

ARTICLE OPEN



An AI-driven microstructure optimization framework for elastic properties of titanium beyond cubic crystal systems

Yuwei Mao¹, Mahmudul Hasan², Arindam Paul¹, Vishu Gupta¹, Kamal Choudhary^{1,3,4}, Francesca Tavazza^{1,3}, Wei-keng Liao¹, Alok Choudhary¹, Pinar Acar^{1,2} and Ankit Agrawal¹✉

Materials design aims to identify the material features that provide optimal properties for various engineering applications, such as aerospace, automotive, and naval. One of the important but challenging problems for materials design is to discover multiple polycrystalline microstructures with optimal properties. This paper proposes an end-to-end artificial intelligence (AI)-driven microstructure optimization framework for elastic properties of materials. In this work, the microstructure is represented by the Orientation Distribution Function (ODF) that determines the volume densities of crystallographic orientations. The framework was evaluated on two crystal systems, cubic and hexagonal, for Titanium (Ti) in Joint Automated Repository for Various Integrated Simulations (JARVIS) database and is expected to be widely applicable for materials with multiple crystal systems. The proposed framework can discover multiple polycrystalline microstructures without compromising the optimal property values and saving significant computational time.

npj Computational Materials (2023)9:111; <https://doi.org/10.1038/s41524-023-01067-8>

INTRODUCTION

Understanding the relationship between processing, structure, properties, and performance (PSPP) is critical in material science^{1–7}. The relationships from processing to performance are the cause-effect relationships, which are forward problems, e.g., property prediction. The relationships from performance to processing are goals-means relationships, that are inverse problems, e.g., microstructure optimization. Variation in microstructure leads to a wide range of material properties, which in turn impacts the performance. Thus, optimization of the microstructure can significantly improve the materials' performance, accelerate materials discovery, and help us design new materials with target engineering properties.

Titanium (Ti) alloys are widely used in many engineering fields, such as aerospace, chemical, biomedical and marine applications. Optimizing the property of Ti is necessary for improving the safety and performance of associated devices and machinery^{8–11}. Generally, pure Ti can crystallize in two crystal systems: α titanium and β titanium. When it crystallizes at low temperatures (room temperature), the hexagonal close-packed (HCP) structure of alpha titanium is formed. While it crystallizes at high temperatures, the body-centered cubic (BCC) structure of beta titanium is formed. In this study, we use the pure Ti datasets in the Joint Automated Repository for Various Integrated Simulations (JARVIS) database¹² as an application. The JARVIS is an integrated infrastructure to accelerate materials discovery and design using density functional theory (DFT), classical force-fields (FF), and machine learning (ML) techniques, that are publicly available at the website: <https://jarvis.nist.gov>. The Ti dataset used in this work includes one cubic material and six hexagonal materials.

One of the challenges of microstructure optimization is multi-objective design. The microstructure that provides the maximum value of the desired property (e.g., stiffness C_{11}) may not be an optimal solution for another property (e.g., C_{12}). One of the major goals of materials design optimization is the trade-off of

properties based on prioritizing one design goal over others^{13,14}. Selection of preferred orientations of various crystals constituting the polycrystalline materials is a popular technique, which allows tailoring of properties of polycrystalline alloys^{15–17}. A polycrystalline material is made up of many crystals, each of which has its unique crystallographic orientation, which determines the microstructural texture. The Orientation Distribution Function (ODF) is used to characterize the microstructure in this work, which describes the volume density of each unique crystal orientation. In this work, we aim to discover the optimum (maximum and minimum) elastic property values along with the corresponding microstructures defined in terms of the ODFs.

The desired properties of this work are the elastic properties stiffness (C_{11}) and Young's modulus (E_{11}), which can be computed using the microstructure homogenization expression (for example, $C_{11} = P_{11}^T \mathbf{A}$, where P_{11} is the property matrix of the single-crystal values for C_{11} , \mathbf{A} is the column vector of the ODF values assigned to the independent nodal points of the microstructure mesh). The Young's modulus (E_{11}), on the other hand, is inversely related to the stiffness when upper bound averaging is used as it is given by $E_{11} = \frac{1}{S_{11}}$, where $S_{11} = S(1, 1)$ while \mathbf{S} is the compliance matrix defined as $\mathbf{S} = \mathbf{C}^{-1}$. Therefore, the relationship between Young's modulus and ODF is non-linear because Young's modulus is found by inverting the homogenized stiffness matrix obtained with the linear homogenization expression. More details can be found in supplementary material. Moreover, the ODF should satisfy the normalization constraint, which is defined as: $\mathbf{q}^T \mathbf{A} = 1$. The normalization constraint is mathematically equivalent to the fact that the sum of the probabilities for having all crystallographic orientations is one. Here, \mathbf{q} is a constant column vector that is obtained from the finite element discretization. The present study describes the microstructure modeling briefly. Interested readers are referred to ref. ¹⁸ for more information.

Researchers have developed and used different techniques to discover microstructures with optimal material properties. A linear

¹Department of Electrical and Computer Engineering, Northwestern University, Evanston, IL, USA. ²Virginia Tech, Blacksburg, VA, USA. ³Materials Measurement Laboratory, National Institute of Standards and Technology, Gaithersburg, MD 20899, USA. ⁴Theiss Research, La Jolla, CA 92037, USA. ✉email: ankitag@eecs.northwestern.edu

programming algorithm was used to discover microstructural textures with optimal properties using the idea of building a reduced-order design space, called the property closure^{3,18,19}. Acar et al.¹⁸ used this approach to find the best microstructure design of an airframe panel for obtaining the maximum buckling temperature. This process was extended to find the maximum yield strength of the Galfenol alloy while considering the constraints for the vibration¹⁹. This approach was able to save significant computational time but can only discover one optimal solution and most discovered solutions are single crystals or near single crystals (only one dimension has a significant non-zero value). However, the polycrystalline microstructures are known to be advantageous over single-crystal designs in terms of cost, performance, ease of manufacturing, homogeneity and good control over composition²⁰. Moreover, multiple microstructures could improve the manufacturability of the microstructures. Paul et al.²¹ designed two data sampling algorithms to sample the entire microstructure space with vibrational tuning constraints for galfenol alloy. This method could discover multiple microstructures and may discover polycrystals. But these data sampling algorithms are time-consuming, and they are not effective for discovering maximum and minimum properties.

With the rapid development of artificial intelligence and machine learning (AI/ML) techniques^{22–28}, and the increasing availability of data from the first three paradigms of science (experiments, theory, and simulations), the fourth paradigm of science, i.e., data-driven science and discovery, is playing an important role in materials science². Many machine learning and deep learning techniques have been extensively used in materials science^{22,29–33} to enhance materials property prediction, discovery, and design. Recent research has demonstrated the potential of machine learning (ML) techniques to predict and optimize the properties of titanium (Ti) alloys. McElfresh et al.³⁴ used a set of machine learning techniques to develop predictive tools relating the yield strength and hardening rate of Ti-6Al-4V alloys to a set of input parameters covering extensive ranges. Liu et al.³⁵ proposed a machine learning method to predict microstructures and Young's moduli of biomedical β -Ti alloys, which can accelerate the design process of such alloys with low moduli, and successfully develops a Ti-13Nb-12Ta-10Zr-4Sn alloy with desired properties. Zou et al.³⁶ use data mining and machine learning approaches to reveal the atomic and electronic insights of the composition-structure-property relationships to improve the strength and ductility of Titanium (Ti) alloys.

There are several works on the application of machine learning methods for microstructure optimization. Liu et al.^{37,38} used four random data sampling algorithms and combined them with machine learning methods for the optimization of five design objectives. Paul et al.³⁹ used a machine learning-based feedback-aware data generation algorithm to discover optimal microstructures for HCP Ti alloy with constrained design objectives stiffness and compliance. Hasan et al.⁴⁰ used gradient-based and ML-based methods to enhance homogenized linear and non-linear properties of cubic microstructures. These ML-based methods used data sampling algorithms to randomly generate data for the machine learning model. Then random forest is used for microstructure search space reduction to identify a minimal subset of ODF dimensions. However, these ML-based methods are time-consuming and need more human knowledge because they are not providing an end-to-end framework, and there are many hyperparameters in different steps. Inadequate choice of hyperparameters and insufficient run time may not get optimal solutions.

In this work, an end-to-end AI-driven optimization framework is proposed to explore the microstructure optimization problems for elastic properties of materials with multiple crystal systems. We take Ti as an application, which includes two crystal systems, cubic and hexagonal in the JARVIS database. The desired properties are

stiffness C_{11} and Young's modulus E_{11} . Single crystal solutions could provide optimal properties, but polycrystalline designs are advantageous over single crystals in terms of better manufacturability. The gradient-based method could provide optimal solutions but could only find one solution for one material, and some solutions are near-single crystal solutions (only one dimension has a significant non-zero value) for some materials. Although previous ML-based methods⁴⁰ were able to discover multiple polycrystalline solutions, the running time was very high and the solutions were usually not optimal solutions. Compared to these methods, the proposed end-to-end AI-driven framework could discover multiple polycrystalline solutions without compromising the optimal property values. The present work builds upon the previous ML-based methods for microstructure design, and could be implemented end-to-end for multiple materials with different crystal systems. And the running time of the proposed framework is significantly less than the previous ML-based methods⁴⁰ while getting better properties.

Example application of microstructure design for a rotating beam

Ti alloys with hexagonal crystal structures have applications in aerospace systems, such as in turbine blades. In this application, a blade is modeled as a rotating cantilever beam. The rotation causes an in-plane axial load (P) as demonstrated in Fig. 1. The effects of this axial load can be included with the geometric stiffness. Accordingly, the effective stiffness (k_e) of the beam is given by $[k_e] = [k_b] - [k_g]$, where k_b denotes bending (flexural) stiffness and k_g is the geometric stiffness under the axial (compression) load. In this formulation, the effective stiffness is a function of Young's modulus and the axial load. The critical buckling load is solved as equal to the axial load ($P_{cr} = P$) when the following condition is satisfied: $\det[k_e] = 0$. As demonstrated in the previous work of the authors⁴¹, the critical buckling load (P_{cr}) is maximized when Young's modulus, E_{11} , is maximized. Therefore, the presented formulation in this work for the maximization of Young's modulus can also be used to improve the buckling performance of engineering systems, such as aerospace components.

RESULTS

Overview of AI-driven microstructure optimization framework

Figure 2 presents the flow-diagram for the proposed optimization framework. The framework inputs include constant column vector (i.e., \mathbf{q} vector) and a nodal point property matrix (i.e., \mathbf{P} matrix). The framework is composed of three data sampling algorithms. These data sampling algorithms' objective is to generate instances of microstructure representations, i.e., the multidimensional ODFs. Here, the ODF could be represented by the vector $\mathbf{x} = [x_1, x_2, \dots, x_D] \subseteq R^D$ and needs to satisfy the constraints $\sum_{i=1}^D x_i \times q_i = 1$ and $x_i \geq 0$. The details of the proposed framework are introduced in the Method section.

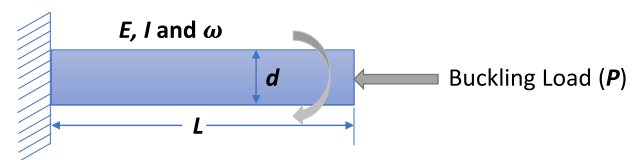


Fig. 1 Schematic of the rotating beam. The rotation of the beam translates into a compressive axial force that can cause the failure of the system due to buckling. E denotes Young's modulus (equal to E_{11}), I is the moment of inertia, ω is the rotation frequency, L is the length of the beam, and d is the diameter of the circular cross-section of the beam.

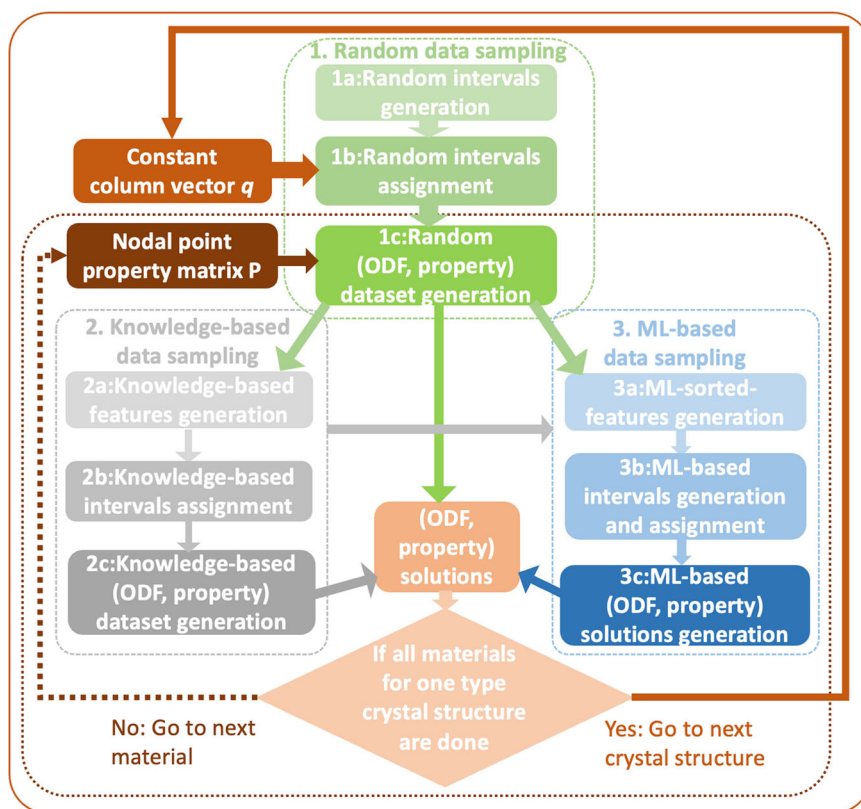


Fig. 2 Flow-diagram for the proposed AI-driven end-to-end optimization framework. The framework inputs include constant column vector (i.e., q vector) and a nodal point property matrix (i.e., P matrix). The framework is composed of three data sampling algorithms, whose objective is to generate instances of microstructure representations, i.e., the multidimensional ODFs.

The goal of the proposed end-to-end AI-driven microstructure optimization framework is generating multiple optimal and near-optimal polycrystalline solutions. Ti data in JARVIS is used as an application in this study, which includes two crystal systems, cubic and hexagonal. There are one cubic Ti material and seven hexagonal Ti materials. The cubic Ti has 76 independent ODF values, and the hexagonal Ti has 50 independent ODF values. The objective properties are linear property value stiffness (C_{11}) and non-linear property value in-plane Young's modulus (E_{11}).

We use best-of-|ODF| method, gradient-based optimization and the previous ML-based optimization method⁴⁰ for comparison. The best-of-|ODF| method calculates every single crystal solution's properties. Then, the maximum and minimum properties and the corresponding single crystal microstructures are selected as the optimal solutions. The gradient-based method uses the sequential quadratic programming (SQP) algorithm and is applied to solve the optimization problem. We use the `fmincon` function in Matlab to implement it. The hyperparameters of the previous ML-based method are the same as in ref.⁴⁰, and it has a running time of about one week. In comparison, the proposed AI-driven framework takes only a couple of days to discover more optimal solutions, as described next.

Optimality and polycrystallinity

Table 1 shows the solutions discovered by different methods with optimal values boldfaced. Different font styles in the tables represent the crystallinity (underline represents single crystal and italic represents polycrystal). We consider the values within 0.01% of the numerically optimal value to also be optimal for practical purposes. Thus bolditalic values are desirable.

Table 1 shows the optimal E_{11}^{min} and E_{11}^{max} solutions obtained from different methods. The gradient-based method discovers the

optimal property values in this table. The best-of-|ODF| method, gradient-based method, and the proposed AI-driven framework could all discover the optimal E_{11}^{min} . However, the solutions of the best-of-|ODF| and the gradient-based method for all materials are single crystal solutions. Both the proposed AI-driven framework and the previous ML-based method could discover polycrystalline solutions for all materials. As mentioned before, polycrystals are better because they are advantageous for manufacturing. However, the property values of ODFs discovered using the previous ML-based method are less optimal than those from the proposed AI-driven method. The proposed AI-driven framework thus can discover solutions that are both optimal as well as polycrystalline. Similar to what was observed for E_{11}^{min} optimal target, the gradient-based method discovers the optimal property values, and both the best-of-|ODF| method and the proposed AI-driven framework could also discover the optimal E_{11}^{max} value. The optimal E_{11}^{max} solutions for JARVIS IDs 14732 and 84837 are single-crystal solutions as discovered by the gradient-based method, while others are polycrystal solutions. Both the proposed AI-driven framework and the previous ML-based method could discover polycrystalline solutions for all materials, but the ones from the AI-driven framework are significantly higher, i.e., more optimum.

Additional observations for optimal C_{11}^{min} and C_{11}^{max} solutions are offered in Supplementary Table 1. Both the best-of-|ODF| method and the gradient-based method discover the optimal properties values, and the proposed AI-driven framework could also discover the optimal solutions within an error of 0.001%. The solutions of the gradient-based method for all materials are single crystal solutions, just like those from the best-of-|ODF| method. These tables demonstrate that the proposed AI-driven framework could consistently discover optimal polycrystalline solutions.

Table 1. Optimal E_{11} obtained using different methods.

Optimal target	Crystal System	ID	Best-of- $ ODF $ method	Gradient-based method	Previous ML-based ⁴⁰ method	Proposed AI-driven framework
E_{11}^{min}	Cubic	15716	50.2676	50.2676	60.5366	50.2697
		Hexagonal	1028	145.2473	145.2473	146.6405
	Hexagonal	1029	134.1967	134.1967	135.3297	134.1969
		1030	146.3204	146.3204	147.3352	146.3208
		14732	88.9550	88.9550	88.9552	88.9551
		14815	134.1967	134.1967	135.3297	134.1970
		84837	148.0852	148.0852	149.5147	148.0854
		84936	140.5261	140.5261	141.8266	140.5265
E_{11}^{max}	Cubic	15716	130.4218	130.9160	127.0722	130.9160
		Hexagonal	1028	192.4707	192.4747	176.4714
	Hexagonal	1029	181.6082	181.6254	165.3472	181.6254
		1030	199.4137	199.4209	181.0323	199.4209
		14732	116.5855	116.5855	106.4076	116.5850
		14815	181.6082	181.6254	165.3472	181.6254
		84837	197.7685	197.7685	181.1660	197.7682
		84936	187.4975	187.5048	171.4899	187.5048

Multiple polycrystalline solutions

Table 2 shows the number of solutions for E_{11} with different number of real non-zero dimensions (two and three) within different neighborhood thresholds of the numerically optimal value (Supplementary Table 2 shows the number of solutions for C_{11}). To calculate the number of real non-zero dimensions for an ODF $\mathbf{x} = [x_1, x_2, \dots, x_D]$, we first normalize it to get $\mathbf{x}_{norm} = [x_{norm_1}, x_{norm_2}, \dots, x_{norm_D}]$, where $x_{norm_i} = \frac{x_i}{\max(\mathbf{x})}$. Then for every dimension i , if $x_{norm_i} > 0.0001$, we consider it to be a real non-zero dimension. This is done since some of the dimensions of the obtained ODF solutions can be infinitesimally small (e.g. of the order of $1e-20$ or lower), which is practically insignificant to be counted as a real non-zero dimension.

As represented by these tables, the proposed AI-driven framework successfully discovers multiple near-optimal polycrystalline solutions for all materials, especially for two real non-zero dimensions. And when the number of non-zero dimensions is larger than three, we still can discover several solutions within 0.01% error for many material-property combinations. As mentioned earlier, it is important to discover multiple near-optimal solutions for improving the efficiency of manufacturing. The traditional low-cost manufacturing processes can only generate a limited set of microstructures. Multiple near-optimal solutions can increase the variability of optimal designs, and accelerate materials development efforts further.

Parameter and running time analysis

In the proposed end-to-end framework, the parameter T represents the number of iterations used in data sampling to generate orientation distribution function (ODF) vectors. Larger T can generate more ODF vectors and increase the number of microstructures with desired material properties. We select material ID=15716 to analyze the effect of T on the number of microstructures with the desired property value and the running time of the proposed end-to-end framework. Figure 3a–d show the number of solutions of E_{11}^{min} for different values of parameter T and neighborhood thresholds. For example, Fig. 3a we plot the number of microstructures with the property values within 0.01% error of the optimal E_{11}^{min} . As T increases from 100 to 10,000,000, the number of microstructures satisfying the desired property value increases from zero to ten. This figure shows that as T

increases, the number of solutions increases significantly, which is expected. It also demonstrates that knowledge-based data sampling is better than the random data sampling method because of using single-crystal knowledge. ML-based data sampling algorithm is better than random data sampling and knowledge-based data sampling because it builds upon these two data sampling information.

Figure 3e–h show the number of solutions of E_{11}^{max} for different values of parameter T and neighborhood thresholds. We could get similar conclusions as from the previous figure. Interestingly, we can observe that the proposed end-to-end framework could discover more optimal E_{11}^{max} solutions than optimal E_{11}^{min} solutions for this material.

Figure 4a shows the running time of different steps. Step 1a needs to be run only once for all crystal systems and all materials. Step 1b needs to be run only once for all materials with a given crystal system. Thus, the running time of these two steps could be saved when there are multiple crystal systems and multiple materials. The running times of steps 2a and 3a are close to 0% compared to the whole running time. ML-based data sampling is the most powerful and takes the longest time because it includes interval generation and assignment twice. Given its contribution to discovering multiple optimal solutions as demonstrated earlier, it is valuable to run ML-based data sampling for a relatively long time.

Figure 4b shows the number of solutions for different thresholds with different running times. Here y-axis represents the number of solutions. Clearly, by running longer, we can discover more solutions. It is also observed that the number of solutions increases rapidly when we switch to knowledge-based data sampling, and again increases rapidly when we switch to ML-based data sampling.

Optimized microstructures

Figure 5a–f show optimized microstructures in orientation space for E_{11}^{max} of cubic material ID=15716. Figure 5a–c show ODFs discovered by best-of- $|ODF|$ method, gradient-based method and previous ML-based method. Figure 5d–f show some example ODFs discovered by the proposed framework. The optimal E_{11}^{max} value is 130.9160 GPa discovered by the gradient-based method and the proposed framework. The proposed framework can discover multiple solutions that provide properties that are close

Table 2. Number of multiple polycrystalline solutions of E_{11} with different number of real non-zero dimensions.

No. of non-zero dimensions			≥ 2				≥ 3			
Optimal target	Crystal System	ID	Threshold				Threshold			
			0.01%	0.1%	0.2%	0.5%	0.01%	0.1%	0.2%	0.5%
E_{11}^{min}	Cubic	15716	10	112	227	559	0	0	0	5
		Hexagonal	1028	253	2378	4858	14488	0	68	332
	Hexagonal	1029	302	3013	6340	18668	1	83	401	2923
		1030	315	3809	8119	24225	0	100	519	3865
		14732	5937	59897	121391	319939	0	133	665	7284
		14815	679	6152	12819	37890	1	181	833	6046
		84837	813	6662	13567	39069	1	169	795	5587
		84936	998	8762	18054	54057	1	285	1301	9448
E_{11}^{max}	Cubic	15716	15785	26578	38765	58515	12	540	1511	8320
		Hexagonal	1028	459	1773	3061	6530	5	51	131
	Hexagonal	1029	621	2372	4008	8223	1	52	170	761
		1030	789	2898	4958	10390	1	69	209	955
		14732	98	2115	4399	11568	0	24	135	819
		14815	1455	5431	8999	18657	1	115	401	1797
		84837	424	4216	7979	19230	0	67	308	1666
		84936	2125	7162	11805	24658	7	176	581	2487

to the optimal property, such as E_{11}^{max} value equal to 130.2197 GPa and 129.5569 GPa in Fig. 5e, f. These results demonstrate that the proposed framework could discover multiple polycrystals that are advantageous for manufacturing.

Similar insights could be observed from Fig. 5g–l, which shows the solutions for hexagonal Ti ID=14815. Figure 5g–i show ODFs discovered by best-of-|ODF| method, gradient-based method and previous ML-based method. Figure 5j–l show some example ODFs discovered by the proposed framework. The optimal E_{11}^{max} value is 181.6254 GPa discovered by the gradient-based method and the proposed framework. And the proposed framework can discover multiple solutions that provide properties that are close to the optimal property, such as E_{11}^{max} value equal to 180.2922 GPa and 178.4170 GPa in Fig. 5k, l.

DISCUSSION

Microstructure optimization is vital for material design. Existing state-of-the-art methods can either discover one microstructure with optimum property rapidly or discover multiple polycrystalline microstructures with less optimum properties slowly. In this study, we proposed an end-to-end AI-driven microstructure optimization framework to identify two optimal properties (stiffness C_{11} and Young's modulus E_{11}) for Ti, which includes two crystal systems (cubic and hexagonal). Compared to the single crystal solutions, gradient-based optimization method, and the previous ML-based method⁴⁰, the proposed AI-driven end-to-end framework could efficiently discover multiple polycrystalline microstructures without compromising the optimal properties. The ability to discover multiple near-optimal polycrystal solutions is greatly advantageous for enhancing manufacturability. Furthermore, it is worth noting that the proposed framework has several significant advantages over previous ML-based methods⁴⁰, as demonstrated by our experimental results. Specifically, our approach incorporates three data sampling algorithms and provides an end-to-end automated framework, which sets it apart from existing methods. In addition, we have shown that by leveraging the results of the other two data sampling techniques, our ML-based data sampling method can achieve better optimization and faster runtime. Overall, these improvements have resulted in a more efficient and

effective approach to microstructure optimization. The proposed framework could thus accelerate the manufacturing of materials by increasing the variability of optimal design solutions.

This framework of microstructure optimization only considers the crystallographic orientations of the grains using ODF to represent microstructure and does not account for other microstructural features such as grain size, shape, and boundary characteristics, which can also have some impact on material properties. Therefore, in the future, we will incorporate additional microstructural features into the optimization process to develop a more comprehensive model. This framework could also be used for other materials with multiple crystal systems as well as to optimize microstructures for other properties. In the future, this framework could be further developed to discover optimal microstructures with improved or maximized polycrystallinity according to different applications. Moreover, we plan to explore more diverse polycrystallinity solutions. Another potential area of future work is to investigate the impact of manufacturing and processing conditions on the optimized microstructures. This would involve incorporating manufacturing constraints, such as grain growth and recrystallization, into the optimization process to ensure the feasibility of the optimized microstructures. The proposed framework can also be integrated with other tools and methodologies, such as experimental testing and process simulation, to further refine the design and optimization of materials for specific applications.

METHODS

Random data sampling

Random data sampling is the first data sampling algorithm and it consists of three steps. The first step is random intervals generation. In this step, the unit length 1 is randomly divided into K random intervals to generate an interval vector $\mathbf{y} = [y_1, y_2, \dots, y_K] \subseteq R^K$, where $\sum_{j=1}^K y_j = 1$. We traverse every integer from 2 to 10 as K and repeat T times for each K to generate $9 \times T$ interval vectors. These vectors are saved as an interval vectors list for the following framework steps for all materials. We can reduce the running time by reusing the generated interval vectors list for all materials of multiple crystal systems.

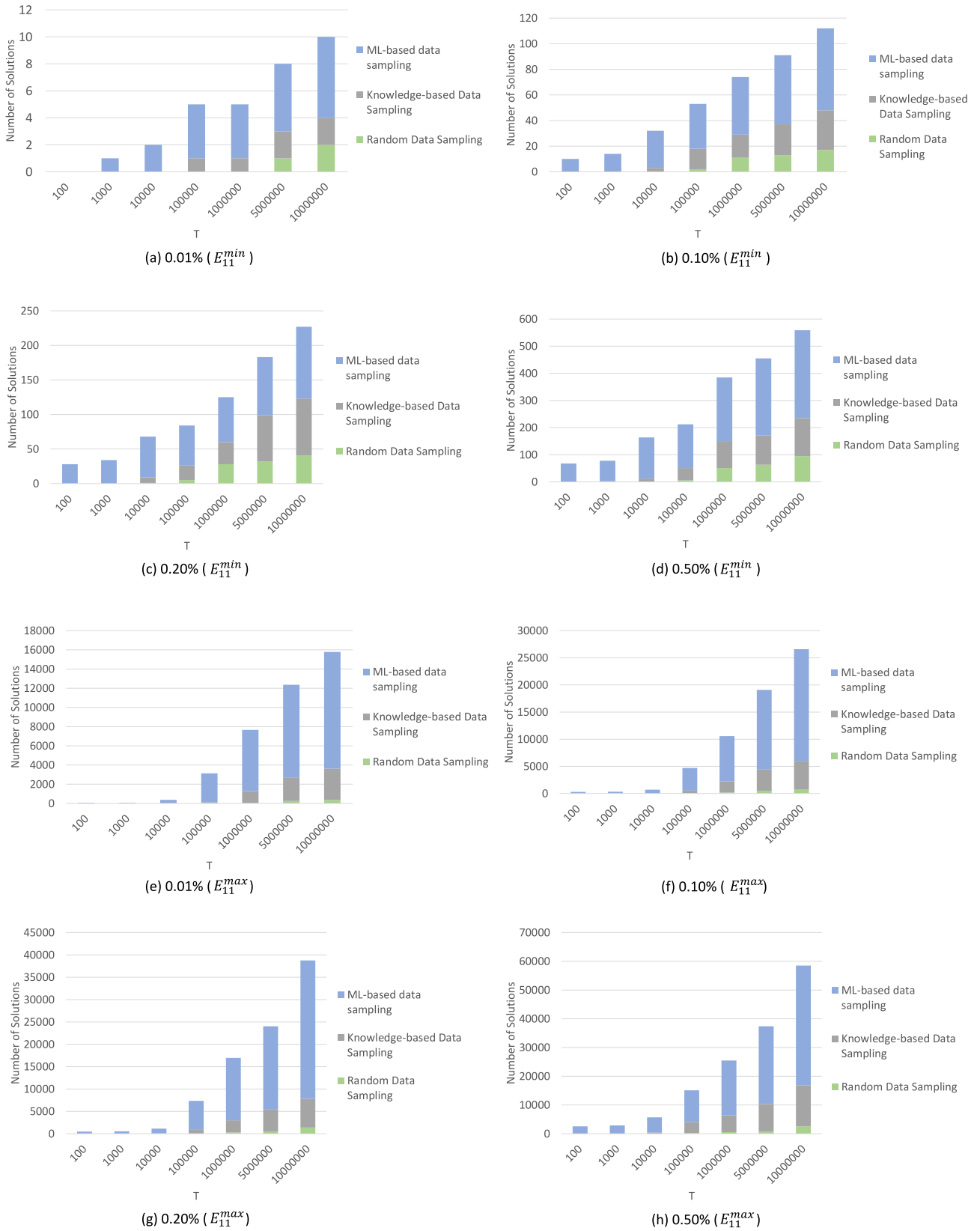
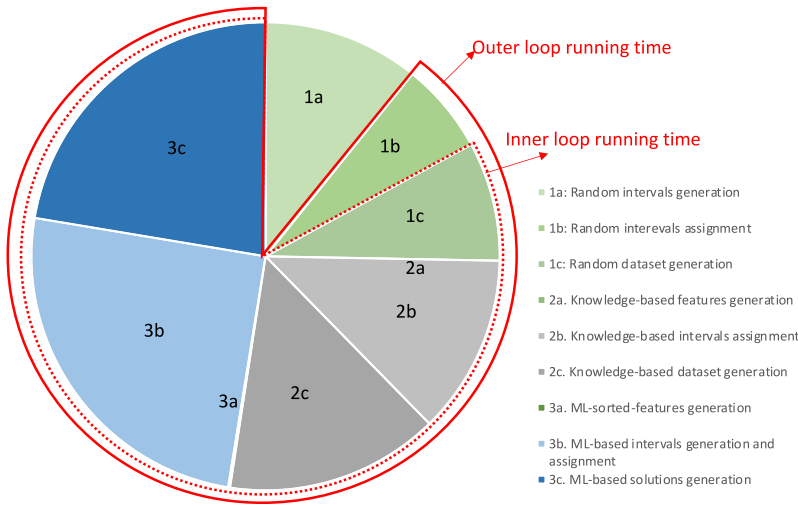
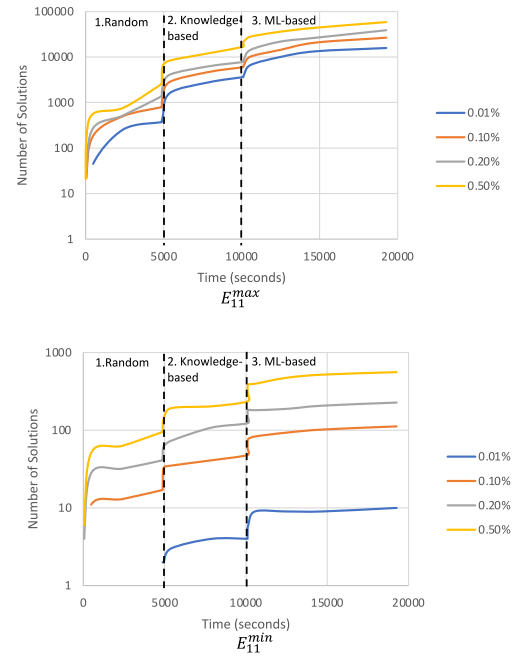


Fig. 3 Number of solutions for different number of iterations T . The neighborhood threshold is set to **a** 0.01%, **b** 0.10%, **c** 0.20%, **d** 0.50% for E_{11}^{min} and **e** 0.01%, **f** 0.10%, **g** 0.20%, **h** 0.50% for E_{11}^{max} .



(a) Running time of different steps (Steps 2a and 3a take negligible time.)



(b) Log(number of solutions)-time curves for different neighborhood thresholds.

Fig. 4 Running time analysis. **a** The running time for different steps of the proposed AI framework (Steps 2a and 3a take negligible time). **b** Log(number of solutions)-time curves for different neighborhood thresholds.

The second step is the random intervals assignment. The constant column vector (i.e., vector $\mathbf{q} = [q_1, q_2, q, \dots, q_D] \subseteq R^D$) of the current crystal system, and the interval vectors list obtained by the previous step are inputs of this step. We generate one corresponding ODF vector \mathbf{x} for each interval vector \mathbf{y} . Specifically, the dimension of \mathbf{x} is D , which is the same as that of \mathbf{q} . We traverse every value y_j ($1 \leq j \leq K$) in interval vector \mathbf{y} , and randomly assign it to i^{th} dimension in \mathbf{x} to calculate x_i by using formula $x_i = \frac{y_j}{q_i}$. Because $K < D$, only K interval values are randomly assigned to K random dimensions in ODF vector \mathbf{x} , and the values of the remaining $D - K$ dimensions are zero. The formula $x_i = \frac{y_j}{q_i}$ is to make sure the ODF vector \mathbf{x} satisfies the constraint $\sum_{i=1}^D x_i \times q_i = 1$. This step can be reused for any material in the same crystal system, thus can also reduce the time consumption for all materials in the same crystal system.

Finally, the corresponding property of the generated ODFs could be calculated using the nodal point property matrix (i.e., \mathbf{P} matrix). The (ODF, property) solutions generated in this algorithm are saved as candidates $DATA_R$, where only T_{HIGH} (ODF, property) pairs with the highest properties and T_{LOW} with the lowest properties are saved for the following steps in order to save storage space.

Knowledge-based data sampling

Knowledge-based data sampling is the second data sampling algorithm, which is an improvement of the random data sampling method. The random interval vectors list generated in the first data sampling method is also reused here. Instead of random intervals assignment, here we want to make use of single-crystal solutions' knowledge to assign intervals.

Different dimensions will have different importance for getting optimal properties. The data sampling method would be more effective if we could assign intervals to more important dimensions to get optimal properties. Single-crystal ODF is a vector that only has one non-zero dimension. The values of other dimensions are

zero. There are D single-crystal ODFs for each material. They are easy to obtain and could provide some knowledge about the dimensions' importance. We assume that if one single-crystal ODF could get more optimal properties, the corresponding non-zero dimension of this vector is more important even in polycrystal solutions. Thus, after ranking the properties of these single-crystal solutions, we could get an ordered list. Single-crystal ODFs at the top of the list could provide higher properties, and single-crystal ODFs at the bottom of the list could provide lower properties. We generate a vector $\mathbf{z} = [z_1, z_2, \dots, z_D]$ according to this list, where z_i is non-zero dimension of the i^{th} ODF in the list. We call it knowledge-based features.

After that, we reuse the interval vectors obtained by the first step of random data sampling and randomly assign each interval vector twice according to the knowledge-based features. The first one is assigning K intervals to K random dimensions from the first D_{KB} values of knowledge-based features. The second one is assigning K intervals to K random dimensions from the last D_{KB} values of knowledge-based features. Finally, we calculate the corresponding properties and save (ODF, property) pairs with the highest properties and lowest properties as the knowledge-based (ODF, property) candidates $DATA_{KB}$.

ML-based data sampling

The third data sampling is the machine learning (ML)-based data sampling algorithm, which builds upon random data sampling and knowledge-based data sampling. In this method, we generate ML-sorted features first. Then, we randomly generate intervals and assign them to ODF dimensions simultaneously. We combine the ML-based features and knowledge-based features for assignment in order to make use of both single-crystal knowledge and ML-based knowledge.

First, the ML-sorted features list is generated using the ML method. The candidates $DATA_R$ and $DATA_{KB}$ from previous data sampling could be combined as the inputs of the ML model. Each (ODF, property) pair is a sample, where the ODFs are considered as the features. We

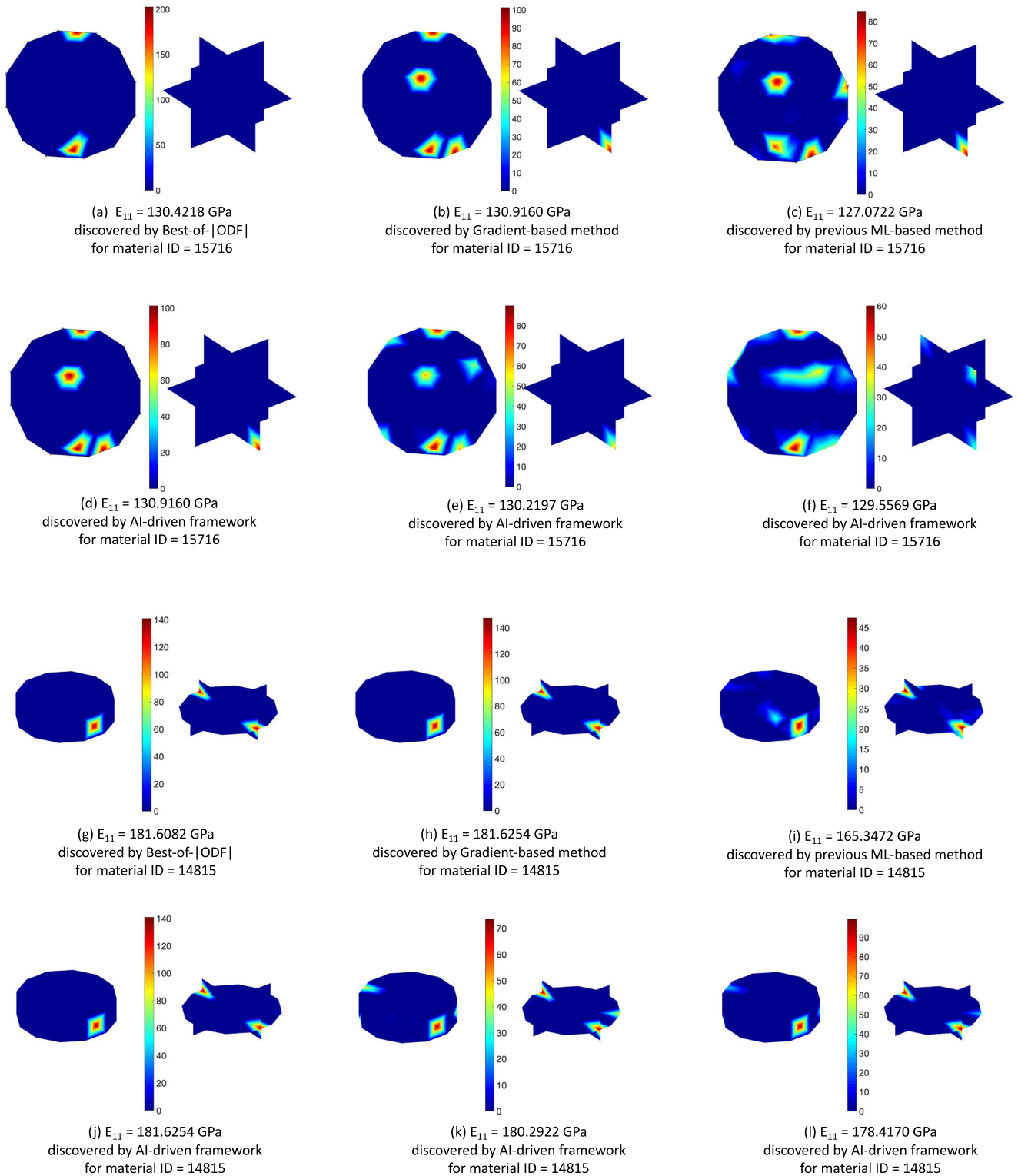


Fig. 5 Optimized E_{11}^{max} microstructures (ODFs) in the orientation space for an example of cubic material and an example of hexagonal material discovered by different methods. An example of cubic material ID = 15716 discovered by **a** Best-of-|ODF|, **b** Gradient-based method, **c** Previous ML-based method, **d-f** AI-driven framework. An example of hexagonal material ID = 14815 discovered by **g** Best-of-|ODF|, **h** Gradient-based method, **i** Previous ML-based method, **j-l** AI-driven framework.

labeled the sample with the higher property as “1” and the sample with the lower property as “-1”. This creates a two-class classification problem. Random forest-based⁴² model could be used for classification problems to learn the feature importance. In this study, we use the random forest model to rank ODF dimensions in the order of their

importance. The purpose of this step is to select the ODF dimensions that are more important for generating optimal properties. We call the obtained list that includes the order of ODF dimensions according to the importance as ML-sorted features list, where features at the top of the list are more important for predicting optimal property.

Then, we randomly generate K intervals and assign the intervals twice. The first one is assigning K intervals to K random dimensions from the first D_{ML} values of ML-sorted features and not in the first D_{KB} values of knowledge-based features. We want to generate solutions that have minimum properties for this assignment. The second one is assigning K intervals to K random dimensions from the first D_{ML} values of ML-sorted features and not in the last D_{KB} values of knowledge-based features to generate solutions that have maximum properties. This step could be repeated T times.

After that, we could calculate the corresponding properties to generate (ODF, property) solutions and save solutions with the highest properties and the lowest properties as the ML-based (ODF, property) candidates $DATA_{ML}$. This data sampling algorithm is expected to be more effective because it not only makes use of single-crystal knowledge but also the ML method to find the dimensions that are more important to generate optimal solutions.

End-to-end optimization framework

We combine all (ODF, property) candidates dataset $DATA_R$, $DATA_{KB}$, and $DATA_{ML}$ and save (ODF, property) pairs with the highest and lowest properties as the final solutions. We observed that different hyper-parameters could affect the results and running time. Larger K may discover ODFs with more non-zero dimensions but less optimal properties. Larger T increases the time of sampling, which may lead to better results. ML-based data sampling method could generate data in a reduced and effective space, thus can reduce the time while getting optimal properties.

After getting the results of one material, we could go to the inner loop for the next material of the same crystal system. In this inner loop, we do not need to repeat the first and second steps of random data sampling again. Thus we could reduce the time consumption significantly. If all materials of the same crystal system are done, we can go to the next crystal system, i.e., go to the outer loop to run the whole framework except for the first step of random data sampling.

Experimental settings

The parameter T equals 10,000,000 in the proposed end-to-end framework to discover these solutions. Both parameters T_{HIGH} and T_{LOW} equal 100,000. T_{HIGH} is the number of (ODF, property) pairs with the highest properties, and T_{LOW} is the number of (ODF, property) pairs with the lowest properties in Random Data Sampling. The parameter D_{KB} is equal to 15. D_{KB} is the number of values of knowledge-based features that is used for assignment in Knowledge-based Data Sampling and ML-based Data Sampling. The parameter D_{ML} is equal to 10. D_{ML} is the number of values of ML-sorted features that is used for assignment in ML-based Data Sampling.

Nomenclature

A	Column vector of the ODF values at the independent nodes in the ODF mesh
P	Property matrix of the single-crystal values
q	Constant column vector that is obtained from the finite element discretization
x_{norm}	Normalized ODF vector, $x_{norm} = [x_{norm_1}, x_{norm_2}, \dots, x_{norm_D}]$, where $x_{norm_i} = \frac{x_i}{\max(x)}$
x	ODF vector, $x = [x_1, x_2, \dots, x_D] \subseteq R^D$
y	Interval vector $y = [y_1, y_2, \dots, y_K] \subseteq R^K$, where $\sum_{j=1}^K y_j = 1$ in random data sampling
z	Knowledge-based features vector $z = [z_1, z_2, \dots, z_D]$, where z_i is non-zero dimension of the i^{th} ODF in the list
C_{11}	Stiffness

Table continued

D_{KB}	The number of first and last values of knowledge-based features selected.
D_{ML}	The number of first and last values of ML-sorted features selected.
$DATA_{KB}$	The (ODF, property) candidates solutions generated in knowledge-based data sampling algorithm
$DATA_{ML}$	The (ODF, property) candidates solutions generated in ML-based data sampling algorithm
$DATA_R$	The (ODF, property) candidates solutions generated in random data sampling algorithm
E_{11}	Young's modulus
K	The number of random intervals in random data sampling
S_{11}	$S(1, 1)$, while S is the compliance matrix defined as $S = C^{-1}$
T	The parameter T represents the number of iterations to generate orientation distribution function (ODF) vectors.
T_{HIGH}	(ODF, property) pairs with the highest properties
T_{LOW}	(ODF, property) pairs with the lowest properties
D	The dimension of x

DATA AVAILABILITY

The data generated by the proposed AI-driven end-to-end microstructure optimization framework in this study is available at <https://github.com/MaoYuwei/AI-Driven-Microstructure-Optimization-Framework/tree/main/data>.

CODE AVAILABILITY

The codes of the proposed AI-driven end-to-end microstructure optimization framework used in this study is available at <https://github.com/MaoYuwei/AI-Driven-Microstructure-Optimization-Framework>.

Received: 18 November 2022; Accepted: 9 June 2023;

Published online: 23 June 2023

REFERENCES

- Olson, G. B. Computational design of hierarchically structured materials. *Science* **277**, 1237–1242 (1997).
- Agrawal, A. & Choudhary, A. Perspective: materials informatics and big data: realization of the “fourth paradigm” of science in materials science. *APL Mater.* **4**, 053208 (2016).
- Sundararaghavan, V. & Zabarav, N. Linear analysis of texture–property relationships using process-based representations of rodrigues space. *Acta Mater.* **55**, 1573–1587 (2007).
- Agrawal, A. et al. Exploration of data science techniques to predict fatigue strength of steel from composition and processing parameters. *Integr. Mater. Manuf.* **3**, 90–108 (2014).
- Ward, L., Agrawal, A., Choudhary, A. & Wolverton, C. A general-purpose machine learning framework for predicting properties of inorganic materials. *npj Comput. Mater.* **2**, 1–7 (2016).
- Liu, R. et al. Context aware machine learning approaches for modeling elastic localization in three-dimensional composite microstructures. *Integr. Mater. Manuf.* **6**, 160–171 (2017).
- Agrawal, A. & Choudhary, A. Deep materials informatics: applications of deep learning in materials science. *MRS Commun.* **9**, 779–792 (2019).
- Brewer, W. D., Bird, R. K. & Wallace, T. A. Titanium alloys and processing for high speed aircraft. *Mater. Sci. Eng. A.* **243**, 299–304 (1998).
- Moiseyev, V. N. *Titanium alloys: Russian aircraft and aerospace applications* (CRC press, 2005).
- Bratukhin, A., Kolachev, B., Sadkov, V. et al. *Technology of production of titanium aircraft structures* (Mashinostroenie, Moscow, 1995).
- Boyer, R. Titanium for aerospace: rationale and applications. *Adv. Perform. Mater.* **2**, 349–368 (1995).
- Choudhary, K. et al. The joint automated repository for various integrated simulations (jarvis) for data-driven materials design. *npj Comput. Mater.* **6**, 1–13 (2020).

13. Grandhi, R. V., Modukuru, S. C. & Malas, J. C. Integrated strength and manufacturing process design using a shape optimization approach. In *IDETC-CIE*, vol. 5213, 265–272 (American Society of Mechanical Engineers, 1990).
14. Xu, H., Li, Y., Brinson, C. & Chen, W. A descriptor-based design methodology for developing heterogeneous microstructural materials system. *J. Mech. Des.* **136**, 051007 (2014).
15. Heinz, A. & Neumann, P. Representation of orientation and disorientation data for cubic, hexagonal, tetragonal and orthorhombic crystals. *Acta Crystallogr.* **47**, 780–789 (1991).
16. Randle, V. & Engler, O. *Introduction to texture analysis: macrotexture, microtexture and orientation mapping* (CRC press, 2000).
17. Kocks, U. F., Tomé, C. N. & Wenk, H.-R. *Texture and anisotropy: preferred orientations in polycrystals and their effect on materials properties* (Cambridge university press, 1998).
18. Acar, P. & Sundararaghavan, V. Utilization of a linear solver for multiscale design and optimization of microstructures. *AIAA J.* **54**, 1751–1759 (2016).
19. Acar, P. & Sundararaghavan, V. Linear solution scheme for microstructure design with process constraints. *AIAA J.* **54**, 4022–4031 (2016).
20. Mezeix, L. & Green, D. J. Comparison of the mechanical properties of single crystal and polycrystalline yttrium aluminum garnet. *Int. J. Appl. Ceram.* **3**, 166–176 (2006).
21. Paul, A. et al. Data sampling schemes for microstructure design with vibrational tuning constraints. *AIAA J.* **56**, 1239–1250 (2018).
22. Mao, Y. et al. A deep learning framework for layer-wise porosity prediction in metal powder bed fusion using thermal signatures. *J. Intell. Manuf.* **34**, 1–15 (2022).
23. Dong, S., Wang, P. & Abbas, K. A survey on deep learning and its applications. *Comput. Sci. Rev.* **40**, 100379 (2021).
24. Esteve, A. et al. Deep learning-enabled medical computer vision. *npj Digit. Med.* **4**, 1–9 (2021).
25. Baek, M. & Baker, D. Deep learning and protein structure modeling. *Nat. Methods* **19**, 13–14 (2022).
26. Torres, J. F., Hadjout, D., Sebaa, A., Martínez-Álvarez, F. & Troncoso, A. Deep learning for time series forecasting: a survey. *Big Data* **9**, 3–21 (2021).
27. Baldock, R., Maennel, H. & Neyshabur, B. Deep learning through the lens of example difficulty. *NeurIPS* **34**, 10876–10889 (2021).
28. Su, X. et al. A comprehensive survey on community detection with deep learning. *IEEE Trans. Neural. Netw. Learn. Syst.* 1–21 (2022).
29. Yang, Z. et al. Microstructural materials design via deep adversarial learning methodology. *J. Mech. Des.* **140**, 111416 (2018).
30. Jha, D. et al. Enabling deeper learning on big data for materials informatics applications. *Sci. Rep.* **11**, 1–12 (2021).
31. Gupta, V. et al. Cross-property deep transfer learning framework for enhanced predictive analytics on small materials data. *Nat. Commun.* **12**, 1–10 (2021).
32. Gupta, V., Liao, W.-k., Choudhary, A. & Agrawal, A. Brnet: Branched residual network for fast and accurate predictive modeling of materials properties. In *Proc. SIAM Int. Conf. Data Min.*, 343–351 (SIAM, 2022).
33. Jha, D., Gupta, V., Liao, W.-k., Choudhary, A. & Agrawal, A. Moving closer to experimental level materials property prediction using ai. *Sci. Rep.* **12**, 1–9 (2022).
34. McElfresh, C., Roberts, C., He, S., Prikhodko, S. & Marian, J. Using machine-learning to understand complex microstructural effects on the mechanical behavior of ti-6al-4v alloys. *Comput. Mater.* **208**, 111267 (2022).
35. Liu, X. et al. Machine learning assisted prediction of microstructures and young's modulus of biomedical multi-component β -ti alloys. *Metals* **12**, 796 (2022).
36. Zou, C. et al. Integrating data mining and machine learning to discover high-strength ductile titanium alloys. *Acta Mater.* **202**, 211–221 (2021).
37. Liu, R. et al. A predictive machine learning approach for microstructure optimization and materials design. *Sci. Rep.* **5**, 1–12 (2015).
38. Liu, R., Agrawal, A., Liao, W.-k., Choudhary, A. & Chen, Z. Pruned search: A machine learning based meta-heuristic approach for constrained continuous optimization. In *Int. Conf. Contemp. Comput.*, 13–18 (IEEE, 2015).
39. Paul, A. et al. Microstructure optimization with constrained design objectives using machine learning-based feedback-aware data-generation. *Comput. Mater.* **160**, 334–351 (2019).
40. Hasan, M. et al. Data-driven multi-scale modeling and optimization for elastic properties of cubic microstructures. *Integr. Mater. Manuf.* **11**, 1–11 (2022).
41. Hasan, M. & Acar, P. Machine learning reinforced microstructure-sensitive prediction of material property closures. *Comput. Mater. Sci.* **210**, 110930 (2022).
42. Liaw, A. & Wiener, M. et al. Classification and regression by randomforest. *R news.* **2**, 18–22 (2002).

ACKNOWLEDGEMENTS

This work was supported primarily by National Science Foundation (NSF) CMMI awards 2053929/2053840. Partial support from NIST award 70NANB19H005 and DOE awards DE-SC0019358, DE-SC0021399 is also acknowledged.

AUTHOR CONTRIBUTIONS

Y.M. designed and carried out the implementation and experiments for the end-to-end AI-driven microstructure optimization framework under the guidance of A.A., A.C., and W.L. P.A., M.H., K.C., and F.T. provided the necessary domain expertise for this work. Y.M., A.A., M.H., P.A., A.P., V.G. wrote the manuscript. All authors discussed the results and reviewed the manuscript.

COMPETING INTERESTS

The authors declare no competing interests.

ADDITIONAL INFORMATION

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41524-023-01067-8>.

Correspondence and requests for materials should be addressed to Ankit Agrawal.

Reprints and permission information is available at <http://www.nature.com/reprints>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2023