective functions.

Moments encode the spatial distribution of mass and are differentiable with respect to design parameters. Such derivatives are used to generate optimized integration rules that accurately assemble sensitivities. This idea is applied to the problem of assembling the adjoint form of a PDE-constrained objective sensitivity. The key property of moment derivatives is that they do not depend on the objective function at hand, and can be reused to assemble any physical property matrix. Similarly, the adjoint variable need only be computed once for a given objective function and does not depend on the design parameter, and can thus be reused for any design parameter. By divorcing the objective sensitivity computation from the design space representation, our framework allows any design space which supports moment derivatives to be paired with any objective function formulated in the adjoint form without requiring extra code.

Outline

The necessary background and prior art in sensitivity analysis and moments is provided in Section 4.2. The formulation and framework details for sensitivity computation using moments is presented in Section 4.2. Examples and their results are shown Section 4.2. Finally, conclusions and future directions are presented in Section 4.2.

Terminology and Notation

We will use the term *design space* to mean the tuple $\{G, M, P\}$, where G is the geometric representation of the domain, M is the material property field over that domain, and P is a set of design variables which can affect either or both of G and M. Some examples of design spaces are:

- SIMP voxels, where G is a set of voxels, M is a penalized stiffness field, and P the set of voxel densities which parameterize M.
- B-Rep models, where G is the boundary represented model, M is constant, and P are spline parameters for the surface of the model.
- Levelsets, where G is a levelset surface of some implicit function, M is constant, and P parameterize the implicit function.
- Tessellated surfaces with a functionally-graded material field, where G is defined as the solid implied by the triangulated surface, and M is a graded field parameterized by P.
- Truss models, where G is a set of implicit cylinders modeled by 1-dimensional line segments, M is constant, and P parameterizes the implicit thickness or profile of the trusses.
- Thin shell models, where G is a surface with implied thickness, M is constant, and P parameterizes the implicit thickness of the surface facets.

A *design problem* consists of a design space and an objective function. Examples of objective functions are: structural compliance, maximum stress, or modal frequency. A *design optimization program* consists of a choice of design problem and optimization algorithm.

We will use bold letters to denote vector or matrix quantities. We will use p to signify a design parameter and u to signify a state variable, such as the displacement variable in the context of compliance minimization. We will use J as a quantity of interest we wish to minimize or maximize, and will use $J = \hat{J}(u(p), p)$ as a pure function of state variable and design parameters. We will use $\boldsymbol{F}(\boldsymbol{u}, p) = \boldsymbol{K}(p)\boldsymbol{u} - \boldsymbol{v}(p) = 0$ as the PDE that constrains the optimization, where $\boldsymbol{K}(p)$ is the physical property matrix (e.g. the stiffness matrix, in general a function of the design parameters), $\boldsymbol{u}$ is the state variable (e.g. displacement), and $\boldsymbol{v}(p)$ is a load function (e.g. applied tractions, also possibly a function of the design parameters). Throughout the paper, we use the terms sensitivity and derivatives interchangeably.

Background and Related Work

Gradient-Based Design Optimization

Design optimization is the generic term for routines which explore a design space in search of elements which optimize one or more objective functions. This paper is focused on PDE-constrained optimization in which finite element analysis must be performed to solve the PDE. A brief summary of some popular design spaces for gradient-based optimization follows.

Shape Optimization Gradient-based shape optimization is a generic technique to search a parametric design space using one or more objective functions and their gradients to guide the optimization routine[24, 45]. Often the parameterization of the design space is inherited directly from the features of a CAD model[2]. The design space may be formulated either as parametric features[45], spline parameters[93], or as free-form optimization[22].

Topology Optimization Topology optimization is a form of generative design which allows for free-form search across candidates with differing topological structure. This allows more design flexibility than feature-based parametric optimization, at the cost of a higher-dimensional design space[130]. Two popular methods for topology optimization are the single isotropic material with penalty (SIMP) method and the level set method.

The SIMP method assumes a power-law relationship between the density design variable and the effective Young's modulus as $E = \rho^r E_0$, where E_0 is some nominal modulus and $r > 1$ [112]. These density variables are typically allocated on a voxel grid that is also used for finite element analysis.

Some approaches to topology optimization involve mapping high-level design parameters onto a non-body-fitted mesh, which have been called feature-mapped methods in the literature[151]. Such methods use either a pseudodensity method or an immersed boundary method to perform numerical integration. In either case the sensitivity analysis involves propagating the effects of the high-level parameters to the fixed grid in order to assemble the adjoint form via numerical integration.

Level set methods operate on a design space of surfaces defined as the zero-set of an implicit function to generate candidate shapes and search for optimal candidates[88, 76]. Implicit representations define a shape $\Omega \subseteq D$ in terms of non-negative values of some function $\Phi(x)$ of the spatial variable x as $\Omega = \{x \in D \mid \Phi(x) >= 0\}$, where D is some predefined reference domain that contains all possible shapes of interest. The boundary $\partial\Omega$ of the shape is the zero level set of the function $\partial\Omega = \{x \in D \mid \Phi(x) = 0\}$[3, 19]. One common technique for analyzing level set geometries is to voxelize the domain and apply a homogenized fictitious material to partially-filled voxels[130]. Other approaches include boundary-aligned meshing and immersed boundary methods.

Material Field Optimization In addition to geometric optimization, there have been several approaches to the more generic goal of material field optimization[18, 148]. Some

approaches have opted for gradient-free optimization techniques specifically due to the difficulty in implementing sensitivity computation[66]. Most approaches involve augmenting a given geometric representation with material data and then modulating the parameters of the representation directly[128, 85].

Numerical Integration

Numerical integration is a key subproblem in both solving the PDE and assembling the adjoint form of the sensitivity. The particular choice of integration routine is important because there are two competing goals. First, it must be sufficiently accurate to compute sensitivity for the purposes of optimization. Second, the integration must be automated to a degree where it can be performed within a design optimization loop as the geometry changes. The automation requirement means that many approaches to design optimization opt for meshfree methods, particularly fictitious domain methods.

Explicit Spatial Discretization The simplest approach is to use the finite element discretization itself to perform numerical integration. Thus integration is performed by accurately meshing local features or by choosing homogenized material properties.

Meshing This method involves generating a volumetric polyhedral mesh for each candidate in the design space. In some design spaces, this procedure can be automated using assumptions about the design candidates[126]. One approach that allows for more automated meshing is isogeometric analysis[147]. However isogeometric analysis requires B-Rep splines to function and thus is only suitable for B-Rep shape optimization. Integration over polyhedral meshes is a trivial procedure involving pre-computed quadrature rules that are mapped from reference elements, but this simplicity comes at the upfront cost of generating the explicit mesh.

Homogenization The homogenization method (which is also known as the pseudo-density method) is a common approach for numerical integration that involves replacing partially-full cells with a cell-wise constant material property. While this approach has the benefit of efficient automation, it gives no guarantees on integration accuracy.

Integration over Partially-Full Finite Elements Alternatively, the finite element discretization may not conform to the geometric domain (see Figure 4.12), which makes the generation of a finite element discretization much simpler at the cost of a more complicated integration problem. Such integration problems can be solved via either geometric adaptation or moment-based methods.

Geometrically-adaptive Integration In adaptive integration, subidvisions of cells are generated such that the lowest levels of the subdivision tree consist of simple geometries

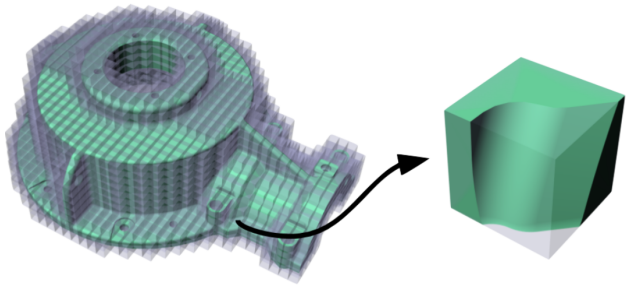


Figure 4.12: Illustration of a finite element discretization which does not conform to the geometric domain

which can be readily parameterized and integrated over[81, 53]. This approach requires tuning heuristics for the subdivision and incurs potentially unbounded computational cost, but is more accurate than the homogenization method.

Moment-based Numerical Integration Moment-based integration has been shown to be an accurate method for evaluating volume integrals for a wide range of geometric representations[137]. While our approach does not require a solver which uses moment-based numerical integration to solve the PDE, such a solver could reuse much of the code and data required to compute moments for the purposes of computing moment derivatives.

Let F be the L^2-space of functions from $\Omega \to \mathbb{R}$. The i-th moment m_i of a scalar field $f \in F$ with respect to a basis $\hat{\boldsymbol{b}} = [b_0, b_1, b_2, \ldots]$ of F is defined as follows

$$m_i(f) = \int_\Omega f b_i \, d\Omega \quad i \in [0, 1, \ldots]. \tag{4.18}$$

Given a finite subset of the basis $\hat{\boldsymbol{b}}$, the n-dimensional moment-vector $\boldsymbol{m}(f)$ is defined as

$$\boldsymbol{m}(f) = \begin{bmatrix} m_1(f) \\ m_2(f) \\ \vdots \\ m_n(f) \end{bmatrix}. \tag{4.19}$$

Once the moment-vector $\boldsymbol{m}(f)$ has been computed, one can generate quadrature rules at a set of prescribed points $\boldsymbol{x}_i, i \in [1, 2, \ldots m]$ using the following equation

$$\boldsymbol{A}\boldsymbol{w} = \boldsymbol{m}(f) \tag{4.20}$$

$$\boldsymbol{A} = \begin{bmatrix} b_1(\boldsymbol{x}_1) & b_1(\boldsymbol{x}_2) & \dots & b_1(\boldsymbol{x}_m) \\ b_2(\boldsymbol{x}_1) & b_2(\boldsymbol{x}_2) & \dots & b_2(\boldsymbol{x}_m) \\ \vdots & \vdots & \ddots & \vdots \\ b_n(\boldsymbol{x}_1) & b_n(\boldsymbol{x}_2) & \dots & b_n(\boldsymbol{x}_m) \end{bmatrix} \tag{4.21}$$

$$\boldsymbol{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_m \end{bmatrix}, \ \boldsymbol{m}(f) = \begin{bmatrix} \int_\Omega f b_1 \, d\Omega \\ \int_\Omega f b_2 \, d\Omega \\ \vdots \\ \int_\Omega f b_n \, d\Omega \end{bmatrix} \tag{4.22}$$

Solving this equation will produce weights w_i which, along with the points $\boldsymbol{x}_i$, form a quadrature rule to integrate over the domain Ω with weighting function f[53]. In the case where $f = 1$ and the moments capture purely geometric information, we may also write $\boldsymbol{m}(\Omega)$ to emphasize that the moments are a function of the domain.

Moment-based integration has been shown to be an efficient method for interfacing with a wide variety of different geometric representations[137]. Moment-fit quadrature rules are exact for any integrand in the span of the moment basis, and have been shown to have good accuracy for practical cases in which the integrand is not exactly in the span of the basis[68, 53].

While in general the quadrature points $\boldsymbol{x}_i$ can be treated as unknowns in the moment-fitting system (thereby making it a nonlinear system), another approach is to prescribe points according to a known quadrature rule such as Gauss-Legendre and then solving the resulting linear system for the weights. In fictitious domain methods, this has proven to be accurate even when the integration points are allocated in the fictitious domain[52, 137].

Design Derivatives

The most straightforward approach to computing sensitivities is via finite difference techniques, which can be easily implemented for any black-box function[21, 141]. In particular, one could consider the entire simulation system as a black-box function and then apply finite differences to each design parameter. Alternatively, intermediate design sensitivities can be computed from a black-box CAD system via finite-difference techniques[2, 108, 142] and used in other symbolically-derived formulations downstream. However, as an approximation of the derivative operator, one must run the finite difference routine again for each design parameter, which can be prohibitively expensive in the typical case of design optimization where the number of design parameters can be in the thousands. Furthermore, the step size and number of points in the finite difference stencil are parameters which must be tuned and can have a huge effect on the accuracy and performance of the approximation[134].

CAD-based approaches to computing sensitivities integrate directly with CAD kernels to provide shape sensitivity data using the geometric primitive APIs provided by the kernels. Specialized CAD kernels have been formulated to compute design derivatives via automatic

differentiation[93, 46, 6, 119]. Note that these approaches are specially-formulated for certain boundary representations and do not generalize to non-geometrical design spaces such as volumetric material field optimization.

Automatic differentiation has been used to derive exact sensitivities for complicated programs[103, 141, 49, 7]. Forward-mode automatic differentiation shares the same drawbacks as the finite difference method, in that such a routine must be run once for every input variable. Conversely, reverse-mode automatic differentiation scales with the output dimension and is the algorithmic derivative version of the adjoint formulation. However, both techniques require modification of internal data structures of systems, which can be prohibitively expensive to develop.

Meshfree methods have been applied to shape optimization[22]. In particular, a recent contribution[44] introduces the notion of computing derivatives of quadrature rules which we will expand in Section 4.2. However these approaches involve specialized implementations of integration routines that only apply to certain geometric representations or design parameterizations. For instance, [44] applies specifically to B-Rep models represented by non-uniform rational B-splines (NURBS).

Many design optimization applications rely on adjoint methods to compute sensitivities, as such methods scale with the number of output dimensions rather than the number of input dimensions[141, 30]. What follows is a review of the adjoint formulation that this paper will build upon in Section 4.2.

Adjoint-based Sensitivity Computation Consider an optimization problem that is constrained by the following generic PDE

$$\boldsymbol{F}(\boldsymbol{u}, p) = \boldsymbol{K}(p)\boldsymbol{u} - \boldsymbol{v}(p) = \boldsymbol{0}. \tag{4.23}$$

This constraint is applied to a system that optimizes an objective function $\hat{J}(\boldsymbol{u}(p), p)$, such as structural compliance or total strain energy.

The adjoint-based formulation for the sensitivity of the objective begins by augmenting the objective function with the PDE such as

$$\tilde{J} = J + \boldsymbol{\lambda} \cdot (\boldsymbol{K}\boldsymbol{u} - \boldsymbol{v} - \boldsymbol{F}), \tag{4.24}$$

and then taking the total derivative of the augmented objective function $\tilde{J}$ and utilizing the chain rule we have

$$\begin{aligned} \frac{d\tilde{J}}{dp} = \frac{\partial J}{\partial \boldsymbol{u}}\frac{\partial \boldsymbol{u}}{\partial p} + \frac{\partial J}{\partial p} + \\ \boldsymbol{\lambda}^T \left(\frac{\partial \boldsymbol{K}}{\partial p}\boldsymbol{u} + \boldsymbol{K}\frac{\partial \boldsymbol{u}}{\partial p} - \frac{\partial \boldsymbol{v}}{\partial p} - \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{u}}\frac{\partial \boldsymbol{u}}{\partial p} - \frac{\partial \boldsymbol{F}}{\partial p} \right). \end{aligned} \tag{4.25}$$

As a reminder, $\boldsymbol{u}$ is the state vector and p represents a particular input parameter. Note that all $\frac{\partial \boldsymbol{\lambda}}{\partial p}$ terms vanish since $\boldsymbol{K}\boldsymbol{u} - \boldsymbol{v} - \boldsymbol{F} = \boldsymbol{0}$. Now, since $\frac{\partial \hat{J}}{\partial \boldsymbol{u}} + \boldsymbol{\lambda}^T\boldsymbol{K} - \boldsymbol{\lambda}^T\frac{\partial \boldsymbol{F}}{\partial \boldsymbol{u}} = \boldsymbol{0}$ then

we can eliminate all $\frac{\partial \boldsymbol{u}}{\partial p}$ terms. Further note that since $\frac{\partial \boldsymbol{F}}{\partial \boldsymbol{u}} = \boldsymbol{K}$ the term $\boldsymbol{K}\frac{\partial \boldsymbol{u}}{\partial p}$ is cancelled out. Therefore, the sensitivity expression in Equation 4.25 becomes

$$\frac{d\tilde{J}}{dp} = \frac{\partial J}{\partial p} + \boldsymbol{\lambda}^T \left(\frac{\partial \boldsymbol{K}}{\partial p}\boldsymbol{u} - \frac{\partial \boldsymbol{v}}{\partial p} - \frac{\partial \boldsymbol{F}}{\partial p} \right). \tag{4.26}$$

Now, computing $\frac{d\tilde{J}}{dp}$ has been reduced to computing $\boldsymbol{\lambda}$ and $\boldsymbol{u}$ (which come from solving the PDE), and the derivative quantities $\frac{\partial J}{\partial p}$, $\frac{\partial \boldsymbol{K}}{\partial p}$, $\frac{\partial \boldsymbol{v}}{\partial p}$, and $\frac{\partial \boldsymbol{F}}{\partial p}$.

Moment-based Sensitivity Computation

In this section, we detail the method for computing derivatives of integral quantities using moment derivatives. We then apply this concept by expanding the adjoint formulation of the sensitivity problem by the chain rule in terms of moments and reduce the problem to computing moment derivatives. Then, we formulate moment derivatives for different geometric and material field representations.

Derivative Quadrature Rules

Let $\Omega(p)$ be a geometric domain parameterized by the parameter p. Consider an integral quantity $F(p)$ that is defined as follows

$$F(p) = \int_{\Omega(p)} f \, d\Omega \tag{4.27}$$

We can approximate $F(p)$ as follows

$$F(p) \approx \sum_{i=0}^{n} w_i f(\boldsymbol{x}_i). \tag{4.28}$$

Now to approximate $\frac{dF}{dp}$ we take derivatives on both sides

$$\frac{dF}{dp} \approx \sum_{i=0}^{n} \left[\frac{dw_i}{dp} f(\boldsymbol{x}_i) + w_i \frac{df}{dp}(\boldsymbol{x}_i) \right]. \tag{4.29}$$

In a design optimization context, the first term represents the geometric contribution of the design parameter, and the second term represents the material contribution. In the special case where the design parameter only affects the geometry, the second term is zero and we have simply

$$\frac{dF}{dp} \approx \sum_{i=0}^{n} \frac{dw_i}{dp} f(\boldsymbol{x}_i). \tag{4.30}$$

Note that in this special case the quadrature points do not change locations as a function of p because we are evaluating infinitesimal variations of Ω to calculate the derivative. We have

reduced the problem of evaluating $\frac{dF}{dp}$ to computing derivatives of the quadrature weights with respect to p, which can be done via moment-fitting. We refer to such a quadrature rule with weights $\frac{dw_i}{dp}, i \in \{0, 1, \ldots, n\}$ as a *derivative quadrature rule*.

Fitting Moment Derivatives

Consider the moment-fitting system $\boldsymbol{A}\boldsymbol{w} = \boldsymbol{M}(\boldsymbol{p})$, where $\boldsymbol{M}$ and therefore $\boldsymbol{w}$ are functions of the design parameters $\boldsymbol{p} = [p_1, p_2, \ldots, p_n]$. The quantity we wish to solve for is $\frac{\partial \boldsymbol{w}}{\partial \boldsymbol{p}}$, i.e. the derivative of each quadrature weight with respect to each design parameter. Since each design parameter can be considered independently, we will assume that the design vector consists of a single parameter p. The extension to vector-valued parameters is a straightforward application of this result to each component of the design vector.

Taking the derivative with respect to p on both sides of the moment-fitting system produces the following

$$\frac{\partial \left(\boldsymbol{A}\boldsymbol{w}\right)}{\partial p} = \frac{\partial \boldsymbol{M}}{\partial p}. \tag{4.31}$$

Since $\boldsymbol{A}$ is not a function of p, we have

$$\boldsymbol{A}\frac{\partial \boldsymbol{w}}{\partial p} = \frac{\partial \boldsymbol{M}}{\partial p} \tag{4.32}$$

which is a moment-fitting system, but with moment derivatives on the right hand side. Examples of design-space-specific methods for computing $\frac{\partial \boldsymbol{M}}{\partial p}$ are described in detail in Section 4.2. Note that the moment matrix $\boldsymbol{A}$ can be reused from the original moment-fitting procedure. A typical method for solving this linear system is the Moore-Penrose pseudoinverse which can be numerically computed via QR decompositions. Since QR decompositions can be cached for any moment matrix $\boldsymbol{A}$, it is possible to precompute such decompositions and store them in memory to optimize the moment-fitting procedure [140].

Accuracy of Derivative Quadrature Rules Moment-fit quadrature rules have strong guarantees on accuracy for any integrand which is well-approximated by the moment basis[137, 68, 53]. In the case of derivative quadrature rules, the same logic as in [137, 68] applies.

Consider an objective function $F = \int_\Omega f \, d\Omega$, whose derivative $\frac{dF}{dp}$ we wish to approximate with a derivative quadrature rule. If the integrand f can be expanded in terms of the moment basis $\boldsymbol{b}$ as $f = \sum_{i=0}^{n} c_i b_i$, we have that

$$\frac{dF}{dp} = \frac{d}{dp}\int_\Omega f \, d\Omega = \sum_{i=0}^{n} c_i \frac{d}{dp}\int_\Omega b_i \, d\Omega, \tag{4.33}$$

and therefore the quadrature error on the left hand side is equal to the error on the right hand side. Therefore any derivative quadrature rule that can accurately integrate $\frac{d}{dp}\int_\Omega b_i \, d\Omega$ will accurately integrate $\frac{dF}{dp}$, but this is simply the definition of a quadrature rule which solves the moment fitting system $A\frac{\partial w}{\partial p} = \frac{\partial M}{\partial p}$.

Adjoint Formulation with Moment Derivatives

In this section we apply the ideas developed so far to the problem of assembling the adjoint form to compute sensitivities. From Equation 4.25, the adjoint formulation of the sensitivity is $\frac{d\hat{J}}{dp} = \boldsymbol{\lambda}^T \left(\frac{d\boldsymbol{K}}{dp}\boldsymbol{u} - \frac{d\boldsymbol{v}}{dp} - \frac{\partial \boldsymbol{F}}{\partial p} \right) + \frac{dJ}{dp}$. Each of these terms is straightforward to compute for any design space except for $\frac{d\boldsymbol{K}}{dp}$.

Recall that the matrix $\boldsymbol{K}$ is assembled via a quadrature rule as follows

$$\boldsymbol{K} = \sum_{i=1}^{n} w_i k(\boldsymbol{x}_i), \tag{4.34}$$

where $k(\boldsymbol{x}_i)$ is a function of space and represents the integrand of the governing bilinear form of the differential equation. Therefore we have

$$\frac{d\boldsymbol{K}}{dp_j} = \sum_{i=1}^{n} \frac{dw_i}{dp_j} k(\boldsymbol{x}_i) + w_i \frac{dk(\boldsymbol{x}_i)}{dp_j}. \tag{4.35}$$

Thus, all that remains is actually computing $\frac{d\boldsymbol{m}}{d\boldsymbol{p}}$, from which we can compute $\frac{d\boldsymbol{w}}{d\boldsymbol{p}}$ via moment-fitting.

Moment Derivatives for Specific Design Spaces

Now that the sensitivity problem has been reduced to computing $\frac{d\boldsymbol{m}}{d\boldsymbol{p}}$, we will demonstrate how to formulate moment derivatives on three different design spaces. We submit that the method described in earlier sections can accommodate new design spaces without any changes for the cost of implementing a routine for computing moment derivatives.

Voxels with Density As discussed in Section 4.2, density-based topology optimization routines typically discretize design candidates using voxels with density values. Let ρ_j be the constant scalar design variable (i.e. density) associated with voxel j. Recall the i^{th} moment of a set $\{v_j\}$, $j \in [0, 1, \dots n]$ of voxels is defined as

$$m_i(\boldsymbol{p}) = \sum_{j=1}^{n} \int_{v_j} b_i \rho_j \, dV. \tag{4.36}$$

It follows that the derivative of Equation 4.36 with respect to the k^{th} design parameter is

$$\frac{\partial m_i(\boldsymbol{p})}{\partial p_k} = \frac{\partial}{\partial p_k} \left(\sum_{j=1}^{n} \int_{v_j} b_i \rho_j \, dV \right) = \sum_{j=1}^{n} \int_{v_j} b_i \frac{\partial \rho_j}{\partial p_k} \, dV. \tag{4.37}$$

In the case where the design parameter is identified with the voxel density, the final expression simplifies due to the identity $\frac{\partial \rho_j(\boldsymbol{p})}{\partial p_k} = \delta_{jk}$ as follows

$$\sum_{j=1}^{n} \int_{v_j} b_i \frac{\partial \rho_j}{\partial p_k} \, dV = \int_{v_k} b_i \, dV. \tag{4.38}$$

In the context of SIMP topology optimization applications, the stiffness field is penalized by the density according to $E(\rho) = \rho^r E_0$, where $r > 1$. In this case, the moment derivative formulation in Equation 4.37 becomes

$$\frac{\partial m_i(\boldsymbol{p})}{\partial p_k} = \sum_{j=1}^{n} \int_{v_j} b_i \frac{\partial \left(\rho_j^r\right)}{\partial p_k} \, dV = \int_{v_k} b_i r \rho_k^{r-1} \, dV. \tag{4.39}$$

In the case of feature-mapped systems which operate in a pseudodensity framework, the high-level parameters effect the density in potentially complicated ways, but this complexity is completely encapsulated in the $\frac{\partial \rho_j}{\partial p_k}$ term.

Shape Sensitivity of Boundary Representations The following derivation is inspired by the shape sensitivity treatment in [64, 56]. Consider a parameterized geometric domain $\Omega(t)$. Denote δt as a small variation of the parameter t. Assume there is a domain mapping between $\Omega(t)$ and $\Omega(t + \delta t)$ denoted as $\Omega(t + \delta t) = T(\Omega(b), \delta t)$. With sufficient regularity assumptions, we can expand the Taylor series of T as follows

$$T(\Omega(t), \delta t) = T(\Omega(t), 0) + \delta t \frac{dT}{d(\delta t)}|_{\delta t=0} + \mathcal{O}(\delta t^2) \tag{4.40}$$

Define the design velocity vector as $\boldsymbol{V}(\Omega_0) = \frac{dT}{d(\delta t)}|_{\delta t=0}$. Ignore the higher order terms of the Taylor expansion as they will not contribute to the derivative. Denote the transformed coordinates as $\boldsymbol{x}(t) \in \Omega(t)$. Then we have that $\boldsymbol{x}(t+\delta t) = \boldsymbol{x}(t) + \boldsymbol{V}\delta t$. The Jacobian of this transform is $J(T) = \nabla T = \boldsymbol{I} + \delta t \nabla(\boldsymbol{V})$. Let $F(t) = \int_{\Omega(t)} f(\boldsymbol{x}(t)) \, d\Omega$ be an intergal quantity of interest, where f is some differentiable function of space. Then we have that

$$F(t + \delta t) = \int_{\Omega(t+\delta t)} f(\boldsymbol{x}(t + \delta t)) \, d\Omega \tag{4.41}$$

$$= \int_{\Omega(t)} f(\boldsymbol{x}(t) + \boldsymbol{V}\delta t)|J| \, d\Omega \tag{4.42}$$

$$\Rightarrow \frac{dF}{dt} = \lim_{\delta t \to 0} \frac{F(t + \delta t) - F(t)}{\delta t} = \int_{\Omega(t)} f \frac{d|J|}{dt} \, d\Omega \tag{4.43}$$

$$= \int_{\Omega(t)} \nabla f^T \cdot \boldsymbol{V} + f div(\boldsymbol{V}) \, d\Omega \tag{4.44}$$

$$= \int_{\Omega(t)} \nabla(f\boldsymbol{V}) \, d\Omega \tag{4.45}$$

Now applying the divergence theorem to convert this volumetric integral into a surface integral over the boundary $\partial\Omega(t)$

$$\frac{dF}{dt} = \int_{\partial\Omega(t)} fV_n \, d\Gamma, \tag{4.46}$$

where V_n is the normal component of the design velocity.

This applies to any differentiable function of space f, and in particular it applies to the moment basis. Letting $F(t) = m_i(\Omega(t)) = \int_{\Omega(t)} b_i \, d\Omega$ we have

$$\frac{dm_i(\Omega(t))}{dt} = \int_{\partial\Omega(t)} b_i V_n d\Gamma \tag{4.47}$$

In the case of level sets, the design velocity of the level set ϕ with respect to a design parameter p_k is[102]

$$V_n = \frac{\partial\phi(\boldsymbol{x})}{\partial p_k} \left|\nabla\phi(\boldsymbol{x})\right|^{-1}, \tag{4.48}$$

The design velocity of B-Rep models can be found by either automatic differentiation of the spline parameters or finite differentiation, as described in Section 4.2.

Functionally-Graded Material Fields In addition to geometric sensitivities, the moment derivatives formulation can be extended to derivatives of functionally-graded material (FGM) fields as well. This demonstrates that the material field representation need not be tied to the geometric representation, since the material properties may be formulated via an algebraic function (such as B-spline) and the geometry may be represented via triangulated surface mesh, voxels, etc.

Define a scalar material property $\rho(\boldsymbol{c})$ that is a function of the coefficients $\boldsymbol{c} = \{c_j \mid j \in [0, 1, \ldots n]\}$. Such a scalar field can also be characterized by its moments for the purposes of analysis[68]. The i-th component of the moment derivative of this field with respect to c_j is computed as follows

$$\frac{\partial m_i(\rho)}{\partial c_j} = \int_\Omega b_i \frac{\partial\rho}{\partial c_j} \, d\Omega \approx \sum_k^n w_k b_i(\boldsymbol{x}_k) \frac{\partial\rho}{\partial c_j}(\boldsymbol{x}_k). \tag{4.49}$$

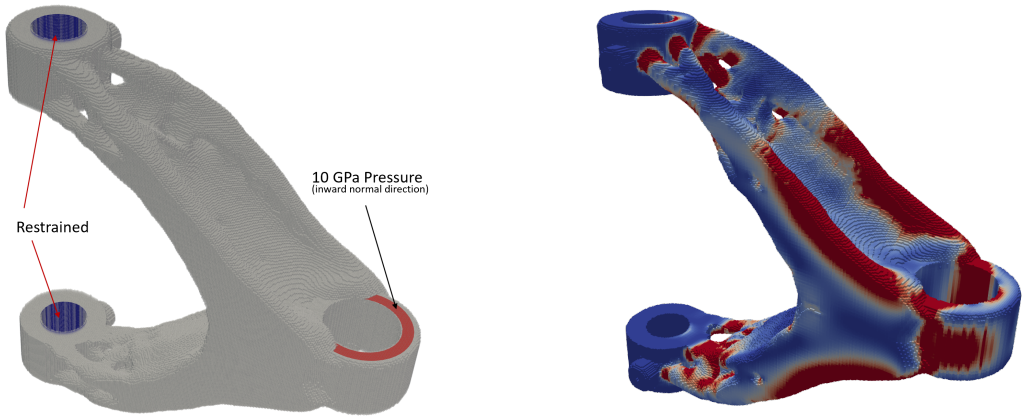
This field may take the form of algebraic functions such as splines[104], distance fields[11], acquired data sets[146], or other exotic representations[29, 55]. The key point is that the only data required to calculate the moment derivatives are the values of $\frac{\partial\rho}{\partial c_j}$ at each quadrature point. Furthermore, extension of this principle to matrix-valued material properties is straightforward, since the components of the matrix can be treated individually as scalar fields.

Results and Examples

In this section, we present examples using our moment sensitivity approach for topology and material properties optimization for various design spaces.

Implementation details of simulation system These examples are computed using a meshfree finite element method. The finite elements are allocated over a fictitious domain that is a superset of the true geometric domain[36]. The Dirichlet boundary conditions are enforced via the solution structure method[115]. The numerical integration is performed using moment-fitted quadrature rules[137]. Finally, we used linear basis functions for the finite elements and monomials up to order 5 as a basis for moment fitting.

SIMP Voxel Models



(a) (b)

Figure 4.13: Illustration of the voxel structural compliance sensitivity example. Red regions signify greater magnitude and blue less magnitude. **(a)** Voxelized model of a bracket with structural boundary conditions. **(b)** Compliance sensitivity field visualized over a voxel geometry.

The SIMP method, as discussed in Section 4.2, defines the design space via a voxel decomposition with associated density values assigned to each voxel. These density values scale the nominal stiffness matrix in structural problems as $\boldsymbol{E}(\rho) = \rho^r \boldsymbol{E}_0$. We demonstrate our approach on this design representation with a voxel bracket model consisting of 30,000 voxels, as pictured in Figure 4.13. Over a random subset of 1000 voxels, we computed an average sensitivity difference between a 2-point FD and moment derivatives of -0.009%.

Meshes with Surface Velocity

Since both level set methods and shape optimization can be formulated as surface meshes with nodal velocities, we will demonstrate and benchmark moment sensitivities using this design space representation. We compare the compliance sensitivity as computed via moment derivatives versus those computed via 2-point, 4-point, and 6-point central finite differences. Note that the moment derivatives approach can be slightly more expensive than the 2-point finite difference technique, but has accuracy closer to the 6-point stencil. Furthermore, care must be taken in finite difference routines to choose the step size and stencil, whereas the moment derivatives approach involves no numerical tuning.

Hole Radius Sizing This scenario setup is illustrated in Figure 4.14. Let the radius of the bridge section of the bracket be the design variable p. In this case the velocity vectors point radially inwards. The results of the moment sensitivities versus finite difference approaches are detailed in Table 4.6.

Fandisk This scenario setup is illustrated in Figure 4.15. The top surface of the fandisk is loaded with a downwards force, and the bottom surface is restrained. The results of the moment sensitivities versus finite difference approaches are detailed in Table 4.7.

Table 4.6: Comparison of moment derivatives vs. finite differences for the hole sizing example.

Derivative Type	Computed Value	Δ vs. moments (%)
Moments	-5.719×10^{-5}	0
2-point FD	-5.718×10^{-5}	-5.0×10^{-3}
4-point FD	-5.719×10^{-5}	7.6×10^{-4}
6-point FD	-5.719×10^{-5}	8.5×10^{-4}

Table 4.7: Comparison of moment derivatives vs. finite differences for the fandisk example.

Derivative Type	Computed Value	Δ vs. moments (%)
Moments	-1.0061×10^{-6}	0
2-point FD	-1.0061×10^{-6}	0.0019
4-point FD	$-1.006\,09 \times 10^{-6}$	-5.13×10^{-5}
6-point FD	$-1.006\,09 \times 10^{-6}$	-5.22×10^{-5}

Functionally Graded Material Field

To demonstrate the efficacy of moment derivatives for FGM fields, we take a wing modeled by a surface mesh and define a polynomial density field over its interior (Figure 4.16). We

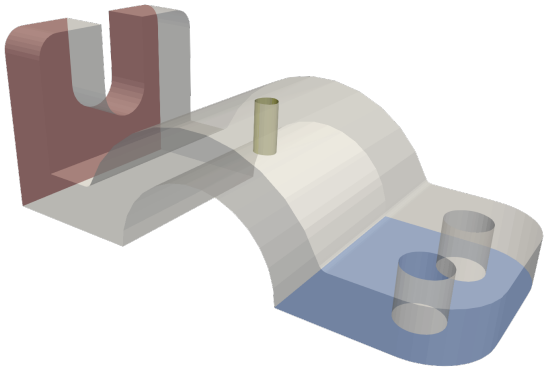


Figure 4.14: Scenario setup for the bracket hole sizing example. The red surface is loaded with a compressive 100 MPa pressure load and the blue surface is restrained. The yellow surface represents the hole whose radius is the design parameter.

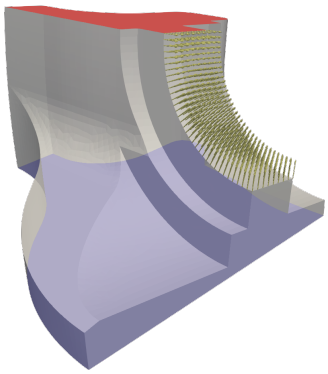
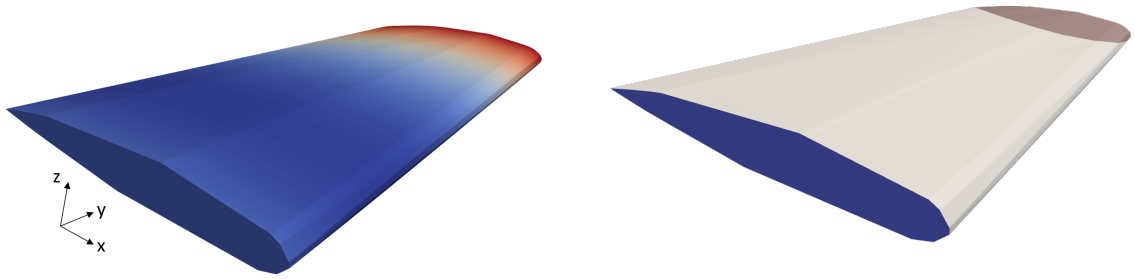


Figure 4.15: Illustration of the fandisk example, with the velocity field illustrated as vectors on the surface. The top (red) surface is loaded with 100 MPa downwards pressure and the bottom surface (blue) is restrained.

then assume an effective stiffness matrix that is proportional to a nominal stiffness of $\boldsymbol{E}_0$ as $\boldsymbol{E}(y) = \rho(y)\boldsymbol{E}_0$. This polynomial material field is $\rho(y) = c_2 y^2 - c_1 y + c_0$ where $c_2 = 0.05, c_1 = -1, c_0 = 20$. Then we compute the sensitivity of the compliance with respect to the coefficient c_2.

The elasticity scenario is a cantilever beam type problem in which the thick side of the wing is restrained and the thin side is loaded to produce a bending of the wing (Figure 4.16b). The results of the moment sensitivities versus finite difference approaches are detailed in Table 4.8.

Table 4.8: Comparison of moment derivatives vs. finite differences for shape sensitivity.



(a) (b)

Figure 4.16: Scenario setup for the FGM wing example. **(a)** Density field visualized over the wing surface mesh. **(b)** The close (blue) end is restrained and the far (red) end is loaded with a transverse 1000 kN force.

Derivative Type	Computed Value	Δ vs. moments (%)
Moments	-47279.66	0
2-point FD	-47295.51	0.03
4-point FD	-47279.63	-6.4×10^{-5}
6-point FD	-47279.66	2.55×10^{-7}

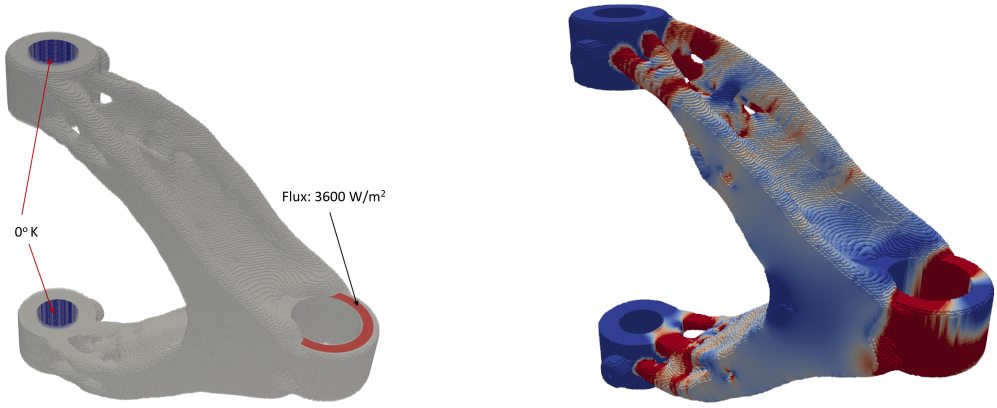
Thermal Compliance Sensitivity

Since the adjoint equation formulated in Section 4.2 can be applied to any PDE-constrained optimization problem, it can be applied to other objective functions and PDEs[155, 127]. To demonstrate this, we define the design space using a penalized thermal material model with voxel densities as in SIMP topology optimization. The bracket geometry and boundary conditions are shown in Figure 4.17a. The thermal compliance sensitivity with respect to density is shown in Figure 4.17b. Over a random subset of 1000 voxels, we computed an average sensitivity difference between a 2-point FD and moment derivatives of 0.0203%.

Mixed Design Spaces

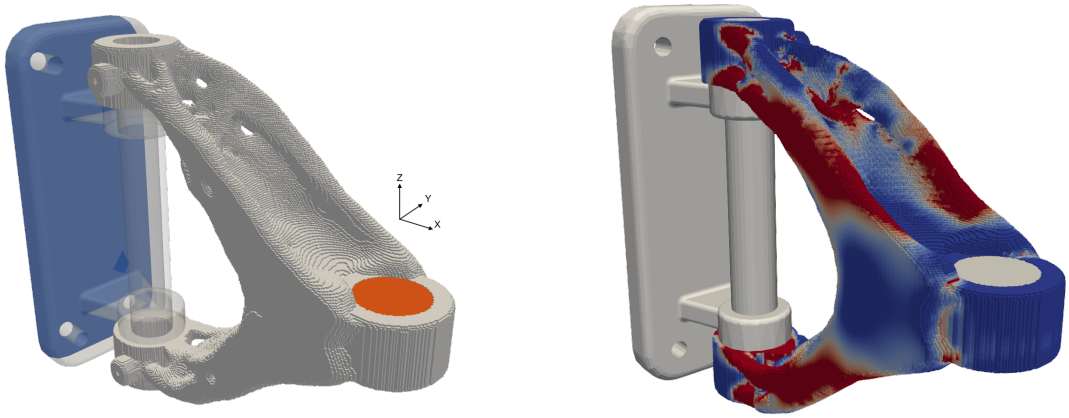
Since our finite element method is meshfree and uses moment-based numerical integration to evaluate the variational form, we can operate on heterogeneous design spaces in a single system. This can be useful in many ways including allowing simultaneous B-Rep shape optimization and topology optimization within an assembly.

To demonstrate this, we performed sensitivity analysis on a SIMP voxel model that is in an assembly with a tessellated B-Rep model. The scenario setup and sensitivity results are detailed in Figure 4.18. Note that in addition to supporting voxel density sensitivity, we can simultaneously query for surface sensitivities in the same assembly.



(a) (b)

Figure 4.17: Illustration of the thermal compliance sensitivity example. **(a)** Voxelized model of a bracket with thermal boundary conditions. **(b)** Thermal sensitivity field visualized over a voxel geometry. Red regions signify greater magnitude and blue less magnitude.



(a) (b)

Figure 4.18: Illustration of the mixed design space structural compliance sensitivity example. **(a)** Scenario setup for a mixed assembly. The blue surface is restrained and the red surface has a torque applied on the red surface of 100N$\cdot$ m^2 about the Z axis. **(b)** Structural sensitivity field visualized over the voxel geometry. Red regions signify greater magnitude and blue less magnitude.

Sensitivity Example Conclusion and Future Work

We have presented a technique for sensitivity computation involving derivatives of integral quantities called moment derivatives. These moment derivatives form a moment-fitting system that is used to generate derivative quadrature rules, which give strong guarantees on

their accuracy. This approach is applied to the problem of computing sensitivities via the adjoint form. These derivative quadrature rules are physics-independent, and can be reused for any PDE-constrained objective function expressed in the adjoint form.

This work suggests new possibilities in optimization that leverage mixed design spaces, either in an assembly together as in Section 4.2 or by switching design spaces on-the-fly during optimization.

4.3 Variability of Targeted Material Thermal Responses to Laser-Induced Heating in AM

In the following example [164], we prepare the groundwork for our UQ study on the 3D DE-FD thermal model of the SLS manufacturing process. As such, we begin by simplifying the domain down to a single region of interest, i.e. a single voxel or particle, as a completely homogeneous lumped mass of material with a single measure of temperature .

We begin with the reminder that the widespread use of lasers in AM is to induce a given temperature and a phase transformation in materials deposited onto a substrate. For a laser to induce a phase transformation in the material, the power intensity must be sufficiently high to induce melting and, in all cases, stay below a vaporization or burn-off temperature of the target material. Oftentimes, there is variability in the laser input to the target zone. For a process designer, a central question is to determine the uncertainty of the resulting target state, i.e. temperature and state (solid or melted), due to uncertainty in the energy (laser) input. This motivates the present work, which integrates relatively fundamental heat transfer models that describe the thermal effects due to

- Laser irradiation,
- Heat conduction into the surface of deposition,
- Infrared radiation outwards into the surroundings,
- Convection due to an exhaust apparatus to control the cooling of the system, and
- Phase transformations, for a dry Nylon 6 powder as a sample material.

One key advantage of this framework is that it is amenable to a sensitivity and uncertainty analysis with respect to any of its parameter inputs.

Accordingly, UQ studies are also undertaken to ascertain the relationship between variation in laser input to variation in the processed material state. Examples will be presented to illustrate the thermal behavior of the numerical model. Due to its simplicity, this framework is designed to be computationally implemented in a straightforward fashion. The model allows for rapid computation and sensitivity analyses, which are provided as numerical examples. Extensions are also given to include mass transport (losses) due to ablation of the target material.

Introduction

In this study, we will focus on a thermal model for SLS processing applied to a Nylon 6 polymer. However, this lumped mass model can be useful to simulate any metal, polymeric, or ceramic based AM process, as governed by the First Law of Thermodynamics, by employing relevant input material properties.

Heating of materials by way of laser-based processes is attractive in manufacturing due to the precision that it offers, with a nominal focus diameter not larger than 0.5 mm. Lasers offer a highly controllable method to deliver the necessary amount of localized energy to process powdered material due to its collimated (parallel) and monochromatic energy, and to its ability to pulse at a prescribed frequency by beam chopping or by modulating the voltage. Some typical lasers used are Yttrium Aluminum Garnett (YAG) for processing ceramics and metals and Carbon Dioxide (CO_2) for polymeric materials. Industrial lasers range in power, approximately from 1 to 10,000 Watts for polymeric materials. However, if not properly controlled, laser procedures can cause excessive variability in the quality of the as-built parts as well as thermal damage to the material and the substrate. These concerns have motivated the present work, in which we build a lumped mass thermal model based to simulate temperature changes of dry Nylon 6 powder due to input power from a laser beam, based on the SLS process. This model will allow us to quantify the sensitivity of the material temperature based on perturbations of the input laser power. We explore the thermal sensitivity on two lumped mass geometries: a cylindrical and a spherical domain, embedded in a previously-solidified substrate.

The outline of the paper is as follows. Initially, we must build a thermal model that allows us to achieve a target temperature *without phase transformations*, by energy balance of a control volume that accounts for

- incoming laser irradiance,
- heat conduction into the substrate,
- forced convection from an exhaust system, and
- infrared radiation to the surroundings.

We then represent this thermal model numerically, with expanded functionality to include phase changes.

An uncertainty quantification is carried out by non-dimensional parameters that describe the relative contribution of each thermal term. One key advantage of this framework is that it is amenable to a sensitivity analysis. Accordingly, uncertainty quantification studies are also undertaken to ascertain the effects of laser input perturbations on variations in the material processed state, with the potential correlation to as-built part quality. Examples will be presented to illustrate the thermal behavior of the numerical model. Due to its simplicity, this framework is designed to be computationally implemented in a straightforward fashion so that it can be easily implemented by researchers in the field, in particular those working with AM technologies.

Contributing thermal terms

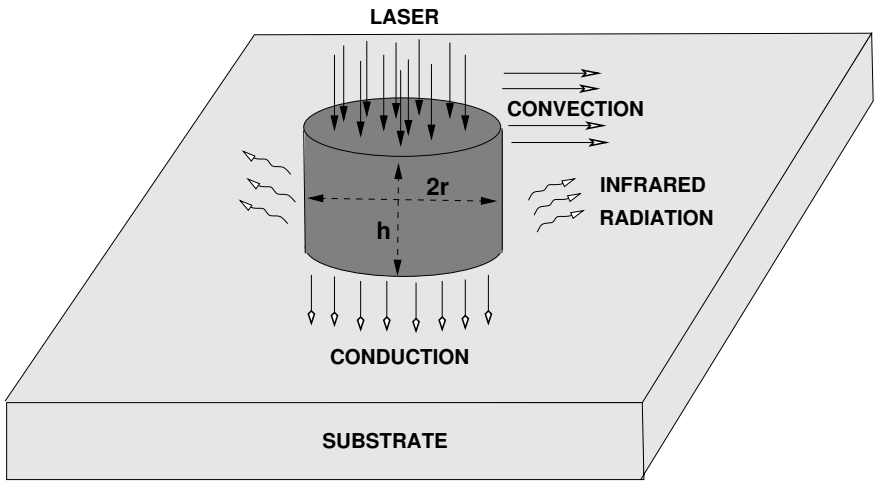


Figure 4.19: An idealized powder particle domain deposited on a substrate material and irradiated by laser energy.

Defining a "building block," we begin by considering a control volume of a lumped mass for a sample of material after it is deposited on a substrate with a prescribed temperature, as shown in Figure 4.19. For now we prescribe *no phase transformations or mass losses* in order to properly estimate the power intensity of the laser that will induce a temperature rise in within a specific time duration, or dwell time, over the sample. This thermal model will serve as the basis for this research. The First Law of Thermodynamics serves as our governing equation in order to perform an overall balance to quantify the rate of change of the stored energy ($\dot{W}$) and the heat input, which in the simple case of no mass transfer and constant heat capacity reads as $\left(\dot{W} = \frac{d}{dt}(MHT) = MH\dot{T}\right)$

$$MH\dot{T} = \mathcal{L}_1 + \mathcal{L}_2 + \mathcal{L}_3 + \mathcal{L}_4, \tag{4.50}$$

where M is the mass inside of the control volume, and H is the thermodynamic heat capacity. More about H in Section 4.3. Both M and H are assumed constant, initially. We have the following contributing effects (denoted $\mathcal{L}_1 \rightarrow \mathcal{L}_4$):

- Absorbed incident optical radiation (laser beam): $\mathcal{L}_1 \overset{\text{def}}{=} I^a A_I = (I^i - I^r)A_I$, where I^a is the absorbed radiation per unit area, $I^r = RI^i$ is the reflected radiation, R is the reflectivity of the material, I^i is the incident radiation per unit area, and A_I is the area of the exposed (irradiated) surface.

- Conduction to the substrate (base): $\mathcal{L}_2 \overset{\text{def}}{=} \frac{\mathbb{K}}{L}(T_B - T)A_B$, where T and T_B are the temperatures of the material and of the substrate (base) onto which the target material

is deposited, respectively, A_B is the area of contact between the base and the target material, $\mathbb{K}$ is the thermal conductivity of the material, and L is the length scale across which thermal conduction occurs.

- Emitted thermal radiation: $\mathcal{L}_3 \overset{\text{def}}{=} \epsilon\sigma(T_S^4 - T^4)A_{RS} + \epsilon\sigma(T_B^4 - T^4)A_{RB}$, where T_S is the temperature of the surrounding air in the build chamber, $0 \leq \epsilon \leq 1$ is the target material emissivity, $\sigma = 5.670373 \times 10^{-8} \frac{W}{m^2 \cdot K^4}$ is the Stefan-Boltzmann constant, A_{RS} is the area of the radiative surface exposed to the surroundings and $A_{RB} = A_B$ is the area of the radiative surface in contact with the substrate material.

- Convection (e.g. from an exhaust system): $\mathcal{L}_4 \overset{\text{def}}{=} h_C(T_S - T)A_C$, where h_C is the effective surface convection coefficient, and A_C is the area of the convective surface exposed to the surroundings.

Relative contributions: order of magnitude analysis

In this section, we create non-dimensional parameters for each thermal contribution so that we can ultimately quantify the sensitivity of the model due to each term. We achieve this by normalizing the governing equation (Equation 4.50), yielding

$$\dot{T} = \frac{I^a A_I}{MH}\left(1 + \frac{\mathcal{L}_2}{\mathcal{L}_1} + \frac{\mathcal{L}_3}{\mathcal{L}_1} + \frac{\mathcal{L}_4}{\mathcal{L}_1}\right). \tag{4.51}$$

The non-dimensional expressions of the contributing thermal terms are defined according to Figure 4.19, leaving behind the pertaining temperatures, the dimensions of the target control volume, and the laser-power as variables,

- The ratio of conductive to laser contributions is:

$$\begin{aligned} \frac{CONDUCTION}{LASER} &= \frac{\mathcal{L}_2}{\mathcal{L}_1} = \frac{\mathbb{K}(T_B - T)A_B}{LI^a A_I} \\ &\approx \frac{\mathcal{O}(10^{-1})(T_B - T)A_B}{LI^a A_I} \\ &\approx \frac{T_B - T}{LI^a}\mathcal{O}(10^{-1}), \end{aligned} \tag{4.52}$$

  where $0.1 \leq \mathbb{K} \leq 0.5 \frac{W}{m \cdot K}$ for typical polymeric materials, and the ratio of the areas is assumed to be of order unity ($A_I \approx A_B$).

- The ratio of infrared radiation to laser contributions is:

$$\begin{aligned}\frac{RADIATION}{LASER} &= \frac{\mathcal{L}_3}{\mathcal{L}_1} = \frac{\epsilon\sigma(T_S^4 - T^4)A_{RS}}{I^a A_I} + \frac{\epsilon\sigma(T_B^4 - T^4)A_{RB}}{I^a A_I} \\ &\approx \frac{\mathcal{O}(10^{-1})\mathcal{O}(10^{-8})(T_S^4 - T^4)(A_{RS} + A_{RB})}{I^a A_I} \\ &\approx \frac{T_S^4 - T^4}{I^a}\mathcal{O}(10^{-9}),\end{aligned} \tag{4.53}$$

  where the ratio of the areas is once again assumed to be of order unity $(A_I \approx A_{RB} + A_{RS})$.

- The ratio of convective to laser contributions is:

$$\begin{aligned}\frac{CONVECTION}{LASER} &= \frac{\mathcal{L}_4}{\mathcal{L}_1} = \frac{h_C(T_S - T)A_C}{I^a A_I} \\ &\approx \frac{\mathcal{O}(10^{-1})(T_S - T)A_C}{I^a A_I} \\ &\approx \frac{T_S - T}{I^a}\mathcal{O}(10^{-1}),\end{aligned} \tag{4.54}$$

  again, the ratio of the areas is assumed to be of order unity $(A_I \approx A_C)$.

Even though we assume that the polymer material absorbs a smaller portion of the incident irradiance than the reflected irradiance, we can still rewrite the absorbed irradiance per unit area in terms of the power input, $I^a \approx \frac{P\beta}{A_I}$ where $\beta = \frac{\mathcal{L}_1}{P}$ denotes the fraction of absorbed power, yielding

- The ratio of conductive to laser contributions is:

$$\frac{CONDUCTION}{LASER} \approx \frac{T_B - T}{L\frac{P\beta}{A_I}}\mathcal{O}(10^{-1}), \tag{4.55}$$

- The ratio of infrared radiation to laser contributions is:

$$\frac{RADIATION}{LASER} \approx \frac{T_S^4 - T^4}{\frac{P\beta}{A_I}}\mathcal{O}(10^{-9}), \tag{4.56}$$

- The ratio of convective to laser contributions is:

$$\frac{CONVECTION}{LASER} \approx \frac{T_S - T}{\frac{P\beta}{A_I}}\mathcal{O}(10^{-1}). \tag{4.57}$$

Using dermatological applications as a guiding standard for the laser power input, we can say that it ranges as $P\beta = 10^n$ Watts $(0 \leq n \leq 3)$. Assuming that we can focus the entire irradiance per unit area on the target we can express this quantity as

$$\frac{1}{A_I} \leq I^a \leq \frac{10^n}{A_I}, \tag{4.58}$$

and so if we assume $I^a \approx \frac{10^n}{A_I}$, we can generate the expressions for the dimensionless ratios for each thermal contribution compared to the laser energy with an explicit particle size factor, A_I.

- The ratio of conductive to laser contributions is:

$$\frac{CONDUCTION}{LASER} \approx \frac{(T_B - T)A_I}{L}\mathcal{O}(10^{-1-n}), \tag{4.59}$$

- The ratio of infrared radiation to laser contributions is:

$$\frac{RADIATION}{LASER} \approx (T_S^4 - T^4)\ A_I\ \mathcal{O}(10^{-9-n}), \tag{4.60}$$

- The ratio of convective to laser contributions is:

$$\frac{CONVECTION}{LASER} \approx (T_S - T)\ A_I\ \mathcal{O}(10^{-1-n}). \tag{4.61}$$

Clearly, for small particle size (or control volume) L, as it is the case in this study, conduction can play a significant role. The idealized protrusion in Figure 4.19 serves as an example, with geometric quantities

- Area in contact with base material: $A_B = \pi r^2$,
- Area exposed to the surroundings: $A_C = 2\pi rL + \pi r^2$,
- Area of radiative surface exposed to surroundings: $A_{RS} = 2\pi rL + \pi r^2$,
- Area of radiative surface exposed to base material: $A_{RB} = \pi r^2$ and
- Area exposed to laser energy irradiation: $A_I = \pi r^2$.

For $r \approx L$, this yields $\frac{A_I}{L} \approx L$ and for $r > L$ yields $\frac{A_I}{L} > r$. These relationships hint at the fact that, for a small target measuring $r \approx 10^{-4}\ m$, the thermal conduction term will dominate the energy balance, and it could become on the order of the laser input.

Note that these comparative ratios are dynamic due to changing temperature difference factor during material cooling or heating. Most notably, the infrared radiation ration expression will undergo the most change since it is a function of temperatures raised to the fourth power. Therefore, we expect that radiation effects will not be significant when the material maintains a low temperature, but it will become a major contributor to the thermal field during higher material temperatures.

Analytics: thermal responses for special cases

In order to obtain insight into the sensitivity of the system response, we now consider special cases that are amenable to analytical methods. Afterwards, we will treat the system numerically, with phase transformations included.

Special Thermal Cases

Case 1: Laser irradiance and conduction Consider now a special case of the general set of thermal models where we only consider laser irradiation and conduction. Equation 4.50 simplifies to

$$MH\dot{T} = I^a A_I + \frac{\mathbb{K} K A_B}{L}(T_B - T), \tag{4.62}$$

and it can be solved analytically, assuming $T\,(t=0) = T_B$, yielding

$$T(t) = T_B + \frac{I^a A_I L}{\mathbb{K} K A_B}\left(1 - e^{-\frac{\mathbb{K} K A_B t}{MHL}}\right). \tag{4.63}$$

This relationship allows us to observe that:

- The rise time for the temperature is dictated by the ratio of the conduction to the thermodynamic heat capacity, $\frac{\mathbb{K} K A_B}{MHL}$.
- At steady-state, $e^{-\frac{\mathbb{K} K A_B t}{MHL}} \to 0$, and

  $$T_\infty = T_B + \frac{I^a A_I L}{\mathbb{K} K A_B}, \tag{4.64}$$

  which indicates that the ratio of I^a to $\frac{\mathbb{K} K}{L}$ dictates the steady state temperature T_∞ (assuming $A_I \approx A_B$).
- A substrate design or material that highly promotes thermal conduction, where the conductive losses of the control volume are instantaneous, we have $\mathbb{K} K \to \infty$ and $T(t) = T_B$.
- A substrate design that is highly insulating, where the conductive losses are zero, we have $\mathbb{K} K \to 0$ and $T(t) = T_B + \frac{I^a A_I t}{MH}$. Here, the control volume will trap a maximum amount of heat and the radiative term will be increasingly more important in the energy balance equation.

Case 2: Laser irradiance, conduction, and convection Convection is an ever-present phenomenon and an integral part of AM processes since it helps to control the cooling profile of the part, both during and after printing. Accordingly, we add convection to Case 1, leading to

$$MH\dot{T} = I^a A_I + \frac{\mathbb{K} K A_B}{L}(T_B - T) + h_C A_C (T_S - T), \tag{4.65}$$

which yields a known analytical solution of

$$T(t) = \left(T_B - \frac{B}{A}\right) e^{-At} + \frac{B}{A}, \tag{4.66}$$

where we have

$$A = \left(\frac{\mathbb{K} K A_B}{MHL} + \frac{h_C A_C}{MH}\right), \tag{4.67}$$

and

$$B = \left(\frac{\mathbb{K} K A_B T_B}{MHL} + \frac{h_C A_C T_S}{MH} + \frac{I^a A_I}{MH}\right). \tag{4.68}$$

The quotient is

$$\frac{B}{A} = \frac{\left(\frac{\mathbb{K} K A_B T_B}{MHL} + \frac{h_C A_C T_S}{MH} + \frac{I^a A_I}{MH}\right)}{\left(\frac{\mathbb{K} K A_B}{MHL} + \frac{h_C A_C}{MH}\right)} = \frac{\left(\frac{\mathbb{K} K A_B T_B}{L} + h_C A_C T_S + I^a A_I\right)}{\left(\frac{\mathbb{K} K A_B}{L} + h_C A_C\right)} \tag{4.69}$$

Inverting the solution to solve for the time for the temperature to reach a critical temperature, such as a known burning temperature, $T(t) = T^*$, yields

$$t = t^* = ln\left(\frac{T^* - \frac{B}{A}}{T_B - \frac{B}{A}}\right)^{-\frac{1}{A}}. \tag{4.70}$$

Note that when the argument of the logarithmic function is negative, the system can never reach the burning temperature. For the range of parameters in this study, the denominator always remains negative, thus the sign of the numerator dictates whether the laser power is sufficient to reach the burning temperature. Explicitly,

$$T^* - \frac{B}{A} \leq 0 \Rightarrow \frac{B}{A} = \frac{\left(\frac{\mathbb{K} K A_B T_B}{L} + h_C A_C T_S + I^a A_I\right)}{\left(\frac{\mathbb{K} K A_B}{L} + h_C A_C\right)} \geq T^*. \tag{4.71}$$

Very strong conduction and/or convection at the laser intensity setting chosen can cause an inability to meet this criterion. According to Equation 4.70, as the convective coefficient approaches infinity, $h_C \to \infty$, the time to burning also increase to infinity, $t^* \to \infty$. Furthermore, from Equation 4.69, as the convective coefficient increases to infinity, $h_C \to \infty$, then $\frac{B}{A} \to T_S$. In this case, a specified critical temperature, T^*, can never be reached.

Sensitivity Analyses

An important question to address is the sensitivity of the thermal response to the system parameters. In this study we pay special attention to laser power input perturbations. Since we assume that the power delivery rate is uniform for a desired simulated dwell time $\tilde{t}$, then the laser must be stationary above the domain. In the case of a laser in motion, the power delivery rate during $\tilde{t}$ would be variable according to a Gaussian shape of typical laser beam. Specifically, consider the following cases:

Case 1: Laser irradiance and conduction

- Sensitivity to laser energy input

  We find the sensitivity of the temperature of the material to the laser power input I^a with Equation 4.63 such that,

$$\frac{\partial T(t)}{\partial I^a} = \frac{A_I L}{\mathbb{K} A_B}\left(1 - e^{-\frac{\mathbb{K} A_B t}{MHL}}\right). \tag{4.72}$$

  As $t \to \infty$,

$$\frac{\partial T(t)}{\partial I^a} \to \frac{A_I L}{\mathbb{K} A_B} = \frac{A_I}{A_B}\frac{L}{\mathbb{K}}. \tag{4.73}$$

  which clearly shows the sensitivity's dependence on the dimensionless ratio of the area of the laser and the area on the conductive base.

- Process variation

  A variation of the laser power ΔI^a is one parameter that can cause process variation ΔT. This variation could arise from extrinsic factors such as human error, gradual fouling of the laser lens, fluctuations of power in the laboratory, etc. To understand this effect, we consider ΔI^a as the sole source of process variation. We express this as

$$\Delta T = \frac{\partial T(t)}{\partial I^a}\Delta I^a = \underbrace{\left(\frac{A_I L}{\mathbb{K} A_B}\left(1 - e^{-\frac{\mathbb{K} A_B t}{MHL}}\right)\right)}_{\alpha(t)}\Delta I^a \tag{4.74}$$

  As $t \to \infty$,

$$\Delta T = \frac{\partial T(t)}{\partial I^a}\Delta I_a = \underbrace{\left(\frac{A_I}{A_B}\frac{L}{\mathbb{K}}\right)}_{\alpha^{\infty}}\Delta I^a. \tag{4.75}$$

  Predefining a laser dwell time over the material $\tilde{t}$, we can define the process variation as

$$\Delta T\left(\tilde{t}\right) = \frac{\partial T(\tilde{t})}{\partial I^a}\Delta I^a = \underbrace{\left(\frac{A_I L}{\mathbb{K} A_B}\left(1 - e^{-\frac{\mathbb{K} A_B \tilde{t}}{MHL}}\right)\right)}_{\alpha(\tilde{t})}\Delta I^a. \tag{4.76}$$

  Explicitly, $\Delta T\left(\tilde{t}\right)$ represents the range of temperatures expected, across all samples, at the end of the dwell time $\tilde{t}$ if we expect the laser intensity to vary within the predetermined perturbation range ΔI^a. Assuming the laser intensity parameter variation has a uniform distribution, a standard deviation of the variation response $T\left(\tilde{t}\right)$ can be quantified.

Case 2: Laser irradiance, conduction, and convection

- Sensitivity to laser energy input

  We find the sensitivity of the final temperature of the material due to laser power input perturbations ∂I^a using Equation 4.66 such that,

$$\frac{\partial T(t)}{\partial I^a} = \frac{A_I(1-e^{-At})}{\frac{\mathbb{K}A_B}{L} + h_C A_C}. \tag{4.77}$$

  As $t \to \infty$,

$$\frac{\partial T(t)}{\partial I^a} \to \frac{A_I}{\frac{\mathbb{K}A_B}{L} + h_C A_C} \tag{4.78}$$

  which clearly shows the sensitivity's dependence on the dimensionless ratio of the area of the laser, the area on the conductive base and the convective area.

- Process variation

  Again, we consider the process variation due to predetermined perturbations of laser intensity from extrinsic factors such that,

$$\Delta T = \frac{\partial T(t)}{\partial I^a}\Delta I_a = \underbrace{\left(\frac{A_I(1-e^{-At})}{\frac{\mathbb{K}A_B}{L} + h_C A_C}\right)}_{\alpha(t)} \Delta I_a. \tag{4.79}$$

  As $t \to \infty$,

$$\Delta T = \frac{\partial T(t)}{\partial I^a}\Delta I_a \to \underbrace{\left(\frac{A_I}{\frac{\mathbb{K}A_B}{L} + h_C A_C}\right)}_{\alpha^\infty} \Delta I_a, \tag{4.80}$$

  which clearly shows the sensitivity's dependence on the dimensionless ratio of the area of the laser, the area on the conductive base, and the convective area.

  Finally, we define the process variation for a certain laser irradiation dwell time,

$$\Delta T\left(\tilde{t}\right) = \frac{\partial T(\tilde{t})}{\partial I^a}\Delta I_a = \underbrace{\left(\frac{A_I(1-e^{-A\tilde{t}})}{\frac{\mathbb{K}A_B}{L} + h_C A_C}\right)}_{\alpha(\tilde{t})} \Delta I_a. \tag{4.81}$$

In all cases, the factor α is the multiplier which indicates the scaling of the laser input perturbation. In all cases, high conductivity and high convection leads to less sensitivity due to laser perturbations.

General time-transient simulations with phase-transformations

Until this point in the analysis, we have not included phase transformations and latent heats. At this point, we must be more careful with the definition of the heat capacity. For example, consider a widely used model, writing the thermodynamic stored energy rate directly as

$$\frac{dW}{dt} = \frac{d(Mw)}{dt} = \frac{dM}{dt}w + M\frac{dw}{dt} = \frac{dM}{dt}w + M\frac{\partial w}{\partial T}\frac{dT}{dt} = \frac{dM}{dt}w + MH\frac{dT}{dt}. \tag{4.82}$$

Here, w is the energy per unit mass and $H = \frac{\partial w}{\partial T}$ represents the *thermodynamic* specific heat capacity. Thus,

$$\dot{W} = \dot{M}w + MH\dot{T} = \mathcal{L}_1 + \mathcal{L}_2 + \mathcal{L}_3 + \mathcal{L}_4. \tag{4.83}$$

Remark. *An alternative simple model, used often in continuum mechanics applications, is to write the thermodynamic stored energy directly as*

$$W = MCT, \tag{4.84}$$

which is rather phenomenological. The heat capacity parameter C is time dependent (implicitly through T-dependency). The rate of energy reads as

$$\frac{dW}{dt} = \dot{M}CT + M\dot{C}T + MC\dot{T}, \tag{4.85}$$

thus

$$\dot{W} = \frac{d}{dt}(MCT) = \dot{M}CT + M\dot{C}T + MC\dot{T} = \mathcal{L}_1 + \mathcal{L}_2 + \mathcal{L}_3 + \mathcal{L}_4. \tag{4.86}$$

Equations 4.83 and 4.86 are mathematically equivalent. To see this, one can write:

$$\dot{W} = \dot{M}w + MH\dot{T} = \dot{M}CT + M\dot{C}T + MC\dot{T}. \tag{4.87}$$

For example, in the case where $\dot{M} = 0$

$$\dot{W} = MH\dot{T} = M\dot{C}T + MC\dot{T} \Rightarrow H\dot{T} = \dot{C}T + C\dot{T}, \tag{4.88}$$

or, alternatively

$$\frac{\partial w}{\partial T} = H = \frac{\partial C}{\partial T}T + C. \tag{4.89}$$

The advantages of one representation over another depend on the data available. Despite some mathematical advantages in using the model in Equation 4.84 (in cases of mass transfer), we will continue to use the model in Equation 4.83, due to more readily available data.

Phase transformations

To include phase transformations, for example melting and solidification, we include a thermally-dependent heat capacity (due to latent heats, and the material phase) by utilizing the following cases:

- *Solid* $\rightarrow$ *solid*: If $T(t) < T_m$ and $T(t + \Delta t) < T_m$ then no melting occurs, so we use $H(T(t)) = H_S$ in the governing equation.

- *Solid* $\rightarrow$ *liquid*: If $T(t) < T_m$ and $T(t + \Delta t) \geq T_m$ then melting occurs, so we use $H(T(t)) = H_S + \frac{\delta \mathcal{P}^{S \rightarrow L}}{\delta T}$ in the governing equation.

- *Liquid* $\rightarrow$ *liquid*: If $T(t) \geq T_m$ and $T(t + \Delta t) \geq T_m$ then material remains melted, so we use $H(T(t)) = H_L$ in the governing equation.

- *Liquid* $\rightarrow$ *solid*: If $T(t) \geq T_m$ and $T(t + \Delta t) < T_m$ then solidification occurs, so we use $H(T) = H_L + \frac{\delta \mathcal{P}^{L \rightarrow S}}{\delta T}$ in the governing equation.

Here, H_S is the solid phase heat capacity and H_L is the liquid phase heat capacity and

- $0 < \delta \mathcal{P}^{S \rightarrow L}$ represents the latent heat of melting,

- $0 < \delta \mathcal{P}^{L \rightarrow S}$ represents the latent heat of solidification, and

- $0 < \delta T$ represents a "bandwidth" over which the phase transformation takes place.

Remark. *Latent heats are defined to represent the energy absorbed or released to transform the material from one initial phase to the new phase during a* constant-temperature *process for a certain time interval. In this case we want to observe the effects of a constant energy input rate on the material temperature.*

The latent heats modify the heat capacity term H inside of the temperature bandwidth δT so that physically correct effects of a phase change are captured. For example, in the case of melting (solid to liquid), the latent heat of melting increases the heat capacity during the phase transformation and so a low temperature rise can be observed[2]. This model is relatively straightforward to include within the upcoming computational framework.

[2]In the idealized limit, the temperature would be constant.

System discretization

We employ an implicit Trapezoidal Rule time integration scheme for solving the balance of energy (assuming that the mass is constant). This time integration scheme calculates the material temperature at a half time step ($\phi = 0.5$) with contributions from the temperatures at the current time step and at the next time step. We first guess the temperature at the next time step and then we iterate until we converge on a solution, within a prescribed tolerance. Normally, we were able to gain convergence well within 6 iterations. In this study, we chose ($\phi = 0.5$), meaning that the contributions from the temperatures at the current and at the next time step have equal weights. Again, the governing equation for energy balance is,

$$MH\dot{T} = \mathcal{L}_1 + \mathcal{L}_2 + \mathcal{L}_3 + \mathcal{L}_4, \tag{4.90}$$

thus,

$$\dot{T}|_{t+\phi\Delta t} \approx \frac{T(t+\Delta t) - T(t)}{\Delta t} = \frac{1}{MH(t+\phi\Delta t)} \left((\mathcal{L}_1 + \mathcal{L}_2 + \mathcal{L}_3 + \mathcal{L}_4)\,|_{t+\phi\Delta t} \right),$$

leading to,

$$T(t+\Delta t) = T(t) + \left(\frac{\Delta t}{MH} \mathcal{L}_1 + \mathcal{L}_2 + \mathcal{L}_3 + \mathcal{L}_4 \right) |_{t+\phi\Delta t}. \tag{4.91}$$

The solution procedure is straightforward. At a given time step:

- STEP 1: Update temperature:

$$T^{k+1}(t+\Delta t) = T(t) + \left(\frac{\Delta t}{MH} \left(\mathcal{L}_1 + \mathcal{L}_2 + \mathcal{L}_3 + \mathcal{L}_4 \right) \right)^k |_{t+\phi\Delta t} \tag{4.92}$$

- STEP 2: Check for phase-transformations and update $H(t + \phi\Delta t)$.
- STEP 3: Iterate until convergence (k=1, ...) to within a predetermined tolerance, and time is advanced.
- STEP 4: Repeat STEPS 1-3 at the next time-step.

Numerical examples with laser perturbations

We consider the idealized lumped mass cylindrical domain shown in Figure 4.19, irradiated by a prescribed laser power P in Watts. The parameters used pertained to material properties of a typical polyamide Nylon 6, which is widely used in SLS processes.

Lumped mass problem definition

The material properties and process parameters were:

- $L = 80.1\,\mu m$,
- $r = 80.1\,\mu m$,
- $T_B = 295\,K$,
- $T_S = 295\,K$,
- $T^* = 10^6\,K$,
- $\rho = 1130\,\frac{kg}{m^3}$,
- $\mathbb{K} = 0.23\,\frac{W}{m \cdot K}$,
- $\epsilon = 0.75$, and
- $h_C = 5.7\,\frac{W}{m^2 \cdot K}$,

where the particle size geometries L and r are based on a mean particle size of a proprietary dry Nylon 6 powder formulation commercialized by BASF for SLS processing; the temperature parameters T_B, T_S, and T^* were based on real process settings typically employed by researchers at BASF laboratories; the thermal conductivity ϵ value follows work from experimental methods by Okada et. al [98]; the density ρ and conductivity $\mathbb{K}$ values taken from data published by the Polymer Data Handbook [84]; and the convection coefficient assumes free convection values based on Equation 1 as reported by Sartori [121] for a vertical plate.

A population of these idealized lumped mass domains were subjected to laser power perturbations. Generated P_i parameter input values, where $i = 1, 2, ...N$, were based on an evenly spaced uniform distribution with a range of $\pm\Delta P$ and mean $\bar{P}$ to simulate perturbations in the laser power input from extrinsic effects (again, human error or lens fouling). Furthermore, a battery of cases at various $\bar{P}$ levels were run, each corresponding to a laser power intensity level I^a, in order to quantify its effects on the sensitivity of the thermal field at time $\tilde{t}$. For these stochastic studies, we used the following laser input and sample population parameters:

- $N = 100$ samples,
- $0.004\,\frac{W}{m^2} \leq I^a \leq 1.5\,\frac{W}{m^2}$,
- $\frac{\Delta P}{\bar{P}} = 0.1$,
- $\tilde{t} = 1\,s$,

For the possible phase transformations, we used Nylon 6 thermodynamic data reported by studies performed by Millot et al. [87] and data published by the *Polymer Data Handbook* [84]. These parameters are:

- $T_m = 493\,K$,
- $H_S\,(T) = 4.502 \cdot T + 138.7\,\frac{J}{kg \cdot K}$,
- $H_L\,(T) = 1.348 \cdot T + 1979.3\,\frac{J}{kg \cdot K}$,
- $\frac{\delta \mathcal{P}^{S \to L}}{\delta T} = 1\,\frac{J}{kg \cdot K}$,
- $\frac{\delta \mathcal{P}^{L \to S}}{\delta T} = 1\,\frac{J}{kg \cdot K}$.

Also, we assumed that the surrounding substrate and environment are heat sinks whose temperatures are constant. The time-step size Δt was chosen to be sufficiently small so that the results are not sensitive to further reductions of Δt. This indicates that we expect the results to contain negligible numerical error.

Uncertainty quantification

This section gives a brief overview of the moments of distributions for developing uncertainty calculations. For any quantity of interest Y with a distribution of values (Y_i, where $i = 1, 2, \ldots, N$, where $N = \#samples$) about an arbitrary reference point, denoted $Y^\star$, as follows:

$$\mathbf{Z}_r^{Y_i - Y^\star} \stackrel{\text{def}}{=} \frac{\sum_{i=1}^{N_p} a_i (Y_i - Y^\star)^r}{\sum_{i=1}^{N_p} a_i} \stackrel{\text{def}}{=} \overline{(Y_i - Y^\star)^r}, \tag{4.93}$$

where r denotes the "r$^{\text{th}}$" moment of the distribution. The various moments characterize the distribution, for example:

1. $\mathbf{Z}_1^{Y_i - \mu}$ equals zero as it measures the first deviation about the mean μ,
2. $\mathbf{Z}_1^{Y_i - 0} \stackrel{\text{def}}{=} \frac{\sum_{i=1}^{N_p} a_i (Y_i - 0)}{\sum_{i=1}^{N_p} a_i} \stackrel{\text{def}}{=} \overline{(Y_i - 0)} = \mu$, is the mean
3. $\mathbf{Z}_2^{Y_i - \mu}$ is the standard deviation,
4. $\mathbf{Z}_3^{Y_i - \mu}$ is the skewness (third moment), which is an indicator of the asymmetry, or bias, and
5. $\mathbf{Z}_4^{Y_i - \mu}$ is the kurtosis (fourth moment), which indicates the "tightness" of the distribution.

The two most important quantities are the mean (μ) and the standard deviation (S), and a key dimensionless quantity is S/μ which can be interpreted either as the quantification of the uncertainty around the mean, or as a percentage error. For the purposes of this analysis, $a_i = 1$, $\mu = \bar{T}\left(\tilde{t}\right)$, and $S = \Delta T\left(\tilde{t}\right)$. We must quantify $\frac{\Delta T}{\bar{T}}$ at each laser power intensity level I^a based on the distribution of the final material temperatures $T\left(\tilde{t}\right)$ across the N samples. This $\frac{\Delta T}{\bar{T}}$ quantification serves as a gauge for the process sensitivity; a higher value corresponds to a wider spread of final temperatures.

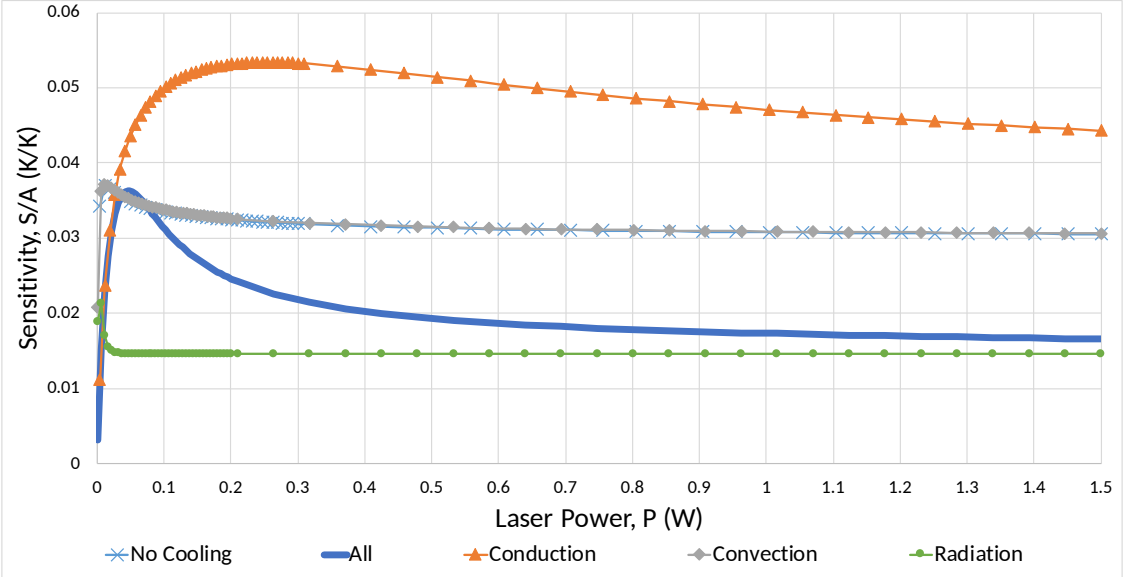
Numerical results

Figures 4.20a and 4.20b depict the process sensitivity results varying the input wattage between $0.004 - 1.5$ W. These plots show the normalized standard deviation $\frac{\Delta T}{\bar{T}}$, otherwise known as the sensitivity, of the final temperatures from five cases:

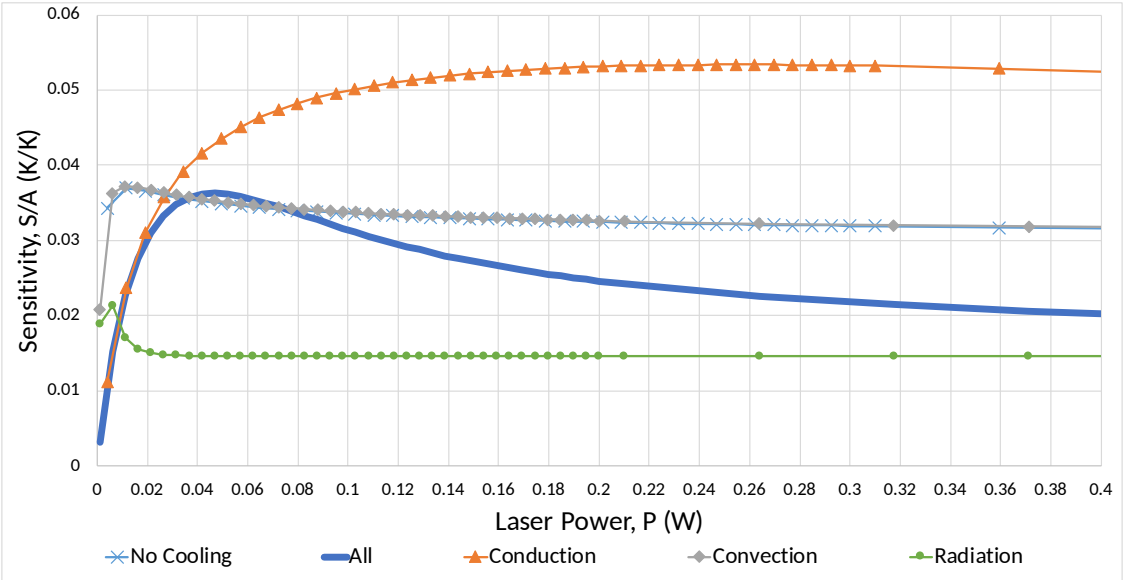
1. Isolated thermal contributions from conduction,
2. Isolated thermal contributions from convection,
3. Isolated thermal contributions from radiation,
4. No cooling, and
5. All thermal contributions (conduction, convection, and radiation).

Table 4.9 illustrates the average resulting temperature $\bar{T}$, the standard deviation ΔT, and the normalized standard deviation, or sensitivity, with perturbations of the laser input power. The peak normalized standard deviation occurs at approximately 0.05 W, and so material samples will experience an irradiation within the range of 0.05 W $\pm 10\%$. Material samples experiencing laser irradiation closer to the higher bound will have fully attained their steady state temperature, whereas the samples that experience laser inputs closer to the lower bound are in the process of heating up towards their steady state temperature. As expected, the steady state temperature is unique for each laser power input level, and it is based on the cooling effects from radiated heat at higher material temperatures.

Samples that have reached the steady state temperature have transitioned from chiefly conductive cooling to radiative cooling. As the laser input power continues to be increased, more and more material samples will have reached their steady state temperature and so the range of the final temperatures gets increasingly tighter. To reinforce these conjectures, we must focus on the trends on Figures 4.20a and 4.20b, especially on the bold line that represents the process sensitivity for the case when all thermal contributions are considered. Initially, between 0.004 and 0.04 W, this bold line closely follows the trend when *only conduction* is considered, serving as evidence that the effects due to conduction indeed dominate the thermal balance within this laser power input range. However, with higher laser power the material is allowed to reach higher final temperatures, and radiation takes over as this term is a function of temperature raised to the fourth power. The bold line now deviates



(a) Sensitivity of final temperatures due to laser irradiation perturbations.



(b) A more detailed view of the high-sensitivity area.

Figure 4.20: Cylindrical Domain: Sensitivity of final temperatures due to laser irradiation perturbations for a lumped mass for cases (1) without thermal contributions (no cooling), (2) with all thermal contributions, and (3)-(5) with isolated thermal contributions.

from the conduction-only case and approaches the radiation-only trend. Ultimately, the effects due to radiation will overcome the effects by conduction and convection.

Table 4.9: Cylindrical Domain: Thermal variation as a function of laser input power with perturbations. The peak normalized standard deviation occurs when some sample have reached their steady state temperature and others have not, at approximately 0.05 W.

$\tilde{P}$, Avg. Input (W)	$\tilde{T}$, Avg. Final Temp. (K)	ΔT, Std. Deviation (K)	$\frac{\Delta T}{\tilde{T}}$
0.001	312.06	1.00	0.0032028
0.04	962.74	34.77	0.036123
0.08	1,451.62	48.35	0.033308
0.12	1,797.73	52.48	0.029197
0.16	2,028.36	54.05	0.026650

It is interesting to note that for the surface convection coefficient h_C chosen, its effects are negligible. We can make this claim since the sensitivity trend for the case with just convection follows closely the trend for the case without cooling effects. In this study, the surface convection coefficient was set at $h_C = 5.7\frac{W}{m^2 \cdot K}$, yielding sensitivity differences between these cooling cases within 1.3×10^{-5}. We must also make note that these results chiefly pertain to the material of choice: Nylon 6.

The temperatures are, of course, unrealistically high after melting (and would probably vaporize), and are just for illustration purposes. In these cases, one should consider vaporization and mass transfer, which is discussed in the conclusions.

A modification for a typical powder particle geometry

A particular example of interest in additive manufacturing is a particle embedded in a powder bed as shown in Figure 4.21, which is being irradiated in order to sinter it to the system. This is, of course, the basis of additive manufacturing. The previous algorithm is the same, except that the volume and surface areas are different than the cylindrical domain example discussed previously. We can now define the surface areas in this model problem that affect the energy balance terms.

- Area in contact with base material: $A_B = \frac{1}{2} \times 4\pi r^2$,
- Area exposed to the surroundings: $A_C = \frac{1}{2} \times 4\pi r^2$,
- Area of radiative surface exposed to surroundings: $A_{RS} = \frac{1}{2} \times 4\pi r^2$,
- Area of radiative surface exposed to base material: $A_{RB} = \frac{1}{2} \times 4\pi r^2$ and
- Effectively projected area exposed to laser energy irradiation: $A_I = \pi r^2$.

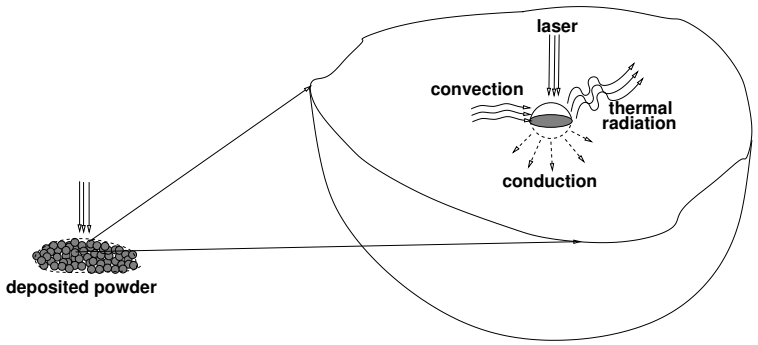


Figure 4.21: An idealization of an embedded particle in a powder pack.

This yields $\frac{A_I}{r} \approx \pi r$. As an example, assuming a relatively small target with a radius of $r \approx 10^{-3}\ m$ the conductive term will be highly significant, as compared to the convective and radiative terms in the energy balance. Additionally, the conductive term could be on the order of the laser input. Thus, fundamentally, the results should be similar to the case with the cylindrical domain with a qualitative shift of the exact power input setting that corresponds to the maximum sensitivity value. In fact, as seen in Figure 4.22 and Table 4.10, the maximum sensitivity has shifted to approximately $0.11W$. This is primarily because of the slight difference of geometric terms that control the relative contributions of convection, conduction, and thermal radiation. Following the same analysis done for a cylindrical domain in Equation 4.59, we can quantify the order of magnitudes of the energy balance terms for a sphere.

- The ratio of conductive to laser contributions is:

$$\frac{CONDUCTION}{LASER} \approx (T_B - T)\pi r\ \mathcal{O}(10^{-1-n}), \tag{4.94}$$

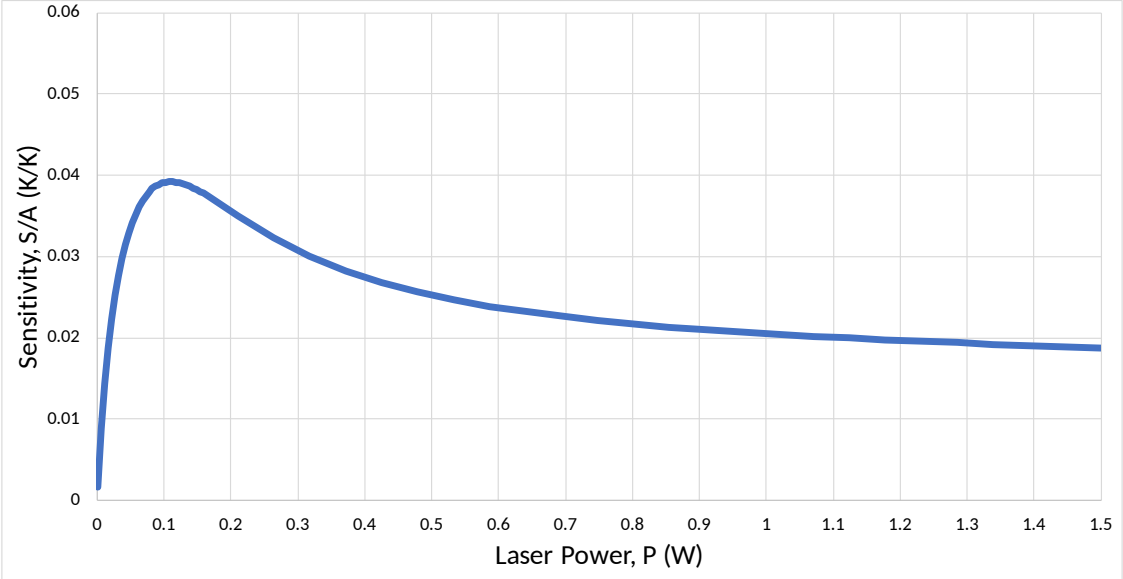
- The ratio of infrared radiation to laser contributions is:

$$\frac{RADIATION}{LASER} \approx (T_S^4 - T^4)\pi r^2\ \mathcal{O}(10^{-9-n}), \tag{4.95}$$

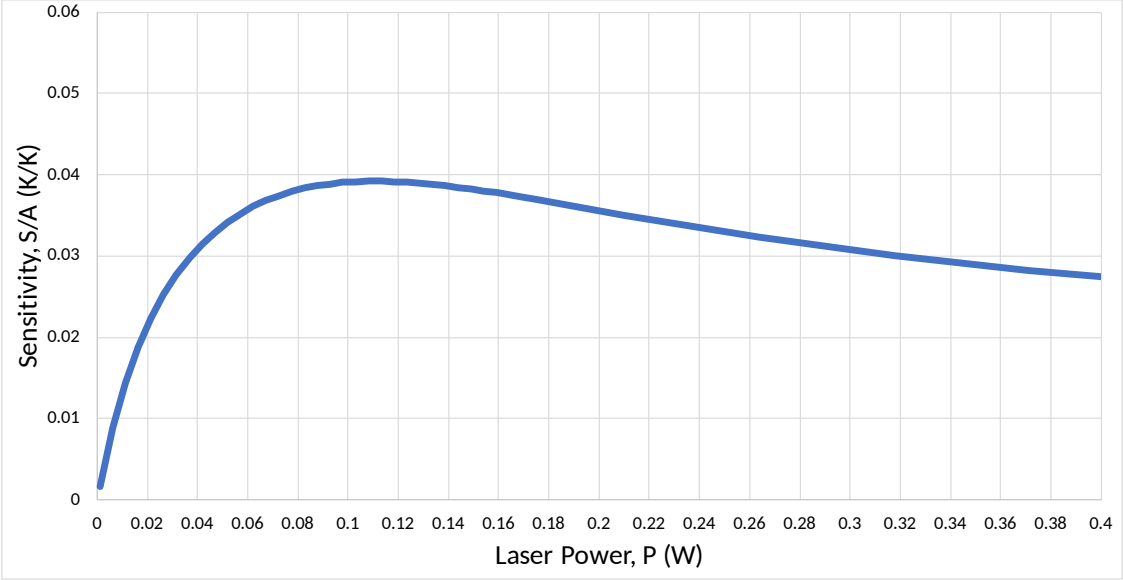
- The ratio of convective to laser contributions is:

$$\frac{CONVECTION}{LASER} \approx \mathcal{O}(10^{-1-n})(T_S - T)\pi r^2\ \mathcal{O}(10^{-1-n}). \tag{4.96}$$

Fundamentally, the trends are the same. However, these two cases deviate when we consider ablation, in the extension section below.



(a) Sensitivity of final temperatures due to laser irradiation perturbations.



(b) A more detailed view of the high-sensitivity area.

Figure 4.22: Spherical Domain: Sensitivity of final temperatures due to laser irradiation perturbations for a lumped mass with all thermal contributions.

Conclusions and further work

While there is certainly a great utility of models for design purposes and for ascertaining trends, there is, however, a limit. To reliably extend this work, more detailed optical and

Table 4.10: Spherical Domain: Thermal variation as a function of laser input power with perturbations. The peak normalized standard deviation occurs when some samples have reached their steady state temperature and others have not, at approximately 0.11 W.

$\bar{P}$, Avg. Input (W)	$\bar{T}$, Avg. Final Temp. (K)	ΔT, Std. Deviation (K)	$\frac{\Delta T}{\bar{T}}$
0.001	303.59	0.50	0.001658
0.04	650.46	20.46	0.031462
0.08	980.20	37.56	0.038326
0.12	1,279.68	49.98	0.039059
0.16	1,511.87	57.08	0.037760

thermal modeling (see, for example, Zohdi [159] and Gross [40] for overviews) should be pursued if it is desirable to understand the laser ray scattering properties of the lumped mass. Techniques that involve functionalized dyes benefit laser-based processes greatly as their enhanced properties can increase the thermal absorption into the material. Electric currents (electrophoresis) direct the application of the dyes in order to precisely target the deposition and interactions, allowing for the operator or researcher to tailor the process for each unique application.

Additionally, the material model can be further refined in order to calculate physical and thermodynamic properties with high fidelity. Using a differential scanning calorimetry curve (DSC) derived from experimental data for the material at hand, and coupled with a thermodynamic model that stochastically calculates the crystalline structure of the solid, this model would integrate the material properties based on the local temperature. This method would allow for the accurate definition of the relationship between the thermodynamic specific heat capacity and the material temperature.

Most critically, the spatial time step sizes for discretization of the thermal fields in a one-, two-, or three-dimensional domain are of interest, and require the use of techniques based on, for example, combinations of Finite Difference (FD), Finite Element (FE), and Discrete Element Methods (DEM). Furthermore, an aspect of key interest is the characterization of the movement of the domain and potentially hazardous particulate ejecta. Ganeriwala and Zohdi [37], and Zohdi [159, 163, 160, 161] present a numerical simulation and a comprehensive model of detailed, computationally-intensive, modeling of the dynamics and thermodynamics of these phenomena. There are many physical mechanisms that conduce to the creation of ejecta. Electronically induced molecular and macro-molecular vibrations are caused by photochemical and electronic ablation, hydrodynamical ablation pertains to the flowing of molten micrometer droplets, and laser energy liberates atomic bonds in photo-thermal ablation. An example of a purely mechanical process can include exfoliation in which material is removed in flakes by erosion. These mechanisms are not mutually exclusive.

4.4 UQ in the SLS Process

Let us now discuss the sources of variability in the SLS process and in the numerical model described in Chapter 3. This AM process is governed by six different areas of study that define the type of model that is to be used:

1. Energy consumption models estimate the amount of energy used for running the SLS printer as a function of the input process parameters [8]. This includes powering the laser, actuating the powder bed pistons and other components, and powering the resistive heaters and lamps. These energy models will inform the part designer about the print job orientation and the heating and cooling strategy for the powder bed.

2. Heat source models focus on the power delivery to the heat source and ultimately to the powdered material. These models inform the nature and beam shape of the laser to deliver the required energy to impart phase change, thus, the careful design of the heat source model will significantly impact the extent of material that does fuse together to build the part. Additionally, these models do have to consider the absorptivity of the material, a material property that is difficult to measure and accurately model.

3. Powder bed models are used to understand the highly-dynamic behavior of the particles as they absorb heat during processing. Models must make sure that the shape and size distribution of the particles approximates the actual powder and the material properties are modeled well across a temperature range, including phase changes. Also, the final packing of the particles is to be modeled via kinematic simulations, such as nonlinear Hertzian models [57], or random particle generation algorithms [143].

4. Melt pool models focus on the thermal or thermo-mechanical changes in the powder during heat absorption from the heat source. Phenomena such as phase transitions, material coalescence to form the melt pool, and material flow are captured byt his model, allowing for the evolution of the microstructure of the material. Models range in complexity from simple assumption-ridden FD models limited to resolving temperatures, to highly complex computational fluid dynamical (CFD) models [62] that capture violent mechanical phenomena such as the Marangoni surface tension effect. The computational resources to run these models is immense and so they require techniques to reduce the domain or impart simplifying assumptions.

5. Solidification models will look at the crystallization and grain growth of the microstucture of the material after the heat source is removed or moves sufficiently away from the material. These models will help to predict the thermo-mechancial properties of the bulk material for gauging the performance of the final printed part. These models will ultimately inform the cooling strategy post-print.

6. Residual stress models study the final state of mechanical properties of the part, governed by the final shape due to material deformation during cooling. Models that

capture the deformation of the part well are able to identify internal residual stresses and the behavior of the material while in operation. These behaviors may include fatigue life, stress concentrations, and gauging a good or bad part on the manufacturing floor.

More detailed information can be found in [50]. Identifying and quantifying all sources of variation within these six model families is the holy grail for UQ of AM processes. However, it is impractical to focus on building a model that is able to incorporate and handle accurate probabilistic distributions for all process inputs and propagating them through the solvers. Therefore, we must focus on a single type of model of the SLS process.

In this section, we will study the variation of the melt pool geometry from epistemic effects stemming from the random and unique arrangement of particles within a powder layer. Mathematically, epistemic uncertainty relates to effects or processes that behave based on an unknown distribution. For instance, based on Algorithm 1, the arrangement of particles is governed by a succession of events that sample a new particle's radius from a Gaussian distribution and places it depending on the arrangement of particles that have previously been placed. Thus, we must distill our focus to a more localized arrangement, to define it and attempt to identify a distribution of a heuristic.

Applying Random Perturbations

Physically, we notice that a variation of local particles within a small neighborhood will vary the local depth of the powder layer. This perspective prevents us from defining the powder depth in globally with a mean. Rather, we define this powder depth locally with a measure as follows. First, take any voxel at the top surface of the substrate and define its particle neighborhood as the collection of particles within a radial distance d away from the voxel. This distance d must be larger than the largest possible radius, $r_{max} = 40$ µm, thus, $d = 1.1r_{max}$. Then, the mean diameter of the particles in the neighborhood is computed, yielding the local powder layer depth at the voxel. In general, a collection of large particles will yield a locally deeper powder layer and, conversely, a collection of smaller particles will yield a locally shallower powder layer.

The concept of a local powder depth field will impact the laser energy depth multiplier term $e^{-\beta\hat{z}}$ in both Equations 3.36 and 3.46 to ensure that we account for all the laser heat locally. Although Gusarov's extinction coefficient β is not affected, since we choose to assume global material volume fraction and mean particle size measures, the local powder depth does add variation to the depth position variable $\hat{z}$. The distribution of the local powder depth across the domain is then propagated through the laser energy depth multiplier term $e^{-\beta\hat{z}}$, which then propagates its distribution through the laser term and the numerical time integration expressions in Equations 3.52 and 3.53 for the particles and the voxels, respectively. The numerics will simulate a final temperature distribution that highlights a range of possible temperatures for a particle or voxel, possibly straddling the boundary between solid and melted material. Thus, examining the hull of the melt pool, we can expect

geometric irregularities that differ from the idealized, symmetric shape approximations that are typical in literature [28, 78, 138].

In this UQ study we will focus on the temperature data that is output form the computational SLS model at the substrate's voxels, and we will treat the domain of the particles as extrinsic. This is simply due to the spatially ordered data, which makes the data useful to analyze. Thus, take the temperature contribution T_ℓ by the laser heat input term in the FD time stepping scheme for the voxels in Equation 3.53, in K:

$$T_\ell = \hat{\ell}\frac{\Delta t}{\rho C} = \alpha\ I_0\ e^{-2\frac{d^2}{\omega^2}}\ e^{-\beta\hat{z}}\ \frac{A_v}{V_v}\ \frac{\Delta t}{\rho C} = \gamma\ e^{-\beta\hat{z}}, \tag{4.97}$$

where γ is the scalar multiplier that gathers all terms that are not function of $\hat{z}$. Define now an upper and lower bound for the local powder depth as $\hat{z}^+$ and $\hat{z}^-$. The bounds for T_ℓ are governed by the bounds of $\hat{z}$ as

$$T_\ell^+ = \gamma\ e^{-\beta\hat{z}^-} \tag{4.98}$$

$$T_\ell^- = \gamma\ e^{-\beta\hat{z}^+}. \tag{4.99}$$

The inverse relationship between $\hat{z}$ and T_ℓ will be discussed later by Equation 4.100.

We are also interested in the output variation of the temperature contribution T_ℓ with respect variations of the local powder depth $\hat{z}$. Thus, we compute its partial derivative as

$$\frac{\partial T_\ell}{\partial\hat{z}} = \alpha\ I_0\ e^{-2\frac{d^2}{\omega^2}}\ \frac{A_v}{V_v}\ \frac{\Delta t}{\rho C}\ \frac{\partial\left(e^{-\beta\hat{z}}\right)}{\partial\hat{z}} = \gamma\ \frac{\partial\left(e^{-\beta\hat{z}}\right)}{\partial\hat{z}}$$

$$\Rightarrow\ \frac{\partial T_\ell}{\partial\hat{z}} = -\gamma\ \beta\ e^{-\beta\hat{z}}, \tag{4.100}$$

where V_v is the volume of a voxel and we have gained a $-\beta$ term. This inverse relation is intuitive: the deeper the local powder depth with an increase of $\hat{z}$, the smaller the contribution of the laser term on the temperature change during time integration. Equation 4.100 is useful since it relates the effect of a small perturbation of $\hat{z}$ with an output variation on T_ℓ. Thus, we can relate linear changes in $\hat{z}$ with changes in the temperature contribution term as

$$\Delta T_\ell = -\gamma\ \beta\ e^{-\beta\hat{z}}\Delta\hat{z}. \tag{4.101}$$

A discussion on the orders of magnitude of the terms in Equation 4.101 is in order. If a voxel is centered with this laser beam, then

$$\begin{aligned}
\gamma &\approx \mathscr{O}\left(10^{-1}\right)\ \mathscr{O}\left(10^{10}\right)\ \mathscr{O}\left(10^{5}\right)\ \mathscr{O}\left(10^{-13}\right) \approx \mathscr{O}\left(10^{1}\right)\\
\beta &\approx \mathscr{O}\left(10^{4}\right)\\
e^{-\beta\hat{z}} &\approx \mathscr{O}\left(-1\right)\\
\Rightarrow\ \frac{\partial T_\ell}{\partial\hat{z}} &\approx -\mathscr{O}\left(10^{4}\right).
\end{aligned} \tag{4.102}$$

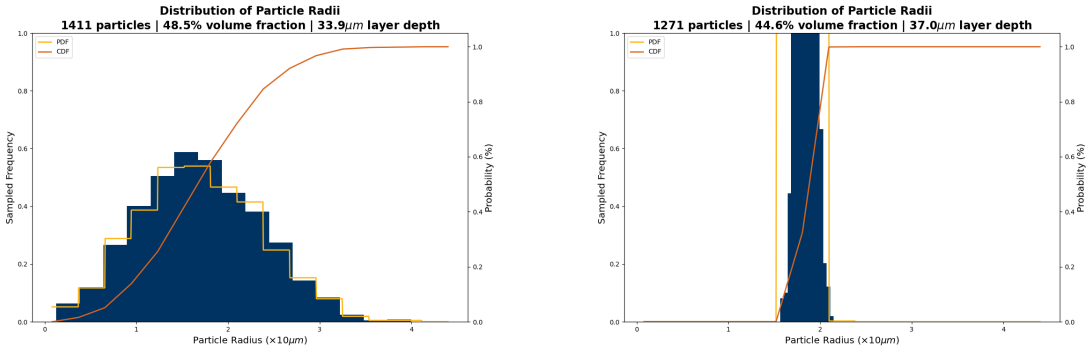
This result highlights the high inverse sensitivity of the temperature contribution with perturbations of the local powder depth. If we perturb the powder depth by about -10 µm, then we can expect an increase in temperature in the order of $\mathscr{O}(10^{-1})$ per time step from the laser term. Compounded with about 1,000 time steps, we would expect a variation in T_ℓ of 100 – 1,000 K throughout the simulation. This variation is significant enough to affect a region of the domain where the material may be a solid or it may be in its melted phase, which can cause variations in the melt pool hull geometry.

Table 4.11: Summary of experiments for quantifying the effects of local depth variations across the domain on the temperature contributions by the laser beam.

	Mean Size	**FWHM**	**Laser Power** (W)	**Laser Speed** (m/s)
Case 1	18.5	15.0	200	2.0
Case 2	18.5	2.0	200	2.0

To study this effect on the melt pool geometry, we can run two experiments that yield variation bounds for the local powder layer depth measure. One experiment is setup with a low spatial variation of $\hat{z}$ across the domain, and another example with high spatial variation. Table 4.11 summarizes the inputs for both cases. Note that they differ only in the FWHM range, which is the full-width at half maximum parameter of a normal distribution and is related to the standard deviation by a factor of about 2.3548. See equation 3.60 for more details. These examples yielded the particle size distributions in Figure 4.23. Both distributions are centered around the same mean size of 18.5 µm, but we can expect both larger and smaller particles in the domain of Case 1. This correlates directly with a wider range of local powder depths than in Case 2. Indeed, the availability of more small particles will skew the computation of the local powder depth towards the lower side of the range and, conversely, larger particles will effectively increase the local powder depth.

We now define a domain measuring 801×101 in the x and y directions, respectively. The z dimension is omitted. Figure 4.24 shows the arrangement of the local powder depth across the domain after computing the particle neighborhood of each voxel. Since this neighborhood is radial, we can see evidence of these circular regions in the local powder depth plots with color bars representing a shallow powder layer in light yellow and a deep one in black. The colors are reminiscent of light shining through a partially translucent material. The same color ranges have been used to show the local powder depth plots, and so we can identify visually that the local powder depths are relatively homogeneous across the domain, and that we can expect higher local depth variability for Case 1. The distribution of the local powder depths are further evidence of the spread. Case 2 has powder depths ranging in the 17 – 20 µm range with a mean at about the prescribed 18 µm, whereas the distribution in Case 1 varies between 10 – 25 µm, which reinforces the fact that the local depth bounds differ between both cases. Furthermore, the manner of rising and leveling-off of the CDF line is an indication that the distributions are normal.



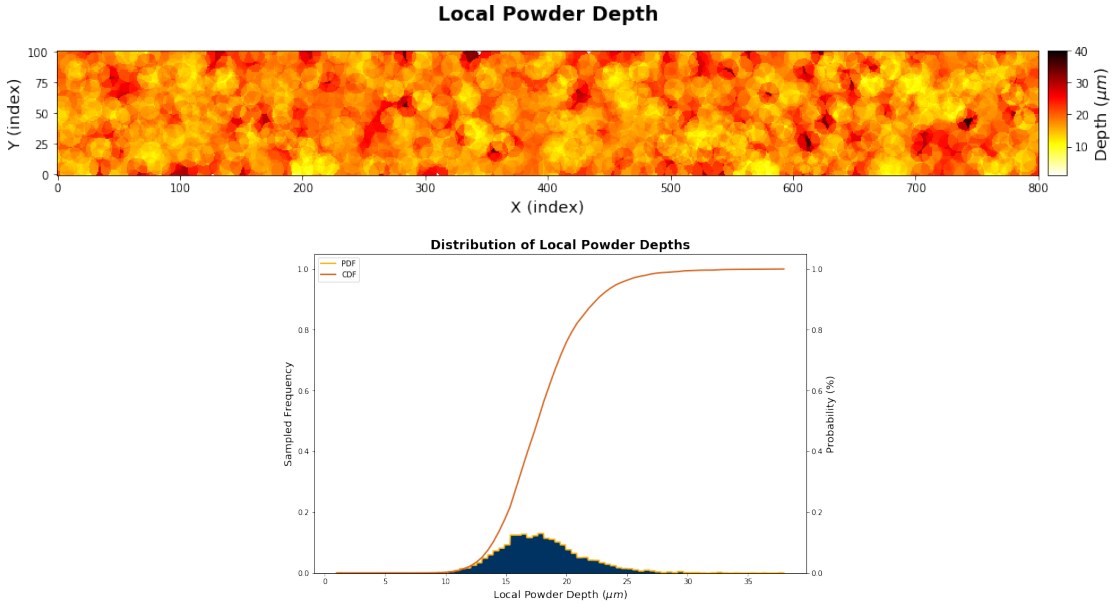
(a) Case 1 (b) Case 2

Figure 4.23: Sampled particle size distributions for the temperature field variations study. The cases are summarized in Table 4.11.

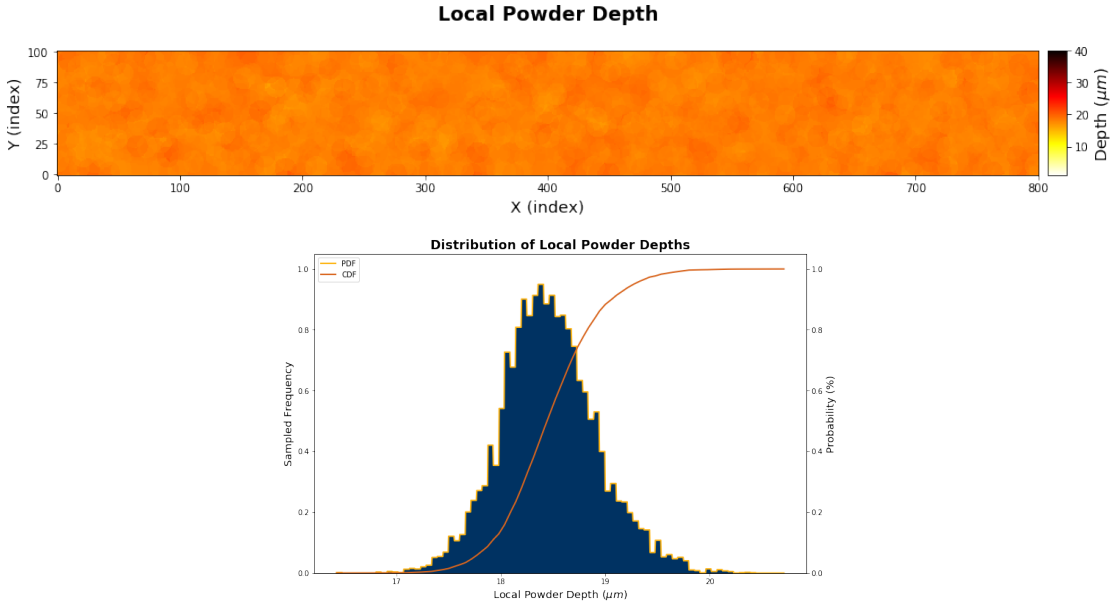
To study the variation on the melt pool width, Figure 4.25 shows voxel-wise maximum and minimum temperature plots across all time steps within the steady state portion of the SLS simulation, along with a temperature difference plot between the maximum and minimum plots. Only the top plane of the substrate is shown. The color bars for the maximum and minimum plots are identical, which demonstrate extreme sizes and shapes of the melt pool for the particular case parameters. Note that Case 1 displays a larger variability in the melt pool size since its maximum malt pool is larger than the maximum melt pool for Case2, but its minimum melt pool is smaller than the minimum melt pool of Case 2. The difference plot on the far right show that the temperature deviations occur along the laser path and with horizontal streaks. These streaks pertain to the effect of extremely sized particles grossly affecting the local powder depth and also conducting heat as they migrate to the left. In fact, the streaks only arise beyond the laser beam area. The difference plot in Figure 4.25b depicts the temperature differences at the laser beam and clear separation with the streak patterns from the particles. The difference plots are surrogates for areas of high temperature data variation, and we see high variation at the location of the laser beam spot and along the laser path.

The local powder depth range for Case 1 is about 15 µm and for Case 2 we have about 3 µm. Therefore, based on the order of magnitude discussion in Equation 4.102 we should expect between 100 – 1,000 K of temperature variation ΔT_ℓ. To correlate this expectation, the difference plots in Figure 4.25 show temperature ranges of less than 800 K for Case 1 and less than 300 K for Case 2, showing agreement with the calculated ΔT_ℓ values.

The upper and lower bounds for the melt pool width can be measured. Tracing both the maximum and minimum temperature data across a constant x value of about 15 – 30 µm (3 – 6 indices) behind the laser beam, which coincides with the peak temperature within the melt pool, we can identify variations on the width. Figure 4.26 shows three plots. The maximum, the minimum, and an time-averaged temperature . Also, we have a dashed horizontal line

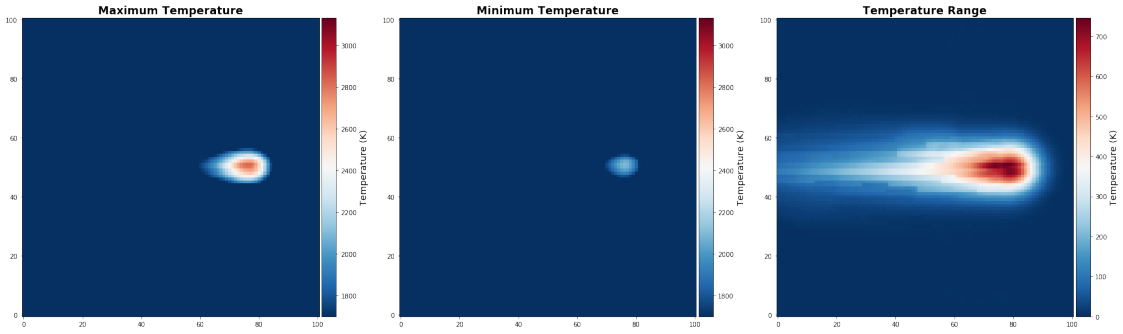


(a) Case 1

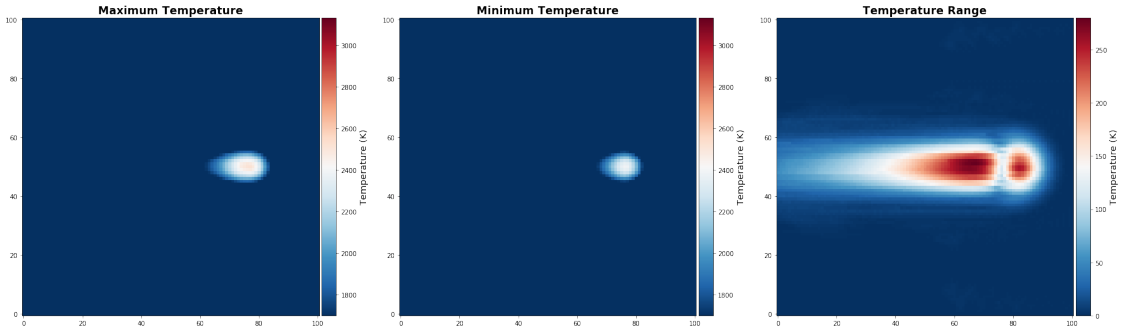


(b) Case 2

Figure 4.24: Local powder depths across a spatial domain measuring 801×101 in the x and y directions, respectively, and their corresponding voxel-wise distribution.



(a) Case 1



(b) Case 2

Figure 4.25: Voxel-wise maximum temperature and minimum temperature across time steps in the steady state stage. The top row is for Case 1 and the bottom for Case 2. The far-right plots show the regions of high temperature variation via the difference between the maximum temperature field and the minimum temperature field.

at the melting temperature , 1,700 K. Thus, the horizontal gap between the intersection of the melting line and each of the maximum and minimum lines yield a representation of the variation on the width due to the range throughout the local powder depth field. Due to a wide range of the local powder depth, Case 1 shows a large variation of the melt pool width, and Case 2 shows a narrow variation.

To discover the relationship between the spread of the particle sizes and the variation of the melt pool within a simulation, a series of 10 cases were ran across the FWHM range ranging from 2 to 20 µm, while holding the mean particle size constant at 18.5 µm. At each FWHM case, two samples were run, from which 2 width measurements can be taken, one for each side of the melt pool across the bisecting y-plane. Thus, the mean value and a maximum and minimum variation can be gathered in µm. Figure 4.27 shows the linear relationship between the FWHM range and the variation. The plot shows error bars, which represent the range of gaps between the maximum and minimum temperature arrays. When particle sizes are homogeneous, we see no variation of the width dimension, but as the particle sizes get

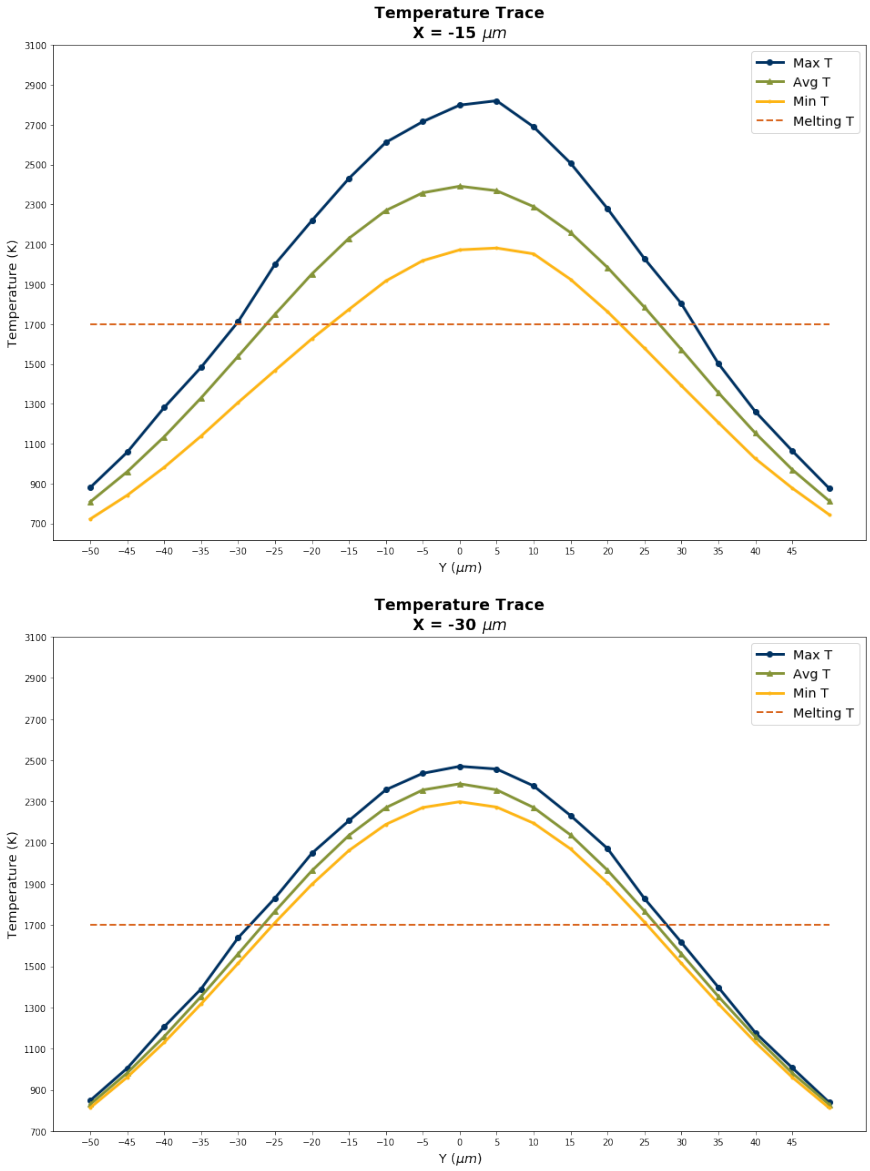


Figure 4.26: Traces of the maximum and minimum temperature along a line with a fixed x value, and through the peak temperature within the melt pool. The horizontal gap between the maximum and minimum lines represent actual variations in the melt pool width.

more diverse, then we do begin to encounter larger variations of the width and even wider fluctuations of the gap. The largest fluctuation occurred at a FWHM range of 16 µm, where we saw possible width variations of up to 50 µm with an average of 20 µ. This 30-µm swing is attributed to the localized powder depth contributions from both very small particles in the 1 to 10 µm range and from large particles in the 30 to 40 µm range.

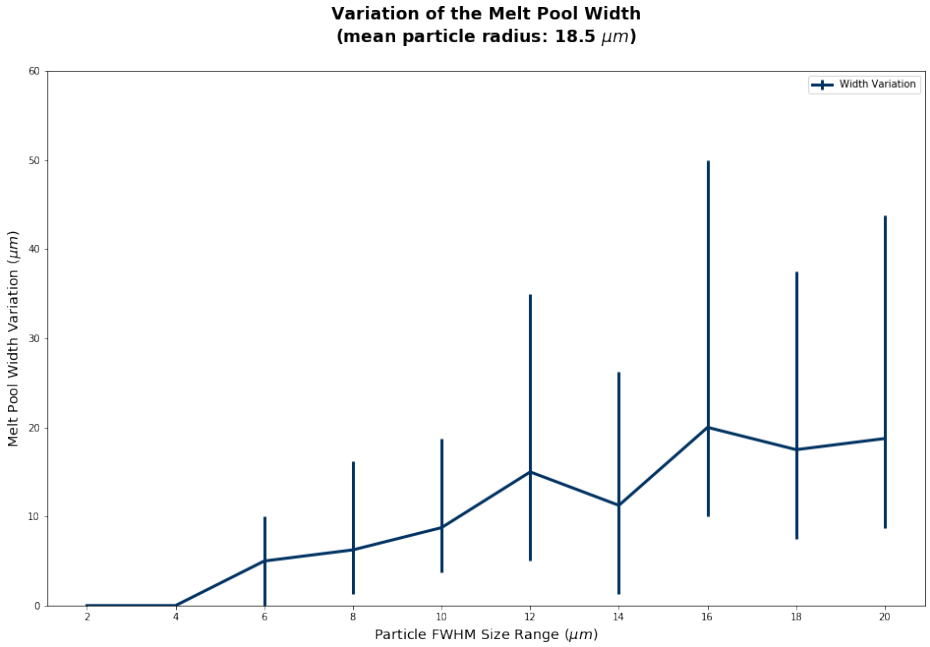


Figure 4.27: An idealization of an embedded particle in a powder pack.

In conclusion, the inclusion of larger and smaller particles in Case 1 yielded a wider range in the local powder depth $\hat{z}$, which propagated variation onto the temperature field via variations on the laser input heat term in the discretized differential equation. Finally, the variation of the voxel-wise temperature created a spatial variation in the width of the melt pool. A wide distribution of particle sizes will yield higher variation in the melt pool width. This is highlighted by the linear trend behavior between the powder particles size spread and the melt pool width variations in Figure 4.27. This result admonishes caution to the engineer or scientist upon purchasing raw powdered material with a wide tolerance in powder particles sizes.

Chapter 5

Reducing the Model

5.1 A Proper Orthogonal Decomposition

The study of fluid turbulence was the breeding ground for the proper orthogonal decomposition (POD) by Lumley in 1967 [82]. This work attempted to study the random vector field of turbulent flows by employing deterministic functions, which individually capture a portion of the fluctuating kinetic energy of the flow. The idea was to identify general behaviors that arise in these types of flows and to classify which behaviors dominate. These functions is what are called the POD modes.

In the case of fluid dynamics, the field data that is typically analyzed with a POD is the velocity vector $\boldsymbol{u}(\boldsymbol{x},t) = (u,v,w)$ of the flow. The notation for position in this chapter is $\boldsymbol{x} = (x,y,z)$ and for time t. The goal with POD is to separate the random velocity vector field into a spatial function $\boldsymbol{\Phi}(\boldsymbol{x})$ and a temporal function $\boldsymbol{V}(t)$ as:

$$\boldsymbol{u}(\boldsymbol{x},t) = \sum_{k=1}^{\infty} V_k(t)\Phi_k(\boldsymbol{x}), \tag{5.1}$$

where $V_k(t)$ are the temporal coefficients, or bases, and $\Phi_k(\boldsymbol{x})$ are the spatial POD modes. Treating the spatial part of the field as the POD modes is characteristic of a POD. A snapshot is the data of the field at a particular time. As the name suggests any POD is proper, or optimal, such that its modes are arranged with the mode that captures the most energy first and then in decreasing order. In other words, $\sum_{k=1}^{n} V_k(t)\Phi_k(\boldsymbol{x})$ will maximize the amount of energy for the first n modes. The POD is also orthonormal:

$$\int_{\boldsymbol{x}} \Phi_{k_1}(\boldsymbol{x})\Phi_{k_2}(\boldsymbol{x})\, d\boldsymbol{x} = \begin{cases} 1 & \text{if } k_1 = k_2, \\ 0 & \text{if } k_1 \neq k_2 \end{cases}, \tag{5.2}$$

which implies that each spatial mode $\Phi\boldsymbol{\Phi}_k(\boldsymbol{x})$ only depends on a single temporal coefficient $V_k(t)$, thus, multiplying Equation 5.1 with the spatial modes and integrating over space we

have

$$V_k(t) = \int_{\boldsymbol{x}} \boldsymbol{u}(\boldsymbol{x}, t)\Phi_k(\boldsymbol{x})\ d\boldsymbol{x}, \tag{5.3}$$

yielding a convenient result for computing the spatial coefficients by employing the temporal POD modes.

Section 5.2 will dive deeper on how the field data is pre-processed for performing a POD to compute the spatial modes and the random temporal coefficients.

Although the POD is widely used by engineers, this same procedure is refer to as a principal component analysis (PCA) by statisticians and the singular value decomposition (SVD) by mathematicians. Although PCA is an alternate name for POD, the SVD of a real matrix $\boldsymbol{U}$ with size $m \times n$ is

$$\boldsymbol{U} = \boldsymbol{L}\boldsymbol{\Sigma}\boldsymbol{R}^T, \tag{5.4}$$

where $\boldsymbol{L}$ is an $m \times m$ orthogonal matrix, $\boldsymbol{\Sigma}$ is an $m \times m$ diagonal matrix, and $\boldsymbol{R}$ an $n \times n$ orthogonal matrix. The non-zero diagonals of $\boldsymbol{\Sigma}$ are a set of $r = \min(m, n)$ positive scalars, which are the singular values of $\boldsymbol{U}$ in decreasing order. Note that the SVD procedure is able to diagonalize any rectangular matrix. The POD and SVD procedures are related since $\boldsymbol{R} = \boldsymbol{\Phi}$ and the eigenvalues of the data $\boldsymbol{U}$ are the square of the singular values. Making this connection allows us to note that the POD modes $\boldsymbol{\Phi}(\boldsymbol{x})$ are the right eigenvectors of the data matrix, are simply a new set of bases upon which we can project our data, and are optimal for capturing a maximal amount of energy at each eigenvector direction.

Utilizing the POD or the SVD method is equivalent in the end. Both procedures allow us to define an optimal set of bases $\boldsymbol{\Phi}$ that is conducive to building a reduced order model (ROM).

5.2 Applying a POD to the SLS Thermal Data

We now concentrate on applying a POD to the temperature scalar field data from running a single SLS model simulation run. We recall that the domain is divided into $801 \times 101 \times 61$ voxels, in the x, y, and z directions, respectively, but we have a simulation window that follows the position of the laser spot. The simulation window handles $101 \times 101 \times 61$ voxels at each time step. In our ROM discussion, we will only focus on handling the voxel thermal field and we ignore the particle thermal field, treating it as an extraneous effect on the substrate domain.

We call this scalar data $\breve{\boldsymbol{T}}(\boldsymbol{x}, t)$ and it is arranged in an array with shape $n_t \times n_x$, where $n_t = 2,101$ is the number of time steps and $n_x = 101 \times 101 \times 61 = 622,261$ is the total number of voxels within the simulation window. However, we first remove all voxels that represent Dirichlet boundaries to focus on data with degrees of freedom (dofs); thus, we remove all surfaces of the reduced domain except the top surface, since this one enjoys a Neumann flux-type boundary condition and we modify $n_x = 99 \times 99 \times 60 = 588,060$. Additionally,

the sPOD excels at finding fluctuations in the data. Since we have random effects in the local powder layer depth, as explained in Section 4.4, we want to capture any fluctuations of the temperature data around a mean. We then normalize the data at each time step with a time averaged temperature value $\bar{\boldsymbol{T}}(\boldsymbol{x})$. Thus, we have our pre-processed temperature data $\boldsymbol{T}(\boldsymbol{x}, t)$:

$$\boldsymbol{T}(\boldsymbol{x}, t) = \breve{\boldsymbol{T}} - \bar{\boldsymbol{T}} = \left[\left\{\breve{T}_0\right\} \quad \left\{\breve{T}_1\right\} \quad \cdots \quad \left\{\breve{T}_{n_x}\right\}\right]_{n_t \times n_x} - \left[\bar{T}_0 \quad \bar{T}_1 \quad \cdots \quad \bar{T}_{n_x}\right]_{1 \times n_x}, \tag{5.5}$$

where subscripts pertain to indices for a vectorized representation of the degrees of freedom and the quantities in curly brackets represent a column vector with the temperature of each degree of freedom across time. Note that in the normalization computation, each component of $\bar{\boldsymbol{T}}$ gets broadcasted to the entire column of $\breve{\boldsymbol{T}}$.

The temperature data $\boldsymbol{T}$ is a normalized thermal field in space and for an POD to give useful results we must make sure that the data is correlated across space at any arbitrary time step. If so, then we have a semblance of continuity across space even in the presence of random effects. We can verify this correlation by computing a square $n_x \times n_x$ covariance matrix $\boldsymbol{C}$ of the data $\boldsymbol{T}$:

$$\boldsymbol{C} = \frac{1}{n_t - 1} \boldsymbol{T}^T \boldsymbol{T}. \tag{5.6}$$

Conversely, we can also measure the temporal correlation of the data for an arbitrary voxel by computing the following covariance matrix:

$$\boldsymbol{C}_s = \frac{1}{n_t - 1} \boldsymbol{T} \boldsymbol{T}^T, \tag{5.7}$$

a $n_t \times n_t$ matrix. The subscript s pertains to the left SVD decomposition of the data $\boldsymbol{T}$, which is related to the snapshot POD (sPOD) procedure.

Remark. *We point out that using the left eigenvectors $\boldsymbol{L}$ of the factorization of the SVD method is called the snapshot POD (sPOD) method. In sPOD, we build temporal POD modes and random spatial coefficients. There is an advantage of the sPOD over the direct POD method: the covariance matrix in that case would be $n_t \times n_t$, a matrix with a number of components that is 6 orders of magnitude lower. This makes the sPOD method faster to compute than the POD method since $n_x > n_t$. The sPOD method was developed by Sirovich [131] in 1987 with the purpose of speeding up the direct POD computations. However, sPOD is not used in this case: we are interested in performing a time integration of the reduce model, based on our FOM formulation, which necessitates spatial POD modes to reconstruct the data or predict new data with the ROM.*

The diagonals of $\boldsymbol{C}$ (and similarly of $\boldsymbol{C}_s$) are the variances of each DOF and the off-diagonals are the covariances between pairs of DOFs. Non-zero values on the diagonals does provide a measure for the correlation between DOF variables and these will allow us to solve

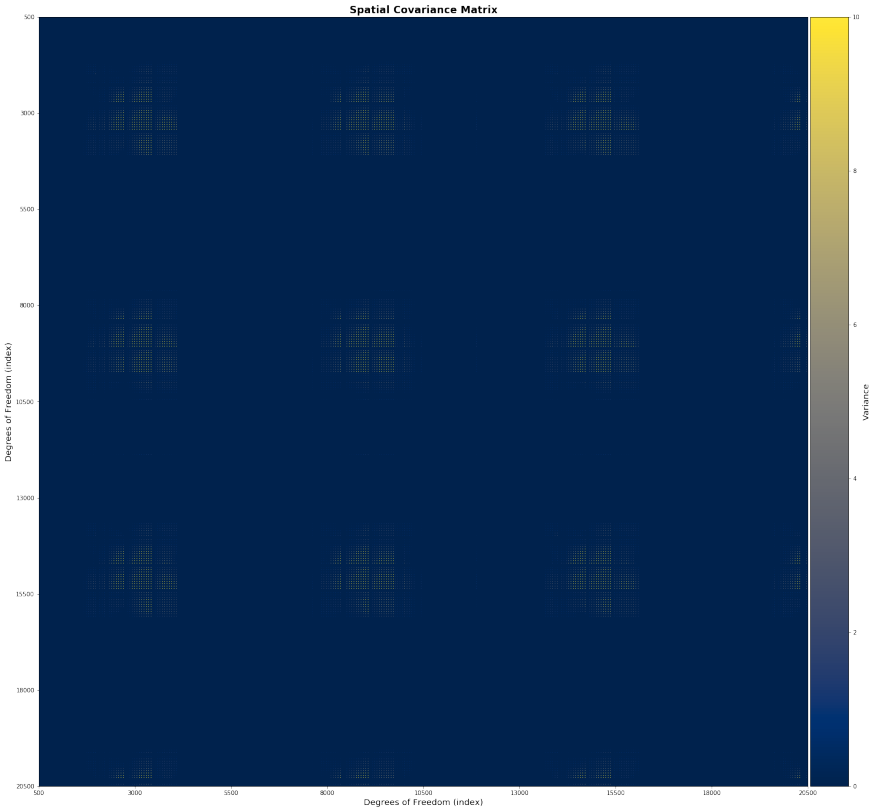


Figure 5.1: A visualization of the spatial covariance matrix $\boldsymbol{C}$. Only a $2,500 \times 2,500$ portion is shown as computing the full $n_x \times n_x$ matrix is computationally prohibitive.

the eigenvalue problem. In the case that we have zeros everywhere except for the diagonals, then the temperature data is uncorrelated and the POD cannot make new inferences when building the modes. In that case, the eigenvalues of $\boldsymbol{C}$ are its diagonal entries and so the modes $\boldsymbol{\Phi}$ project the data $\boldsymbol{T}$ back to itself.

Figure 5.1 is a visualization of a relatively small portion of the spatial covariance matrix $\boldsymbol{C}$ for the model validation run presented in Section 3.3. Due to the computationally massive computation of its $n_x^2 \approx 346$ billion elements, the temperature data $\boldsymbol{T}$ matrix can be truncated so that a number of contiguous columns are used for the computation of a portion of $\boldsymbol{C}$. In this case, $\boldsymbol{T}$ was truncated to a size of $n_t \times 2,500$, yielding a a $2,500 \times 2,500$ portion of $\boldsymbol{C}$. Exploring different regions of the full covariance matrix we do see a patter arising, which Figure 5.1 represents well. The geometric pattern is intuitive from Fourier's Law of heat conduction: due to the flattening strategy of snapshot temperature data into rows to form $\boldsymbol{T}$, a particular voxel will be correlated to other voxels with nearby indices in the z-direction, about $\pm n_z = 60$ indices away for its y-direction neighbors, and about $\pm n_y \times n_z = 5,940$ indices away for its x-direction neighbors. Note that the cluster of yellow-tinted squares repeat every 6,000 spatial indices, a pattern which repeats throughout the matrix. These

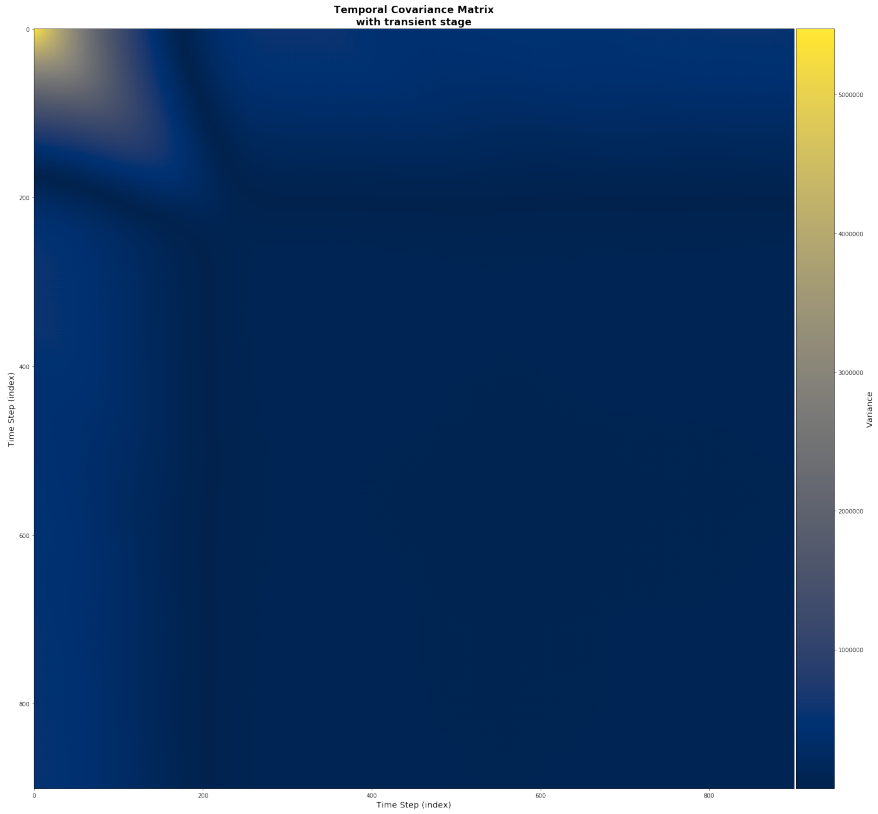
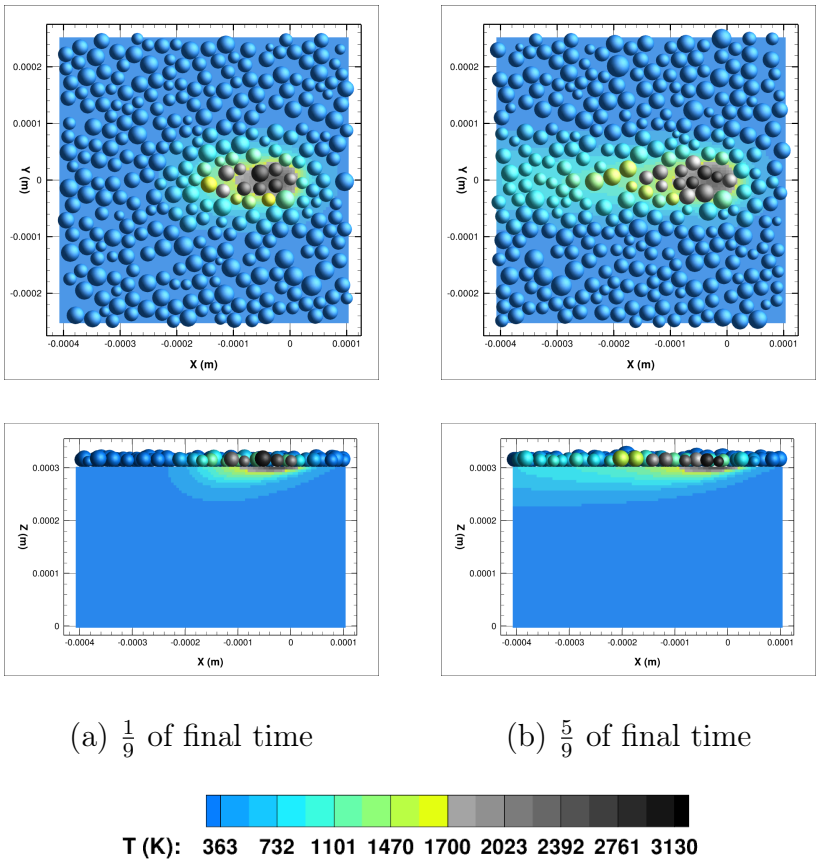


Figure 5.2: A visualization of the temporal covariance matrix $\boldsymbol{C}_s$, built with the transient and the steady state stages of the SLS simulation. Time step indices up to 901 are included.

non-zero regions pertain to the voxels that have incurred a temperature change away from initial conditions and the blue regions are for voxels far away from the laser path to not incur any temperature change.

Remark. *The temperature data set shape for a single simulation is* $n_t \times n_x = 353,424,060$. *This data takes about 15 GB of memory. A full* $n_x \times n_x$ $\boldsymbol{C}$ *matrix takes more than a TB of memory to store and to handle.*

Figure 5.2 shows a visualization of the temporal covariance matrix $\boldsymbol{C}_s$ for the same model validation run. We see that we have very high temporal correlation between the initial time steps and a much lower correlation between the rest of the time steps, with a short transition in between. Note that Figure 5.3 showcases a thermal field that is in its transient developing state as the laser begins to move away from the left wall ($x = 0$ plane). This transient state lasts for about $200 - 300$ time steps while the laser position moves to the right and its initial position exits the simulation window to the left. Afterwards, the simulation enters a pseudo steady state, where only the random effects provide the fluctuations between time steps. To prevent the POD procedure from confounding the transient effects with the random effects,



(a) $\frac{1}{9}$ of final time (b) $\frac{5}{9}$ of final time

Figure 5.3: Sample screenshots for a top and a sectioned side view of the temperature field at **(a)** 100 and **(b)** 500 time steps.

the data for the first 300 time steps is removed. Additionally, to minimize computational resources so that we can study the POD method on this temperature data set, we limit our focus on the first 901 time steps. Therefore, we modify the number of time steps to $n_t = 601$.

Figure 5.4 shows the temporal covariance matrix $\boldsymbol{C}_s$ for the "steady state" portion of the SLS thermal data, which reveals a pattern with a re-scaled color range, as compared to Figure 5.2. We gain confidence on the POD procedure the data is correlated in time for a given voxel location due to the non-zero values throughout the off-diagonal positions. The texture in the color visualization is due to the nature of the simulation window rolling forward by one voxel in the x-direction every 3 time steps. This creates a step change in the spatial indexing of the data leading to the periodic loss of correlation every 3 time steps indices when computing $\boldsymbol{C}_s$. Lastly, the image in Figure 5.4 might contain some aliasing due to the high resolution of the image, a phenomenon that exacerbates the appearance of the texture.

To find the POD modes, we perform an eigenvalue decomposition of $\boldsymbol{C}$, where the eigen-

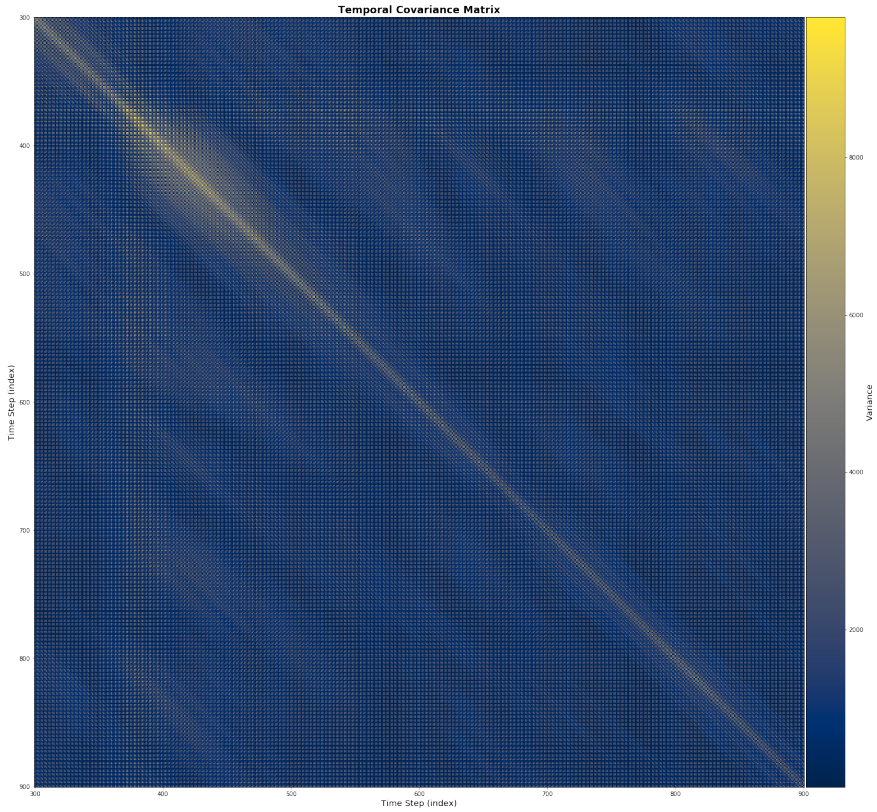


Figure 5.4: A visualization of the temporal covariance matrix $\boldsymbol{C}_s$, built with the steady state stage only of the SLS simulation. Note that time step indices 300 to 901 are included.

vectors are the POD spatial modes $\boldsymbol{\Phi}$. The columns of $\boldsymbol{\Phi}$ must be arranged based on the decreasing order of their associated eigenvalues. Thus, the first mode (first column of $\boldsymbol{\Phi}$) will explain the temperature data more strongly than the other modes, and each subsequent mode will have diminishing returns. Thus, we arrive at a decomposition of $\boldsymbol{T}$ in n_t-dimensional space, with n_t eigenvalues and eigenvectors. To obtain the temporal coefficients $\boldsymbol{V}$, we project the temperature data onto the modes by computing

$$\boldsymbol{V} = \boldsymbol{T}\boldsymbol{\Phi}. \tag{5.8}$$

The temporal coefficients matrix has n_t coefficients in its columns, and its shape is $n_t \times n_x$.

The temperature data can be reconstructed perfectly with the spatial modes and the temporal coefficients. Using Equation 5.8 above and noting that $\boldsymbol{\Phi}$ is a unitary matrix,

$$\begin{aligned} \boldsymbol{V} = \boldsymbol{T}\boldsymbol{\Phi} \;&\Rightarrow\; \boldsymbol{T} = \boldsymbol{V}\boldsymbol{\Phi}^{-1} = \boldsymbol{V}\boldsymbol{\Phi}^{T} \\ &\Rightarrow\; \boldsymbol{T}^{T} = \boldsymbol{\Phi}\boldsymbol{V}^{T} \end{aligned} \tag{5.9}$$

Algorithm 2 A POD Algorithm
Inputs: A square or rectangular data array `T`.
Output: A POD spatial modes matrix Φ, a temporal coefficients matrix `V`, and a diagonal matrix Λ with the eigenvalues.

1: Compute the covariance matrix `C`
2: Solve the eigenvalue problem for `T` to compute Φ and Λ
3: Sort the columns of Φ based on the decreasing order of Λ
4: Compute the temporal coefficients $\texttt{V} = \texttt{T}\Phi$
5: Optional: Reconstruct $\texttt{T} = \texttt{V}\Phi^{\texttt{T}}$
6: **return** Φ, `V`, Λ

Algorithm 3 A Singular Value Decomposition Algorithm for sPOD
Inputs: A square or rectangular data array `T`.
Output: A sPOD temporal modes matrix `V`, a spatial coefficients matrix Φ, and a diagonal matrix Λ with the eigenvalues.

1: Compute the singular value decomposition of `T` for the right-decomposition matrix `R` and the singular values Σ
2: $\Phi \leftarrow \texttt{R}$
3: Sort the columns of Φ based on the decreasing order of Σ
4: Compute the temporal coefficients $\texttt{V} = \texttt{T}\Phi$
5: Compute the diagonal matrix of eigenvalues $\Lambda = \texttt{diag}(\Sigma)\texttt{\^{}2}$
6: **return** Φ, `V`, Λ

The algorithm for decomposing the temperature data is shown in Algorithm 2, which includes the step for reconstructing the data. As mentioned before, we can also obtain the same decomposition of $\boldsymbol{T}$ with an SVD procedure. See Algorithm 3.

The POD spatial modes $\boldsymbol{\Phi}$ are difficult to reconcile with a physical representation. To do so, however, we must picture each mode as a basis of an n_x-dimensional space upon which we project the columns of the temperature data $\boldsymbol{T}$ in the form of n_x-dimensional vectors. In other words, $\boldsymbol{\Phi}^T$ maps $\boldsymbol{T}^T$ onto a different representation $\boldsymbol{V}^T$, a matrix with the same dimensions as the transposed version of original data $\boldsymbol{T}$. Thus, we have $\boldsymbol{T}^T \mapsto \boldsymbol{\Phi}^T\boldsymbol{T}^T$ where we define a mapping $\mathbb{R}^{n_x \times n_t} \to \mathbb{R}^{n_x \times n_t}$. This mapping preserves the dimensions of both representations.

A Local Reduced Model

We are now interested in the development of a data-driven model that produces reconstructed and predicted temperature data with a reduced number of DOFs. This "light-weight" model is a reduced order model (ROM), which achieves $\boldsymbol{T}$ data within an acceptable degree of error. ROMs are useful to speed up computations significantly for producing new data efficiently that approximates the true data without running the full order model (FOM).

The POD is effective in yielding a ROM since the modes are conveniently arranged in order of maximal energy. Thus, we may simply ignore a certain number of modes that may not be significant and reconstruct $\boldsymbol{T}$ with more significant nodes. Assuming a target error tolerance, we can successively reconstruct the temperature data by using an increasing number of modes until the error between the FOM data and the reconstruction data is low and within the acceptable threshold. Let's define the approximated temperature data $\tilde{\boldsymbol{T}}$ as a reconstruction of the data by using a reduced amount of modes. Explicitly,

$$\boldsymbol{T}^T \approx \tilde{\boldsymbol{T}}^T = \hat{\boldsymbol{\Phi}}\hat{\boldsymbol{V}}^T, \tag{5.10}$$

where $\hat{\boldsymbol{\Phi}}$ is a truncated POD spatial modes matrix with dimensions $n_x \times r$, $\hat{\boldsymbol{V}}^T$ is a truncated temporal coefficient matrix with dimensions $r \times n_t$, and $r < n_x$ is the minimum number of modes that must be employed for reconstructing the data within the error tolerance.

Noting that Equation 5.10 is a $\mathbb{R}^{n_t \times r} \to \mathbb{R}^{n_t \times n_x}$ mapping since we project the transpose of the truncated temporal coefficients $\hat{\boldsymbol{V}}^T$ onto a truncated number of modes $\hat{\boldsymbol{\Phi}}$ to get a transposed data approximation $\tilde{\boldsymbol{T}}^T$, we can say that $\hat{\boldsymbol{V}}$ is a reduced representation of $\boldsymbol{T}$. Now, we can define this reduced representation as $\hat{\boldsymbol{T}} = \hat{\boldsymbol{V}}$. For completion, Equation 5.10 becomes

$$\boldsymbol{T}^T \approx \tilde{\boldsymbol{T}}^T = \hat{\boldsymbol{\Phi}}\hat{\boldsymbol{T}}^T. \tag{5.11}$$

Thus, we have a reduced model where $\hat{\boldsymbol{T}}^T \mapsto \hat{\boldsymbol{\Phi}}\hat{\boldsymbol{T}}^T$.

In this work, the FOM is the finite difference numerical scheme for the substrate with an explicit Forward Euler time integrator as presented in Equation 3.53. We can simplify the equation as a system of linear equations,

$$\{T\}^{n+1} = [A]\{T\}^n + \{L\}^n, \tag{5.12}$$

where quantities in $\{\cdot\}$ represent column vectors and quantities in $[\cdot]$ represent matrices. The superscripts represent the time step index and L is the vector for the laser heat input, which changes at each time step due to local powder depth effects. Thus, the system dynamics matrix $\boldsymbol{A}$ includes the material properties and all coefficients of the numerical scheme, and the temperature data at the current time step $\boldsymbol{T}^n$ has been factored out. In this FOM, the temperature column vector has a length of $622,261$ components, and, thus, $\boldsymbol{A}$ contains $622,261 \times 622,261 = 387,208,752,121$ components. To handle the FOM in memory, sparse matrix computations had to be employed when implementing the algorithm since the finite difference scheme is sparse by definition.

Recall from Equation 5.11 that we can approximate $\boldsymbol{T}$ with a reduced set of basis and this mapping is substituted into the FOM in Equation 5.12, which yields

$$\begin{aligned} \left(\boldsymbol{T}^T\right)^{n+1} &= \boldsymbol{A}\left(\boldsymbol{T}^T\right)^n + \boldsymbol{L}^n \\ \hat{\boldsymbol{\Phi}}\left(\hat{\boldsymbol{T}}^T\right)^{n+1} &= \boldsymbol{A}\hat{\boldsymbol{\Phi}}\left(\hat{\boldsymbol{T}}^T\right)^n + \boldsymbol{L}^n \end{aligned} \tag{5.13}$$

Note that in this case $\boldsymbol{L}$ is the laser input power term, not the left factorization matrix for the SVD procedure. Now, since this is an overdetermined system of equations, we perform a Galerkin projection by pre-multiplying both sides of Equation 5.13 by $\hat{\boldsymbol{\Phi}}^T$ and then redefine reduced terms to formulate the time stepping scheme of ROM:

$$\begin{aligned}\hat{\boldsymbol{\Phi}}^T\hat{\boldsymbol{\Phi}}\left(\hat{\boldsymbol{T}}^T\right)^{n+1} &= \hat{\boldsymbol{\Phi}}^T\boldsymbol{A}\hat{\boldsymbol{\Phi}}\left(\hat{\boldsymbol{T}}^T\right)^{n} + \hat{\boldsymbol{\Phi}}^T\boldsymbol{L}^n \\ \left(\hat{\boldsymbol{T}}^T\right)^{n+1} &= \hat{\boldsymbol{A}}\left(\hat{\boldsymbol{T}}^T\right)^{n} + \hat{\boldsymbol{L}}^n,\end{aligned} \tag{5.14}$$

where $\hat{\boldsymbol{\Phi}}^T\hat{\boldsymbol{\Phi}}$ computes to the indentity matrix with shape $r \times r$ due to the orthonormality of the POD modes, $\hat{\boldsymbol{A}} = \hat{\boldsymbol{\Phi}}^T\boldsymbol{A}\hat{\boldsymbol{\Phi}}$ with shape $r \times r$, and $\hat{\boldsymbol{L}}^n = \hat{\boldsymbol{\Phi}}^T\boldsymbol{L}^n$ with shape $r \times 1$, which completes the Galerkin projection procedure. Equation 5.14 is now a system of r equations, not n_x equations, since we are solving for $\hat{\boldsymbol{T}}$ with shape $r \times n_t$. As a reminder, the superscript n is the time step index, and we have assumed a general formulation that treats the laser heat term as time-dependent.

Reconstruction of the Temperature Data

Alluding to the input parameters defined for numerical example in Section 3.3, the SLS thermal simulation is run once and the raw thermal field $\breve{\boldsymbol{T}}(\boldsymbol{x}, t)$ is recorded and pre-processed as explained in Section 5.2 to obtain temperature data $\boldsymbol{T}(\boldsymbol{x}, t)$ in an $n_t \times n_x$ array. This data is now primed for a POD procedure and then its reconstruction will be judged based on a maximum mean L_2 norm error, which will be introduced later.

Following Algorithm 2, we compute the covariance matrix $\boldsymbol{C}$, as shown in Figure 5.4, which pertains to a spatial domain within the simulation window and outside of the transient stage of the simulation. We see that, across space, the normalized thermal data correlates well spatially since there are non-zero values on the off-diagonals of the array. This result is clear evidence that the POD method will be able to extract useful results about the fluctuations of the data, and thus the variation in time of the melt pool shape.

Next, we solve the eigenvalue problem for $\boldsymbol{C}$ to obtain the POD spatial modes $\boldsymbol{\Phi}$ and their associated eigenvalues, which will be indicators for the degree of compression that we can achieve. Figure 5.5 is a plot of the eigenvalues in decreasing order. We notice that the POD method will potentially compress the temperature data due to the reverse hockey stick trend, yielding a high explanation of the data by the first spatial modes and an insignificant contribution by a majority of the modes.

The Reconstruction Error We must measure the error between the FOM data and the reconstructed data in order to gauge the effectiveness of the ROM. Since we are reconstructing data that was used to build the spatial modes and the temporal coefficients, then, we can define an absolute error metric for reconstruction:

$$e_R^n = \frac{1}{n_t}\left\|\tilde{\boldsymbol{T}}^n - \boldsymbol{T}^n\right\|_2, \tag{5.15}$$

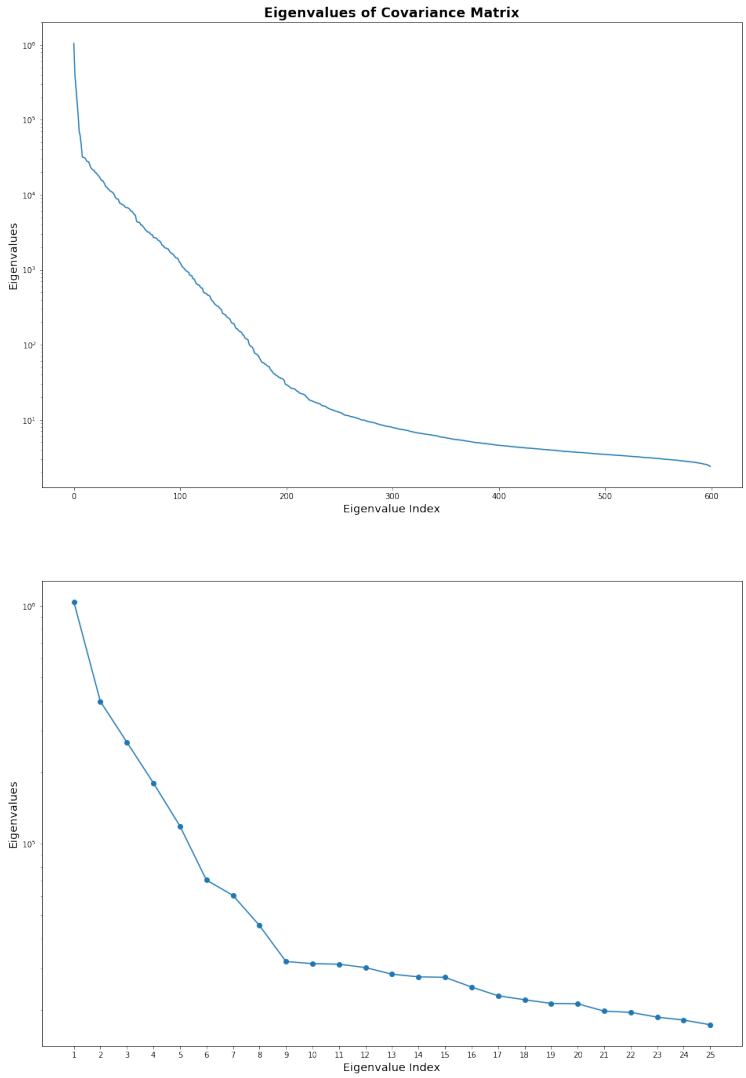


Figure 5.5: (Top) Sorted eigenvalues of $\boldsymbol{C}$ associated with the eigenvectors (spatial modes) in $\boldsymbol{\Phi}$. (Bottom) Same plot showing the region with the most significant eigenvalues.

where we take the mean L_2-norm of the absolute difference between the reconstructed and the FOM data at time step n. Then, we take the maximum time step error as the reconstruction error,

$$e_R = \max_{1 \leq n \leq n_t} e_R^n, \tag{5.16}$$

which yields a maximum mean L_2-norm error.

Determining a ROM Dimension An experiment must now be run to search for a ROM dimension r that yields a reconstruction error below an acceptable tolerance. Let us define

$\tilde{\boldsymbol{T}}^{(r)}$ as the reconstructed $n_t \times n_x$ normalized thermal data using the first r modes, so that the first r columns of $\boldsymbol{\Phi}$ and $\boldsymbol{V}$ are used in Equation 5.10. As such, we reconstruct the data $\tilde{\boldsymbol{T}}^{(r)}$ with $r = 1, 2, \ldots, n_t$, increasing r until the reconstruction error reaches below the predetermined error tolerance.

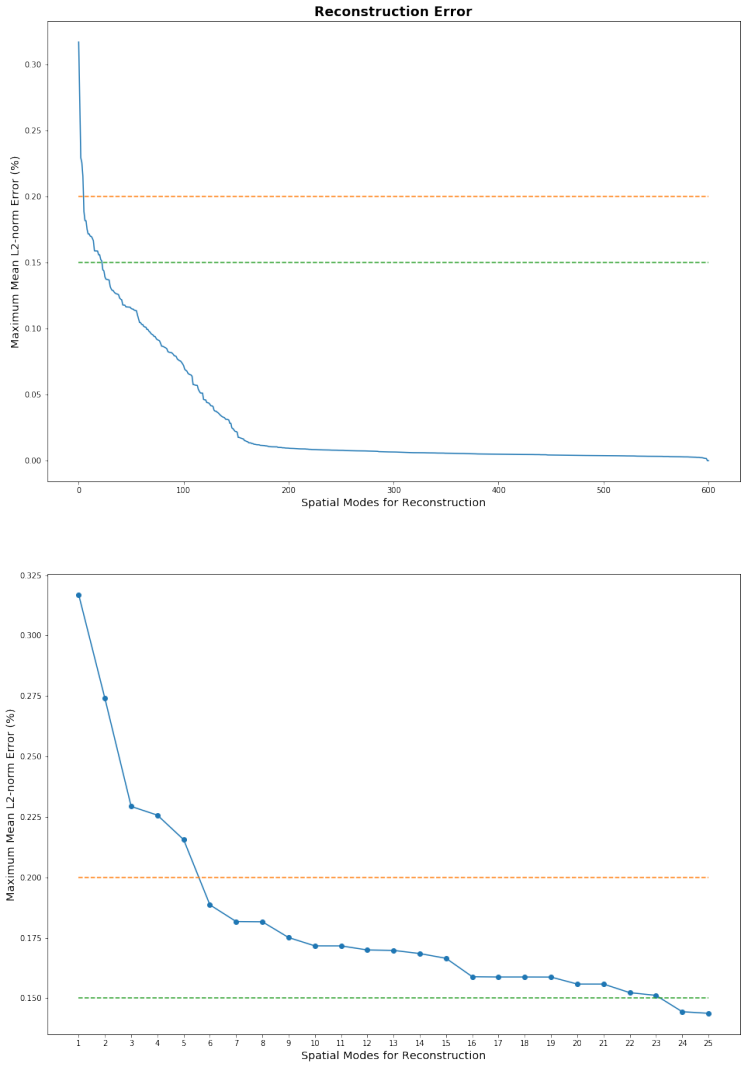


Figure 5.6: (Top) Reconstruction error e_R by incrementally employing spatial modes. (Bottom) Same plot showing the region with the most significant spatial modes. The dashed horizontal lines highlight the additional modes necessary to decrease e_R from 0.20% to 0.15%.

Figure 5.6 shows the evolution of e_R as the number of modes for reconstruction is increased; the top and bottom plots show the same error data trend with a modified horizontal axis range. The reconstruction error is significantly reduced by using the first 3 modes from around 0.32% to 0.23%, with a slightly reduced improvement when employing up to mode

6 (0.19%). Then, we see diminishing returns for e_R improvement as more modes are used for reconstruction beyond the 7^{th} mode. The bottom portion of Figure 5.6 showcases this behavior up to the 25^{th} mode. The dashed lines on the plots illustrate the trade-off between minimizing the reconstruction error tolerance but increasing the computational cost due to the handling of larger $\boldsymbol{\Phi}$ and $\boldsymbol{V}$ and a slightly higher error tolerance with faster computations. Thus, meeting an error tolerance of 0.20% (orange dashed line) will necessitate a ROM with 6 modes and a 0.15% tolerance (green dashed line) necessitates at least 24 modes.

Moving forward, let us assume a 0.20% error tolerance based on the SLS process capability and acceptable variability for the manufacturing application at hand. The process engineer or scientist will have to make a judgement to determine this error tolerance before carrying on with the ROM definition process. Thus, we define a reconstructed data with 6 modes as $\tilde{\boldsymbol{T}}^{(r=6)}$ using a POD spatial modes $\boldsymbol{\Phi}$ with shape $n_x \times 6$ and a reduced temperature data $\hat{\boldsymbol{T}}$ with shape $n_t \times 6$. As a comparison with the FOM temperature data $\boldsymbol{T}$ with a shape of $n_t \times n_x$, this amounts to a data compression factor of $\frac{n_x}{r} = \frac{588{,}060}{6} \approx 98,000$ with a loss of 0.19% of information. This compression factor amounts to the fact that the FOM temperature data is such that the POD is able to effectively capture temporal fluctuations of the data stemming from the random effects from the local powder depth metric.

Visualizing the Error To analyze the reconstruction error further, we visualize the spatial distribution of the reconstruction error across the SLS thermal simulation domain. Furthermore, to simplify notation, $\tilde{\boldsymbol{T}}^n$ will henceforth imply reconstruction with 6 modes at a time step with index n. An instantaneous absolute error $e_A^n = \tilde{\boldsymbol{T}}^n - \boldsymbol{T}^n$ allows us to compare the reconstructed $n_t \times n_x$ temperature data at a particular time step with the FOM temperature data with $r = 6$. Note that the absolute error is computed at each time step for the calculation e_R^n, before taking its mean L_2-norm. Also, due to its nature the absolute error has units of temperature ; in this case, Kelvin.

Figure 5.7 showcases the absolute error distribution at an arbitrary time step index $n = 400$ and across the plane (z-index plane = 60) of the SLS domain, which pertains to the top surface of the substrate which contacts the particles. Voxels with a blue color denote regions that the reconstruction procedure has approximated to be colder than the FOM, and, conversely, red areas denote voxels with a warmer reconstructed temperature data. We notice various structures of the error that are interesting. For one, the error of reconstruction happens across the path of the laser, which is where we see large temperature gradients and transient behaviors due to the random effects. Areas well away from the path of the laser beam spot show virtually no reconstruction error since these areas do not experience any temperature changes during the simulation, and so the POD procedure is able to compress this data successfully. Within the zone of the laser path we also see a spotted pattern overlaying a softer regions with errors. In general, reconstruction errors are proxies for areas with temporal fluctuations. In other words, the reconstruction of data with a truncated number of modes will shed less significant modes that explain different fluctuation behaviors. As such, areas with high fluctuations will locally require a higher

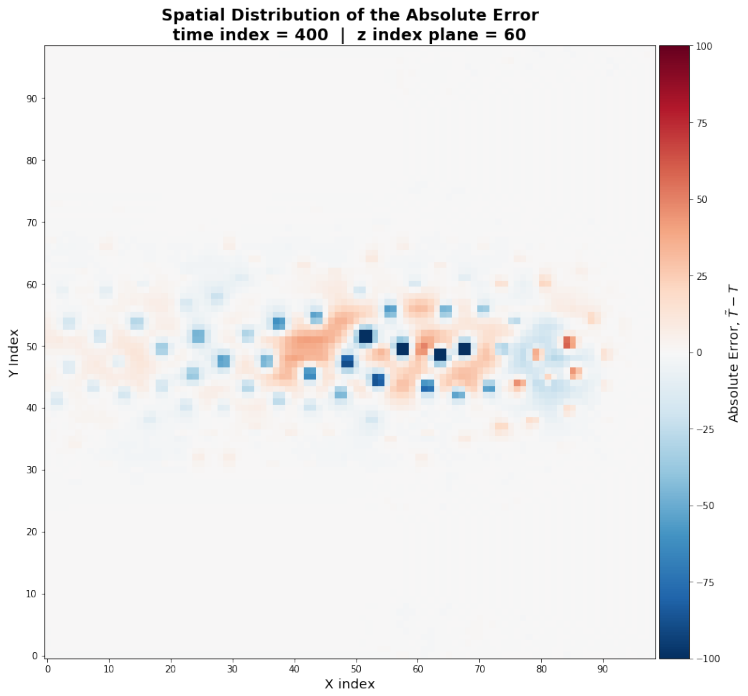
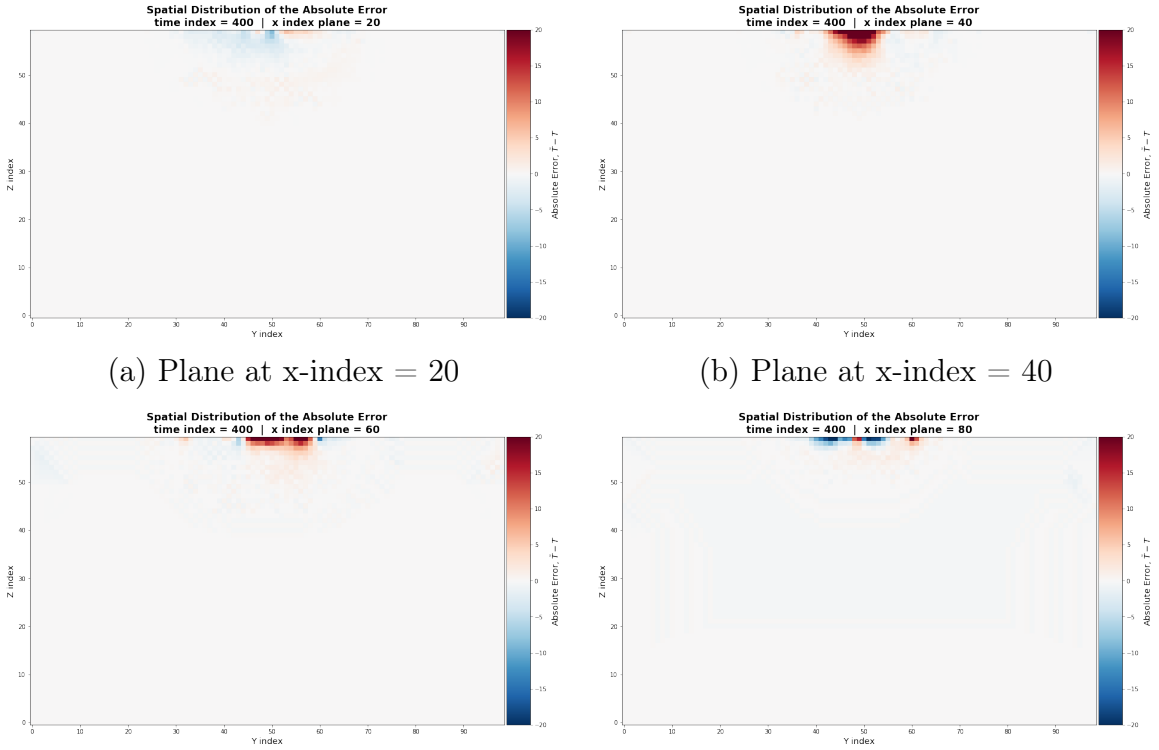


Figure 5.7: Absolute error e_A^n distribution at the plane where z-index = 60 of the SLS thermal simulation domain and at an arbitrary time step. Blue voxels pertain to an underestimation of the temperature and red areas pertain to an overestimation of the temperature .

amount of modes for meeting the error tolerance upon reconstruction as compared to areas with lower fluctuations in the data.

The dotted pattern is directly related to the thermal coupling between the voxel domain and the particle domain, which we decided to treat as extrinsic for building the ROM. This coupling provides a significant amount of conductive heat, which locally affects the voxels in contact with the particles. Additionally, since the simulation window follows the laser beam spot, then the particles travel across the domain and with them high thermal fluctuations with high thermal gradients. Thus, we would need a considerably higher amount of POD modes to explain these fluctuations. By truncating the modes from $n_x = 588,060$ down to $r = 6$, we remove the ability for the reconstruction procedure to approximate these critical regions with a low local error. Additionally, we see that the temperature data is grossly underestimated in dark blue for voxels in contact with particles that are behind the laser (x-index planes 80 and before) and overestimated in red for particles in front of the laser (x-index planes 82 and beyond). We learn from this pattern that particles that are cooling down after the laser has moved on will yield negative errors at their contacting voxels and the inverse is true for particles that are heating up in front of the laser. The stronger the rate of cooling or heating (closer to the laser spot) the darker the error hue. This behavior makes sense based on our previous assessment: higher changes between time steps is associated



(a) Plane at x-index = 20

(b) Plane at x-index = 40

(c) Plane at x-index = 60

(d) Plane at x-index = 80

Figure 5.8: Absolute error e_A^n distribution at various x-index planes of the SLS thermal simulation domain and at an arbitrary time step. Blue voxels pertain to an underestimation of the temperature and red areas pertain to an overestimation of the temperature data.

with a higher amount of modes needed for a reconstruction with lower errors.

The areas with errors dispersed in between the dotted pattern pertain to thermal changes due to the conduction of heat from the hotter voxels that are in contact with the particles and the surrounding voxels. We notice that a reconstruction with 6 modes effected significant errors in this area, but not as severe as the effects from the particle-voxel thermal coupling. The irregular pattern of the error does allude to the irregular shape of the thermal field, and thus the melt pool boundary, from the random effects by the local powder depth metric. These effects disrupt the symmetry in the thermal field data with respect to the bisecting y-plane.

The absolute errors are also visualized into the substrate domain at four different y-z planes in Figure 5.8, where the x indices equal 20, 40 ,60, and 80. Index 80 coincides with the plane at the laser center. Intuitively, the errors do arise within the laser track region, where voxels have incurred a temperature change. These errors pertain to red areas for regions where the substrate has received a relatively high amount of laser heat, mostly due to a shallow local powder depth. Thus, we see Figures 5.8b and 5.8c with deep red areas,

which are zones with a relative lack of particles in its vicinity. Conversely, Figures 5.8a and 5.8d do showcase the blue zones that are characteristic of voxels in contact with particles overhead. Lastly, this absolute error visualization lends itself to highlight the effects of heat conduction into the material. Figure 5.8a is a clear indication that the particle has heated up two voxels on the top surface with the deep blue color and this heat has diffused in a cone pattern with the fading blue hue.

In closing, the POD method is an effective method for compressing temperature data by projecting the data onto another set of orthonormal bases with directions that optimally explain a maximum amount of information. Thus, the least significant POD bases (modes) can be ignored for reconstructing the temperature data with slight errors. The success of the POD procedure to reconstruct 99.8% of the data with 6 modes is in part due to the setup of the substrate domain and the definition of the simulation window. This arrangement yielded a temperature fluctuation region that qualitatively resembles the velocity vector field of a fluid flowing around a cylinder, a problem that engendered the POD method.

Chapter 6

Conclusions

Additive Manufacturing (AM) processes have carved out their niches in industry, which include the need for part customization and complex geometries, high precision manufacturing for tooling and jigs, a heightened efficiency for industrial good production, or the need to print a wide array of materials. These relatively new capabilities in manufacturing allow for added freedom in the space of part design and function. The selective laser sintering/melting (SLS/SLM) process is a AM process that achieves production-ready parts since its high resolution yields detailed part features and precise surface finishes. This is due to the employment of a laser as the heat delivery system for a powder material. Through a series of mirrors and lenses actuated by servo motors, the laser is scanned over the top surface of the powder bed, melting a track of material that is less than 50 µm in width and melting material that is 30 – 150 µm deep. Thus, the laser must move in a back-and-forth pattern to stitch together these molted tracks to form a single layer of consolidated material. Then, more powder is applied on top of the layer and the laser repeats the procedure for a new layer. The printing of a full part is possible when multiple layers are build until the full volume is finished.

In this work, a single track of the laser is modeled and simulated using a coupled discrete element–finite difference (DE-FD) method. This method solves the thermal balance differential equation shown in Equation 3.24 and captures the evolution of the temperature across the particles that make up the powder layer and the substrate, which represents an underlying layer that has previously re-solidified after processing. The particles are modeled as lumped capacitance thermal masses and heat is allowed to be exchanged as discrete elements between particle-particle pairs, the environment inside the build chamber, and the substrate underneath. The substrate is broken up by a three-dimensional uniform mesh. At each mesh node, a cubic voxel is centered and it is able to exchange heat with neighboring voxels through Fourier's Law of thermal conduction. This exchange is discretized with a 7-point central difference stencil. To balance out the coupling between the particle and the substrate domain, the voxels also receive heat from the particles overhead through a prescribed heat flux term through the boundary that separated both domains. Temperatures are then integrated across time using an explicit forward Euler time-stepping scheme. In

the simulation, the laser moves at a prescribed speed, melting the powder and some of the underlying substrate material.

Temperature data for both material domains and all time steps is the extracted to measure the shape and size of the melt pool along the laser track. The melt pool depth, width, and length were measured for cases where the mean particle size μ and the full-width at half-maximum range of the distribution of particle sizes (FWHM) were varied. This facilitated a parameter study on the relative sensitivities of the melt pool size parameters due to these variations. The results indicated that the overall size of the melt pool is significantly dependant on μ and less so on the FWHM range. Furthermore, varying the volumetric heat delivered to the material via the laser power setting parameter yielded expected results: a higher power setting formed larger melt pools and a lower stetting formed smaller melt pools. In one case, the power was low enough that no material was melted.

Uncertainty of the Melt Pool Shape The variations in the melt pool size within a single laser track was also explored. These shape variations stemmed from a random scalar function applied at each voxel on the top surface of the substrate. This function calculated a measure of the local powder bed depth at each voxel based on the mean diameter of a neighborhood of particles within a certain radial distance. This heuristic accounted for the fact that a group of relatively small particles leads to a locally shallow powder layer and a group of larger particles amount to a locally deeper powder layer. This variation in depth across the substrate domain caused a variation in laser energy that is able to penetrate through the layer onto the top surface of the substrate. This phenomenon can lead to material at the boundary of the melt pool to either melt or remain solid, thus, causing a shape variation in the melt pool hull.

The variation of the melt pool shape was captured algorithmically by using a bounds approach. The maximum, average, and minimum temperature at each voxel for a full simulation is found and then the melt pool shape parameters for each set of data is extracted. The range of melt pool shapes can be quantified for a particular set of input parameters on the particle size distribution: μ and the FWHM range. A set of experiments varying these parameters showed that the mean particle size did not cause the melt pool shape to vary. On the contrary, the FWHM range of the particle size distribution did contribute to the variation of the melt pool shape.

Although promising results on the uncertainty quantification of the melt pool size were presented, these were confined to the depth, width, and length parameters. Other parameters that can be explored are the incident angle of the melt pool with the substrate top surface or the height of the melt pool [61, 62, 58]. However, this UQ study can be expanded to the three-dimensional melt pool hull, possibly defined by a triangular mesh or a similar representation to define the spatial variation at each node. This approach would produce variational results that are comprehensive across the melt pool geometry.

Reducing the SLS Thermal Model The temperature data from the SLS model was used for creating a reduced order model in the interest of achieving faster simulation runtimes with minimal losses. The approach that was chosen was the proper orthogonal decomposition (POD) procedure, a technique that takes a temperature data matrix that includes all degrees of freedom and all time steps. An eigendecomposition is performed on the covariance matrix of the temperature data to extract optimal modes and coefficients. These modes are a new set of bases to represent the same data and the coefficients modulate the contribution of each mode at each time step. Thus, the coefficients are an alternate representation of the temperature data. Choosing only the most significant modes (based on their respective sorted eigenvalues) that yield data reconstruction losses up to an application-dependent error level. This procedure is what reduced the order of the model. Indeed, the eigendecomposition produces $n_x = 588,060$ modes, but their contribution to the minimization of the reconstruction error decreases exponentially. This was the case with the temperature data from the SLS thermal simulation: it was found that using 6 modes achieved a reconstruction error below 0.20% and 24 modes below 0.15%, for a model compression surpassing 98,000 times and 25,000 times, respectively. In both cases, the time integration portion of the SLS simulation was reduced from about 130 s to fractions of a second.

Bibliography

[1] Moloud Abdar et al. "A review of uncertainty quantification in deep learning: Techniques, applications and challenges". In: *arXiv preprint arXiv:2011.06225* (2020).

[2] Dheeraj Agarwal et al. "Parametric design velocity computation for CAD-based design optimization using adjoint methods". In: *Engineering with Computers* 34.2 (2018), pp. 225–239.

[3] Grégoire Allaire, François Jouve, and Anca-Maria Toader. "Structural optimization using sensitivity analysis and a level-set method". In: *Journal of computational physics* 194.1 (2004), pp. 363–393.

[4] B. Avci and P. Wriggers. "A DEM-FEM coupling approach for the direct numerical simulation of 3D particulate flows". In: *Journal of Applied Mechanics* 79 (2012), 010901-(1–7).

[5] Devin Balkcom. "Robotic origami folding". PhD thesis. Citeseer, 2004.

[6] Mladen Banovic. "Efficient algorithmic differentiation of CAD frameworks". In: ().

[7] J-FM Barthelemy and Laura E Hall. "Automatic differentiation as a tool in engineering design". In: *Structural optimization* 9.2 (1995), pp. 76–82.

[8] Martin Baumers et al. "Sustainability of additive manufacturing: measuring the energy consumption of the laser sintering process". In: *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 225.12 (2011), pp. 2228–2239.

[9] Roland Becker and Rolf Rannacher. "An optimal control approach to a posteriori error estimation in finite element methods". In: *Acta numerica* 10 (2001), pp. 1–102.

[10] P. Bertrand et al. "Ceramic components manufacturing by selective laser sintering". In: *Applied Surface Science* 254.4 (2007), pp. 989–992.

[11] Arpan Biswas, Vadim Shapiro, and Igor Tsukanov. "Heterogeneous material modeling with distance fields". In: *Computer Aided Geometric Design* 21.3 (2004), pp. 215–242.

[12] Mathis Bode et al. "Using physics-informed super-resolution generative adversarial networks for subgrid modeling in turbulent reactive flows". In: *arXiv preprint arXiv:1911.11380* (2019).

[13] D. S. Bolintineanu et al. "Particle dynamics modeling methods for colloid suspensions". In: *Computational Particle Mechanics* 1.3 (2014), pp. 321–356.

[14] Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. "Discovering governing equations from data by sparse identification of nonlinear dynamical systems". In: *Proceedings of the national academy of sciences* 113.15 (2016), pp. 3932–3937.

[15] Kevin Bulthuis et al. "Uncertainty quantification of the multi-centennial response of the Antarctic ice sheet to climate change". In: *The Cryosphere* 13.4 (2019), pp. 1349–1380.

[16] J. M. Carbonell, E. Oñate, and B. Suárez. "Modeling of ground excavation with the particle finite element method". In: *Journal of Engineering Mechanics, ASCE* 136 (2010), pp. 455–463.

[17] Nicolás Castrillón, Avery Rock, and Tarek I Zohdi. "Thermal Modeling And Uncertainty Quantification of Tool For Automated Garment Assembly". In: (2021).

[18] Jiaqin Chen and Vadim Shapiro. "Optimization of continuous heterogeneous models". In: *Heterogeneous objects modelling and applications*. Springer, 2008, pp. 193–213.

[19] Jiaqin Chen et al. "Shape optimization with topological changes and parametric control". In: *International journal for numerical methods in engineering* 71.3 (2007), pp. 313–346.

[20] Wei Chen, Ruichen Jin, and Agus Sudjianto. "Analytical global sensitivity analysis and uncertainty propagation for robust design". In: *Journal of quality technology* 38.4 (2006), pp. 333–348.

[21] Kyung K Choi, JLT Santos, and Marjorie C Frederick. "Implementation of design sensitivity analysis with existing finite element codes". In: (1987).

[22] Sachin D Daxini and Jagdish M Prajapati. "Parametric shape optimization techniques based on Meshless methods: A review". In: *Structural and Multidisciplinary Optimization* 56.5 (2017), pp. 1197–1214.

[23] C. Deckard. *Method and apparatus for producing parts by selective sintering*. U.S. Patent 4,863,538. 1986.

[24] Yunliang Ding. "Shape optimization of structures: a literature survey". In: *Computers & Structures* 24.6 (1986), pp. 985–1004.

[25] Fabrizio Donati et al. "Non-invasive relative pressure estimation using 4D PC-MRI". PhD thesis. King's College London, 2016.

[26] Dorin Drignei et al. "Reallocation of testing resources in validating optimal designs using local domains". In: *Structural and Multidisciplinary Optimization* 50.5 (2014), pp. 825–838.

[27] Y Du, T Mukherjee, and T DebRoy. "Physics-informed machine learning and mechanistic modeling of additive manufacturing to reduce defects". In: *Applied Materials Today* 24 (2021), p. 101123.

[28] TW Eagar, NS Tsai, et al. "Temperature fields produced by traveling distributed heat sources". In: *Welding journal* 62.12 (1983), pp. 346–355.

[29] Ben Ezair, Daniel Dikovsky, and Gershon Elber. "Fabricating functionally graded material objects using trimmed trivariate volumetric representations". In: *Proceedings of SMI*. 2017.

[30] Patrick E Farrell et al. "Automated derivation of the adjoint of high-level transient finite element programs". In: *SIAM Journal on Scientific Computing* 35.4 (2013), pp. C369–C393.

[31] Scott Ferson and William L Oberkampf. "Validation of imprecise probability models". In: *International Journal of Reliability and Safety* 3.1-3 (2009), pp. 3–22.

[32] Scott Ferson, William L Oberkampf, and Lev Ginzburg. "Model validation and predictive capability for the thermal challenge problem". In: *Computer Methods in Applied Mechanics and Engineering* 197.29-32 (2008), pp. 2408–2430.

[33] Scott Fish et al. "Design and subsystem development of a high temperature selective laser sintering machine for enhanced process monitoring and control". In: *Additive Manufacturing* 5 (2015), pp. 60–67.

[34] Ronald Aylmer Fisher. "The design of experiments." In: *The design of experiments.* 1st Ed (1935).

[35] Imola K Fodor. *A survey of dimension reduction techniques*. Tech. rep. Lawrence Livermore National Lab., CA (US), 2002.

[36] Michael Freytag, Vadim Shapiro, and Igor Tsukanov. "Finite element analysis in situ". In: *Finite Elements in Analysis and Design* 47.9 (2011), pp. 957–972.

[37] Rishi Ganeriwala and Tarek I Zohdi. "A coupled discrete element-finite difference model of selective laser sintering". In: *Granular Matter* 18.2 (2016), pp. 1–15.

[38] Ian Gibson et al. *Additive manufacturing technologies*. Vol. 17. Springer, 2021.

[39] Anthony Giunta and Layne Watson. "A comparison of approximation modeling techniques-polynomial versus interpolating models". In: *7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*. 1998, p. 4758.

[40] H. Gross. *Handbook of optical systems. Fundamental of technical optics*. Wiley-VCH, 2005.

[41] AV Gusarov and J-P Kruth. "Modelling of radiation transfer in metallic powders at laser treatment". In: *International Journal of Heat and Mass Transfer* 48.16 (2005), pp. 3423–3434.

[42] AV Gusarov et al. "Model of radiation and heat transfer in laser-powder interaction zone at selective laser melting". In: *Journal of heat transfer* 131.7 (2009).

[43] Jan Hendrik Klaas Haertel. "Design of Thermal Systems Using Topology Optimization". In: (2018).

[44] Christian Hafner et al. "X-CAD: optimizing CAD models with extended finite elements". In: *ACM Transactions on Graphics (TOG)* 38.6 (2019), pp. 1–15.

[45] Raphael T Haftka and Ramana V Grandhi. "Structural shape optimization—a survey". In: *Computer methods in applied mechanics and engineering* 57.1 (1986), pp. 91–106.

[46] Robert Haimes and John Dannenhoffer. "The engineering sketch pad: A solid-modeling, feature-based, web-enabled system for building parametric geometry". In: *21st AIAA Computational Fluid Dynamics Conference*. 2013, p. 3073.

[47] Erik Holmberg, Bo Torstenfelt, and Anders Klarbring. "Stress constrained topology optimization". In: *Structural and Multidisciplinary Optimization* 48.1 (2013), pp. 33–47.

[48] R. Householder. *Molding Process*. U.S. Patent 4,247,508. 1979.

[49] Paul D Hovland et al. "Sensitivity analysis and design optimization through automatic differentiation". In: *Journal of Physics: Conference Series*. Vol. 16. 1. IOP Publishing. 2005, p. 466.

[50] Zhen Hu and Sankaran Mahadevan. "Uncertainty quantification and management in additive manufacturing: current status, needs, and opportunities". In: *The International Journal of Advanced Manufacturing Technology* 93.5 (2017), pp. 2855–2874.

[51] Changwu Huang, Bouchaïb Radi, and Abdelkhalak El Hami. "Uncertainty analysis of deep drawing using surrogate model based probabilistic method". In: *The International Journal of Advanced Manufacturing Technology* 86.9 (2016), pp. 3229–3240.

[52] Simeon Hubrich and Alexander Düster. "Numerical integration for nonlinear problems of the finite cell method using an adaptive scheme based on moment fitting". In: *Computers & Mathematics with Applications* 77.7 (2019), pp. 1983–1997.

[53] Simeon Hubrich et al. "Numerical integration of discontinuous functions: moment fitting and smart octree". In: *Computational Mechanics* 60.5 (2017), pp. 863–881.

[54] SS Isukapalli, A Roy, and PG Georgopoulos. "Stochastic response surface methods (SRSMs) for uncertainty propagation: application to environmental and biological systems". In: *Risk analysis* 18.3 (1998), pp. 351–363.

[55] Todd Robert Jackson. "Analysis of functionally graded material object representation methods". PhD thesis. Massachusetts Institute of Technology, 2000.

[56] Sokolowski Jan and JP Zolesio. *Introduction to shape optimization: shape sensitivity analysis*. 1992.

[57] Kenneth Langstreth Johnson and Kenneth Langstreth Johnson. *Contact mechanics*. Cambridge university press, 1987.

[58] Chandrika Kamath. "Data mining and statistical inference in selective laser melting". In: *The International Journal of Advanced Manufacturing Technology* 86.5 (2016), pp. 1659–1677.

[59] Anuj Karpatne et al. "Theory-guided data science: A new paradigm for scientific discovery from data". In: *IEEE Transactions on knowledge and data engineering* 29.10 (2017), pp. 2318–2331.

[60] Dmitriy Kats et al. "A physics-informed machine learning method for predicting grain structure characteristics in directed energy deposition". In: *Computational Materials Science* 202 (2022), p. 110958.

[61] Saad A Khairallah and Andy Anderson. "Mesoscopic simulation model of selective laser melting of stainless steel powder". In: *Journal of Materials Processing Technology* 214.11 (2014), pp. 2627–2636.

[62] Saad A Khairallah et al. "Laser powder-bed fusion additive manufacturing: Physics of complex melt flow and formation mechanisms of pores, spatter, and denudation zones". In: *Acta Materialia* 108 (2016), pp. 36–45.

[63] I. S. Kinstlinger et al. "Open-Source Selective Laser Sintering (OpenSLS) of nylon and biocompatible polycaprolactone". In: *PLoS ONE* 11.2 (2016), pp. 1–25.

[64] Vadim Komkov, Kyung K Choi, and Edward J Haug. *Design sensitivity analysis of structural systems*. Vol. 177. Academic press, 1986.

[65] Juho Könnö, Dominik Schötzau, and Rolf Stenberg. "Mixed finite element methods for problems with robin boundary conditions". In: *SIAM journal on numerical analysis* 49.1 (2011), pp. 285–308.

[66] XY Kou, Geoffrey T Parks, and Sooi Thor Tan. "Optimal design of functionally graded materials using a procedural model and particle swarm optimization". In: *Computer-Aided Design* 44.4 (2012), pp. 300–310.

[67] J. Kruth et al. "Consolidation of polymer powders by selective laser sintering". In: *International Conference on Polymers and Moulds Innovations* (2008).

[68] Goldy Kumar and Andrew Taber. "An integral representation of fields with applications to finite element analysis of spatially varying materials". In: *Computer-Aided Design* (2020), p. 102869.

[69] C. Labra and E. Oñate. "High-density sphere packing for discrete element method simulations". In: *Communications in Numerical Methods in Engineering* 25.7 (2009), pp. 837–849.

[70] Kookjin Lee and Kevin T Carlberg. "Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders". In: *Journal of Computational Physics* 404 (2020), p. 108973.

[71] A. Leonardi et al. "Coupled DEM-LBM method for the free-surface simulation of heterogeneous suspensions". In: *Computational Particle Mechanics* 1.1 (2014), pp. 3–13.

[72] Kai Li et al. "Bayesian uncertainty quantification and propagation for prediction of milling stability lobe". In: *Mechanical Systems and Signal Processing* 138 (2020), p. 106532.

[73] Xun Li et al. "Transition from factor-driven to innovation-driven urbanization in China: A study of manufacturing industry automation in Dongguan City". In: *China Economic Review* 59 (2020), p. 101382.

[74] Bin Liang and Sankaran Mahadevan. "Error and uncertainty quantification and sensitivity analysis in mechanics computational models". In: *International Journal for Uncertainty Quantification* 1.2 (2011).

[75] You Ling and Sankaran Mahadevan. "Quantitative model validation techniques: New insights". In: *Reliability Engineering & System Safety* 111 (2013), pp. 217–231.

[76] Chang Liu et al. "Additive manufacturing-oriented design of graded lattice structures through explicit topology optimization". In: *Journal of Applied Mechanics* 84.8 (2017).

[77] Rui Liu, Sen Liu, and Xiaoli Zhang. "A physics-informed machine learning model for porosity analysis in laser powder bed fusion additive manufacturing". In: *The International Journal of Advanced Manufacturing Technology* 113.7 (2021), pp. 1943–1958.

[78] Weiping Liu and JN DuPont. "Effects of melt-pool geometry on crystal growth and microstructure development in laser surface-melted superalloy single crystals: Mathematical modeling of single-crystal growth in a melt pool (part I)". In: *Acta materialia* 52.16 (2004), pp. 4833–4847.

[79] Felipe Lopez, Paul Witherell, and Brandon Lane. "Identifying uncertainty in laser powder bed fusion additive manufacturing models". In: *Journal of Mechanical Design* 138.11 (2016).

[80] Bill Lozanovski et al. "A Monte Carlo simulation-based approach to realistic modelling of additively manufactured lattice structures". In: *Additive Manufacturing* 32 (2020), p. 101092.

[81] Brian Luft, Vadim Shapiro, and Igor Tsukanov. "Geometrically adaptive numerical integration". In: *Proceedings of the 2008 ACM symposium on Solid and physical modeling*. ACM. 2008, pp. 147–157.

[82] John Leask Lumley. "The structure of inhomogeneous turbulent flows". In: *Atmospheric turbulence and radio wave propagation* (1967), pp. 166–178.

[83] Jeremy Maitin-Shepard et al. "Cloth grasp point detection based on multiple-view geometric cues with application to robotic towel folding". In: *2010 IEEE International Conference on Robotics and Automation*. IEEE. 2010, pp. 2308–2315.

[84] J. E. Mark. *Polymer Data Handbook*. Oxford University Press, 1999.

[85] Fady Massarwi and Gershon Elber. "A B-spline based framework for volumetric object modeling". In: *Computer-Aided Design* 78 (2016), pp. 36–47.

[86] C McCallen. "ALE3D: arbitrary lagrange eulerian three-and two dimensional modeling and simulation capability". In: *Lawrence Livemore National Laboratory, Livermore, CA* (2012).

[87] C. Millot et al. "Assessment of polyamide-6 crystallinity by DSC: Temperature dependence of the melting enthalpy". In: *Journal of Thermal Analysis and Calorimetry* 122.1 (2015), pp. 307–314.

[88] Amir M Mirzendehdel and Krishnan Suresh. "A Pareto-optimal approach to multimaterial topology optimization". In: *Journal of Mechanical Design* 137.10 (2015).

[89] Arvind T Mohan and Datta V Gaitonde. "A deep learning based approach to reduced order modeling for turbulent flow control using LSTM neural networks". In: *arXiv preprint arXiv:1804.09269* (2018).

[90] Colt Montgomery et al. "Process mapping of Inconel 625 in laser powder bed additive manufacturing". In: *2015 International Solid Freeform Fabrication Symposium*. University of Texas at Austin. 2015.

[91] Daniel Moser et al. "Multi-layer computational modeling of selective laser sintering processes". In: *ASME International Mechanical Engineering Congress and Exposition*. Vol. 46438. American Society of Mechanical Engineers. 2014, V02AT02A008.

[92] Alkhair Almabrouk Mousa. "Experimental investigations of curling phenomenon in selective laser sintering process". In: *Rapid Prototyping Journal* 22.2 (2016), pp. 405–415.

[93] Orest Mykhaskiv et al. "NURBS-based and parametric-based shape optimization with differentiated CAD kernel". In: *Computer-Aided Design and Applications* 15.6 (2018), pp. 916–926.

[94] Saideep Nannapaneni and Sankaran Mahadevan. "Uncertainty quantification in performance evaluation of manufacturing processes". In: *2014 IEEE International Conference on Big Data (Big Data)*. IlEEE. 2014, pp. 996–1005.

[95] Rajkishore Nayak and Rajiv Padhye. "Introduction to automation in garment manufacturing". In: *Automation in garment manufacturing*. Elsevier, 2018, pp. 1–27.

[96] Anthony O'Hagan. "Fractional Bayes factors for model comparison". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 57.1 (1995), pp. 99–118.

[97] Assad A Oberai, Nachiket H Gokhale, and Gonzalo R Feijóo. "Solution of inverse problems in elasticity imaging using the adjoint method". In: *Inverse problems* 19.2 (2003), p. 297.

[98] T. Okada, R. Ishige, and S. Ando. "Analysis of Thermal Radiation Properties of Polyimide and Polymeric Materials Based on ATR-IR spectroscopy". In: *Journal of Photopolymer Science and Technology* 29.2 (2016), pp. 251–254.

[99] E. Oñate et al. "Advances in the particle finite element method for the analysis of fluid multibody interaction and bed erosion in free surface flows". In: *Computer Methods in Applied Mechanics and Engineering* 197.19–20 (2008), pp. 1777–1800.

[100] E. Oñate et al. "Lagrangian analysis of multiscale particulate flows with the particle finite element method". In: *Computational Particle Mechanics* 1.1 (2011), pp. 85–102.

[101] E. Oñate et al. "Possibilities of the Particle Finite Element Method for fluid-soil-structure interaction problems". In: *Computational Mechanics* 48 (2011), pp. 307–318.

[102] Stanley Osher and Ronald P Fedkiw. "Level set methods: an overview and some recent results". In: *Journal of Computational physics* 169.2 (2001), pp. 463–502.

[103] Ikuo Ozaki and Takao Terano. "Applying an automatic differentiation technique to sensitivity analysis in design optimization problems". In: *Finite elements in analysis and design* 14.2-3 (1993), pp. 143–151.

[104] Xiaoping Qian and Debasish Dutta. "Physics based B-Spline heterogeneous object modeling". In: *ASME Design Engineering Technical Conference*. 2001.

[105] Paul Raccuglia et al. "Machine-learning-assisted materials discovery using failed experiments". In: *Nature* 533.7601 (2016), pp. 73–76.

[106] Ramesh Rebba and Sankaran Mahadevan. "Computational methods for model reliability assessment". In: *Reliability Engineering & System Safety* 93.8 (2008), pp. 1197–1207.

[107] Markus Reichstein et al. "Deep learning and process understanding for data-driven Earth system science". In: *Nature* 566.7743 (2019), pp. 195–204.

[108] Trevor T Robinson et al. "Optimizing parameterized CAD geometries using sensitivities based on adjoint functions". In: *Computer-Aided Design and Applications* 9.3 (2012), pp. 253–268.

[109] J. Rojek. "Discrete element thermomechanical modeling of rock cutting with valuation of tool wear". In: *Computational Particle Mechanics,* 1.1 (2014), pp. 71–84.

[110] Jerzy Rojek et al. "Comparative study of different discrete element models and evaluation of equivalent micromechanical parameters". In: *International Journal of Solids and Structures* 49.13 (2012), pp. 1497–1517.

[111] Marleen Rombouts et al. "Photopyroelectric measurement of thermal conductivity of metallic powders". In: *Journal of applied physics* 97.2 (2005), p. 024905.

[112] George Rozvany. "The SIMP method in topology optimization-theoretical background, advantages and new applications". In: *8th Symposium on Multidisciplinary Analysis and Optimization*. 2000, p. 4738.

[113] Samuel H Rudy, J Nathan Kutz, and Steven L Brunton. "Deep learning of dynamics and signal-noise decomposition with time-stepping constraints". In: *Journal of Computational Physics* 396 (2019), pp. 483–506.

[114] Marc Russell. "A Smoothed Particle Hydrodynamics Model for the Simulation of Laser Fusion Additive Manufacturing Processes". PhD thesis. University of California, Berkeley, 2018.

[115] Vladimir L Rvachev et al. "On completeness of RFM solution structures". In: *Computational Mechanics* 25.2-3 (2000), pp. 305–317.

[116] Jerome Sacks et al. "Design and analysis of computer experiments". In: *Statistical science* (1989), pp. 409–423.

[117] M. Salloum et al. "Adaptive wavelet compression of large additive manufacturing experimental and simulation datasets". In: *Computational Mechanics* (2018).

[118] Andrea Saltelli et al. "Why so many published sensitivity analyses are false: A systematic review of sensitivity analysis practices". In: *Environmental modelling & software* 114 (2019), pp. 29–39.

[119] Ismael Sanchez Torreguitart, Tom Verstraete, and Lasse Mueller. "Optimization of the LS89 axial turbine profile using a cad and adjoint based approach". In: *International Journal of Turbomachinery, Propulsion and Power* 3.3 (2018), p. 20.

[120] Robert G Sargent. "Verification and validation of simulation models". In: *Proceedings of the 2010 winter simulation conference*. IEEE. 2010, pp. 166–183.

[121] E. Sartori. "Convection coefficient equations for forced air flow over flat surfaces". In: *Solar Energy* 80 (2006), pp. 1063–1071.

[122] Mark J Schervish. "P values: what they are and what they are not". In: *The American Statistician* 50.3 (1996), pp. 203–206.

[123] Gabriel R Schleder et al. "From DFT to machine learning: recent approaches to materials science–a review". In: *Journal of Physics: Materials* 2.3 (2019), p. 032001.

[124] Stephanie Schrage and Kristin Huber. "15 Living Wages in International Supply Chains and the Capability Approach". In: *New frontiers of the capability approach* (2018), p. 351.

[125] Kristof Schütt et al. "Schnet: A continuous-filter convolutional neural network for modeling quantum interactions". In: *Advances in neural information processing systems* 30 (2017).

[126] Ney R Secco et al. "Efficient mesh generation and deformation for aerodynamic shape optimization". In: *AIAA Journal* (2020), p. 15.

[127] Carolyn Conner Seepersad et al. "Multifunctional topology design of cellular material structures". In: *Journal of Mechanical Design* 130.3 (2008).

[128] GK Sharma and B Gurumoorthy. "A hybrid approach to define and represent material distribution in heterogeneous objects". In: *Computer-Aided Design and Applications* 14.1 (2017), pp. 70–82.

[129] Takayuki Shiraiwa, Fabien Briffod, and Manabu Enoki. "Uncertainty quantification of fatigue life prediction in welded structures using microstructure-based simulations". In: *International Conference on New Trends in Fatigue and Fracture*. Springer. 2017, pp. 329–334.

[130] Ole Sigmund and Kurt Maute. "Topology optimization approaches". In: *Structural and Multidisciplinary Optimization* 48.6 (2013), pp. 1031–1055.

[131] Lawrence Sirovich. "Turbulence and the dynamics of coherent structures. I. Coherent structures". In: *Quarterly of applied mathematics* 45.3 (1987), pp. 561–571.

[132] J. Smith et al. "Linking process, structure, property, and performance for metal-based additive manufacturing: computational approaches with experimental support". In: *Computational Mechanics* 57.4 (2016), pp. 583–610.

[133] Christian Soize. "Stochastic models of uncertainties in computational mechanics". In: American Society of Civil Engineers. 2012.

[134] John C Strikwerda. *Finite difference schemes and partial differential equations*. SIAM, 2004.

[135] Shoujin Sun, Milan Brandt, and MJLAM Easton. "Powder bed fusion processes: An overview". In: *Laser additive manufacturing* (2017), pp. 55–77.

[136] Andrew Taber, Nicolas Castrillon, and Goldy Kumar. "Sensitivity Computation of Integral Quantities with Moment Derivatives". 2020.

[137] Andrew Taber et al. "A moment-vector approach to interoperable analysis". In: *Computer-Aided Design* 102 (2018), pp. 139–147.

[138] Ming Tang, P Chris Pistorius, and Jack L Beuth. "Prediction of lack-of-fusion porosity for powder bed fusion". In: *Additive Manufacturing* 14 (2017), pp. 39–48.

[139] Gustavo Tapia et al. "Uncertainty propagation analysis of computational models in laser powder bed fusion additive manufacturing using polynomial chaos expansions". In: *Journal of Manufacturing Science and Engineering* 140.12 (2018).

[140] Vaidyanathan Thiagarajan and Vadim Shapiro. "Shape aware quadratures". In: *Journal of Computational Physics* 374 (2018), pp. 1239–1260.

[141] Daniel A Tortorelli and Panagiotis Michaleris. "Design sensitivity analysis: overview and review". In: *Inverse problems in Engineering* 1.1 (1994), pp. 71–105.

[142] Anh H Truong, David W Zingg, and Robert Haimes. "Surface mesh movement algorithm for computer-aided-design-based aerodynamic shape optimization". In: *AIAA Journal* 54.2 (2016), pp. 542–556.

[143] Roberto Rosello Valera et al. "Modified algorithm for generating high volume fraction sphere packings". In: *Computational Particle Mechanics* 2.2 (2015), pp. 161–172.

[144] Felipe AC Viana, Gerhard Venter, and Vladimir Balabanov. "An algorithm for fast optimal Latin hypercube design of experiments". In: *International journal for numerical methods in engineering* 82.2 (2010), pp. 135–156.

[145] T. Voisin et al. "Defects-dictated tensile properties of selective laser melted Ti-6Al-4V". In: *Materials & Design* 158 (2018), pp. 113–126.

[146] David W Wagner, Derek P Lindsey, and Gary S Beaupre. "Deriving tissue density and elastic modulus from microCT bone scans". In: *Bone* 49.5 (2011), pp. 931–938.

[147] Wolfgang A Wall, Moritz A Frenzel, and Christian Cyron. "Isogeometric structural shape optimization". In: *Computer methods in applied mechanics and engineering* 197.33-40 (2008), pp. 2976–2988.

[148] Michael Yu Wang and Xiaoming Wang. "A level-set based variational method for design and optimization of heterogeneous objects". In: *Computer-Aided Design* 37.3 (2005), pp. 321–337.

[149] Z. Wang et al. "Powder-scale multiphysics modeling of multi-layer multi-track selective laser melting with sharp interface capturing method". In: *Computational Mechanics* (2018).

[150] Zhuo Wang et al. "Uncertainty quantification in metallic additive manufacturing through physics-informed data-driven modeling". In: *Jom* 71.8 (2019), pp. 2625–2634.

[151] Fabian Wein, Peter D Dunning, and Julián A Norato. "A review on feature-mapping methods for structural optimization". In: *Structural and multidisciplinary optimization* (2020), pp. 1–42.

[152] Jared Willard et al. "Integrating physics-based modeling with machine learning: A survey". In: *arXiv preprint arXiv:2003.04919* 1.1 (2020), pp. 1–34.

[153] Jun Wu, Anders Clausen, and Ole Sigmund. "Minimum compliance topology optimization of shell–infill composites for additive manufacturing". In: *Computer Methods in Applied Mechanics and Engineering* 326 (2017), pp. 358–375.

[154] Tianfang Xu and Albert J Valocchi. "Data-driven methods to improve baseflow prediction of a regional groundwater model". In: *Computers & Geosciences* 85 (2015), pp. 124–136.

[155] Takayuki Yamada, Kazuhiro Izui, and Shinji Nishiwaki. "A level set-based topology optimization method for maximizing thermal diffusivity in problems including design-dependent effects". In: *Journal of Mechanical Design* 133.3 (2011).

[156] Qiming Zhu, Zeliang Liu, and Jinhui Yan. "Machine learning for metal additive manufacturing: predicting temperature and melt pool fluid dynamics using physics-informed neural networks". In: *Computational Mechanics* 67.2 (2021), pp. 619–635.

[157] Navid Zobeiry and Keith D Humfeld. "A physics-informed machine learning approach for solving heat transfer equation in advanced manufacturing and engineering applications". In: *Engineering Applications of Artificial Intelligence* 101 (2021), p. 104232.

[158] T. I. Zohdi. "Additive particle deposition and selective laser processing–a computational manufacturing framework". In: *Computational Mechanics* 54 (2014), pp. 171–191.

[159] T. I. Zohdi. "Computation of the coupled thermo-optical scattering properties of random particulate systems". In: *Computer Methods in Applied Mechanics and Engineering* 195 (2006), pp. 5813–5830.

[160] T. I. Zohdi. "Construction of a rapid simulation design tool for thermal responses to laser-induced feature patterns". In: *Computational Mechanics* (2017).

[161] T. I. Zohdi. "Laser-induced heating of dynamic depositions in additive manufacturing". In: *Computer Methods in Applied Mechanics and Engineering* (2017).

[162] T. I. Zohdi. "Modeling and simulation of cooling-induced residual stresses in heated particulate mixture depositions". In: *Computational Mechanics* 56 (2015), pp. 613–630.

[163] T. I. Zohdi. "Rapid simulation of laser processing of discrete particulate materials". In: *Archives of Computational Methods in Engineering* (2013), pp. 1–7.

[164] TI Zohdi and N Castrillon. "Variability of Targeted Material Thermal Responses to Laser-Induced Heating in Additive Manufacturing". In: *Journal of Manufacturing Science and Engineering* 141.8 (2019), p. 081012.

[165] Jonathan Zornow. *Sewbo, Inc.* 2020. URL: `https://www.sewbo.com/`.