Data Driven Modeling, Statistical Analysis, and Machine Learning for Additive Manufacturing

N.S. Johnson,[1] C.K.H. Borg,[2] J. Saal,[2] R. Liu,[1] X. Zhang,[1] C.A. Brice,[1]
H. Wang,[3] J. Ling,[2] B. Meredig,[2] B. Kappes,[1] and A.P. Stebner*[1]

[1] *Department of Mechanical Engineering, Colorado School of Mines, Golden, CO 80401*
[2] *Citrine Informatics, Redwood City, CA 94063*
[3] *Department of Computer Science, Colorado School of Mines, Golden, CO 80401*

In metal additive manufacturing (AM), materials and components are concurrently made in a single process as layers of metal are fabricated on top of each other in the (near) final topology required for the end-use product. Consequently, a large number of processing degrees of freedom (tens to hundreds) must be simultaneously controlled and understood; hence, metal AM is a highly interdisciplinary technology that requires synchronized consideration of physics, chemistry, materials science, physical metallurgy, computer science, electrical engineering, and mechanical engineering. The use of modern statistics-based approaches to modeling data sets with many degrees of freedom (known as machine learning) with metal AM can reduce the time and cost to elucidate and optimize the complex multidisciplinary phenomena. Machine learning techniques have been used in materials science for several decades. Most prolifically, the density functional theory community (DFT) rapidly adopted machine learning and has used it since the early 2000s for evaluating many combinations of elements and crystal structures to discover new materials. This focused review examines the potential of machine learning in metal AM, highlighting the many parallels to previous efforts in materials science and manufacturing, and discusses new challenges specific to metal AM.

CONTENTS

I. MOTIVATION

Metals additive manufacturing (AM) enables paradigm shifts in how metallic products are manufactured, provides versatility in the type and design of parts, it enables materials and parts to be produced by a single machine and enables novelty in the microstructure design of the parts resulting in new and enhanced properties. Although decades of scientific and engineering work in industry, academia, and government have resulted in the commercialization of AM, there have been significant roadblocks to fully realize AM's potential, particularly in the control of consistency and quality in part production and in the development of materials amenable to the AM process. Integrated Computational Materials Engineering (ICME) approaches have proven to accelerate development and adoption of materials technologies. Traditionally ICME approaches incorporate physics-based experiments and simulations across many fields and scientific methods. However, for metals AM, much of the physics are still being discovered, hence physics-first approaches to ICME are still immature. The diverse array of promises and problems in AM has resulted in a field of study which is rich with data – so much so that our ability to store and analyze the data is challenged. At the same time, this wealth of data is motivating a paradigm

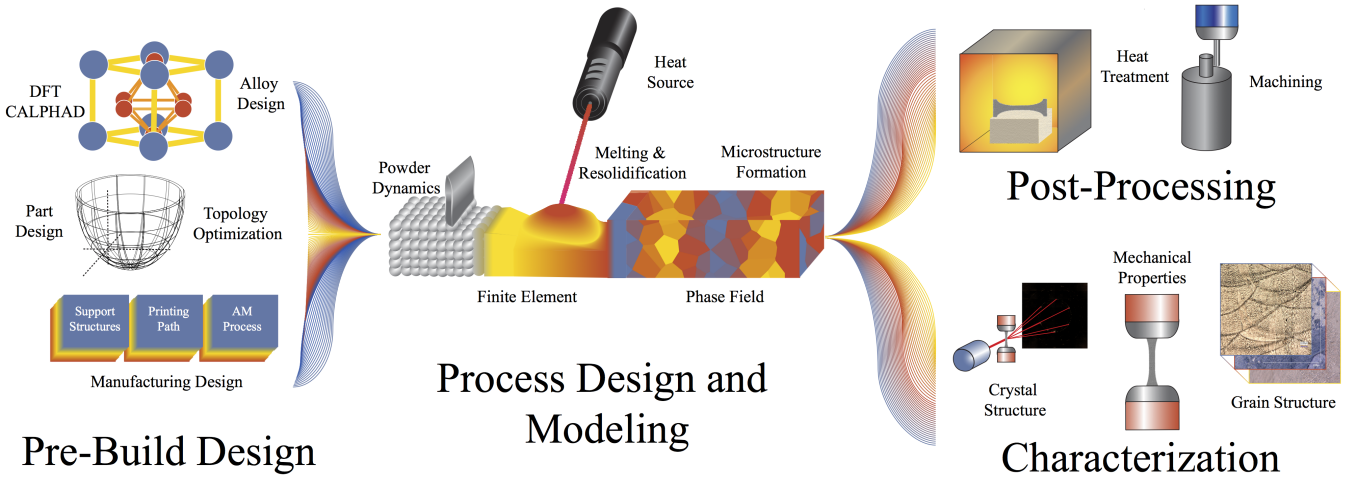

FIG. 1. The design space of additive manufacturing. The study and use of AM generates data at all length and time scales throughout the manufacturing pipeline, from pre-process engineering, to process monitoring, to post-manufacture characterization and implementation. This data can be implemented in a larger integrated computational materials engineering ecosystem that optimizes process design at a single step within the manufacturing pipeline and links materials properties across different steps in the pipeline.

shift in ICME toward data-driven, statistics based machine learning ICME approaches.

A. Background

The 20th century saw the maturation of materials science and engineering as a field of study, enabling targeted materials discoveries and innovations for specific applications. Over the past decades, materials development has greatly accelerated by formulating materials problems through the process-structure-property-performance paradigm [**?** **?**].

The process-structure-property-performance (PSPP) paradigm is a core philosophy in materials science that governs how the manufacturing of a material determines its ability to be used in different engineering applications. First introduced by Olson in 2000 [**?**], the PSPP relationships break down materials development into four key areas of scientific and engineering interest. Various phenomenon and measurements of interest for an AM-specific PSPP paradigm can be seen in Figure **??**.

The **processing** of a material is the thermal, mechanical, and chemical behaviors experienced by an AM part throughout the duration of its manufacture. The design parameters chosen, and subsequent thermal history experienced by the part, are most often thought of as an AM part's processing history. Controllable machine parameters like energy density, the order in which part layers are manufactured, or the location of parts on the build plate are determining factors in the part's processing. The actual processing history, however, is better described by the thermal history of the build volume, both during manufacture and post-processing, the mechanical forces it experiences, and any chemical reactions that occur in or on the part. Processing *routes* are often discussed in AM and typically refer to beneficial or detrimental processing histories that impact the part's structure.

The **structure** of a material is a wide-ranging concept that spans all length scales of materials science. Structure can refer to the crystallographic structure of a metal, at the atomic scale, to the morphology and orientation of metal grains, at the mesoscale, to the geometry of the part being manufactured, at the macroscale. *Micro*structure is a term often used in materials science referring to a specific subset of the material structure. Microstructure for metals most commonly refers to grain and sub-grain level information like material phases, grain morphologies, texture, and any defects like pores or cracks that might be present. The structures encompassing microstructure are so often referred to because they are a primary determinant of a material's properties.

The **properties** of a material are characteristics which determine its quality or qualification of use in an engineering application. Properties for AM parts are also wide-ranging and the 'properties of interest' vary depending on the desired engineering application of the part. Mechanical properties are some of the most studied material properties in AM because AM has a lot to offer for improving mechanical performance of parts. However, AM currently has equally as many drawbacks for mechanical properties which is why it has requires so much scientific and engineering attention. Other properties of interest include thermal properties, such as the heat transfer through an AM part, chemical properties, like corrosion resistance, and optical properties, like reflectance. Measuring material properties are the first step

in evaluating an engineering part's performance.

The **performance** of a part is its ability to be successfully implemented in an engineering application. Performance can be viewed through the lifetime of an AM part in deployment. While an AM part may have improved static properties over traditionally manufactured metals, it is not necessarily guaranteed to be more useful in an application. A common performance metric for materials is fatigue life, or the ability of a part to consistently perform over many application cycles. Other performance characteristics include thermal and chemical stability in dynamic environments. Performance can also be thought of as the environment in which an AM part will be employed, including the mechanical, thermal, chemical, etc. forces it will experience during use.

The process-structure-property-performance paradigm has been presented (and is often though of) as a cause-and-effect relationship between the different levels. In reality, it is a complex relationship with many materials models/theories that interconnect and bridge each level. Different processing routes will result in different properties, the performance of a part can change its structure, the structure of a material can determine its performance, and so on. The ICME approach to materials science is focused on modeling, bridging, and predicting relationships throughout the PSPP paradigm.

Computational materials science has enabled the prediction of microstructure from processing and of properties from microstructure, reducing the need for costly and time consuming experimentation. Today ICME approaches tightly integrate physics-based computational models into the industrial design process, allowing the desired performance requirements of a part to guide the design of a novel material. Examples specific to alloys include low-RE Ni superalloys for better turbine performance [1] and lower cost and radioactive element free Ferrium S53 alloy designed for corrosion-resistant landing gears [2]. Both cases took materials innovation timelines from decades to years, demonstrating the practical capability of designing new materials within an industrial product timeframe. Generalizing this capability to more industries and further accelerating the process is the primary goal of the Materials Genome Initiative (MGI) [3].

Many of the current limitations of AM are difficult to solve with existing physics-based ICME approaches. The physics of AM processes are more complex than traditional casting methods as they involve rapid solidification, vaporization and ingestion of volatile elements, and complex thermal history that consists of dozens of heating and cooling cycles, each one different. Furthermore, all of these additional complexities vary from one location to another within a part and from part to part within a build volume. For AM, physics-based ICME tools have been mostly developed through attempts to bootstrap legacy models formulated for traditional manufacturing of alloys to AM data, with some success. However, the lower cost and time barrier to entry for performing AM has enabled the rapid accumulation of experimental data, enabling a high-throughput approach for finding optimized AM processing methods and parameters of existing alloys. As such, current AM materials development is largely combinatorial and consists of adopting AM processing of legacy alloys that were developed for other types of manufacturing using extensive design of experiments.

It is with awareness of the grand amounts of data being generated in AM that machine learning (ML) can accelerate AM innovations and their commercialization. ML as a method for model development has shown wide application in recent years, including finance [4], molecule design for genomics, chemistry and pharmacology [5], social networking [6] and, most importantly to this review, materials science. Still, the use of ML in materials science has been relatively limited for a variety of reasons, especially the lack of large curated datasets amenable to existing ML methodologies. Through the work done under the MGI, this data limitation was identified as a primary impediment to future materials innovations [3]. In response, there has been significant recent investment in infrastructure development for materials databases suitable for data informatics innovations. It is now recognized and accepted that ML frameworks can couple legacy physics-based ICME tools with experimental data to produce more accurate process-structure-property models and to automate the iteration of designed experiments for model improvement and optimized materials [144**? ? ?**]

We proceed to review how the paradigm shift from physics-based to data-driven ICME approaches can be made through solving metals AM challenges. We begin by phrasing terms and ideas from additive manufacturing in ways that are compatible with machine learning. We provide basics of machine learning algorithms and how they can be interpreted in an additive manufacturing study. Following this introduction to using ML for AM problems, we review other uses of machine learning in materials science and state the uses of such approaches for solving AM challenges. We conclude with an outlook for how database-driven design of materials for AM can accelerate material creation and part qualification.

II. PHRASING ADDITIVE MANUFACTURING AS A MACHINE LEARNING PROBLEM

Data-driven modeling and machine learning have been employed to great success across several fields including materials science. While machine learning may seem abstract at first, it can be expressed and understood in plain terms. Many of the tenets and frameworks for machine learning are based in mathematical operations which may be familiar to the reader, but applied in new ways. In this section, we proceed to define the basic terminology and classes of machine learning and data. Specific ML algorithms are then introduced specific to contextual AM examples in Section III.

A. The Design Space of Additive Manufacturing

The **design space** of additive manufacturing is the set of all manufacturing parameters and material properties. More generally, the term 'design space' will be used throughout this article to discuss the set of all additive manufacturing data which can be used with data-driven methods and machine learning algorithms. An example design space for laser powder bed fusion metals, the most prolific of current technologies, is depicted in Figure 1. A complementary example of controllable manufacturing parameters is given in Table I. Observable manufacturing phenomena may link the manufacturing parameters to the resulting materials properties, hence they may also be used to augment the manufacturing parameters and material properties within the design space. Examples include melt pool morphology, temperature and cooling rates.

A single set of machine parameters, observed process phenomenon, and measured material properties can be considered as a *coordinate*, or point, in the design space. A single set of manufacturing parameters – for example, a fixed laser power, scan speed, scan path, and layer thickness in LPBF – is often thought of as a design space coordinate. Single coordinates, defined this way, can sometimes lead to a multitude of material properties due to latent variables, unforseen complications, and the stochasticity of the process. Include manufacturing phenomenon in the design space coordinate can more accurately identify truly unique points in the design space. Under a broader definition, any part which is manufactured under a single set of conditions and is observed to have a set thermal history and material properties can be considered to be manufactured *at that point* in the design space.

B. Additive Manufacturing Data Types and Formats

Data types and sources in additive manufacturing are as widespread as any field of engineering and science. Machine learning algorithms, however, typically operate using specific mathematical representations of data. It is important to recognize all the different sources and formats of AM data and consider how they can be coerced into use for machine learning. Representing data in the design space properly will be a major pre-processing step for machine algorithms to be amenable to AM optimization.

Scalar data are the most commonly obtained data in materials science – one simply needs to feel the weight of an ASME book on measured material properties to see this truth. Scalar data can include material property measurements such as strength, hardness, density and modulus. A single set of manufacturing parameters, such as heat source parameters, can also be represented as a list of scalars. Desired models in AM research include models of the relationships between material properties and manufacturing parameters. In these cases, the machine inputs will most often be represented as a vector, such as

$$\mathbf{x} = [\text{Machine Input 1, Machine Input 2}, \ldots \\ \ldots, \text{Machine Input } n] \quad (1)$$

out of n many scalar machine inputs. Vector representations of the design space are useful in determining manufacturing conditions which will result in similar results, as well as in building regression models of the AM process.

Another common data type often represented by vectors is *time series* data. Time series data are usually measured during the manufacturing process. Highly desired time series data collected during AM are temperature histories; others include videos, spectroscopy, and part deformation and strain. These data can sometimes be used as-is, or operations can be performed to extract scalar data values from the time series. Scalar statistical values such as the maximum, minimum, mean, and standard deviation of a time series signal can be equally useful in machine learning applications. Signal processing of time series data is also an active area of machine learning research which seeks to extract the most useful information from a time series. Pre-processing time series data can result in useful machine learning inputs; separately, signal processing algorithms have been used to reduce the amount of data required for machine learning to be successful **cite**.

Matrices are perhaps the most diverse type of data used in terms of the range of materials science information that the can convey. Large datasets of scalar values measured multiple times will often be represented by a matrix where the columns represent each individual measurement type and the rows are repeated measurements of that value, usually for different samples. Representing this collection of scalar measurements as as a matrix can be useful; one example would be

$$\mathbf{X} = \begin{bmatrix} x_{1,1} & x_{1,2} & \ldots & x_{1,m} \\ x_{2,1} & x_{2,2} & \ldots & x_{2,m} \\ \vdots & \vdots & \vdots & \vdots \\ x_{n,1} & x_{n,2} & \ldots & x_{n,m} \end{bmatrix} \quad (2)$$

where the columns of $\mathbf{X}$ represent the different design space considerations being made, out of m, and there have been n repeated measurements of each. Time series data can be represented in a similar way, with each column being a point in time and each row being a different measurement of the time series data [1].

[1] It is important to note that the row and column definitions given here are not concrete; indeed, the columns could be repeated measurements and the rows could represent different measurement types. If a machine learning algorithm is sensitive to the row-column definitions, the solution is as easy as taking a transpose.

TABLE I: A possible design space for laser powder bed fusion additive manufacturing. There are over 10^9 possible combinations of machine inputs, based on the listed ranges and step sizes. Any possible combination of these parameters is a point in the design space.

Parameter	range	step size	levels
Skin/Contour/Core laser parameters:			
Power	100-200 W	10 W	10
Scan speed	500-1000 mm/s	100 mm/s	5
Spot size	50-100 μm	10 μm	5
Energy density	1-5 J/mm^2	1 J/mm^2	5
Build parameters:			
Polar angle	0-90°	30°	4
Azimuth angle	0-180°	90°	3
Sieve count	0-10	2	5
Amount of recycled powder	0-100%	10%	10
Other parameters:			
Blade direction	0-300 mm	10 mm	30
Transverse direction	0-300 mm	10 mm	30
Hatch spacing	0.1-0.50 mm	0.1 nm	5

Images are also most often represented as matrices. Each entry in the matrix represents something about a corresponding pixel in the image: it could be a grayscale intensity, it could be one of the three RGB channel values, it could be a binary value, etc. Almost all image processing algorithms discussed in this review rely on a matrix representation of images. There are many toolboxes available, both free and commercial, which can preprocess images into the correct form for machine learning algorithms. Examples include the MATLAB Computer Vision Toolbox and the C++/Python OpenCV libraries.

While many machine learning algorithms can operate directly on machine inputs and processing outputs, it can be equally useful to calculate covariances between data points and use those as machine learning inputs. *Covariance* is measured between two data points $\kappa(\mathbf{x}, \mathbf{x}')$, instead of being a property of a single data point. The function $\kappa(\cdot, \cdot)$ is often referred to as a kernel function. The covariance between data points encodes cross-correlated information within the design space. In most cases, kernel functions assess the similarity of design space coordinates. Ways of calculating covariance are many and varied and will be explicitly defined where they are used. The nature of *why* covariance is a useful tool for machine learning algorithms is slightly more complex than this review article allows; however, some of the most powerful and useful machine learning applications in materials science have relied on the calculation of covariance between design space coordinates. Readers who are seeking more information on the use of covariance in machine learning are directed to [**?**].

C. The Assumptions Behind Machine Learning

Consider moving this initial point/paragraph to Section III – we need to introduce the language of ML first; later, we can tie it into process-property relationships. Ultimately, AM users desire a model that relates manufacturing parameters to material properties; i.e., a process-property model. In a functional form, this model can be written as

$$f(x_1, x_2, ..., x_n) = \mathbf{y}, \tag{3}$$

where the inputs x_i are n different manufacturing parameters and the outputs $\mathbf{y}$ are measurable properties.

Without machine learning, humans have adopted the ability to best understand process-property relationships by connecting separate process-structure and structure-property models. For example, individual process-structure models are developed to relate : heat source parameters to melt pool topologies [7] using the finite element method or similar computational techniques; melt pool topologies to solidification routes using thermal- and mass-diffusion models [8]; solidification to microstructure evolution using phase field methods [9]. Separately, structure-property models are developed such as relating grain orientations and stress to material properties using crystal plasticity [10]. It is then by making connections between individual process-structure and structure-property models that researchers relate process parameters to material properties. This understanding-through-sequential-modeling approach has a detriment in that the errors and uncertainty of each approach compound on one another, so that the final result is less accurate, more time consuming, and more expensive than the direct models of process-property relationships.

Machine learning is not a complete replacement for these traditional approaches, but rather a complementary modeling approach that can accelerate or even automate that process of building connections across many degrees of freedom that span the many different individual phenomenon within AM processes.

Start section II.C here, move previous discussion to SectionIII Two assumptions are necessary when using machine learning:

1. *The Similarity Hypothesis*: Parts manufactured at similar points in the design space will have similar properties.

2. *The Relational Hypothesis*: A correlative relationship exists between the data input to the model and the response of the system.

The first hypothesis is used to compare data and datasets, as well as to search and optimize through regression and classification algorithms. The similarity hypothesis is dependent upon mathematical tools for comparing similarity in a sensible way; certain data types and data relationships can make similarity interpretation difficult. This is an active area of research within the data sciences

The second hypothesis is required for finding regression and classification functions that are physically accurate.

There are two types of machine learning covered in this review: unsupervised and supervised. Unsupervised learning will find trends in a dataset that are indicative of the underlying behavior. Supervised learning will learn a form of $f(\mathbf{x}) = y$.

D. Unsupervised Machine Learning

Unsupervised machine learning algorithms are used to identify similarities or draw conclusions from unlabeled data by relying on the similarity hypothesis. Unsupervised approaches are useful for visualizing or finding trends in high dimensional data sets, screening out irrelevant modeling inputs, or finding manufacturing conditions which produce similar material properties.

Consider an experiment that varies three different manufacturing inputs x_1, x_2, x_3 and measures a single material property y. In matrix form, the data are expressed as:

$$\mathbf{X} = \begin{bmatrix} x_{1,1} & x_{2,1} & x_{3,1} \\ x_{1,2} & x_{2,2} & x_{3,2} \\ \vdots & \vdots & \vdots \\ x_{1,m} & x_{2,m} & x_{3,m} \end{bmatrix} \qquad \mathbf{Y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} \tag{4}$$

where $x_{i,j}$ is the j^{th} measurement of the i^{th} manufacturing input. A distance metric can be defined between data points in the design space. For example, data can be collected at two points $\mathbf{a} = (x_1, x_2, x_3)$ and $\mathbf{b} = (x_1 + \delta, x_2, x_3)$. The ℓ_2 norm of $\mathbf{a} - \mathbf{b}$ yields

$$||\mathbf{a} - \mathbf{b}||_2 = \delta. \tag{5}$$

The value and magnitude of δ gives an inclination about how similar $\mathbf{a}$ and $\mathbf{b}$ are. If δ is close to zero, then a researcher can say that they are similar, or even the same if δ is exactly zero. As δ becomes larger a researcher can say $\mathbf{a}$ and $\mathbf{b}$ become more dissimilar. The concept of 'similar' manufacturing conditions may be easy to assess by an experimentalist when tuning only a few parameters at a time. When taking into consideration tens or hundreds of design criteria, sometimes with correlated inputs, elucidating similar manufacturing conditions becomes difficult. This vector distance approach is a simple, yet effective first glance at similarity in a design space and is generalizable to n many design criteria.

Let us say that δ is small and that $\mathbf{a}$ and $\mathbf{b}$ are similar manufacturing conditions. Now, consider a third point in the design space $\mathbf{c} = (x_1 + \delta, x_2 + \delta, x_3)$ that has not yet been measured. Since $\mathbf{c}$ was manufactured at similar conditions to $\mathbf{a}$, as measured by $||\mathbf{c} - \mathbf{a}||_2 = 2\delta$, then we may say that $\mathbf{a}$, $\mathbf{b}$, and $\mathbf{c}$ are all similar to each other. If the similarity hypothesis is correct then manufacturing with conditions $\mathbf{a}$, $\mathbf{b}$ and $\mathbf{c}$ should yield similar measurements of y.

At some point, a researcher will have a set of initial manufacturing inputs $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$, $\mathbf{d}$, etc., and associated property measurements that have been tested. Churning through the remainder of all possible manufacturing conditions becomes expensive and tedious quickly. Instead, researchers can use similarity metrics to determine whether or not a future test is worth running. Comparing the manufacturing inputs through vector distance gives a rough idea of the possible outcome before spending time and resources on running a test. If the intent is exploring design spaces then manufacturing at conditions *furthest away* from previously observed points may be the answer. If looking for local maxima of quality, an operator would want to manufacture at conditions *nearest to* the conditions currently known to have high quality.

Using vector distances as metrics of similarities can produce results that are analogous to creating process maps [11]. Process maps are used to divide 2 dimensional plots of manufacturing inputs into regions of quality, or regions of different material responses. A common application of unsupervised learning is finding clusters in data sets which produce useful partitions of material behavior. Similarly to how process maps define boundaries between material performance and response, clustering with unsupervised learning can identify manufacturing conditions which will result in similar material performance.

E. Supervised Machine Learning

In a *supervised machine learning algorithm* the goal is to find a functional relationship that approximates the underlying physical relationship $f(\mathbf{x}) = \mathbf{y}$ based on previously measurements of $\mathbf{y}$ at point $\mathbf{x}$ in the design space. That is, supervised machine learning algorithms relate manufacturing inputs to labeled data. Functional relationships can take many forms, depending on the specific supervised ML algorithm being used. One method is to model the relationships as a vector product

$$\mathbf{X}T = \mathbf{Y}. \tag{6}$$

where T is a vector of coefficients that weigh the machine inputs to approximate an entry in $\mathbf{Y}$.

A researcher usually seeks this relationship through the measurements they have observed; in this case, the measurements are stored in the matrices of Eqn. 4. A common method to find a vector representation of T, and a critical element in most machine learning algorithms, is through least squares regression. Least squares regression finds T through a minimization problem, given by

$$\min ||\mathbf{X}T - \mathbf{Y}||_2^2. \tag{7}$$

Equation 7 can be interpreted analogously to similarity measurements for unsupervised algorithms: the closer that $\mathbf{X}T - \mathbf{Y}$ is to zero, the more similar T is to $f(\mathbf{x})$.

The methods of solving equation 7 are many and varied; indeed, much of this review will focus on finding solutions to Eqn. 7 for various problems throughout additive manufacturing. The result is an approximation to the functional relationship $f(\mathbf{x}) = \mathbf{y}$. A new point of interest in the design space $\mathbf{x}'$ can be chosen and its associated material property $\mathbf{y}'$ can be predicted by computing

$$\mathbf{x}'T = \mathbf{y}'. \tag{8}$$

This simple example demonstrates how functional relationships can elucidate more information about design spaces from previously generated data.

Commonly used machine learning algorithms in materials science and engineering are given in Table III. Note that the field of Machine Learning is evolving as fast as AM itself; hence Table III is by no means a comprehensive review of all machine learning algorithms, but rather is intended to provide a comparison between the form and function of some of the most widely adopted algorithms in materials science and engineering.

F. Machine Learning Toolboxes

All of the machine learning algorithms and approaches discussed in this review are, in some form, free and openly accessible. Many machine learning packages exist across many different programming languages and platforms. Commercially available platforms like MATLAB also have robust machine learning packages that can be installed. Table II highlights a variety of computational tools and packages and their relevance to AM synthesis optimization.

III. CURRENT ICME TOOLS ARE WELL EQUIPPED TO INTEGRATE WITH AN ML FRAMEWORK

The following section details how machine learning approaches can tie-in to current R&D efforts in AM. Since the study of AM is often focused around specific problems, this section is tailored to many common areas of study within AM. Typical analysis, characterization, and simulation methods for AM tie-in well with different machine learning algorithms. Even more so, ML can be used to automate the generation of knowledge about AM process-structure and process-property relationships.

This article is not an exhaustive review of either machine learning algorithms or additive manufacturing. The algorithms discussed were chosen because they were previously demonstrated in a materials science and engineering applications **or** because the possible application of an algorithm to AM was clear and immediate. Similarly, the additive manufacturing problems addressed are not all-encompassing; they are merely a few that may be immediately addressable with machine learning approaches.

A. Pre-Build Design

1. Alloy Design

Choice of alloy impacts the physics of AM from start to finish, starting with the optics of energy sources incident on feedstock and ending with the material properties of the final part. For example, the reflected/absorbed intensity of lasers on powder beds is determined by the powder's composition [17, 18]. The density of feedstock, both intra- and inter-granular density, plays a role in final part density [19]. Conduction modes in the melt are partially determined by the thermal properties of the alloy [20]. All of this is not to mention that different alloys exhibit different solidification kinetics, which can lead to drastically different microstructures after manufacture [21].

Problems in the additive process can also be linked to composition such as vaporization of constituent elements due to rapid thermal fluxes, impacting the stoichiometry of melt pools and, ultimately, quality [22]. Traditional engineering alloys sometimes need to be altered to improve compatibility with AM. Searching for new alloys specifically for AM may also be fruitful, as unique strengthening mechanisms can arise [22, 23]. Designing alloys for AM – either altering currently used alloys or starting from scratch – requires taking into consideration the compatibility of alloys' physical properties with AM. Alloy design for AM must take into consideration general alloy properties, like melting point, to feedstock-level properties, like vaporization temperature, to bulk alloy properties, like strength. Much of this information has been collated into databases that are compatible with design for AM. Alloy designers from AM should take advantage of these databases to search for new alloys for use in additive.

Databases exist that contain alloy properties ranging from the reflectivity of the alloy to the mechanical properties of alloys in bulk. The International Crystal Structure Database (ICSD) contains crystallographic information

TABLE II:

Language/Platform	Package	Applications
Python	scikit-learn	
	tensorflow	
	keras	
	OpenCV	
MATLAB	Statistics and Machine Learning Toolbox	
	Computer Vision System Toolbox	
C ++	OpenCV	

TABLE III: Several of the most widely used machine learning algorithms that have been used in materials science are compared.

Class of Algorithm	Examples	Applications	Strengths	Constraints
Weighted neighborhood clustering	Decision trees, k-Nearest neighbor	Regression, Classification, Clustering and similarity	These algorithms are robust against uncertainty in data sets and can provide intuitive relationships between inputs and outputs. See Ref. [12] for a primer on clustering.	They can be susceptible to classification bias toward descriptors with more data entries than others.
Nonlinear dimensionality reduction	t-SNE, Kernel ridge regression, Multidimensional metric scaling	Dimensionality reduction, Clustering and similarity, Input/output visualization, Descriptor analysis, Regression, Predictive modeling	These algorithms are robust against nonlinear input/output relationships and can provide intuitive projections of the material input/output space. For accessible examples, see Refs. [13, 14].	Projections can represent unphysical, difficult to interpret relationships. Global relationships can also be lost when nonlinear dimensionality reduction results are projected onto lower-dimensional spaces.
Linear dimensionality reduction	Principle component analysis (PCA), Support vector regression (SVR)	Dimensionality reduction, Clustering and similarity, Input/output visualization, Descriptor analysis, Regression, Predictive modeling	This type of algorithm can produce orthogonal basis sets which reproduce the training data space. They can also provide quick and accurate regression analysis. For a primer on PCA specifically, see Ref. [15].	The relationships studied must be linear in nature, and these algorithms are susceptible to bias when descriptors are scaled differently.
Search algorithms	Genetic algorithms, Evolutionary algorithms	Searching a material space to optimize on a certain condition, Lowest-energy state searches, Crystal structure prediction	Search algorithms are intuitive for material properties that can be described geometrically, such as topology optimization for weight reduction. They are efficient at searching spaces with multiple local extrema, such as finding local maxima of quality in multidimensional design spaces.	These algorithms are highly dependent upon selection and mutation criteria. For a useful application of genetic algorithms to process characterization, see Ref. [16].

for millions of compositions. The Linus Pauling files contains a range of material information, from atomic properties like radius and electron valency to crystallographic level information [24]. In modern day, large databases such as AFLOWLib [25], the Materials Project [26] and more allow users to search through large databases of relevant alloy information to find one that matches a desired property. Searching through large databases of information to find optimal compositions for manufacturing is actually one of the earliest materials informatics problems ever addressed. Methods exist to perform these searches in a fast, automated way. These methods are referred to as database mining, a data-driven materials design approach.

A study by Martin et al. used database mining to find micronucleants for Al alloys in powder bed manufacturing [20]. Part of the design process is identifying which alloy properties are important for the desired application. Heterogeneous nucleation of Al grains was the desired outcome of Martin's study. To induce

such nucleation, Martin et al. searched for possible nucleants whose crystallographic lattice parameters closely matched that of Al. This way, the Al grains would have a low-energy-barrier nucleating site from which to grow heterogeneously. Martin's study employed a search algorithm to search through 4,500 different possible nucleants and identify those with the closest-matching parameters. Ultimately, Zr was found to be the best candidate.

The same process employed by Martin – identify the properties which need to be satisfied, then search for a material that is closest matching – can be extended to many other alloy properties relevant to AM. Database mining was first introduced in material science to predict stable compositions, or estimate material properties from composition. Database mining has been successfully implemented to predict stable crystal structures [27–29] and predict material properties as a function of composition [30–34]. Some specially designed search algorithms have also been designed for improved speed in automated searches [35]. Successes have been found in designing Heusler compounds using high throughput search methods [36]. Reviews of early high-throughput searches for compositions with ideal properties can be found in [37, 38]. The same search algorithms employed in these studies can be extended to the additive case.

A limiting factor in database mining is that designers are limited to properties which have been measured or calculated. We do not have information about the vast space of *possible* materials. Consider a set of alloying elements for Ti such as $\{\mathrm{Al}, \mathrm{V}, \mathrm{Zr}, \mathrm{Cr}\}$. Researchers may need to test the impact of alloying composition on the dendrite arm spacing of Ti alloys. Phase field models exist which simulate the growth of and measure dendrite arm spacing as a function of a continuum of composition. A particularly efficient combined phase field/cellular automata model was implemented by Tan et al. precisely to model dendrite arm spacing in laser manufactured alloys [8].

Modeling all possible combinations of $\{\mathrm{Ti}, \mathrm{Al}, \mathrm{V}, \mathrm{Zr}, \mathrm{Cr}\}$ is possible with coarse additions of alloying elements, but undesirable. Machine learning can aid in the process to find an optimal composition without modeling all possibilities. *Genetic alogrithms* (GA) can search the space of possible alloys to find the optimal dendrite arm spacing. Genetic algorithms have been one of the most-used data driven approaches in materials science over the past few decades [29, 35, 39–43].

The principle of genetic algorithms is to evaluate the *fitness* of a population of candidate alloys against a *fitness function*. The fitness function is a method of evaluating how well a candidate alloy meets a criteria.

As a thought experiment, consider using the CALculation of PHase Diagrams (CALPHAD) method as a fitness function. It can be run for candidate compositions – in this case, various amounts of $\{\mathrm{Al}, \mathrm{V}, \mathrm{Zr}, \mathrm{Cr}\}$ alloyed into Ti – and used to evaluate its compatibility with rapid solidification. This is similar to a study completed in [44]. Once a fitness function has been identified, the next step in a genetic algorithm is to represent candidate alloys as a *gene*.

We can represent a gene as

$$\text{Alloy} = [\chi_1,\ \chi_2,\ \ldots,\ \chi_n]$$

where χ_1 is the species and weight percent of the first element (titanium, in this example), χ_2 is the species and weight percent of the second element, up to n elements. For example, Ti-6Al-4V would be represented as

$$[0.9\ \mathrm{Ti},\ 0.06\ \mathrm{Al},\ 0.04\ \mathrm{V}\ ]$$

The goal is to find the alloy with optimal dendrite arm spacing. First, a population of candidate genes needs to be generated, either randomly or by design. Two examples from a starting population may be

$$\begin{aligned}\text{Alloy 1} &= [0.9\ \mathrm{Ti},\ 0.05\ \mathrm{Al},\ 0.05\ \mathrm{V}\ ]\\ \text{Alloy 2} &= [0.9\ \mathrm{Ti},\ \ 0.1\ \mathrm{Zr}]\end{aligned}$$

The thermodynamic properties of the various compositions, and therefore their compatibility with AM, is estimated by running a CALPHAD model for each composition. It is not guaranteed that the optimal composition is in this starting population.

Genetic algorithms select genes out of the current population – called the parent generation – to proceed to another generation of model assessment – called the child generation. Selection consists of keeping the best performing compositions, say the top 10%, and discarding the rest. Genetic algorithms find optimal locations in the design space by relying on the similarity hypothesis. If one alloy is in the top 10% of genes then it is possible that a similar alloy will also be high performing – it may be even perform better. Once selection is done, the next step is to search the space near the best performing alloys from the parent generation.

Genetic algorithms generate similar compositions from those selected in the parent generation by making alterations to genes. One operation is *mutation*, whereby entries of the genes are changed. For example, we could mutate alloy 1 by changing the composition:

Parent Generation:	Alloy 1 = [0.9 Ti, 0.05 Al, 0.05 V]
Child Generation:	Alloy 1 = [0.9 Ti, 0.02 Al, 0.08 V]

where in the child generation the amount of V was increased, while the amount of Al was decreased. Another operation which may be performed is *crossover* where entries of genes are added or interchanged. For example, one crossover operation may look like

Parent Generation:	Alloy 1 = [0.9 Ti, 0.05 Al, 0.05 V]
	Alloy 2 = [0.9 Ti, 0.1 Zr]
Child Generation:	Alloy 1 = [0.9 Ti, 0.05 Al, 0.05 Zr]
	Alloy 2 = [0.9 Ti, 0.1 V]

where in the second generation V and Zr have been interchanged.

Selection, mutation, and crossover followed by model assessment and further selection, mutation, and crossover continues until the design criteria is met. The intuition behind genetic algorithms is that eventually the selection process is narrowed down to alloys within a given region of the design space such that further mutation and crossover do not produce new genes. Eventually, all the 'fittest' genes as determined by the model will converge to be approximately the same composition.

Genetic algorithms have been applied to alloy design for low and high temperature structural materials [30, 45], ultra high strength steels [46], specific electronic band gaps [47], minimum defect structures [48], exploring stable ternary or higher alloys alloys [41, 49], and more. For a review on the application of GA's to alloy design through the early 2000s see Ref. [50].

Other machine learning algorithms have also been applied for classification and optimization of alloy compositions. Anijdan used a combined genetic algorithm–neural network method to find Al-Si compositions of minimum porosity [48]. Liu et al. applied partial least squares to data mining of structure-property relationships across compositions [51]. Decisions trees have been implemented for a number of different alloy optimization, such as predicting ferromagnetism [52] and the stability of Heusler compounds [53].

In the search for new alloys, some compositions definitely *won't* be compatible with the additive process. It would be useful to identify these alloys up front or as quickly as possible. Additive can also be improved by machine learning algorithms which suggest the best materials or properties to test. A wide range of machine learning algorithms can be implemented to guide the entire experimental design process so that an optimized property is found as quickly as possible.

2. *Design of Experiments*

Parametric analysis, broadly defined, is an experimental method of mapping independent variables to their corresponding dependent parameters. Process-property relationships are typically studied through parametric analysis. Machine learning aids in investigations of AM by reducing the amount of experiments needed to characterize process-property relationships. Machine learning approaches like sequential learning model relationships in parametric studies to discover regions of the parameter space which will produce the most information about process-property relationships.

In additive manufacturing independent parameters such as laser energy, speed, build direction, composition, layer height, and more are varied to study their impact on material properties. Examples include relating build geometry to microstructure or surface roughness [54, 55] or temperature history to microstructure [56, 57], or substrate temperature to residual stress development [22, 58], or even entire manufacturing processes to microstructure [59]. One of the most common types of parametric analysis is relating heat source parameters to all aspects of AM, such as part temperature history [60, 61], microstructure [62, 63], mechanical properties [64, 65], residual stresses [66, 67], and more.

Both engineering and scientific investigations of AM utilize parametric analysis. In the sciences, parametric analysis proceeds until a theory or model can be presented for a process-property relationship. In engineering, parametric analysis continues until an optimality criterion is met, such as maximum strength or minimum porosity. Both disciplines vary independent parameters and measure dependent responses; doing so provides information about the underlying phenomenon.

Information is any observation of process-property relationships. For example, observing that a set of laser parameters results in an equiaxed microstructure can be considered information because the researcher has gained an idea of the properties to expect from set processing conditions. Therefore, *information gain* is any experiment which reveals a previously unobserved process-property relationship. Rigorous mathematical definitions of information and information gain have been defined, typically referencing back to Shannon's original formulation of information theory [68].

Traditional design of experiments maximize information gain by dividing up the design space beforehand to maximize distance between like experimental conditions. Machine learning driven design of experiments makes suggestions for future experiments based on the results of past experiments. It does not make any assumptions up front about correlations between design parameters. Rather, machine learning algorithms model relationships between inputs and outputs and suggest the experiment which is statistically most likely to result in information gain. Design of experiments with machine learning algorithms can be adopted by augmenting traditional design of experiments with a statistical model.

The first step is to identify which parameters which may impact properties, as is done in traditional design of experiments. As more parameters and finer step sizes are added, the size of the design space grows. Once the scope of the design space has been defined, the next step is to generate an initial dataset. This can be done with traditional design of experiments. The process-property relationships revealed in these initial tests will be the basis of an initial statistical model.

After an initial dataset is generated, the researchers need to define a *response function* which interprets the relationship between parameters and material properties.

One example is a regression model of the process parameters and material characteristics. A *random forest* algorithm trains many regression algorithms, each on a subset of the experimental data. Identifying tests of importance begins by identifying all n parameters in the design space that may or may not impact a final manufacturing result y. First, a subset of the n many parameters m_1 is chosen such that at least one parameter in $\mathbf{x}$ is left out. A simple regression function called a decision tree is trained

$$\hat{f}_1(\mathbf{x}_{m_1}) = y \tag{9}$$

on the subset of features m_1. Next, a new random sampling of the n parameters m_2 is chosen, and a second regression problem is solved $\hat{f}_2(\mathbf{x}_{m_2}) = y$. The random forest is the linear combination of all of these individual regression functions

$$\hat{f}(\mathbf{x}) = \frac{1}{B}\sum_{b=1}^{B} \hat{f}_b(\mathbf{x}_{m_b}) \tag{10}$$

out of B many parameter samplings. A standard deviation in the prediction output is typically computed as

$$\sigma = \sqrt{\frac{\sum_{b=1}^{B} \hat{f}_b(\mathbf{x}_{m_b}) - \hat{f}(\mathbf{x})}{B-1}} \tag{11}$$

Some of the features in $\mathbf{x}$ may be irrelevant for studying process-property relationships and some may be very important to model. Random forests test the importance of an AM process parameter in $\mathbf{x}$ by re-training every individual regression function $\hat{f}_b(\mathbf{x}_{m_b})$ with that machine input included. Then, the standard deviation is re-computed. If the inclusion of a parameter in every regression tree significantly changes σ then it plays a large role in prediction of y. Therefore, the feature should be included and tested in experiments. Experiments which vary this parameter are likely to produce information gain; they are chosen as future experiments.

Random forests are easy to use because they do not require tuning many hyper-parameters. Furthermore, random forests are very good at sifting through many irrelevant features to find the features that actually matter. Random forest training is also computationally inexpensive and easily parallelized. Random forests provide two advantages that are particularly important in the context of materials science: the ability to efficiently calculate uncertainty estimates and the added interpretability of feature importance metrics. Based on their ensemble nature, it is possible to generate uncertainty estimates for random forest predictions using jackknife-based procedures [69–71]. They automatically identify the most important features in a given model based on how often a feature is used in splitting criteria and what the aggregated information gain is over those splits. These feature importance metrics can provide insights into how the model is making its predictions.

Ling et al. performed design of experiments to optimize processing of high fatigue strength steels. [72]. In Ling's case, a random forest model was fit to design criteria from a database of steel compositions, processing routes, and fatigue strength. The goal was to find the composition and processing route combination which had the highest fatigue strength in the dataset using as few experiments as possible.

Ling's machine learning-assisted design of experiments proceeded by suggesting experiments with a high uncertainty in their result as modeled by the random forest regression. The intuition is that predictions by the response function which have low uncertainty have enough data to characterize the process-property relationships in that region of the design space. Therefore, researchers can have high confidence in the material properties they will achieve if parts are printed at those conditions. Thus, the process-property relationship likely has enough information to pick an optimal condition or investigate further. Regions of the design space with high uncertainty do not have enough information for the regression model. Thus, machine learning algorithms suggest these regions of the design space for further experiments. When only varying a few parameters at a time, regions of the design space which need characterization can be easily identified. When varying tens of parameters in additive manufacturing these regions of the design space are not apparent. Furthermore, correlated inputs can be masked by the complexity of the process-property relationship. Statistical models can spot these correlated regions of the design space.

Random forests have been applied successfully to a range of applications in materials science. They have been used to discover new Heusler compounds [53] and new thermoelectric materials [73]. They have also been used to model material properties such as thermal conductivity in half-Heusler semiconductors [74] and to break down fields for dielectrics [75]. A review article detailing many optimization algorithms for design of experiments can be found in Shan et al. [76]. Adoption of machine-learning assisted design of experiments algorithms can rapidly increase the rate at which the relationship between AM process parameters and material properties are understood.

3. Topology Optimization

Alloy design and experimental design focus around combinatorial screening of inputs to either search for *new* properties or optimize on current properties. These optimizations reduce manufacturing cost, monetary or otherwise, and maximize performance capability. The same optimization can be applied to mechanical properties of parts. For structural materials, the goal is to optimize load bearing capacity or lifetime while minimizing the amount of material used. For aerospace, the goal is to minimize weight. Unique manufacturing geometries was one of the first intended applications of AM. Topology optimization (TO) focuses around exactly this task – find-

ing optimized topological structures for a given mechanical application.

A filter is a mathematical operation which reveals information about a region of pixels/voxels in a mesh. Filters are most often represented as a product of a filter matrix with a matrix of mesh pixel values. Topology optimization proceeds by generating a CAD model of an AM part and modeling its performance, such as testing performance under mechanical load through an FEA simulation. Filters are applied to the CAD mesh which selectively removes material from the part. Then, the mechanical performance of the new part is modeled, followed by further material removal. This process proceeds until either a minimum weight/volume condition is met or the mechanical performance of the part is degraded.

In additive, topology optimization serves an additional purpose: TO algorithms can find un-printable regions of a part. Unsupported structures, low angle slopes, and certain part orientations during building are prohibited in AM because they will cause part deformation. An unsupported slope at acute angles can lead to part deformation and warpage [77]. Sacrificial support structures also need to be considered during topological optimization, along with the number of free-hanging features and the orientation of the part during manufacture. Langelaar et al. developed an AM-specific TO algorithm which searches for regions of parts that have too little support for manufacture [78, 79]. Other additive specific algorithms have been designed for optimizing density of parts [80]. These algorithms augment the AM process, both by taking advantage of the ability to optimize unique geometries, and also by identifying regions of parts which are incompatible with AM.

B. Process Design

Integrated computational materials engineering, as the name may imply, is primarily focused around *computational* engineering of materials. In experimental and engineering studies, the design space in AM makes choosing useful experiments difficult. Similarly, the wide range of AM physics to consider for modeling makes full-scale, full-physics computational difficult. This section focuses around applying machine learning algorithms to aid in computational studies and design of additive manufacturing.

1. *Model Complexity and Dimensionality Reduction*

Add high throughput grain boundary study of Homer [81]. A bottleneck in ICME approaches is the number of different models required to simulate all physics during additive manufacturing. The spatiotemporal scales that *could* be taken into consideration for AM process modeling span microns to meters. ICME of AM process modeling incorporates phase field, cellular automata, finite element, direct element methods, and more. At the smallest scale the feedstock, heat source, and melt pool dynamics have been modeled by finite element methods [7, 82, 83] or finite volume methods [84]. Microstructure growth from the melt pool is often the next phenomenon to be studied and has been modeled through phase field [9, 85–87], cellular automata [8], or finite element methods [57]. Thermal histories of entire parts or sections can be modeled next, typically by finite element methods. Thermal history models look at heat transfer through the part [88], residual stress build up during manufacturing [10, 89], and thermal history such as cooling rate and temperature gradient [61, 90]. Full ICME approaches involve modeling the AM process at all these steps to determine the entire processing history for a part. Martukanitz et al. published a full ICME investigation of AM [91]. A review of ICME approaches across spatiotemporal scales can be found at [92] and a review of the physics of AM modeling can be found at [93]. A review of finite element methods specifically for AM can be found at [94].

For many engineering studies of AM not all of these physics *need* to be modeled or included. The complexity of AM, however, obscures which physics are relevant to accurately model a given property. A coordinate point in the design space could be 160 variables long if only accounting for tunable machine parameters. Complexity in ICME of AM grows when considering cross-correlated information across different models. The result of one simulation y could be an input $\mathbf{x}$ of another simulation. A thermomechanical model may predict temperature gradient in deposited layers, while a phase field model may use thermal gradient to calculate solidification velocity. This problem motivates using computational models which balance accuracy with complexity.

Dimensionality reduction algorithms identify which parameters are relevant to model in an ICME approach and which are not, informing future ICME investigations of faster simulation routes to achieve the same result. Materials science has long had a need for dimensionally reduced, computationally accurate models. Some of the first applications of machine learning in materials science was for dimensionality reduction [28, 95–97]. The dimensionality reduction techniques covered in this review can be broadly classified into three mathematical frameworks: statistically driven methods, similarity analysis, and matrix factorization.

Statistically driven approaches can be employed to determine the parameters in $\mathbf{x}$ that strongly impact AM model outputs. A commonly used, statistically driven dimensionality reduction method is through random forest networks, as is done in design of experiments. The difference, however, is how uncertainty in the random forest is employed. In design of experiments, **outputs** with high associated uncertainty in the random forest model may be chosen for future tests. In dimensionality reduction, **inputs** with high uncertainty are indicative of irrelevant features.

Kamath utilized random forests to screen out irrelevant modeling parameters for predicting maximum density of additively manufactured parts [98]. Kamath started with an experimental dataset of manufacturing parameters and multiple modeling methods. An Eagar-Tsai simulation of a Gaussian laser beam on a powder bed was used to model thermal conduction during manufacture. The model originally began with four inputs (laser power, speed, beam size, and powder absorptivity) and a design space of 462 possible input combinations. Kamath utilized random forests to determine which input was most important for achieving fully dense parts. If simulations are time-intensive to run then 462 different simulations may be out of the question. Identifying which parameters do not impact the final result reduces the size of input combinations, therefore reducing the number of computations to be performed. As a fictitious example, if the angle between the laser and part surface does not impact the result of a simulation, then simulations do not need to vary the angle. This is valuable, time-saving information in an ICME study.

Kamath identified that laser speed and power were the most important inputs out of the four to determine melt pool depth and shape. Thus, models that vary only these two parameters can likely be relied upon for their results. After determining the most important inputs, the same regression tree was applied in order to find optimized manufacturing conditions for fully dense parts, using the same approach. Instead of identifying which features impacted the model standard deviation, the machine settings which maximized y were found.

Another dimensionality reduction technique is focused around visualizing trends in datasets. Dimensionality reduction through similarity analysis allows visual interpretations of distributions across high dimensional spaces. Visualizing an n-dimensional distribution is difficult. Process maps are commonly employed in AM to visualize 2D slices of the AM design space [11]. Process maps are plots which chart the possible values of machine inputs and identify regions of the design space with similar properties. A commonly employed process map in AM of Ti-6Al-4V describes grain morphology as a function of solidification velocity R and temperature gradient G [99]. Extending process maps to n many process variables would require $\binom{n}{2}$ plots. Similarity analysis can be used to express trends in an n dimensional space in a human interpretable way without relying on multiple 2D process maps.

The 'similarity' in similarity analysis arises from the fact that models are trained on covariances between coordinates in the design space, instead of the coordinates themselves. These approaches combine similarity analysis with regression. Covariance between AM parameters are measured by similarity and these similarity scores are the data for a regression model.

t-distributed Stochastic Neighborhood Embedding (tSNE) is a dimensionality reduction technique which measures distances in a high dimensional space and then projects data points onto a two dimensional plot. The similarity of all data points in the design space with each other is used to fit a distribution of similarities. The tSNE algorithm begins by fitting a probability distribution to all $\mathbf{x}$'s contained in a dataset. Relationships in n dimensional space are assessed through a *kernel function* $\kappa(\mathbf{x}, \mathbf{x}')$ which measures similarity between points in the design space. A commonly employed kernel is the Gaussian kernel

$$\kappa(\mathbf{x}, \mathbf{x}') = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{\mathbf{x} - \mathbf{x}'}{2\sigma^2}\right] \tag{12}$$

where σ is a user-specified or fit standard deviation in the distribution of points in the design space. This kernel function assesses distance in the n dimensional space and assigns a similarity value between $\left[0, 1/\sqrt{2\pi\sigma^2}\right]$ depending on how similar the values are.

After the n dimensional dataset is fit, then a 2 dimensional coordinate $\mathbf{x}^*$ is assigned to each $\mathbf{x}$. The reason for choosing a 2 dimensional coordinate is so that the final result can be visualized on a 2D plot. The tSNE algorithm fits a probability distribution to the n dimensional data set first, then assigns values to each $\mathbf{x}^*$ such that they have the same probability as the associated high-dimensional $\mathbf{x}$. Once the probability distributions have been assigned, the $\mathbf{x}^*$ values can be visualized on a 2D plot to investigate for trends.

The benefit of tSNE is that points which are close together in the n dimensional space appear close together on the 2 dimensional plot. This gives AM modelers an idea of how machine inputs and material behavior are distributed in the n dimensional space through a 2 dimensional visualization. This dimensionality reduction technique guides modelers as to what simulation conditions will reveal similar results. Furthermore, these 2D projections provide clusters of inputs and outputs which are similar in the high dimensional space. Analysis of these clusters can inform modelers as to correlations within their inputs and outputs, identifying which parameters can be screened out to save complexity.

A final dimensionality reduction technique requires expressing model data in a matrix and performing matrix factorization. For a dataset of model inputs and results, a matrix can be formed $\mathbf{X}$ whose rows are coordinates in the design space $\mathbf{x}_i$. Matrix factorization techniques re-represent correlations in large datasets in a simplified way. The matrix $\mathbf{X}^T\mathbf{X}$ is a measure of covariance within $\mathbf{X}$, different than Eqn. 12 but equally valid. The matrix $\mathbf{X}^T\mathbf{X}$ can be very large due to the design space of additive manufacturing. One type of matrix factorization, called Principal Component Analysis (PCA) re-represents the data matrix $\mathbf{X}$ as

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}^T \tag{13}$$

where the rows of Σ are the eigenvectors of $\mathbf{X}^T\mathbf{X}$ and the diagonal entries of $\mathbf{U}$ are called the principal components of $\mathbf{X}$. PCA operates such that the first eigenvector

and principal component indicates the most heavily correlated inputs of $\mathbf{X}$. The length of vectors in Σ is p many variables long. In cases where the original design space size n is large, the size of p is often much less than n, reducing the dimension of the problem. Regression can be performed on one, a few, or many of the eigenvectors in Σ to predict new model results using considerably less information than that contained in $\mathbf{X}$. Some studies have gone as far as to understand which correlations are being represented by the eigenvectors in Σ, revealing inputs or phenomenon which can be ignored during modeling. Such analysis is possible in AM, though requires further processing steps past matrix factorization. Materials science studies have utilized PCA previously to re-represent large datasets in simpler forms, such as predicting the formation energies of crystal structures from a lower dimensional space [100]. A review of applications of PCA in materials science can be found at [101].

In additive manufacturing, PCA can serve as a dimensionally reducing pre-processing technique. For a large data set with many machine parameters the design space can be reduced from vectors of n length to p length. The new vectors can then be used as regression model inputs for prediction of material properties based on trends observed in dataset $\mathbf{X}$. Doing so allows researchers to predict values at unobserved coordinates in the design space without running further tests. These types of models are called *surrogates*.

2. Surrogate Modeling

Phenomenon which are difficult to study experimentally, such as flow within the melt pool, are best studied through modeling approaches. Though expensive, full-physics modeling is quite necessary to understand how physics at different scales interact to impact the AM process. Full physics models, however, can be expensive and time consuming to run. If a model's computational expense is *too* high then performing simulations at all relevant manufacturing conditions can be impossible. Dimensionally reduction may not be an option. While it is useful for optimization and visualization, reduced order models are unlikely to capture the full dynamics of solidification in AM. Machine learning algorithms can use the results of previously run high fidelity simulations to fill in the gaps and reduce development time.

A *surrogate model* is a regression model that interpolates the results of high-cost simulations. Surrogate models are regressed on the inputs and results of previously run simulations. Then, the surrogate model interpolates simulation results at new coordinates the design space. Surrogate models preclude the need for running computationally expensive simulations for every possible manufacturing condition. Surrogates can be as simple as linear regression between simulation inputs and results, but are often more complex. The accuracy of a surrogate model is dependent upon how many previous simulations have been run and at how many different points in the design space.

Tapia et al. built a surrogate model for laser powder bed fusion of 316L stainless steel. They were concerned with predicting the melt pool depth of single-track prints solely from the laser power, velocity, and spot size [102]. The dataset used to build the surrogate was computationally derived, based on previous simulation methods used by the same research team [103]. In particular, they used the results from a computationally expensive but high-accuracy melt pool flow model of Khairallah et al. [7]. They ran powder bed simulations at various laser powers, velocities, and spot sizes, and the model told them the depth of the melt pool, among other information. The datasets provided enough information for a surrogate model to be trained to predict simulation results.

To build surrogate model, Tapia et al. used a machine learning model known as a Gaussian process model (GPM). A common model assumption in Gaussian process modeling is

$$z(\mathbf{x}) = y(\mathbf{x}) + \epsilon(\mathbf{x}) \tag{14}$$

where $y(\mathbf{x})$ is the approximation (surrogate) of the simulation process, $\epsilon(\mathbf{x})$ is a stochastic, randomly distributed noise in measurement, and $z(\mathbf{x})$ is the value given by a simulation. GPMs make the assumption that all finite joint distributions of $y(\mathbf{x})$ are distributed in a multivariate normal manner, also known as a Gaussian distribution. For AM modeling this assumption holds if the model data used for a surrogate is stochastic in nature. The primary goal in GPMs is to find model parameters for the mean process $y(\mathbf{x})$ and a covariance function $\kappa(\mathbf{x}, \mathbf{x}')$, which is a function of similar form to Eqn. 12. Fitting a Gaussian process model often begins with assuming a model function for covariance, fitting the model parameters such as σ to the observed values $z(\mathbf{x})$, then using those model parameters to predict simulation results $y(\mathbf{x})$ at other locations in the design space.

Tapia used Bayesian statistics to develop a probabilistic model that predicted melt pool depth from simulation inputs. They were able to successfully predict the outcomes of both high-fidelity simulations and experimental measurements solely by analyzing trends in previously obtained results. In particular, they were able to accurately predict the melt pool depth at a value that had never been observed before, either computationally or experimentally. For future investigations, predictions by the surrogate model can be relied up on instead of running a simulation or experiment.

Gaussian process models have benefits beyond their surrogate modeling capabilities. GPMs provide robust uncertainty metrics on the predictions they make. Uncertainty in prediction is important in materials informatics because it aids in discriminating against poor prediction accuracy in machine learning. Some machine learning models do not have straightforward ways of assessing model error [104].

Another benefit of GPM is that it aids in inverse design and design space visualization. GPMs can explicitly identify regions of the design space which will maximize or minimize a value. In the case of Tapia et al. response surfaces were created from the GPM which visualized the depth of melt pools as a function of laser power and speed. Doing so allows engineers to identify regions of the design space which provide specific material responses, an important tool in optimization for additive.

Another approach to full scale models in AM is building high-fidelity models from an ensemble of low-fidelity models. Current integrated computational models link phenomena across spatiotemporal scales by running many single-physics models and passing the results from model to model, then comparing with experimental results. An example is the model of Martukanitz et al. that considered the thermal, mechanical, and material response of Ti-6Al-4V alloys manufactured with powder bed processes [91]. Martukanitz's model uses the finite element method to model the thermal and mechanical response based on classical continuum heat transfer equations. At the same time, the Calculation of Phase Diagrams (CALPHAD) method modeled the thermodynamics and mass transfer for each chemical species, while Diffusion-Controlled phase Transformations (DICTRA) is used to model phase formation. As noted by Martukanitz, this approach becomes computationally infeasible as the number of deposition layers increases.

Even a *single* modeling method – just FEA, phase field, CALPHAD, etc – becomes computationally expensive for thousands of layers of material deposition. Based on this scaling, accurately modeling the physics of a full build seems prohibitive, at least without a major leap in computing power. A machine learning method known as *committee voting* or *ensemble modeling* may provide a workaround. These methods rely on sampling many individual models to predict the behavior of a larger class of physics.

In traditional ensemble modeling approaches, many different regression models are trained to model a relationship $y = f(\mathbf{x})$, as in regression trees. For a given input $\mathbf{x_1}$ all regression models are assessed and various outputs predicted, $f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_n(\mathbf{x})$, etc. Each model $f_i(\mathbf{x})$ is a different type of machine learning regression model. Different types of regression approaches are robust for different trends in data sets. The individual regression models could be a random forest, support vector machine, neural network, or any of the type of regression models discussed herein. A linear combination of these models can have a higher prediction accuracy than any individual model.

In AM, the same concept can be extended to datasets sampled from different physical models, instead of an ensemble of different regression models. Each model $f_i(\mathbf{x}_j)$ can be trained on the inputs and outputs of the jth AM simulation. These data sets for additive may be the inputs and solutions of a solidification or heat transfer model, for example, or experimentally obtained relationships.

Ensemble modeling can be used in AM to solve multi-objective optimization problems, such as optimizing the heat transfer through a build and the grain growth simultaneously. Doing so would require training an ensemble model of simulations. The simulations to include could be thermomechanical models of heat transfer, phase field models of grain growth, finite element models of laser absorption, and so forth. The final ensemble model would take the form

$$\mathbf{y}(\mathbf{x}) = \sum_{i=1}^{N} w_i f_i(\mathbf{x}) \tag{15}$$

where $\mathbf{y}$ is a response vector of properties to optimize, $f_i(\mathbf{x})$ is a single simulation of the process out of N, and w_i are weights associated with each simulation. The vector $\mathbf{y}$ would contain many parameters such as temperature $T(t)$, temperature gradient ∇T, solidification velocity $\nu(t)$, and many more phenomenon to monitor from simulations.

In the AM case, where information from models may overlap, this type of approach can screen out noise from simulations or experiments to gauge a more fundamental relationship. The uncertainties associated with a model can be weighted by the experimentally observed data points. Furthermore, predictions can be made across many different physical models, resulting in a predictive method for holistic analysis of the final properties. This method does not provide a picture of how the physics in each simulation interact or disagree. However, it can be used to simultaneously optimize the results of many different simulations and point toward desirable manufacturing conditions. A review of multiobjective optimization functions can be found at [105].

Meredig et al. applied this method to the prediction of ABC ternary compounds, where A, B, and C each represent an element [106]. The space of all combinations of A, B, and C elements is large and would require many, many simulations. Instead of running high-throughput DFT for all possibilities they ran calculations for AB, AC, and BC compounds to generate a database of information. Then, regression trees were trained from the inputs of each binary alloy simulation to predict formation energy. Finally, an ensemble model was trained of the form

$$E(\mathrm{ABC}) = w_{\mathrm{AB}} f(\mathbf{x}_{\mathrm{AB}}) + w_{\mathrm{AC}} f(\mathbf{x}_{\mathrm{AC}}) + w_{\mathrm{BC}} f(\mathbf{x}_{\mathrm{BC}}) \tag{16}$$

where $E()$ is the formation energy, w_{AB} is the weight for a regression model of AB alloys, $f(\mathbf{x}_{\mathrm{AB}})$ is the regression predicting formation energies for AB alloys, and so on. In the end, Meredig's model was able to predict $4,500$ new, stable ternary materials.

Machine learning is not only limited to ex situ experimental investigations or modeling approaches. Machine learning has also made major advances in signal processing and feedback. Many of the same ideas which apply to experimental and modeling studies can also be aids for in

situ analysis and feedback of the manufacturing process. In particular, they can be used for simultaneous feature recognition and processing in images of AM processes.

C. Process Monitoring and Characterization

Computer vision is a class of image recognition algorithms that have been developed for automated feature identification in signals. Intelligent computer vision utilizes machine learning algorithms to identify objects and features in images and time-series data.

Monitoring of AM processes produces data equally numerous to, if not in excess of, the data produced by parametric analysis and modeling. The numerousness of time series data in AM warrants usage of quick, efficient, and robust signal processing methods for process monitoring, feedback, and control. These signal processing algorithms are closely related to machine learning. They serve as tools in their own right, and can also pre-process data for use in other machine learning applications, like clustering and regression.

1. In Situ Process Monitoring and Feedback

Thus far, in situ control in AM has been consistently ranked as one of the most-needed technologies for advancing the technology [107–109]. The combination of rapid solidification and the small length scales of AM solidification can make traditional process monitoring approaches difficult. Machine learning can fill in gaps where human-specified process monitoring models are insufficient.

Process monitoring involves acquisition of real-time signals which can reveal information about a wide variety of phenomenon during manufacturing. McKeown et al. has used dynamic transmission electron microscopy to measure solidification rates in powder bed AM [110]. Bertoli et al. has also characterized cooling rates using high speed imaging [111]. Raplee et al. has used thermography to monitor the solidification and cooling rates of electron beam powder bed fusion, relating the temperature profiles to defect and microstructural characteristics [112]. Distortion of parts due to thermal cycling was investigated by Denlinger et al. by means of thermocouples in contact with the build substrate [67]. Computer vision can be employed in additive manufacturing to monitor the printing process, such as characterizing temperature profiles, identifying abnormal melt pool morphologies, and automatically detecting defect formation. These methods, and many others, are amenable to aid by computer vision and signal processing.

The type of data being collected in situ is most often in the form of time series or image data. Signal processing of either typically involves identifying signals which deviate from a mean, target signal. Examples include a spike in temperature or a sharp change in intensity in an image, both of which may indicate a deviation from a desired processing conditions. Image processing *filters* can be used to selectively modify or extract features AM data. Image processing filters are mathematically analogous to those introduced for topology optimization (Section III A 3).

A filter is implemented as a mathematical operation, a kernel, applied to an area of pixels in an image. Filters attempt to use local spatial information and *a priori* knowledge of the expected properties of the image to improve image quality and extract features, e.g., distinctive characteristics such as edges or regions of similar intensity (domains) that represent the boundaries or spatial extents of objects, phases, etc. In AM, filters are often selected to reduce noise in the raw signal, or identify transitions between visually distinct regions, such as the transition between melt pool and powder bed.

A comprehensive review of image filters is beyond the scope of this review, so the interested reader is directed to the many works on this topic, c.f., **?**]. However, three use cases are especially common and worth specific acknowledgement: reduction of high-frequency noise, also known as salt-and-pepper noise; additive noise reduction; and edge detection.

High frequency noise is characterized by sudden changes in intensity relative to the surrounding field. Although there are a number of possible causes, this may be caused by pixel-level variability or insufficiency in the detector, e.g. “dead pixels” or excessive gain. Median and conservative filters are commonly used when the fraction of noise pixels is large (1%–10%) and small (¡ 1%), respectively.

Additive noise, unlike high frequency noise, is a result of insufficient Poisson (counting) statistics, which may result from insufficient incident intensity, exposure time, or detector efficiency, random variability in the intensity of the input signal (Gaussian noise), or both. A gaussian filter adjusts the intensity of each pixel according to the gaussian-weighted intensities of neighboring pixels. Unlike median and conservative filters, a gaussian filter will soften edges, making adjacent domains less distinct.

Edge detection captures local changes in intensity to identify transitions between adjacent domains. Laplacian filters use a kernel approximation to the second derivative while the Laplacian of Gaussians (LoG) combines a laplacian kernel with a gaussian smoothing kernel. In noisy images these filters generally identify many spurious edges and are often used as part of a larger algorithm, such as Canny edge detection [**?**].

The use of filters alone does not constitute machine learning, but features extracted using these filters can be used as part of the larger machine learning workflow. For example, these features can be used in learning algorithms to correlate characteristic features with particular behaviors in the manufacturing process. In this case, identification of a feature, or set of features, may be sufficient to indicate a particular process outcome.

A computer vision method called *template matching* can be used for automatic identification of signal features.

Template matching involves the comparison of a measured features to a database of pre-identified signal features. The scale-invariant feature transform (SIFT) [113] and a variant, "Speeded-Up Robust Features" (SURF) [114] are both feature identification algorithms that can be used for template matching. One such template matching algorithm is the *bag of visual words* or dictionary method [122].

Would a figure of a neural network help clarify this discussion, or is it unnecessary?

Neural networks (NNs) are particularly well-suited to handle features extracted from images, or simply the images themselves. There are many references that describe neural networks in detail, c.f. **?**], and an increasing number that address the specific challenges associated with neural networks in the larger space of materials science [**?**]. There are several properties of NNs that are worth repeating here, however. Each layer in a NN is connected to the next layer through an affine (linear) transformation. This step stretches, scales, and skews the input vector.

$$\mathbf{z}^{(i+1)} = \boldsymbol{\theta}_i^T \mathbf{x}^{(i)}$$

where $\mathbf{z}^{(i+1)}$ is the input into the $(i+1)$ layer and $\mathbf{x}^{(i)}$ is the output from the previous, i^{th} layer. Then, an activation function, such as those summarized in Figure 2, introduces a non-linearity that can warp distort the vector input to that layer.

$$\mathbf{x}^{(i+1)} = f\left(\mathbf{z}^{(i+1)}\right)$$

By increasing the depth of the NN, that is, adding additional layers, and the width (number of nodes) of those layers, a NN can be used to approximate any function [**?**]. However, the general sparsity of materials data coupled to the complexity of process–structure–process relationship requires an understanding of the tradeoffs and requirements of using NNs in materials science, and in AM more specifically. Beyond the basics of model architecture, overfitting and the bias–variance tradeoff that is part of any machine learning model, a basic understanding of the role of activation functions can help to develop an intuition for the use of NN in materials and manufacturing.

One of the early uses of NNs was in classification. The perceptron, logistic sigmoid (or simply, sigmoid), and hyperbolic tangent are all activation functions that choose between two options (0 or 1, or in the case of tanh, -1 or 1). Although more mathematically complicated than the perceptron, the logistic sigmoid,

$$\sigma(z) = \frac{1}{1+e^z}$$

has the beneficial property that its derivative is an analytical expression of the sigmoid function itself, i.e.

$$\sigma'(z) = \sigma(z)\left(1-\sigma(z)\right)$$

which enables very quick calculation of the Jacobian during backpropagation and training, that is used to fit the layer weights, $\boldsymbol{\theta}_i$, also known as the model hyperparameters. While a binary posit may seem overly limiting, even multinomial classification can be broken down into a sequence of such binary posits: A or not A; and if not A, then B or not B; and if not B, C or not C; etc. However, such a serial solution will require deep neural networks (more hidden layers) and, with more hidden layers, longer training on larger datasets in order to fit all model hyperparameters.

In the case of a multinomial classification problem, a more simple network may be possible by using one-hot encoding. A one-hot encoding vector is defined for N exclusive options: one element in the N-element vector is 1, all other values are 0. Rather than using multiple layers to construct the binomial ladder required to simulate a multinomial decision, the softmax activation function selects one-from-many in a single layer. Since each value in the input vector appears in the softmax exponent; even small differences in the magnitude of z result in large differences in the output of this activation function; one option, represented by one node or neuron in the layer, dominates. Simplification of the network architecture by choosing activation functions that more closely resemble the nature of the problem is one point where domain-specific knowledge is critical.

NN are also used for regression. Unlike classification, regression requires a continuously variable response across the problem domain, and so the binary posit of the logistic sigmoid and the multinomial posit of the softmax are replaced with activation functions such as ELU, softplus, Leaky ReLU, and most commonly, ReLU, the Rectified Linear Unit. While each behaves differently, particularly across the negative domain ($x < 0$), the simplicity and robustness of the ReLU have made it the most commonly used activation function for regression-type layers.

Combining the concepts of neural networks and image processing filters, convolutional neural networks (CNNs) not only learn how to correlate features to results, they are specifically designed to identify the filters that extract those features. These networks require large numbers of hyperparameters, in the tens to hundreds of millions, which introduces an insurmountable training burden due to the sparsity of materials data. However, CNNs trained on natural images have demonstrated a remarkable similarity in their initial layers [**?**]. These first few layers identify basic shapes, edges, and colors that are common to many image types; a phenomenon that many groups have exploited overcome the limitation of data sparsity through transfer learning [72], including specific work in the field of additive manufacturing. Yuan et al [115] were able to successfully monitor melt track width, standard deviation, and continuity of tracks in situ during laser powder bed manufacturing. Scime and Beuth trained a convolutional neural network to identify six different types of defect which are typical of laser powder bed fu-

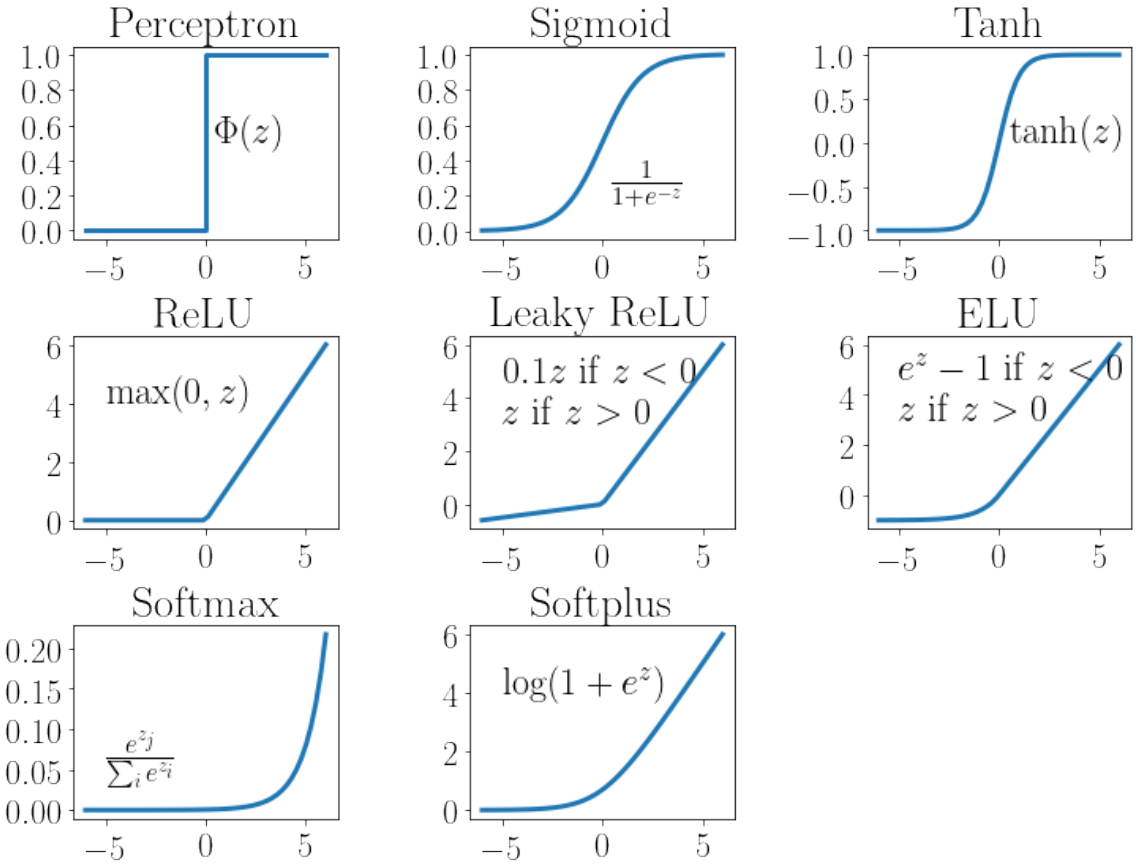

FIG. 2. Common activation functions in artificial neural networks (NNs) that introduce nonlinearity into the NN. The sigmoid is the archetype activation function because the closed form solution for the derivative of the sigmoid, which is used in backpropagation, is an excellent pedagogical tool; however, the rectified linear unit (ReLU) is, at present, the most common activation function in the hidden layers of NN. Uses for the other activation functions are provided in the text.

sion, with fairly good prediction accuracy [116]. Kwon et al. classified melt pool morphologies using a neural network [117]. These studies encompass only a few of the possibilities of in situ process monitoring for AM.

2. *Featurization of Qualitative Image Data*

The use of images in studying additive manufacturing is widespread, common throughout all aspects of the manufacturing process, and provides key information about material properties and processing. As with all aspects of AM, the sheer size of image data to be analyzed is profound due to the large design space of AM. The types of information taken from images includes grain characteristics, like size, orientation, and phase, and defect characteristics, like pore size or crack length. When characterizing all of these features for all possible processing conditions and alloys the size of the problem grows quickly.

Computer vision algorithms have been tested for automation of materials science image classification and analysis. Using these algorithms can speed up the experimental characterization process of AM. Furthermore, computer vision techniques can quantify information which may have otherwise only been used qualitatively or measured by approximation.

It is worthwhile to mention up front that these algorithms have been *tested* on microstructure and, in some cases, additive-specific images. There are few algorithms that can process AM microstructure data 'out-of-the-box.' Rather, these algorithms will need to be tailored in order to quantify AM images specifically. However, the algorithms discussed here have been proven on non-AM microstructure datasets, thus they should be extensible to AM datasets. The computer vision approaches which work for microstructure data are often the same approaches discussed in the previous section for in situ monitoring.

One AM-related application of image characterization is measuring particle size distributions in AM powder feedstock. DeCost and Holm used SIFT with a dictionary classifier, as in template matching, to measure the particle size distribution for a dataset of synthetic powder particles [118]. Particle size distribution plays in several steps across the additive process including energy absorp-

tion and part metrology [17, 119, 120]. DeCost created datasets with six different particle size distributions. Image features were identified and classified using k-means clustering on the features found by SIFT. Then, a classification algorithm known as a support vector machine (SVM) was trained to classify image features into particle sizes. DeCost was able to achieve 89% overall classification accuracy in measuring particle size distribution this way. DeCost et al. later improved upon this powder classification method and were able to achieve higher classification accuracies for real powder images [121].

Strides have been made in automatically identifying and quantifying information from metallographs [72, 122–124]. A good portion of quality control in materials science as a whole, not just AM, involves classifying materials based on metallographs or micrographs of microstructure. Work is being done across materials science to apply machine learning based computer vision to classifying and quantifying information in these microstructural images. Doing so will speed up the process of materials characterization and qualification, while also providing methods of quantifying information which otherwise would have stayed in a qualitative form. Examples include classification of grain stuctures, measurements of grain size, pore size calculations, and more.

An additive-specific image segmentation algorithm was used by Miyazaki et al. [125]. Five image filters were convolved with microstructure images of selective laser melted Ti-6Al-4V. The features identified by these filters were used in a random forest algorithm to segment the image into regions of α phase grains and β phase grains. The algorithm was able to automatically calculate area fraction of primary and secondary α phases that form during cooling. It was also able to calculate the nearest-neighbor distance between grains. Nearest neighbor distance of grains is indicative of grain characteristics like size, morphology, and distribution.

Chowdhury et al. took a more expansive approach to performing feature identification in microstructures. In particular, they were looking to classify microstructures as either dendritic or non dendritic. Chowdhury employed 8 different feature identification methods for a dataset of images. Classification was performed using an ensemble of ML techniques including support vector machines (SVM), Naïve Bayes, nearest neighbor, and a committee of the three previous classification methods [126]. Chowdhury's wide approach to image classification achieved classification accuracies above 90%.

It would be overly burdensome to lay out every *possible* application of computer vision in additive manufacturing. Efforts are underway across materials science to implement computer vision for the automation of materials classification. Rather, the authors would like to refer the reader to reviews on the subject of computer vision for materials science, as well as open libraries listed in Table II. The hope is that readers will discover the many possible uses of computer vision and begin applying methods to their own AM problems.

IV. LEARNING FROM THE PAST: MOVING TOWARDS DATABASE-DRIVEN DESIGN OF ADDITIVE TECHNOLOGIES

The scientific approaches to studying additive manufacturing discussed herein – parametric analysis, computational modeling, in situ monitoring, and the like – produce data. The application of machine learning to these scientific approaches likewise produces data. All of this data contains a slice of the AM design space. AM researchers who have access to more data in the design space can better tailor manufacturing processes and material properties. Furthermore, machine learning algorithms trained on large datasets typically have higher prediction accuracy and a greater range of behaviors they can model. Making AM process-structure and process-property data open and accessible to the scientific public accelerates the rate at which research and engineering advances.

Databases of process-structure and process-property relationships are not a new concept in materials science. Databases like the Linus Pauling Files or International Crystal Structure Database have been widely used for materials design. However, databases of materials information are beginning to expand beyond general materials properties like thermodynamic behavior or crystal structure. Domain-specific databases are being generated from high throughput experimental and computational investigations that have occurred over the past thirty years. Studying and understanding the development of data-driven fields in materials science can aid in the development of data-driven approaches for additive manufacturing.

Experimental high throughput investigations have been used in materials science for many decades [127]. Common deposition techniques have enough degrees of freedom to allow for continuous compositional variation within a single sample, which allows for continuous mapping of composition-structure-property relationships [128–130]. These manufacturing methods have the same design space problem that additive has: the number of possible input combinations obscures many of the important underlying process-structure phenomenon. In cases like this, manufacturing and characterizing large catalogues of samples can lead to optimized properties faster than a theory-driven approach [131, 132]. High throughput deposition studies with chemical vapor deposition, metallorganic chemical vapor deposition, physical vapor deposition, and atomic layer deposition, among other techniques are commonplace for the manufacturing of sensors, batteries, photovoltaics, electronics, and the like [37, 133–135]. Furthermore, the parameters of interest in these studies can sometimes be quickly catalogued using high throughput characterization techniques like laboratory X-ray diffraction and electron probe microanalysis.

These studies culminate in large libraries of material properties listed as a function of composition. As far

back as the 1990s, data-driven algorithms were being applied to these large libraries of composition-property data. Evolutionary and genetic algorithms were trained on composition to predict stable crystal structure and material properties [35, 39, 42, 136, 137]. Even neural networks, which did not have the widespread then use then that they have now, were being applied for the prediction of crystal structures based on composition [138].

Modeling challenges in materials science have also been tackled using large databases with machine learning. Packages such as the Vienna Ab initio Simulation Package (VASP) have been employed for high throughput searches of stable material systems with a wide range of properties. Computational high throughput investigations have proven to closely match experiments in many regimes [139]. High throughput density functional theory (DFT) studies generate quite a bit of data and are therefore well equipped for machine learning and database-driven design. The application of high throughput DFT is widespread for design of materials with all sorts of properties including high temperature superconductors [140], lithium ion batteries [33, 141, 142], molecule design [143, 144], cathode materials [145], piezoelectrics [36], ferroelectrics [146], corrosion resistant films [147], and thermoelectrics [148, 149]. Each of these studies, like parametric studies in additive, vary a set number of model input parameters and measure a lsmaterial property as the dependent response.

Yet many of the same modeling obstacles exist in DFT as in AM, such as a lack of transferability between models and the computational expense of large material systems. The design space problem exists here as well – there are so many possible compositional combinations that knowing *where* to look is difficult. Machine learning was proposed as a solution for obstacles in high throughput DFT as early as 2005 [150]. Large unit cells whose properties cannot be directly calculated using DFT are often approximated using machine learning approaches like neural networks [151], genetic algorithms [43], and principal component analysis [97]. Studies applying machine learning to databases of computational information have gone beyond tackling computational problems. In some cases, the studies have revealed previously unobserved or uncharacterized relationships between crystal structure information and materials properties [152].

Image processing databases have developed outside of materials science specifically, though a database of materials-specific images is highly desirable for the training of recognition algorithms. The collection and distribution of image databases have enabled rapid developments in the field of computer vision. Many of the more common objectives with computer vision – autonomous navigation, face recognition, object recognition, image segementation – have databases which are catalogued in online repositories like CVonline [153] and VisionScience [154]. Open sharing of additive microstructures will aid in the development of segmentation and identification algorithms which are suited for additive specific problems.

In an effort to reduce the design time on material systems, programs like the Materials Project incorporate data taken from a wide range of experimental and computational methods into an open-source, accessible database. The Materials Project also features electronic, structural, and thermodynamic calculations of different materials as well as an automated workflow for doing DFT computations of material systems [26, 155]. Other databases of materials information include AFLOWLib [25, 156], the Harvard Clean Energy Project [157], Japan's National Institute of Material Science [158], and the Open Quantum Materials Database [159]. Some pipelines for high-throughput computation and analysis have included consideration of publication timelines in their processes [160]. These databases offer a multitude of benefits to materials researchers. First and foremost, publicly accessible databases offer an infrastructure for the free flow of experimental and computational results. Synergy between research groups becomes easier as data is shared more freely.

Furthermore, many of these online databases also provide tools for performing material design. The Materials Project offers a design interface, whereby users can specify a set of material properties and are provided with a list of likely candidate materials. Other projects, like AFLOW, allow for fast high-throughput DFT calculations of a wide range of material systems.

The generation of databases that are accessible to the scientific public is a primary step on the roadmap of the Materials Genome Initiative [161]. Much of the development of materials databases have focused on computationally-derived materials information. Infrastructure and standards need to be developed that allow for sharing of experimental data that is understandable and usable by many researchers. Data journals are becoming more common for sharing datasets from scientific investigations and are making strides in standardizing data-sharing infrastructure [162]. The publication of datasets themselves for public use is becoming more common [163, 164].

Having open, accessible databases improves the rate at which machine learning can be applied to design for additive manfuacturing. Machine learning as a tool driving materials design was proposed some time ago. Review articles have explored the many and varied uses of machine learning across materials science, many of the applications finding great success [165–167]. A review article on best practices for machine learning in materials science can be found here [168].

Additive manufacturing should move toward the same types of infrastructure for open data sharing. The combinatorial problems in additive are widespread and cover many, many length scales. Large institutions may have the resources to link time- and length-scales in additive manufacturing. Smaller research groups are often limited to studying a single process phenomenon and do not necessarily have means to integrate their knowledge into other additive manufacturing studies. The generation

of additive databases allows for a democratization of research and an acceleration of the pace at which additive manufacturing advances are made.

V. CONCLUSIONS

Materials informatics has demonstrated great success in design and optimization of many material systems. Additive manufacturing is primed to benefit from the same algorithms and statistical models. Many of the major obstacles that lie ahead in additive manufacturing – full scale modeling, integrated design, feedback and control – can be tackled through the adoption of machine learning. However, machine learning itself is not the end-all-be-all of manufacturing. There are many obstacles in the application of machine learning itself that will need to be addressed along the way.

A. Key Application Areas for Machine Learning in Additive Manufacturing

- **Coupled Physics-Statistics Models:** The original goal of materials informatics, dating back to high throughput thin-film studies in the 1990s, was to model material process-structure-property relationships that were highly complicated and lacked a single governing physical theory. Additive manufacturing is the embodiment of a complicated physical system, where governing equations across optics, fluid mechanics, solid mechanics, thermodynamics, and kinetics have to be incorporated into one model. Machine learning can build computationally accessible surrogate models of more complicated physical systems that are useful for engineering and design.
- **Materials Design:** Materials design through machine learning has already been applied in a wide range of fields cited here, including thermoelectrics, photovoltaics, semiconductors, Heusler compounds, and many, many more. Design in these fields typically focuses around combinatorial studies of compositions, crystal structures, and a material response. Materials are manufactured through a wide variety of techniques but optimization is rarely applied to the manufacturing method itself, just the materials used in manufacturing. In additive, not only does the material system need to be tailored but the conditions of manufacturing also need optimization. Materials properties to consider range from composition and atomic properties to phase kinetics. Manufacturing optimization includes the energy density used, deposition rate, feedstock supply mechanism, and more. Machine learning can integrate optimization across these separate design considerations. Process optimization is likely to include in situ control.
- **Automated Process Control:** There are many variables to monitor and keep track of in the additive processes. There are equally many sensors and measurement techniques to monitor the process. Advancements in signal processing and computer vision must be taken advantage of to build incorporating process control models. Intelligent feedback and control for additive can simultaneously integrate and understand multiple signal types **and** optimize on multiple objective functions simultaneously. Taking full advantage of the promises of AM – topologically optimized geometries, functionally graded materials, minimized design-to-fly time – will require tight control over the manufacturing process.

B. Further Developments are Needed in Both Additive Manufacturing and Machine Learning

- **Data Sharing Infrastructure:** Programs like the Materials Project, AFLOW, and OQMD have accelerated the rate at which materials design can occur, as well as the rate at which scientific data is shared. The democratization of data has allowed many different research teams to search through the materials design space in search of new materials, to great success. The same type of democratization is possible in additive if infrastructure exists for sharing of AM data. However, standardization of AM data types should be addressed before data can be shared in a useful, meaningful way.
- **Curation of Data and AM Standards:** Success in applying data-driven approaches is tied tightly to the quality of data being used. Even data that has been collected with the highest care and precision can be detrimental to a model if it is labeled incorrectly or inconsistently. Work is proceeding in standards development for additive manufacturing [**?**]. However, additive manufacturing technology development has sometimes proceeded faster than standardization. Care needs to be taken in developing AM standards that are consistent across manufacturing devices and can also account for developments in the broader technology.
- **Experimental Measurement and Sensor Development:** While in situ measurement devices are widespread, the time and length scales of additive manufacturing can push the limits of current high-end sensors. Imaging methods that can resolve the fast, dynamic, microscale melt pools of additive would allow for a huge leap in process monitoring and control. Equally important is developing methods of determining temperature history throughout the duration of builds. Both of these technologies are crucial for fine control over the additive process.

- **Physics Informed Data Driven Models:** Additive manufacturing has developed amazingly over the past few decades thanks to traditional scientific and engineering approaches in many different fields. Modeling AM using classical thermal, mechanical and kinetic models has allowed for most of the successes in AM. This review is suggesting that machine learning be used as a complementary tool to these traditional approaches. It would be unwise to completely ignore physical theories which have shown applicability in AM. Rather, machine learning algorithms should be built around currently existing models. There are equally rich mathematical frameworks in both materials science and machine learning which are currently being utilized separately. The physics of AM at all length scales – solidification, phase kinetics, heat transfer, solid mechanics, etc. – should be used as first principles for building physics-informed statistical models. Many in the materials science community have considered how to use domain knowledge to build better informatics models [39, 136, 168**?**]. The same should be applied to additive manufacturing.

[1] T. M. Pollock, Nature Materials **15**, 809 (2016).
[2] G. Olson and C. Keuhmann, Scripta Materialia **70**, 25 (2014).
[3] "Materials genome initiative for global competitiveness," Tech. Rep. (United States National Science and Technology Council, Office of Science and Technology Policy, 2011).
[4] I. Bose and R. K. Mahapatra, Information and Management **39**, 211 (2001).
[5] R. Gómez-Bombarelli, J. N. Wei, D. Duvenaud, J. M. Hernández-Lobato, B. Sánchez-Lengeling, D. Sheberla, J. Aguilera-Iparraguirre, T. D. Hirzel, R. P. Adams, and A. Aspuru-Guzik, ACS Central Science **4**, 268 (2018).
[6] P. Brusilovsky, A. Kobsa, and W. Nejdl, *The Adaptive Web*, edited by P. Brusilovsky, A. Kobsa, and W. Nejdl, Vol. 1 (Springer-Verlag, New York, 2007).
[7] S. A. Khairallah, A. T. Anderson, A. Rubenchik, and W. E. King, Acta Materialia **108**, 36 (2016).
[8] W. Tan, N. S. Bailey, and Y. C. Shin, Computational Materials Science **50**, 2573 (2011).
[9] J. Kundin, L. Mushongera, and H. Emmerich, Acta Materialia **95**, 343 (2015).
[10] D. Pal, N. Patil, K. Zeng, and B. Stucker, Journal of Manufacturing Science and Engineering **136**, 1 (2014).
[11] J. Beuth and N. Klingbeil, Journal of Materials: Laser Processing (2001).
[12] J. Quinlan, Machine Learning **1**, 81 (1986).
[13] J. B. Tenenbaum, V. de Silva, and J. C. Langford, Science **290** (2000).
[14] S. T. Roweis and L. K. Saul, Science **290** (2000).
[15] R. Bro and A. K. Smilde, Analytical Methods Tutorial Review **6** (2014).
[16] J. J. Grefenstette, IEEE Transactions of Systems, Man, and Cybernetics **16** (1986).
[17] C. Boley, S. Mitchell, A. Rubenchik, and S. Qu, Applied Optics **55**, 6496 (2016).
[18] J. Trapp, A. M. Rubenchik, G. Guss, and M. J. Matthews, Applied Materials Today **9**, 341 (2017).
[19] G. Bi, C. N. Sun, and A. Gasser, Journal of Materials Processing Technology **213**, 463 (2013).
[20] J. H. Martin, B. D. Yahata, J. M. Hundley, J. A. Mayer, T. A. Schaedler, and T. M. Pollock, Nature Letters **549**, 365 (2017).
[21] P. Collins, D. Brice, P. Samimi, I. Ghamarian, and H. Fraser, Annual Review of Materials Research **46**, 63 (2016).
[22] C. A. Brice, W. A. Tayon, J. A. Newman, M. V. Kral, C. Bishop, and A. Sokolova, Materials Characterization (2018).
[23] A. Wang, S. Song, Q. Huang, and F. Tsung, IEEE Transactions on Automation Science and Engineering **14**, 968 (2017).
[24] P. Villars, N. Onodera, and S. Iwata, Journal of Alloys and Compounds **279**, 1 (1998).
[25] S. Curtarolo, W. Setyawan, S. Wang, J. Xue, K. Yang, R. H. Taylor, L. J. Nelson, G. L. W. Hart, S. Sanvito, M. Buongiorno-Nardelli, N. Mingo, and O. Levy, Computational Materials Science **58**, 227 (2012).
[26] A. Jain, S. P. Ong, G. Hautier, W. Chen, W. D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, and K. A. Persson, APL Materials **1** (2013), 10.1063/1.4812323.
[27] A. Fransceschetti and A. Zunger, Letters to Nature **402** (1999).
[28] C. C. Fischer, K. J. Tibbetts, D. Morgan, and G. Ceder, Nature Materials **5**, 641 (2006).
[29] A. R. Oganov and C. W. Glass, Journal of Chemical Physics **124** (2006), 10.1063/1.2210932.
[30] Y. Ikeda, Materials Transactions **38**, 771 (1997).
[31] A. M. Gopakumar, P. V. Balachandran, D. Xue, J. E. Gubernatis, and T. Lookman, Nature Scientific Reports **8** (2018).
[32] D. Wu, Y. Tian, L. Zhang, Z. Wang, J. Sheng, W. Wang, K. Zhou, and L. Liu, Materials **11** (2018).
[33] S. Kirklin, B. Meredig, and C. Wolverton, Advanced Energy Materials **3**, 252 (2013).
[34] W. Setyawan, R. M. Gaume, S. Lam, R. S. Feigelson, and S. Curtarolo, ACS Combinatorial Science , 382 (2011).
[35] D. Wolf, O. Buyevskaya, and M. Baerns, Applied Catalysis A: General **200**, 63 (2000).
[36] A. Roy, J. W. Bennett, K. M. Rabe, and D. Vanderbilt, Physical Review Letters **109**, 1 (2012), arXiv:1107.5078.
[37] G. Gilmer, H. Huang, and C. Roland, Computational Materials Science **12**, 354 (1998).
[38] H. Koinuma and I. Takeuchi, Nature Materials **3**, 429 (2004).
[39] J. Morris, D. Deaven, and K. Ho, Physical Review B **53**, R1740 (1996).
[40] K.-M. Ho, A. A. Shvartsburg, B. Pan, Z.-Y. Lu, C.-Z. Wang, J. G. Wacker, J. L. Fye, and M. F. Jarrold,

Nature (London) **392**, 582 (1998).
[41] G. H. Jóhannesson, T. Bligaard, A. V. Ruban, H. L. Skriver, K. W. Jacobsen, and J. K. Norskov, Physical Review Letters **88**, 2555061 (2002).
[42] D. P. Stucke and V. H. Crespi, Nano Letters **3**, 1183 (2003).
[43] G. L. W. Hart, V. Blum, M. J. Walorski, and A. Zunger, Nature Materials **4**, 391 (2005).
[44] S. Li, U. R. Kattner, and C. E. Campbell, Interesting Materials and Manufacturing Innovation **6**, 229 (2017).
[45] A. Kulkarni, K. Krishnamurthy, S. Deshmukh, and R. Mishra, Materials Science and Engineering A **372**, 213 (2004).
[46] W. Xu, P. R.-D. del Castillo, and S. van der Zwaag, Philosophical Magazine **88**, 1825 (2008).
[47] S. Dudiy and A. Zunger, Physical Review Letters **97** (2006).
[48] S. M. Anijdan, A. Bahrami, H. M. Hosseini, and A. Shafyei, Materials and Design **27**, 605 (2006).
[49] G. Hautier, C. C. Fischer, A. Jain, T. Mueller, and G. Ceder, Chemistry of Materials **22**, 3762 (2010).
[50] N. Chakraborti, International Materials Reviews **49**, 246 (2004).
[51] Z.-K. Liu, L.-Q. Chen, and K. Rajan, Journal of Materials: Integrated Computational Materials , 42 (2006).
[52] G. A. Landrum and H. Genin, Journal of Solid State Chemistry **176**, 587 (2003).
[53] A. Oliynyk, E. Antono, T. Sparks, L. Ghadbeigi, M. Gaultois, B. Meredig, and A. Mar, Chemistry of Materials **28**, 7324 (2016).
[54] A. Antonysamy, J. Meyer, and P. Prangell, Materials Characterization **84**, 153 (2013).
[55] G. Strano, L. Hao, R. M. Everson, and K. E. Evans, Journal of Material Processing Technology **21**, 589 (2013).
[56] S. Bontha, N. W. Klingbeil, P. A. Kobryn, and H. L. Fraser, Materials Science and Engineering A **513-514**, 311 (2009).
[57] P. Nie, O. Ojo, and Z. Li, Acta Materialia **77**, 85 (2014).
[58] Y. Chen, F. Lu, K. Zhang, P. Nie, S. R. E. Hosseini, K. Feng, and Z. Li, Journal of Alloys and Compounds **670**, 312 (2016).
[59] B. Baufeld, E. Brandl, and O. Van Der Biest, Journal of Materials Processing Technology **211**, 1146 (2011).
[60] S. Bontha, N. W. Klingbeil, P. A. Kobryn, and H. L. Fraser, Journal of Materials Processing Technology **178**, 135 (2006).
[61] Y. Li and D. Gu, Materials and Design **63**, 856 (2014).
[62] J. Cherry, H. M. Davies, S. Mehmood, N. P. Lavery, S. G. R. Brown, and J. Sienz, International Journal of Advanced Manufacturing Technology **76**, 869 (2015).
[63] Q. Jia and D. Gu, Journal of Alloys and Compounds **585**, 713 (2014).
[64] J. Delgado, J. Ciurana, and C. A. Rodriguez, International Journal of Advanced Manufacturing Technology **60**, 601 (2012).
[65] A. M. Khorasani, I. Gibson, U. S. Awan, and A. Ghaderi, Additive Manufacturing (2018).
[66] A. S. Wu, D. W. Brown, M. Kumar, G. F. Gallegos, and W. E. King, Metallurgical and Materials Transactions A: Physical Metallurgy and Materials Science **45**, 6260 (2014).
[67] E. R. Denlinger, J. C. Heigel, P. Michaleris, and T. Palmer, Journal of Materials Processing Technology **215**, 123 (2015).
[68] C. E. Shannon, The Bell System Technical Journal **27**, 379 (1948), arXiv:9411012 [chao-dyn].
[69] B. Efron, Journal of the Royal Statistical Society Series B (Methodological) , 83 (1992).
[70] B. Efron, Journal of the American Statistical Association **109**, 991 (2014).
[71] S. Wager, T. Hastie, and B. Efron, Journal of Machine Learning Research **15**, 1625 (2014), arXiv:1311.4555v2.
[72] J. Ling, M. Hutchinson, E. Antono, B. DeCost, E. A. Holm, and B. Meredig, Materials Discovery **10**, 19 (2017).
[73] M. Gaultois, A. Oliynyk, A. Mar, T. Sparks, G. Mulholland, and B. Meredig, APL Materials **4**, 053213 (2016).
[74] J. Carrete, W. Li, N. Mingo, S. Wang, and S. Curtarolo, Phys. Rev. X **4**, 011019 (2014).
[75] C. Kim, G. Pilania, and R. Ramprasad, Chemistry of Materials **28**, 1304 (2016).
[76] S. Shan and G. G. Wang, Structural and Multidisciplinary Optimization **41**, 219 (2010).
[77] A. T. Gaynor and J. K. Guest, Structural and Multidisciplinary Optimization **54**, 1157 (2016).
[78] M. Langelaar, Additive Manufacturing **12**, 60 (2016).
[79] M. Langelaar, Structural and Multidisciplinary Optimization **55**, 871 (2017).
[80] T. Zegard and G. H. Paulino, Structural Multidisciplinary Optimization **53**, 175 (2016).
[81] E. R. Homer, Computational Materials Science **161**, 244 (2019).
[82] E. Toyserkani, A. Khajepour, and S. Corbin, Optics and Lasers in Engineering **41**, 849 (2004).
[83] V. Manvatkar, A. De, and T. DebRoy, Journal of Applied Physics **116** (2014).
[84] D. Dai and D. Gu, Materials and Design **55**, 482 (2014).
[85] L.-Q. Chen, Annual Review of Materials Research **32**, 113 (2002).
[86] X. Gong and K. Chou, Journal of Materials **67**, 1176 (2015).
[87] S. Sahoo and K. Chou, Additive Manufacturing **9**, 14 (2016).
[88] P. Michaleris, Finite Elements in Analysis and Design **86**, 51 (2014).
[89] J. Ding, P. Colegrove, J. Mehnen, S. Ganguly, P. S. Almeida, F. Wang, and S. Williams, Computational Materials Science **50**, 3315 (2011).
[90] N. Raghavan, R. Dehoff, S. Pannala, S. Simunovic, M. Kirka, J. Turner, N. Carlson, and S. S. Babu, Acta Materialia **112**, 303 (2016).
[91] R. Martukanitz, P. Michaleris, T. Palmer, T. DebRoy, Z.-K. Liu, R. Otis, T. W. Heo, and L.-Q. Chen, Addtive Manufacturing **1**, 52 (2014).
[92] M. Francois, A. Sun, W. King, N. Henson, D. Tourret, C. Bronkhorst, N. Carlson, C. Newman, T. Haut, J. Bakosi, J. Gibbs, V. Livescu, S. Vander Wiel, A. Clarke, M. Schraad, T. Blacker, H. Lim, T. Rodgers, S. Owen, F. Abdeljawad, J. Madison, A. Anderson, J.-L. Fattebert, R. Ferencz, N. Hodge, S. Khairallah, and O. Walton, Current Opinion in Solid State and Materials Science , 1 (2017).
[93] W. King, A. Anderson, R. Ferencz, N. Hodge, C. Kamath, and S. Khairallah, Materials Science and Technology **31** (2015).
[94] M. Gouge, P. Michaleris, E. Denlinger, and J. Irwin, *Thermo-Mechanical Modeling of Additive Manfuactur-*

ing, Chapter 2: The Finite Element Method for the Thermo-Mechanical Modeling of Additive Manufacturing Processes (Elsevier Inc, 2018).
[95] J. A. Flores-Livas, A. Sanna, and S. Goedecker, Novel Superconducting Materials **3**, 6 (2017).
[96] M. Rupp, A. Tkatchenko, K.-R. Müller, and O. A. von Lilienfeld, Physical Review Letters **208**, 1 (2011), arXiv:1109.2618.
[97] J. C. Snyder, M. Rupp, K. Hansen, K. R. Müller, and K. Burke, Physical Review Letters **108**, 1 (2012), arXiv:1112.5441.
[98] C. Kamath, International Journal of Advanced Manufacturing Technology **10** (2016).
[99] R. R. Dehoff, M. M. Kirka, W. J. Sames, H. Bilheux, A. S. Tremsin, L. E. Lowe, and S. S. Babu, Materials Science and Technology **31** (2015).
[100] S. Curtarolo, D. Morgan, K. Persson, J. Rodgers, and G. Ceder, Physical Review Letters **91**, 135503 (2003), arXiv:0307262 [cond-mat].
[101] K. Rajan, C. Suh, and P. F. Mendez, Statistical Analysis and Data Mining **1**, 361 (2009).
[102] G. Tapia, S. A. Khairallah, M. Matthews, and W. E. King, International Journal of Advanced Manufacturing Technology **10** (2017).
[103] W. E. King, H. D. Barth, V. M. Castillo, G. F. Gallegos, J. W. Gibbs, D. E. Hahn, C. Kamath, and A. M. Rubenchik, Journals of Materials Processing Technology **214**, 2915 (2014).
[104] M. A. Bessa, R. Bostanabad, Z. Liu, A. Hu, D. W. Apley, C. Brinson, W. Chen, and W. K. Liu, Computer Methods in Applied Mechanics and Engineering **320**, 633 (2017).
[105] Y. Jin and B. Sendhoff, IEEE Transactions of Systems, Man, and Cybernetics - Parc C: Applications and Reviews **38** (2008).
[106] B. Meredig, A. Agrawal, S. Kirklin, J. E. Saal, J. W. Doak, A. Thompson, K. Zhang, A. Choudhary, and C. Wolverton, Physical Review B - Condensed Matter and Materials Physics **89**, 1 (2014).
[107] S. Berumen, F. Bechmann, S. Lindner, J.-P. Kruth, and T. Craeghs, Physics Procedia **5**, 617 (2010).
[108] G. Tapia and A. Elwany, Journal of Manufacturing Science and Engineering **136**, 060801 (2014).
[109] M. Mani, B. M. Lane, M. A. Donmez, S. C. Feng, and S. P. Moylan, International Journal of Production Research **55** (2017).
[110] J. T. McKeown, K. Zweiacker, C. Liu, D. R. Coughlin, A. J. Clarke, J. K. Baldwin, J. W. Gibbs, J. D. Roehling, S. D. Imhoff, P. J. Gibbs, D. Tourret, J. M. Wiezorek, and G. H. Campbell, JOM **68** (2016).
[111] U. S. Bertoli, G. Guss, S. Wu, M. J. Matthews, and J. M. Schoenung, Materials and Design **135**, 385 (2017).
[112] J. Raplee, A. Plotkowski, M. M. Kirka, R. Dinwiddie, A. Okello, R. R. Dehoff, and S. S. Babu, Scientific Reports **7**, 1 (2017).
[113] D. G. Lowe, International Journal of Computer Vision **60**, 91 (2004).
[114] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, Computer Vision and Image Understanding **110**, 346 (2008).
[115] B. Yuan, G. M. Guss, A. C. Wilson, S. P. Hau-Riege, P. J. DePond, S. McMains, M. J. Matthews, and B. Giera, Adv. Mater. Technol. , 1 (2018).
[116] L. Scime and J. Beuth, Additive Manufacturing **24**, 273 (2018).
[117] O. Kwon, H. G. Kim, M. J. Ham, W. Kim, G.-H. Kim, J.-H. Cho, N. I. Kim, and K. Kim, Journal of Intelligent Manufacturing (2018).
[118] B. L. DeCost and E. A. Holm, Computational Materials Science **126**, 438 (2017).
[119] J. Zhou, Y. Zhang, and J. Chen, Journal of Manufacturing Science and Engineering **131**, 1 (2009).
[120] C. Boley, S. Khairallah, and A. Rubenchik, Applied Optics **54**, 2477 (2015).
[121] B. L. DeCost, H. Jain, A. D. Rollett, and E. A. Holm, Jom **69**, 456 (2017).
[122] B. L. Decost and E. A. Holm, Computational Materials Science **110**, 126 (2015).
[123] B. L. DeCost, T. Francis, and E. A. Holm, Acta Materialia **133**, 30 (2017), arXiv:1702.01117.
[124] D. S. Bulgarevich, S. Tsukamoto, T. Kasuya, M. Demura, and M. Watanabe, Scientific Reports **8**, 1 (2018).
[125] S. Miyazaki, M. Kusano, D. S. Bulgarevich, S. Kishimoto, A. Yumoto, and M. Watanabe, Materials Transactions (2019).
[126] A. Chowdhury, E. Kautz, B. Yener, and D. Lewis, Computational Materials Science **123**, 176 (2016).
[127] X. D. Xiang, X. Sun, G. Briceno, Y. Lou, K.-A. Wang, H. Chang, W. G. Wallace-Freedman, S.-W. Chen, and P. G. Schultz, Science **268**, 1738 (1995).
[128] C. J. Long, J. Hattrick-Simpers, M. Murakami, R. C. Srivastava, I. Takeuchi, V. L. Karen, and X. Li, Review of Scientific Instruments **78** (2007), 10.1063/1.2755487.
[129] C. J. Long, D. Bunker, X. Li, V. L. Karen, and I. Takeuchi, Review of Scientific Instruments **80** (2009), 10.1063/1.3216809.
[130] A. G. Kusne, T. Gao, A. Mehta, L. Ke, M. C. Nguyen, K.-M. Ho, V. Antropov, C.-Z. Wang, M. J. Kramer, C. Long, and I. Takeuchi, Scientific Reports **4**, 6367 (2015).
[131] G. Ceder, Y.-M. Chiang, D. R. Sadoway, M. K. Aydinol, Y.-I. Jang, and B. Huang, Nature **392**, 694 (1998).
[132] G. Pilania, C. Wang, X. Jiang, S. Rajasekaran, and R. Ramprasad, Scientific Reports **3**, 2810 (2013).
[133] M. J. Hampden-Smith and T. T. Kodas, Chemical Vapor Deposition **1** (1995).
[134] B. Mercey, P. A. Salvador, W. Prellier, T.-D. Doan, J. Wolfman, J.-F. Hamet, M. Hervieu, and B. Raveau, Journal of Materials Chemistry **9**, 233 (1999).
[135] D. B. Mitzi, Chemical Materials **13**, 3283 (2001).
[136] D. M. Deaven and K. M. Ho, Physical Review Letters **75**, 288 (1995), arXiv:9506004 [mtrl-th].
[137] S. M. Woodley, P. D. Battle, J. D. Gale, and C. Richard A. Catlow, Physical Chemistry Chemical Physics **1**, 2535 (1999).
[138] B. G. Sumpter and D. W. Noid, Annual Review of Materials Science **26**, 223 (1996).
[139] S. Curtarolo, D. Morgan, and G. Ceder, Calphad: Computer Coupling of Phase Diagrams and Thermochemistry **29**, 163 (2005), arXiv:0502465 [cond-mat].
[140] A. N. Kolmogorov and S. Curtarolo, Physical Review B - Condensed Matter and Materials Physics **73**, 1 (2006), arXiv:0603304 [cond-mat].
[141] K. Kang, Y. S. Meng, J. Breger, C. P. Grey, and G. Ceder, Science **311**, 977 (2006).
[142] H. Chen, G. Hautier, A. Jain, C. Moore, B. Kang, R. Doe, L. Wu, Y. Zhu, Y. Tang, and G. Ceder, Chemistry of Materials **24**, 2009 (2012).
[143] A. Mannodi-Kanakkithodi, G. Pilania, T. D. Huan,

T. Lookman, and R. Ramprasad, Nature Scientific Reports **6**, 1 (2016).
[144] K. T. Butler, D. W. Davies, H. Cartwright, O. Isayev, and A. Walsh, Nature Reviews **559** (2018).
[145] G. Hautier, A. Jain, T. Mueller, C. Moore, S. P. Ong, and G. Ceder, Chemistry of Materials **25**, 2064 (2013).
[146] J. W. Bennett, K. F. Garrity, K. M. Rabe, and D. Vanderbilt, Physical Review Letters **109**, 1 (2012), arXiv:1206.4732v1.
[147] C. V. Ciobanu, D. T. Tambe, and V. B. Shenoy, Surface Science **582**, 145 (2005).
[148] S. Wang, Z. Wang, W. Setyawan, N. Mingo, and S. Curtarolo, Physical Review X **1**, 1 (2011).
[149] J. Yan, P. Gorai, B. Ortiz, S. Miller, S. A. Barnett, T. Mason, V. Stevanovic, and E. S. Toberer, Energy & Environmental Science **8**, 983 (2015).
[150] D. Morgan, G. Ceder, and S. Curtarolo, Measurement Science and Technology **16**, 296 (2005).
[151] J. Behler, International Journal of Quantum Chemistry **115**, 1032 (2015).
[152] L. M. Ghiringhelli, J. Vybiral, S. V. Levechenko, C. Draxl, and M. Scheffler, Physical Review Letters **114** (2015).
[153] (2019).
[154] .
[155] A. Jain, G. Hautier, C. J. Moore, S. Ping Ong, C. C. Fischer, T. Mueller, K. A. Persson, and G. Ceder, Computational Materials Science **50**, 2295 (2011).
[156] S. Curtarolo, W. Setyawan, G. L. W. Hart, M. Jahnatek, R. V. Chepulskii, R. H. Taylor, S. Wang, J. Xue, K. Yang, O. Levy, M. J. Mehl, H. T. Stokes, D. O. Demchenko, and D. Morgan, Computational Materials Science **58**, 218 (2012), arXiv:1308.5715.
[157] J. Hachmann, R. Olivares-Amaya, S. Atahan-Evrenk, C. Amador-Bedolla, R. S. Sánchez-Carrera, A. Gold-Parker, L. Vogt, A. M. Brockway, and A. Aspuru-Guzik, Journal of Physical Chemistry Letters **2** (2011).
[158] "National institute of materials science," (2017).
[159] J. E. Saal, S. K. nd Murathan Aykol, B. MEredig, and C. Wolverton, JOM **65** (2013).
[160] I. Foster, R. Ananthakrisnan, B. Blaiszik, K. Chard, R. Osborn, S. Tuecke, M. Wilde, and J. Wozniak, Big Data and HPC (2015).
[161] J. J. De Pablo, B. Jones, C. L. Kovacs, V. Ozolins, and A. P. Ramirez, Current Opinion in Solid State and Materials Science **18**, 99 (2014), arXiv:arXiv:1011.1669v3.
[162] M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomber, J.-W. Boiten, L. B. da Silva Santos, P. E. Bourne, J. Bouwman, A. J. Brookes, T. Clark, M. Crosas, I. Dillo, O. Dumon, S. Edmunds, C. T. Evelo, R. Finkers, A. Gonzalez-Beltran, A. J. Gray, P. Groth, C. Goble, J. S. Grethe, J. Heringa, P. A. 't Hoen, R. Hooft, T. Kuhn, R. Kok, J. Kok, S. J. Lusher, M. E. Martone, A. Mons, A. L. Packer, B. Persson, P. Rocca-Serra, M. Roos, R. van Schaik, S.-A. Sansone, E. Schultes, T. Sengstag, T. Slater, G. Strawn, M. A. Swertz, M. Thompson, J. van der Lei, E. van Mulligen, J. Velterop, A. Waagmeester, P. Wittenburg, K. Wolstencroft, J. Zhao, and B. Mons, Scientific Data **3** (2016).
[163] M. de Jong, W. Chen, H. Geerlings, M. Asta, and K. A. Persson, Scientific Data **2**, 150053 (2015).
[164] E. Kim, K. Huang, A. Tomala, S. Matthews, E. Strubell, A. Saunders, A. McCallum, and E. Olivetti, Nature Scientific Data **4** (2017).
[165] S. R. Kalidindi, D. B. Brough, S. Li, A. Cecen, A. L. Blekh, F. Y. P. Congo, and C. Campbell, Materials Research Society Bulletin **41** (2016).
[166] R. Ramprasad, R. Batra, G. Pilania, A. Mannodi-Kanakkithodi, and C. Kim, Nature Computational Materials **3** (2017).
[167] J. Gubernatis and T. Lookman, Physical Review Materials **2** (2018).
[168] N. Wagner and J. Rondinelli, Frontiers in Materials **3** (2016).