Computational Discovery of Microstructured Composites with Optimized Trade-Off between Strength and Toughness

by

Beichen Li

B.S., Tsinghua University (2018)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2021

Author .....
Department of Electrical Engineering and Computer Science
May 20, 2021

Certified by .....
Wojciech Matusik
Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by .....
Leslie A. Kolodziejski
Professor of Electrical Engineering and Computer Science
Chair, Department Committee on Graduate Students

Computational Discovery of Microstructured Composites with Optimized Trade-Off between Strength and Toughness

by

Beichen Li

Submitted to the Department of Electrical Engineering and Computer Science
on May 20, 2021, in partial fulfillment of the
requirements for the degree of
Master of Science in Computer Science

Abstract

The conflict between strength and toughness is critical to practical engineering problems. To create structural composites with extraordinary strength and toughness, previous works mainly drew inspiration from nature and attempted to replicate nacre-like structures in synthetic composites using a brick-and-mortar architecture. There is no microscale control over constituent materials and the designed composites often exhibit anisotropic properties. Recent advances in high-resolution multi-material additive manufacturing enable the creation of high-performance heterogeneous composites through voxel-level microstructure configurations. However, past efforts in exploring these designs suffered from a limited design space and a weak benchmark that only comprised base materials. More importantly, they failed to address the clear discrepancies between simulation predictions and experimental measurements (the sim-to-real gap). To our best knowledge, no work has been successfully conducted to tackle the conflict between strength and toughness while taking on the sim-to-real challenge. In this work, we propose a computational pipeline where microstructures with optimal strength and toughness trade-offs are automatically discovered and analyzed for intrinsic toughening mechanisms. Based on a fast physics-based simulator, it employs a competitive game approach to bridging the gap between simulation and experiment. The pipeline will open the door for reversing the traditional scientific discovery process through analysis-by-synthesis, and potentially generalize to a wide range of applications in material science, chemistry, pharmaceutics, robotics, etc.

Thesis Supervisor: Wojciech Matusik
Title: Professor of Electrical Engineering and Computer Science

Acknowledgments

I would like to express heartfelt gratitude to Prof. Wojciech Matusik, my advisor, for leading me on an unforgettable journey into a challenging yet fundamental and influential research topic. It is his guidance, engagement, encouragement, and patience that walked me through ups and downs in a project that has been long overdue.

I am deeply grateful to my fellow collaborators Dr. Wan Shou, Prof. Tae-Hyun Oh, Dr. Yuanming Hu, and Yiyue Luo, who not only contributed a significant amount of time and effort but also kindly supported me as mentors and friends. It is their professionalism, devotion, and passion that motivated me to be better and march towards higher goals with confidence.

I would also like to thank my labmates Liang Shi, Dr. Kui Wu, Tao Du, and Jie Xu, for providing great advice and support as I navigated through my first three years of PhD life. Many thanks to other lab members as well with whom I learned, chatted, and played.

Finally, I'm very much thankful to my family and friends who were always there when I felt alone and needed someone to talk to. I'm not a problem solver but more than willing to do the same for them.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Trade-offs, or design conflicts, exist ubiquitously in nature and practical engineering problems [38, 34, 4], such as yield and purity in chemical synthesis [41]; cost, mass, volume, power-to-weight ratio, and energy density during material selection [3]. While nature has set many examples to harmonize such conflicts [47, 7] (e.g., strength and toughness in mollusk shells or nacre [35, 40]), engineers are still striving to make a great balance. A classical problem in engineering material design is the conflict between strength and toughness [40, 47]. In particular, designing structural materials that exhibit extremal toughness is vitally important. Although stiff materials are hard to deform under load, they generally have a low resistance to fracture and can cause catastrophic failure, hence resulting in inferior toughness. Various efforts have been attempted to replicate biological structures in synthetic composites due to their excellence in both properties. Inspired by biological materials, Zhao et al. [49] developed a shear-flow-induced alignment of two-dimensional nanosheets at an immiscible hydrogel/oil interface to manufacture layered nanocomposites, which showed remarkable stiffness and toughness. By mimicking the microstructure of natural nacre through stacking engraved glass sheets and transparent thermoplastic elastomer, Yin et al. [48] developed an impact-resistant glass. A synthetic nacre was developed through a mesoscale 'assembly-and-mineralization' approach inspired by the natural process in mollusks [33], and had similar hierarchical structure to natural nacre. However, even with diverse fabrication methods, including ice-templating [33], magnetic field align-

ment [15], and shear-flow-induced alignment [49], the fundamental design principle for fabricating strong and tough composites is the same, namely using a brick-and-mortar architecture. Constituent materials are mainly randomly assembled and there is no precise control at microscale. Besides, these bioinspired materials usually show anisotropic mechanical performance which is extraordinary in one direction but much weaker in another direction [36]. Instead, making heterogeneous composites strong and tough in multi-direction is more appealing to practical applications.

Recent advances in multi-material high-resolution additive manufacturing opened up new possibilities for creating heterogeneous composites with great mechanical performance. Dimas et al. [12] reported natural-material-inspired fracture-resistant composites using multi-material 3D printing at micrometer resolution; however, the design space is limited to a certain fraction of the soft material. Furthermore, simulation and experiment see a huge gap. Despite relatively accurate predictions in ranking, there is no direct comparison between experimental and simulated values. Mirzaali et al. [37] found that a linear graded transition interface between hard and soft materials can increase Young's modulus while at the cost of reduced elongation and fracture toughness. However, their work is limited to interface study, and only five designs were investigated. Kim et al. [29] reported a set of tough isotropic structural composites fabricated by multi-material 3D printing. A lattice spring model (LSM) was developed to capture their toughening mechanism, yet only in a qualitative manner, due to the nonlinear response of materials to crack propagation. Gu et al. [21, 22, 20, 19] conducted a series of studies about bioinspired composites made by multi-material 3D printing. Although multiple materials were used in most cases, researchers mainly compared designed composites with base materials. The performance of homogeneously mixed composites was left unclear, which is a critical benchmark. So far, no work has been successfully conducted to tackle the conflict between strength (e.g., Young's modulus) and toughness via microstructure configurations at voxel-level resolution [5].

With voxel as the building block, the design space of microstructures is combinatorial and often infeasible to probe manually. Therefore, computational exploration

can be an effective alternative [43, 44, 1, 9]. Recently, some machine learning (ML) algorithms have been applied to explore synthetic composites inspired by bio-structures for improved toughness [20, 19]. However, to the best of our knowledge, no computational model can precisely replicate the fracture behaviors of experimental materials [13] because of multiscale features (from hundreds of micrometers to tens of millimeters) and the stochastic nature in crack propagation [25]. Unfortunately, almost all ML-assisted discovery of high mechanical performance composites takes numerical simulations as the ground truth. Thus, a direct comparison of predicted values with experimental measurements is not available to evaluate the feasibility of numerical simulation or learning-based prediction in real applications, leaving the gap between simulation and experiment unattended. Besides, the majority of bio-inspired and ML discovered structures only exhibit satisfactory fracture resistance in a specific direction [22, 20, 19]. Last but not least, a systematic analysis of intrinsic toughening mechanisms in heterogeneous composites is currently absent [24]. Such mechanisms, once uncovered, will likely complement the ones found in homogeneous ones and promote the structural design of composites with extremal toughness [18].

In this work, we present a computational pipeline that automatically discovers microstructured composites with optimal trade-offs between strength and toughness. Then, we reveal the underlying mechanisms contributing to their extremal toughness. To realize our goals, the following challenges are resolved.

- Virtual mechanical testing using FEM is the primary means of evaluating microstructure properties in the pipeline. Thus, we propose a data-driven system identification approach that bridges the gap between simulation and experiment (also known as the sim-to-real gap) for more accurate predictions of Young's modulus and toughness.

- We devise a learning-assisted data-efficient exploration algorithm that discovers microstructures with optimal performance trade-offs within the combinatorial design space. The algorithm is able to cover a sufficiently large performance range with an order of magnitude fewer samples than existing evolutionary

algorithms.

- The best microstructures found by the exploration algorithm are grouped into families by appearance and properties. We inspect each microstructure family and analyze the most representative intrinsic toughening mechanisms.

Meanwhile, the multi-material inkjet 3D printer we use for fabricating microstructured composites can produce homogeneous composites by mixing constituent materials on the fly through droplet-level control. In a quantitative comparison, we demonstrate that the optimal microstructured composites discovered by the pipeline exhibit much better performance trade-offs between Young's modulus and toughness than homogeneous counterparts.

The remaining of the thesis is organized as follows. Chapter 2 describes the implementation of the computational pipeline in comprehensive details, including the overall workflow, problem definition, fabrication and testing methods, FEM simulation, and algorithms used in each step. Chapter 3 demonstrates results on the performance and analysis of discovered microstructures, as well as the evaluation of key algorithms in the pipeline. Chapter 4 makes a summary and discusses possible future works.

Chapter 2

Method

In this work, we develop a machine learning-assisted data-driven computational pipeline to discover microstructured composites with optimal performance trade-offs between Young's modulus and toughness, tackling a classical design conflict in engineering materials. The pipeline is formulated using an interactive proposal-validation scheme that embodies a competitive game between two agents. This chapter introduces the definition of microstructures, methods for manufacturing and mechanical testing, the workflow and key components of the pipeline, and the analysis procedure of optimal microstructures.

2.1 Definition of Microstructures

As the structural unit of heterogeneous composites in this work, a microstructure is digitally represented as a 20×20 binary matrix that uniquely defines the spatial arrangement of two base materials in a composite (Figure 2-1A). The matrix is also called a *microstructure pattern* that represents a single microstructure unit. The soft base material is indicated by a value of `0` in a microstructure pattern while the rigid material corresponds to `1`. As a microstructure pattern is conceptually similar to an image, we refer to its elements as *pixels* to avoid confusion with FEM.

To construct the design space for computational exploration, we additionally pose two types of constraints on microstructure patterns, illustrated in Figure 2-1B and

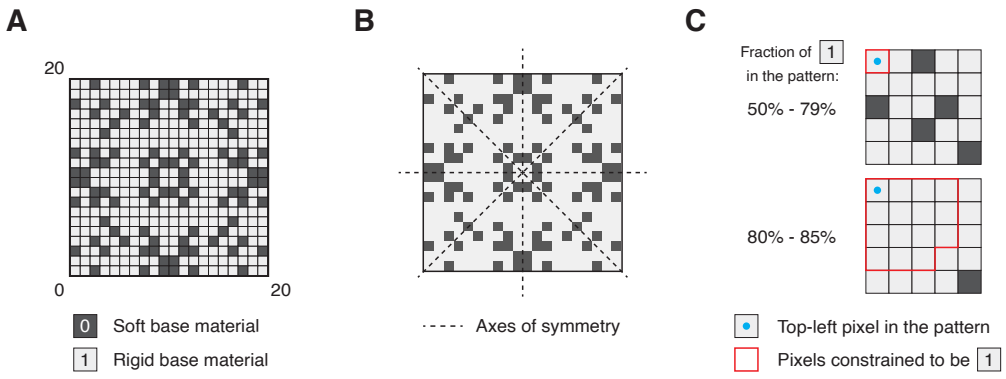


Figure 2-1: A. A microstructure example. B. Symmetry constraints on microstructure patterns. C. Structural constraints on the design space of microstructures.

Figure 2-1C. The symmetry constraints require all microstructures to be invariant to reflections and 90-degree rotations so that they have identical mechanical properties in horizontal and vertical directions. The structural constraints enforce that the volume fraction of the rigid material be between 50% and 85%. Also, pixels around the corner of a pattern are restricted to `1` depending on this fraction. If it is below 80%, only the corner pixel is restricted; otherwise (above 80% but no more than 85%), the rule extends to every pixel whose center lies within a 4-pixel-wide radius from the corner vertex. We empirically observe that these structural constraints are crucial to obtaining reliable and consistent toughness measurements in mechanical testing, because the fracture behaviors of very stiff microstructures tend to be highly stochastic and result in low toughness [21]. All constraints aggregated, the design space of microstructures contains more than 10^{16} possibilities.

2.2 Fabrication and Measurements

Microstructures and homogeneous mixed composites are manufactured on a commercial multi-material 3D printer using a voxel printing approach. Their Young's moduli and toughnesses are physically measured via tensile mechanical testing. To enhance the resistance to fracture in microstructured composites, we engineer the interface between base materials to mimic a 50/50 mixture of base materials.

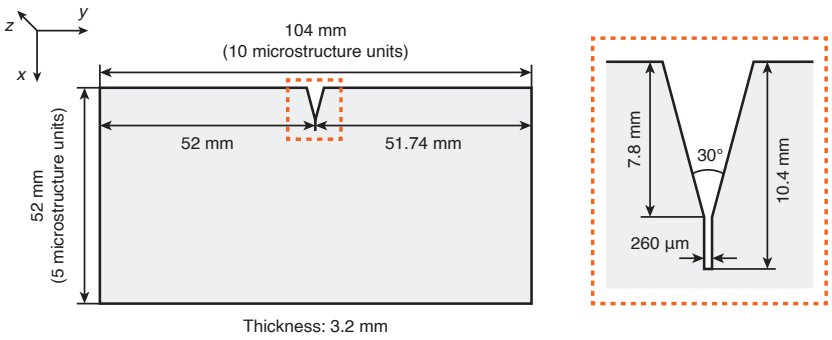


Figure 2-2: Schematics of a toughness specimen. Coordinate axes indicate the orientation of the specimen in 3D printing.

2.2.1 Manufacturing of Specimens

All composites are manufactured on a Stratasys Object 260 Connex multi-material 3D printer with a nominal resolution of 300 dpi, 600 dpi, and 800 dpi in x, y, and z directions, respectively. VeroWhitePlus (`VW+`) and TangoBlackPlus (`TB+`), two acrylic-based photopolymers, are used as base materials. The orientation of models is visualized by the coordinate axes in Figure 2-2. Microstructure patterns are located on the x-y plane and sweep along the z direction, where the longest dimension of the model aligns with the y direction. Regarding the scale of printed specimens, each pixel in a microstructure pattern is calculated to be a 260 μm square, a reasonable approximation to the least common multiple of the droplet dimensions in x and y directions.

To enable droplet-level control, we adopt a voxel printing approach supported by the printer. In voxel printing mode, droplet configurations can be precomputed locally and sent to the printer in real time. This allows for printing homogeneous composites mixed by base materials at a given ratio. For any volume fraction of the rigid base material, denoted by ϕ, a printing model is created by probabilistically assigning base materials to droplets, where each droplet is assigned with the rigid material at a probability of ϕ. Since the base materials are miscible, homogeneous composites are printed as droplets get mixed on the fly. Uniform composite materials composed of 50%, 60%, 65%, 70%, 75%, 80%, 85%, and 90% `VW+` are printed as the benchmark.

2.2.2 Physical Measurements

For Young's modulus measurements, all testing specimens are printed following the dimensions recommended in ASTM D638. For toughness measurements, as there is no standard reference for heterogeneous composites, we devise our specimens shown in Figure 2-2. The dimensions of toughness specimens are 104×52×3.2 mm. A notch with a length of 10.4 mm is printed, equivalent to 20% of the specimen's width. After that, a triangle cut is added using a milling machine at a depth of 7.8 mm, which is 15% of the specimen's width. The length of the gripper area on either side is 25 mm, leaving a 54 mm gauge area in the middle.

After printing, all specimens are carefully cleaned on a waterjet machine to remove support material and dried at room temperature. An ultraviolet (UV) light post-curing process is conducted to allow uniform curing on top and bottom sides. This step is performed in a Fusion UV system with a constant belt speed of around 2 cm/s. Additionally, the printed notch of each toughness specimen is cleaned by a new razor blade to avoid the influence of residual support material at the tip.

All specimens are tested on an Instron 5984 universal testing machine with a maximum load of 150 kN. Tensile tests for Young's modulus measurements are conducted according to ASTM D638. Four specimens are tested for each composite and the results are averaged. On the toughness side, there is no existing standard targeting heterogeneous composites. Thus, we adopt the following tensile test method. The specimen is pulled at a rate of 2 mm/min. Data acquisition stops once the crack propagated entirely through the specimen. Because the crack propagation in microstructured composites is nonlinear, the toughness is defined as the energy absorbed and dissipated per unit volume, namely, the area under the stress-strain curve [35]. At least four specimens are tested for each composite, where at least three specimens that manifest the most common and consistent fracture behaviors are considered valid. The valid toughness measurements are averaged, from which a representative stress-strain curve is selected. The same procedure is followed when testing homogeneous composites for consistency.

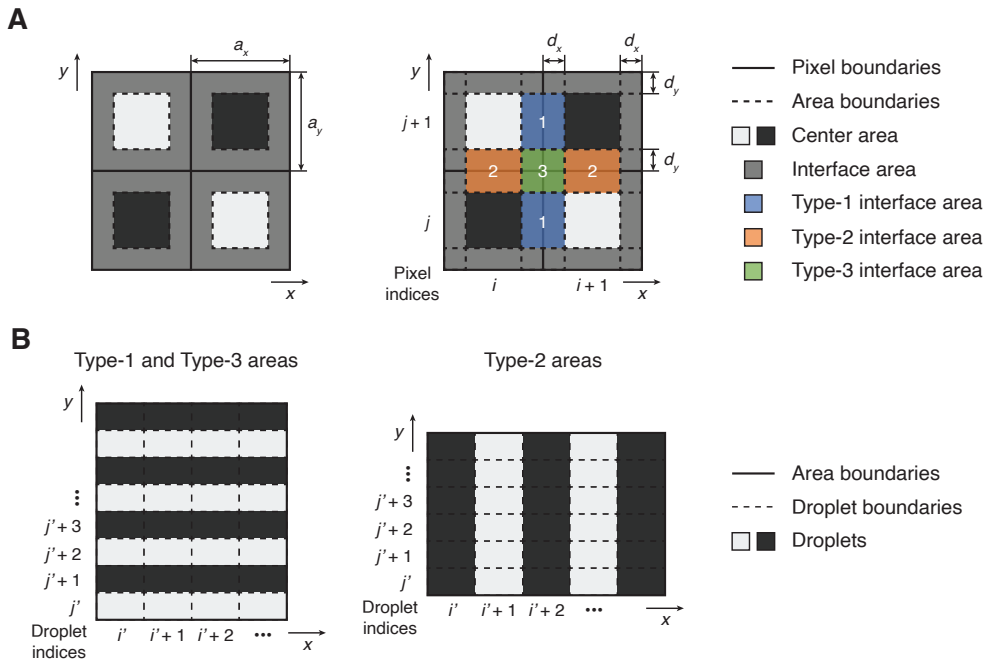


Figure 2-3: Schematics of interface engineering. A. Definition and classification of interface areas. B. Droplet arrangements in interface areas. All coordinate axes conform with the definitions in Figure 2-2.

2.2.3 Interface Engineering

Upon examination on printed specimens, we notice that the interface between base materials is relatively strong but can still be the weakest spot during crack propagation [37]. Thus, we engineered the interface to enhance its strength and improve the toughnesses of microstructures. This is done by applying manually designed droplet patterns in *interface areas*. As illustrated in Figure 2-3, interface areas are located around shared edges or vertices of adjacent pixels, where one base material transitions into the other. There are three types of interface areas in total. Let (i, j) be the x and y indices of a pixel in an entire specimen model. Type-1 areas connect two neighboring pixels (i, j) and $(i+1, j)$ along the x direction, i.e., the direction of crack propagation. Type-2 areas connect two neighboring pixels (i, j) and $(i, j+1)$ along the y direction, i.e., the pulling direction in tensile testing. Type-3 areas join all four surrounding pixels. Whenever an interface area covers two pixels of different material assignments, droplets in that area are specially arranged to approximate a 50/50 mix-

ture of base materials. Similarly, let (i', j') be the x and y indices of a droplet in the specimen model. Droplets appear in alternating rows in Type-1 and Type-3 areas, where each droplet (i', j') follows the parity of $i + j'$. In contrast, droplets appear in alternating columns in Type-2 areas, each following the parity of $j + i'$. Such patterns remain constant in the z direction.

Table 2.1: Possible dimensions of a pixel and its subdivided areas after quantization into droplets for voxel printing. All units are droplets.

Pixel dimensions		**The center area**		**The interface area**	
x (a_x)	y (a_y)	x	y	x (d_x)	y (d_y)
3	5	1	3	1	1
3	6	1	2	1	2
4	5	2	3	1	1
4	6	2	2	1	2

In the context of voxel printing, a pixel may contain 3×5, 3×6, 4×5, or 4×6 droplets after quantization. Table 2.1 provides the quantized dimensions of a pixel's internal area contingent on the pixel size. Quantized dimensions of interface areas are then derived from the pixels they intersect with. Type-1 and type-3 areas may contain 2×2 or 2×3 droplets, while type-2 areas vary among 1×2, 1×3, 2×2, and 2×3 droplets in size.

2.3 Computational Pipeline

In this work, we present an interactive proposal-validation scheme for the computational pipeline of microstructure discovery, conceptualized as a competitive game [17]. Two agents participate with opposite goals and take turns improving their performances. The competition process not only yields microstructures with the best performance trade-offs but also closes the sim-to-real gap, both being fundamental prerequisites for the analysis of toughening mechanisms.

The workflow of the computational pipeline is shown in Figure 2-4. Overall, the game proceeds in an iterative manner. Each iteration starts with the proposal agent

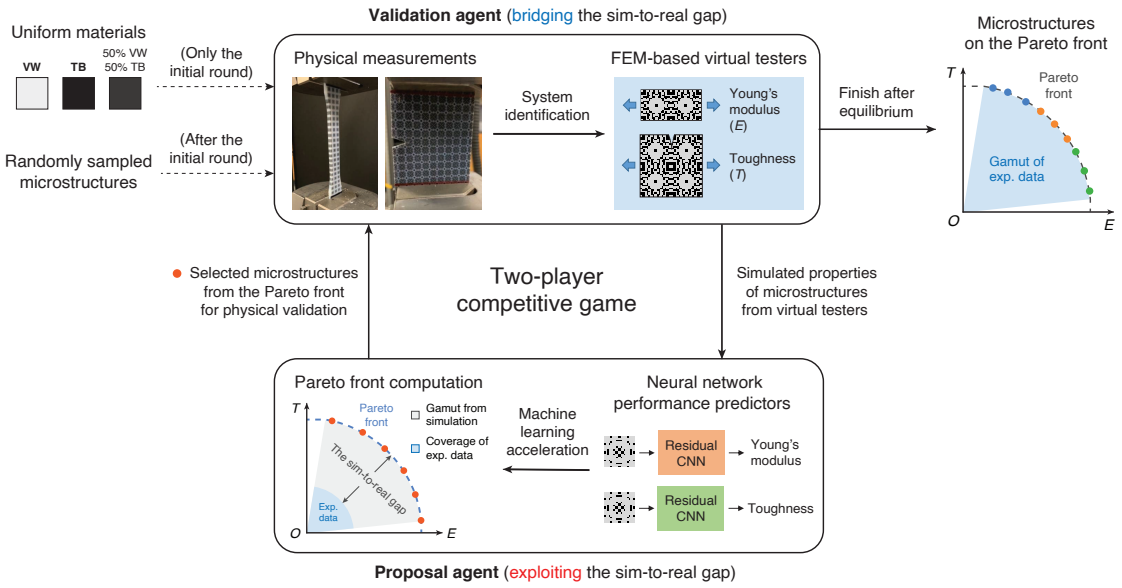


Figure 2-4: The workflow of the computational pipeline in the form of a two-player competitive game.

running multi-objective optimization (MOO) using predictions from a pair of virtual testers. The MOO algorithm computes a performance range of microstructures as large as possible, called the performance gamut (or simply the *gamut*). After that, the proposal agent selects several samples on or near the Pareto front of the gamut where optimal trade-offs are achieved. The validation agent then validates these samples by manufacturing testing specimens and conducting physical measurements. The measured data is used in system identification, which constrains the simulation model by minimizing the prediction error. At last, the updated simulator is passed back to the proposal agent to begin on the next iteration. Due to the sim-to-real gap, MOO can easily discover microstructures with overestimated properties. Conversely, system identification encourages the simulator to make more realistic predictions. Therefore, the proposal agent and the validation agent alternately exploits and bridges the sim-to-real gap, making the gamut shrink over iterations. In the end, the game terminates when an equilibrium is reached.

This section dives deep into the following core components in the pipeline:

- Fast FEM-based virtual testers that predict the Young's modulus and toughness

of microstructures in high throughput.

- System identification that matches the simulation output with physical measurements to bridge the sim-to-real gap.
- A learning-accelerated exploration algorithm that discovers the Pareto front of the gamut using much fewer samples than existing evolutionary algorithms.
- The selection strategy of Pareto-optimal samples for physical validation.

2.3.1 Simulation

We implement a pair of customized virtual testers that estimate the performance of microstructures. Given an input microstructure pattern, the virtual testers simulate the mechanical testing process of real specimens using FEM. Young's modulus and toughness are then computed from recorded stress-strain data. The virtual testers use the same material model but differ in algorithmic details and parameter settings.

The nonlinear stress-strain response of the base materials and the interface between them is approximated using the Neo-Hookean model. The model is set in 2D where all materials are isotropic and compressible. We follow the strain energy density function in [42]:

$$W = \frac{\mu}{2}\left(I_1 - 2 - \ln J\right) + \frac{\lambda}{2}\left(\ln J\right)^2, \tag{2.1}$$

where μ and λ are the Lamé parameters. $J = \det(F)$ is the determinant of the deformation gradient (F), and $I_1 = \mathrm{tr}(C)$ is the first invariant of the right Cauchy-Green deformation tensor ($C = F^{\mathrm{T}}F$). μ and λ are derived from Young's modulus (E) and Poisson's ratio (ν) as follows

$$\begin{aligned} \mu &= \frac{E'}{2(1+\nu')} \\ \lambda &= \frac{E'\nu'}{(1+\nu')(1-2\nu')}, \end{aligned} \tag{2.2}$$

where E' and v' are converted from E and ν as follows [28]

$$\begin{aligned} \nu' &= \frac{\nu}{1+\nu} \\ E' &= \frac{E(1+2\nu}{(1+\nu)^2}. \end{aligned} \tag{2.3}$$

Although Young's moduli and Poisson's ratios of the constituent materials can be physically measured, they are regarded as optimizable parameters in system identification, which will be introduced in Section 2.3.2.

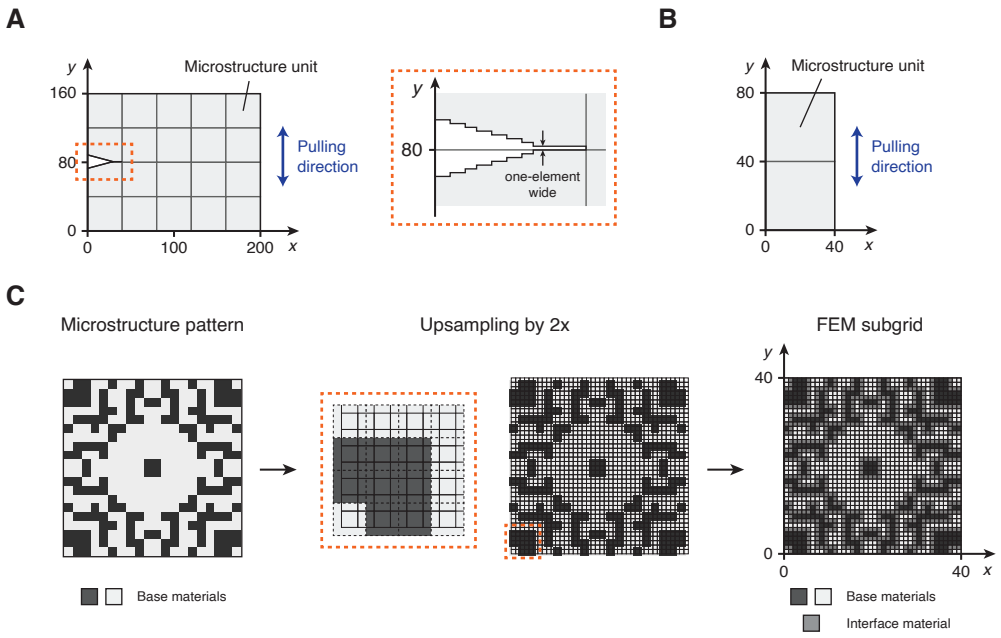


Figure 2-5: A. Dimensions of the FEM grid in the toughness virtual tester. B. Dimensions of the FEM grid in the Young's modulus virtual tester. C. The upsampling process from a microstructure pattern to a 40×40 FEM subgrid.

The virtual tester for toughness operates on a regular grid of 200×160 quadrilateral finite elements (Figure 2-5A) where each node has four degrees of freedom. The grid contains 5×4 tiles of microstructure units where each unit covers a 40×40 sub-grid. Notably, a microstructure unit doubles the resolution of the pattern due to modeling the interface between base materials. The interface is modeled as a third homogeneous material, referred to as the *interface material*. Finite elements of the

interface material and the base materials are equally sized. This is backed by our microscopic observation of printed specimens, which indicates that the width of the interface is typically around 100 μm and comparable to half of the pixel size. The computation of material assignment in a sub-grid is analogous to image upsampling. As is shown in Figure 2-5C, the FEM subgrid is superimposed on the original pattern where the center of the bottom-left element aligns with the center of the bottom-left pixel. An element inherits the corresponding pixel value if it is located entirely within a pixel; otherwise, the material assignment depends on the values of surrounding pixels in the same way as interface engineering. To mimic the pre-cracking of real specimens, a single-element-wide notch and a triangular cut are applied at the grid position of $(0, 80)$. The 'removed' elements are assigned with void material and do not generate nodal forces under deformation. Overall, this configuration correlates well with a printed specimen despite a simpler geometry to reduce computation cost. Dirichlet boundary conditions are enforced on the displacement in the y-direction during testing. Nodal forces in each element are computed from the Neo-Hookean model using four Gaussian quadrature points. Nodal velocities and displacements are then obtained from explicit time integration. Crack initiation and propagation are modeled on a per-element basis. To that end, strain energy densities are computed at the quadrature points together with nodal force contributions. An element is considered failed and reassigned with a void material once the maximum strain energy density at quadrature points exceeds a certain threshold specific to each material. Finally, simulation terminates when the global strain goes beyond 0.3 or the measured stress falls below 20% of the ultimate tensile strength, with the only exception being the soft base material for which the maximum strain is unlimited. In the end, toughness is calculated by integrating the stress-strain curve.

While it is possible to compute nodal velocities and displacements using an implicit solver, we choose an explicit solver since explicit solvers are generally preferred for nonlinear FEM simulations that involve dynamic failure. In particular, the virtual tester must operate at a sufficiently small time step (around 10^{-7} s) to precisely capture the crack propagation in microstructures and maintain numerical stability.

Under such a strict requirement, explicit solvers are more efficient due to simpler implementation and better compatibility with many-core hardware accelerators like graphical computing units (GPU). However, we notice that the virtual tester runs too slow at a strain rate that matches real mechanical testing. In this case, simulating a microstructure till an assumed failure strain of 0.02 requires 1.62×10^8 time steps, taking almost one hour to finish on an NVIDIA Tesla V100 GPU under maximum throughput. The running time is impractical to our computational pipeline because the simulation budget will be too tight to allow for a proper exploration of the combinatorial design space. As a workaround for this challenge, we used a much larger strain rate (equal to 20) to speed up the simulation while introducing system-wide numerical damping to model material viscosity. Once the velocity field is updated in every time step, nodal velocities $v_{i,j}$ are smoothed by a 3×3 filter defined as

$$v'_{i,j} = (1-\alpha)v_{i,j} + \frac{\alpha}{9}\sum_{k=-1}^{1}\sum_{l=-1}^{1} v_{i+k,j+l}, \tag{2.4}$$

where $\alpha = \gamma\Delta t$ is the product of the time step Δt and a constant damping coefficient γ. Since all nodes have identical mass, this step effectively interpolates between the momentum of each node and the mean momentum in a 3×3 neighborhood. A larger γ leads to stronger damping effects and potentially accounts for more material viscoelasticity. Our observations suggest that larger γ values work better with stiffer microstructures in suppressing oscillation. As there is no trivial way to determine the value of γ for all microstructures, it is optimized in system identification instead.

The virtual tester for Young's modulus has a much smaller grid of 80×40 quadrilateral elements, consisting of 2×1 microstructure units (Figure 2-5B). The material assignment to elements is computed using the same image upsampling method and no notch or triangular cut is added. This configuration qualitatively matches the gauge area of an ASTM D638 Type 1 specimen. A constant displacement boundary condition is enforced in the y-direction where the global strain remains fixed at 10^{-4}. Nodal forces are computed from the aforementioned Neo-Hookean model. Explicit time integration is used to update velocities and displacements. Negative exponential

damping is applied to the velocity field as follows

$$v'_{i,j} = v_{i,j} e^{-\gamma \Delta t}, \tag{2.5}$$

where $\Delta t \approx 10^{-7}$ s is used as the time step and the damping coefficient γ is optimized through system identification. The grid is allowed to reach a static equilibrium within 10^4 time steps, after which the Young's modulus is derived from the gauge stress. Although an implicit linear elasticity solver is usually used to virtually measure Young's modulus, we directly reused the explicit solver in the toughness simulator for simplicity. Note that the Young's modulus virtual tester is not the bottleneck of the whole simulation.

In pursuit of maximal computation efficiency, our customized virtual testers were implemented using the Taichi programming language [26], an emerging high-performance programming language tailored to computer graphics applications including physics-based simulation. Taichi compiles our source code written in Python syntax into deeply optimized machine programs that, when executed, exploit the massive parallel computing power of GPUs. However, we find that simulating microstructures sequentially results in low GPU utilization due to inadequate workload. Our solution is adding a batching mechanism to enable concurrent simulation of multiple microstructures, where each microstructure can be evaluated using a unique set of optimizable parameters. This allows for maximum GPU usage regardless of its power, bringing about a much higher throughput.

2.3.2 System Identificaton

System identification finds the optimal parameter settings of the virtual testers to match simulation results with physical measurements (Figure 2-6). We first define some mathematical notations before providing a formal description of the problem. For any microstructure x, let $f(x, \theta) : X \times P \to \mathbb{R}$ denote the virtual tester for Young's modulus with a parameter setting θ, where $X \subset \{0, 1\}^{20 \times 20}$ is the design space of microstructures and $P \subset \mathbb{R}^7$ is the feasible set of θ. The estimated Young's modulus

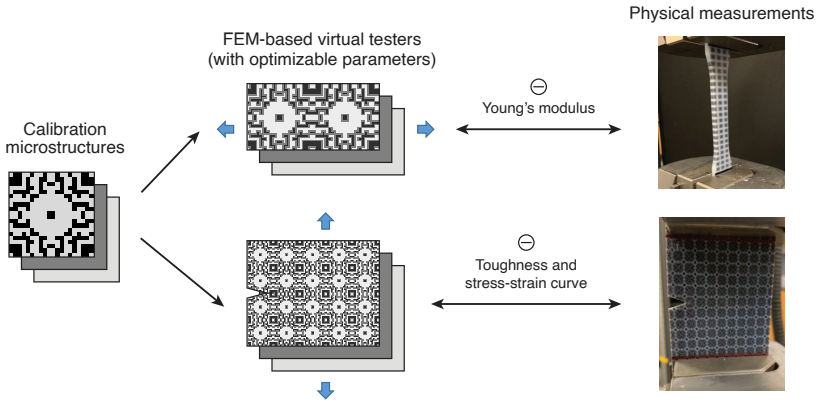


Figure 2-6: Illustration of the system identification problem.

of x is directly obtained from $\hat{E} = f(x, \theta)$. Similarly, let $g(x, \eta) : X \times Q \to C[0, \infty)$ denote the virtual tester for toughness with a separate parameter setting η, where $Q \subset \mathbb{R}^{10}$ is the feasible set of η. The output g is not a single number but a stress-strain curve $\hat{\sigma}(\varepsilon)$ defined as a continuous over non-negative strains. Thus, the toughness of x is estimated by computing the following integral

$$\hat{T} = I[\hat{\sigma}] = \int_0^\infty \hat{\sigma}(\varepsilon)\, d\varepsilon. \tag{2.6}$$

Both θ and η are tuples of optimizable simulation parameters, including

- Reference Young's moduli of base materials (dubbed as TB and VW) and the interface material (dubbed as IF): $E_{\text{TB}}, E_{\text{VW}}, E_{\text{IF}}$;
- Reference Poisson's ratios: $\nu_{\text{TB}}, \nu_{\text{VW}}, \nu_{\text{IF}}$;
- Failure thresholds of strain energy density: $W_{\text{TB}}, W_{\text{VW}}, W_{\text{IF}}$;
- A system-wide damping coefficient: γ.

Since fracture simulation is not involved in predicting Young's modulus, we have

$$\begin{aligned} \theta &= (E_{\text{TB}}, E_{\text{VW}}, E_{\text{IF}}, \nu_{\text{TB}}, \nu_{\text{VW}}, \nu_{\text{IF}}, \gamma) \\ \eta &= (E_{\text{TB}}, E_{\text{VW}}, E_{\text{IF}}, \nu_{\text{TB}}, \nu_{\text{VW}}, \nu_{\text{IF}}, W_{\text{TB}}, W_{\text{VW}}, W_{\text{IF}}, \gamma). \end{aligned} \tag{2.7}$$

Based on the optimizable parameters, the prediction errors of virtual testers are minimized over a group of microstructures $X_c = \{x_i \,|\, i = 1, 2, \cdots, N\}$, called the calibration set. The optimization process solves the following problems

$$\begin{aligned} \theta^* &= \underset{\theta}{\operatorname{argmin}} \sum_{i=1}^{N} w_i \frac{\left|\hat{E}_i - E_i\right|}{E_i} \\ \eta^* &= \underset{\eta}{\operatorname{argmin}} \sum_{i=1}^{N} w_i \frac{\left|\hat{T}_i - T_i\right| + \tilde{\lambda} I\left[|\hat{\sigma}_i - \sigma_i|\right]}{T_i}, \end{aligned} \tag{2.8}$$

where $\hat{E}_i, \hat{T}_i, \hat{\sigma}_i$ are the predicted Young's modulus, toughness, and stress-strain curve of each microstructure x_i; E_i, T_i, σ_i are the corresponding physical measurements. For Young's modulus, the objective is simply the average relative prediction error over the calibration set. For toughness, we not only match the values but also assimilate stress-strain curves as much as possible. Since stress-strain curves are functions, the error between two curves is calculated by integrating their absolute difference. Geometrically, this error represents the symmetric difference of the areas beneath the curves. The two error terms are balanced by a multiplier $\tilde{\lambda} = 1$.

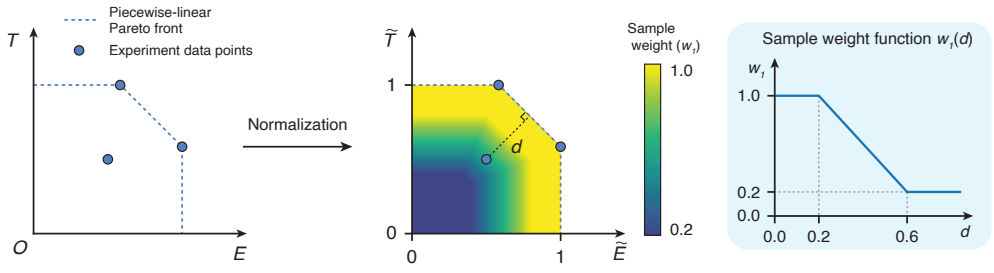


Figure 2-7: An example of deriving the first sample weight component, w_1, in system identification.

Additionally, as the optimizable parameters are much fewer than calibration microstructures, we postulate that the simulation model has an insufficient capacity to capture the variation in experiment data. Thus, each microstructure x_i is weighted by $w_i \in [0, 1]$ to indicate priority. In practice, the weight is computed as the product of two components $w_i = w_{1i} w_{2i}$ that reflect two intuitive principles. The first principle is that microstructures with better mechanical properties should be priori-

tized in accuracy as we are most interested in Pareto-optimal samples. To that end, physical measurements $y = (E, T)$ are normalized into $\tilde{y} = (\tilde{E}, \tilde{T})$ in a unit square and a piecewise-linear Pareto front is computed in the normalized performance space. Then, the weight component w_{1i} of microstructure x_i is set according to the Euclidean distance between $\tilde{y}_i$ and the Pareto front, denoted by d_i, via the following function (Figure 2-7)

$$w_1(d) = \begin{cases} 1 & 0 \leq d < 0.2 \\ 1.4 - 2d & 0.2 \leq d < 0.6 \\ 0.2 & d \geq 0.6 \end{cases} \cdot \tag{2.9}$$

The second principle further demands that, while showing preference over Pareto-optimal samples, system identification should not be severely biased by an uneven sample distribution in the gamut. w_{2i} is a penalty term that mitigates the potential risks under such circumstances. We monitor the sample distribution by computing a density heatmap $\rho(\tilde{y})$ over the normalized performance space. The density heatmap is derived from a Gaussian mixture model

$$\rho(\tilde{y}) = \sum_{i=1}^{N} \exp(-\gamma_d \left\| \tilde{y}_i - \tilde{y} \right\|_2^2) \tag{2.10}$$

where γ_d is a parameter that controls the spread of each Gaussian function. Quantitatively, $\rho(\tilde{y})$ estimates the number of calibration samples in the vicinity of any position $\tilde{y}$ variates smoothly between adjacent samples. In a straightforward manner, w_{2i} is defined as a weight discount for every sample x_i whose density value exceeds a certain threshold, ρ_0 (Figure 2-8).

$$w_2(\tilde{y}) = \min \left\{ \frac{\rho_0}{\rho(\tilde{y})}, 1 \right\} \tag{2.11}$$

In this study, we find that $\gamma_d = 200$ and $\rho_0 = 2$ are a desirable parameter setting. It is noteworthy that, although the parameters of weight components are explicitly tuned for microstructure simulation, the design principles of sample weights generalize to other problem domains. Proper modifications can be made when the computational

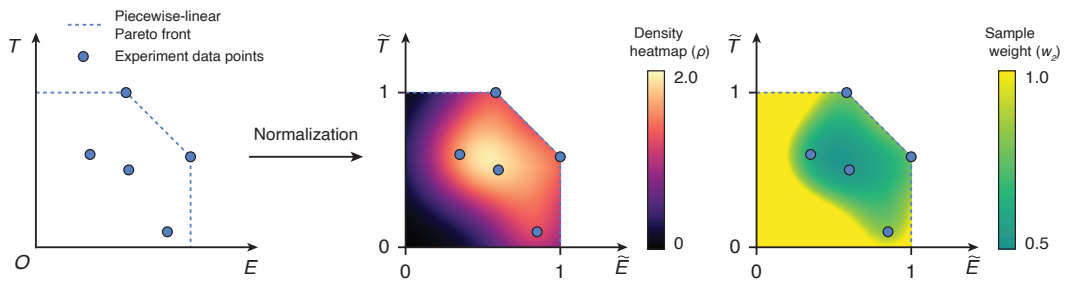


Figure 2-8: An example of deriving the second sample weight component, w_2, in system identification. $\gamma_d = 10$ and $\rho_0 = 1$ are used for illustration purposes.

pipeline is repurposed to a new design optimization task.

We use batched Bayesian Optimization (BO) as the optimizer for both θ and η. The BO algorithm outputs the best solution after 200 iterations with 3 evaluations per iteration, hence performing $600N$ simulations at most. The manually specified feasible ranges of simulation parameters are listed in Table A.1 and A.2. The algorithm is repeated with five different random seeds and the solution that have the smallest loss value is chosen. To harness the high throughput of virtual testers, We implement the algorithm based on the GPyOpt library [16] with a modification that batches all $3N$ simulations in a BO iteration into one simulation run.

2.3.3 Computation of the Pareto Front

Microstructures with optimal trade-offs between simulated Young's modulus and toughness are computed following the system identification step. Essentially, a multi-objective optimization (MOO) problem is solved in the combinatorial design space. Since material assignments to pixels are discrete variables, the MOO problem can be tackled using genetic algorithms like NSGA-II [11]. However, NSGA-II suffers from low sample efficiency attributed to random mutations and crossovers. To reduce the number of samples required, one can use multi-objective Bayesian optimization (MOBO) for a more guided exploration of the design space [31, 8, 32]. Most MOBO algorithms leverage Gaussian process-based (GP-based) surrogate models that make fast and approximative performance predictions without conducting any simulation.

However, the surrogate models are simple and smooth continuous functions and can struggle in approximating sophisticated discrete mappings. Furthermore, since the computation cost of fitting Gaussian processes is cubic to the number of data points, the fitting dataset is often limited to a few hundred in practice, far from adequate to model the complex relationship between microstructure pattern and toughness.

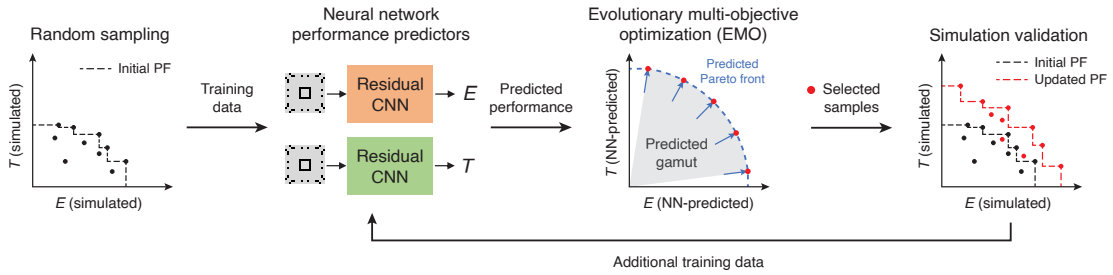


Figure 2-9: Workflow of the learning-accelerated evolutionary MOO algorithm.

Inspired by MOBO, we develop an evolutionary MOO algorithm specially tailored to the design space of microstructure patterns while harnessing the impressive capabilities of deep learning. The workflow of the proposed MOO algorithm is depicted in Figure 2-9. In the beginning, a set of 10 microstructures are randomly sampled from the design space and simulated to obtain their performance, constituting an initial dataset. In every iteration, a surrogate model is fitted to the dataset. 10 predicted Pareto-optimal samples are then computed by solving a surrogate MOO problem where the simulator is replaced by the surrogate model. As the surrogate model is much cheaper to evaluate, we apply NSGA-II to solve the surrogate problem, which runs for 200 generations at a population size of 100. Then, the predicted best samples are verified by simulation before added to the fitting dataset for the next iteration. The algorithm lasts for 500 iterations as dictated by a pre-specified budget of 5000 simulations.

Although the described workflow is similar to MOBO, there several major improvements. The most important one is that the surrogate model is based on convolutional neural networks (CNN), which specialize in image reasoning, instead of Gaussian processes. We model toughness and Young's modulus as functions of a microstructure

pattern using two separate residual networks [23]. Both networks are constructed from the same architecture template demonstrated in Figure 2-10. The template is a concatenation of one stem convolution layer, one or more residual blocks, a global average pooling layer, an optional dropout layer, and a fully connected layer. The network for toughness has a larger capacity, with 5 residual blocks and a penultimate dropout layer for regularization. The network for Young's modulus is much smaller with only 2 residual blocks. In the training phase, both networks are trained on the dataset of microstructures discovered by the algorithm at that moment. The Adam optimizer [30] is used at a learning rate of 1e-3. The network for toughness undergoes 300 training epochs whereas the network for Young's modulus does 100 epochs. The maximum batch size is 2048. To alleviate overfitting on small datasets, the networks are penalized by weight decay coefficients of 1e-2 and 1e-6, respectively.

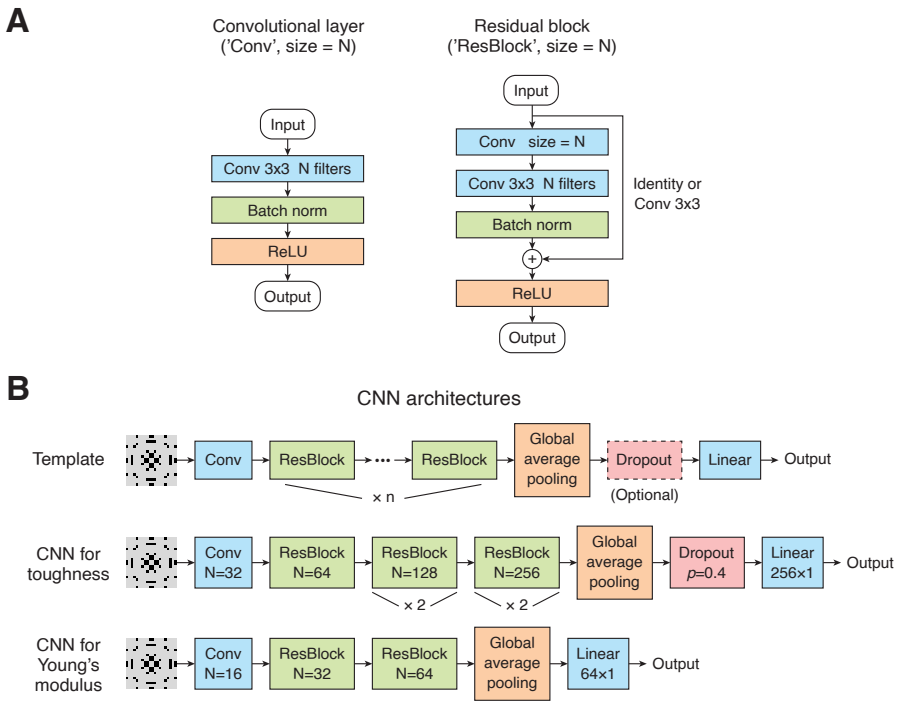


Figure 2-10: CNN architectures used in the learning-accelerated MOO algorithm. A. Definitions of architectural units. B. Prediction networks of Young's modulus and toughness constructed from an architecture template.

The next improvement is a customized mutation operator specially designed for microstructure patterns in NSGA-II. In a canonical NSGA-II implementation, the

mutation operator and the crossover operator strike a strategic balance between exploitation and exploration of the design space. However, for the current task, conventional crossover operators are not very compatible with microstructure patterns as they tend to disrupt key structural features in patterns while introducing changes. Therefore, we opt to omit the crossover step but instead augment the mutation operator to encourage exploration. The customized mutation operator carries out one of the following options stochastically.

- With $p_1 = 0.86$, flip a single random pixel;
- With $p_2 = 0.11$, rewrite a rasterized line of pixels by a unified random value of `0` or `1`, where the starting and ending positions are randomly selected;
- With $p_3 = 0.03$, choose a random pixel as the top-left corner, then rewrite a square region of pixels (the size of the square is randomly chosen) by the complement of the top-left pixel.

After that, symmetry is enforced by mirroring the changes across the entire pattern. Note that the mutated pattern may violate some constraints on the design space. Thus, potential violations are resolved through the procedure below

1. If the occupancy of the stiff base material falls below 50%, the pattern is flipped with four corner pixels reassigned by the stiff base material.
2. If any other violation still exists, the operator keeps on applying new mutation attempts until the violation is cleared.

Our implementation of the mutation operator takes into account that the toughness of a microstructure is more likely affected by geometric structural features in the pattern, such as points, segments, and rectangles. It balances between exploitation and exploration by modifying these structural features at varied scales probabilistically.

We also add a deduplication mechanism in the entire workflow using hash tables. At a high level, it prevents the algorithm from rediscovering previous samples in later iterations. More importantly, it is vital to NSGA-II as an effective means of escaping

local optima when solving surrogate problems. Normally, single-pixel mutations easily result in duplicate samples. In the presence of a hash table, however, the mutation operator automatically switches to larger perturbations once the hash table entries of all neighboring patterns are filled, thus enforcing exploration. The hash value of a microstructure pattern (viewed as a flattened array of binaries $x = (b_1, b_2, \cdots, b_{400})$) is computed as

$$\text{hash}(x) = \left(\sum_{i=1}^{400} b_i y^{i-1} \right) \bmod k \tag{2.12}$$

where $y = 99997$ and $k = 10^8 + 7$ are prime numbers. Each hash table has k entries. While a global hash table is associated with the entire algorithm, each run of the surrogate problem solver uses a separate table. After the predicted Pareto-optimal samples are computed, they are ordered by non-dominated sorting and checked for hash collision inside the global hash table. The first 10 collision-free samples are selected for simulation verification.

2.3.4 Selection of Validation Samples

The gamut given by the MOO algorithm typically contains a large pool of samples and only a few of them will be selected for physical validation due to a limited experiment budget. Overall, the selected samples should cover a wide performance range and maximally distinguish from existing samples in appearance. To fulfill these requirements, we proposed a two-stage selection approach visualized in Figure 2-11.

The first stage extracts samples closest to the gamut boundary. This step can be done in various ways depending on the shape of the gamut. We adopt a simple method where samples in the gamut are ranked in the order of Pareto dominance. By definition, Pareto-optimal samples of the entire gamut are of the 1st rank, and samples that fall into the i-th rank are only dominated by those of higher ranks. We regard samples up to the 3rd rank as near the boundary. Meanwhile, the gamut is intersected by two benchmark lines l_1 and l_2. l_1 is a polyline that connects the physical measurements of homogeneous composites. l_2 is a horizontal line going through the sample with the largest Young's modulus in simulation. Samples that are both near

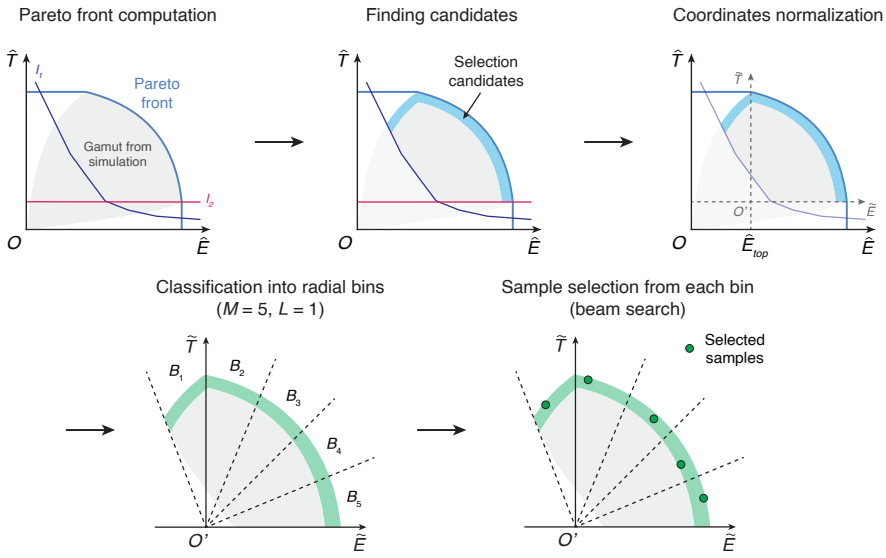


Figure 2-11: The selection process of validation microstructures near the Pareto front. $M = 5$ and $L = 1$ are used in this illustration for simplicity.

the boundary and above the benchmark lines are deemed as candidates for selection.

In the next stage, the candidate samples, $S \subset X$, are classified into two groups of radial bins $B_1, B_2, \cdots, B_L$ and $B_{L+1}, B_{L+2}, \cdots, B_M$. M denotes the total number of samples to select, and L denotes the number of samples to select from the first group and that $0 \leq L < M$. To that end, we map the properties of samples in S to a normalized performance space in $[-1, 1] \times [0, 1]$. While both properties undergo a linear transformation for the mapping, there are two circumstances related to Young's modulus. As shown in Figure 2-11, the Young's modulus of the toughest sample $\hat{E}_{top}$ is mapped to zero by default. However, the resulting image will not fit into $[-1, 1]$ if $\hat{E}_{top}$ exceeds the midpoint of the Young's modulus range, so the midpoint is mapped to zero in that case. Within the normalized performance space, the radial bins are outlined by $M + 1$ rays shooting from the origin, dubbed as $r_1, r_2, \cdots, r_{M+1}$. r_{L+1} and r_{M+1} always align with the coordinate axes no matter what value L might take. If $L > 0$, r_1 passes through the sample in S that has the largest elevation angle φ_{max}. The i-th radial bin B_i refers to the area between r_i and r_{i+1}, and we let S_i denote the subset of S classified into B_i. $B_1, B_2, \cdots, B_L$ are equally-sized and located in

the second quadrant, while the other bins are equally-sized and located in the first quadrant. The value of L is determined by φ_{max} using the rules below

$$L = \begin{cases} 0 & \varphi_{max} < \dfrac{\pi}{2} + \varphi_{th} \\ \left\lceil \left(1 - \dfrac{\phi}{2\varphi_{max}}\right) M \right\rceil & \varphi_{max} \geq \dfrac{\pi}{2} + \varphi_{th} \end{cases}, \tag{2.13}$$

where $\varphi_{th} = \pi/36$ (i.e., 5°). Practically, when φ_{max} is at least 95°, L is a minimal number such that the bins in the second quadrant are no larger than those in the first quadrant.

To decide the choice of sample from each subset, the following combinatorial optimization problem is solved to search for a group of microstructure patterns that differentiate from physically measured ones as much as possible. Let R denote the set of microstructures that have been measured and $\Omega = (x_1, x_2, \cdots, x_M)$ (where $x_i \in S_i$ for $i = 1, 2, \cdots, M$) denote a possible selection. The objective is to maximize the average L1-distance of each x_i to the rest of the samples in $R \cup \Omega$.

$$\Omega^* = \underset{\Omega}{\operatorname{argmax}} \frac{1}{M} \sum_{i=1}^{M} \min_{u \in R \cup \Omega \setminus \{x_i\}} \|x_i - u\|_1 \tag{2.14}$$

As the problem is likely intractable by enumeration when S contains over 100 samples, we use a beam search algorithm to find an approximate solution. The beam size is set to 100 to preserve the quality of the solution at an acceptable computation cost.

2.4 Analysis on Pareto-Optimal Microstructures

Inspired by [10], we postulate that the Pareto front of the gamut comprises multiple families of samples that resemble one another in patterns and properties. Identifying and analyzing such families helps reveal the common intrinsic toughening mechanisms shared across many structurally similar samples. Thus, among all physically measured microstructures discovered by the pipeline, we group the ones with Pareto-optimal Young's modulus and toughness into families, where each microstructure is

expanded into a subfamily with slight pattern variations. By measuring the properties of several more validation samples, we summarize the most representative toughening mechanisms in each family. Furthermore, we compute a low-dimensional embedding space per subfamily and build an interpolation model for neighboring patterns in the embedding space, allowing for an intuitive exploration of microstructure patterns in these families. Figure 2-12 demonstrates the workflow of such procedure.

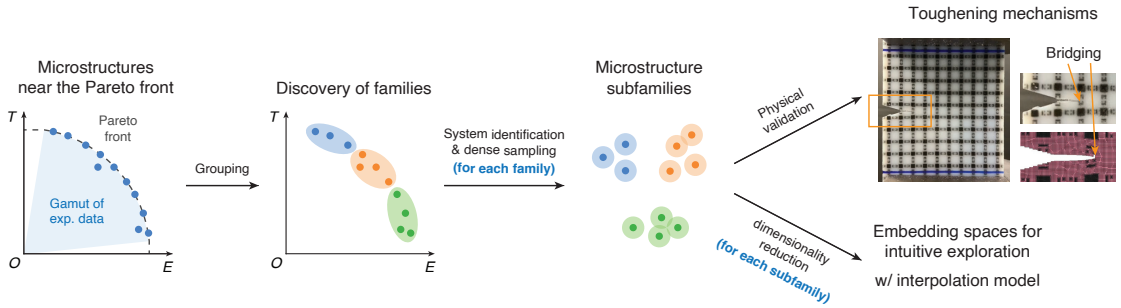


Figure 2-12: Workflow of post-analysis on discovered Pareto-optimal microstructures from the computational pipeline.

2.4.1 Discovery of Microstructure Families

Microstructure families are discovered from microstructures on or near the Pareto front of all physical measurements after the non-cooperative game terminates. We manually group these near Pareto-optimal samples into families by their patterns and mechanical properties. For each family, we run system identification once more to obtain an exclusive simulation model. The system identification step involves all near Pareto-optimal samples but assigning them with different weights. Samples in the family have a constant weight of 1, while others are weighted by 0.1 for regularization purposes. Compared with the global simulation model calibrated on all physically measured microstructures, a family-specific model is better at predicting properties within that family.

To create more pattern variations in a family, we perform dense sampling around initial family members, treating them as seed patterns to mutate and generate variants. The sampling algorithm starts with the seed patterns in the family. Other

patternsare generated iteratively using the mutation operator introduced in Section 2.3.3. The only exception is that mutation attempts are not applied consecutively in the violation of structural constraints. Instead, the operator rolls back changes before proposing a new mutation attempt as larger-scale exploration is unnecessary. New patterns are restricted within a maximum of 16-pixel difference (which is 4% of all pixels) from the closest seed pattern. Likewise, the hash table in Section 2.3.3 is adopted for deduplication. In every iteration, the algorithm keeps generating mutated samples until 10 non-existing samples are found, after which they are evaluated using the family-specific simulation model. As hash table entries are rapidly filled, it becomes increasingly hard to sample non-existing patterns over time. Thus, the algorithm is allowed to generate a maximum number of 3×10^5 mutated samples per iteration, and stops when it fails to complete an iteration under the limit.

Following our practices in the computational pipeline, we validate the simulation model pertaining to a family by conducting physical measurements on samples with the best predictions. After dense sampling, several samples are selected from the family gamut using the algorithm in Section 2.3.4. Toughening mechanisms are analyzed from video recordings of validations samples during mechanical testing through human observation.

2.4.2 Intra-Family Analysis

More in-depth analysis is conducted within each microstructure family to uncover essential toughening mechanisms that contribute to optimal performance trade-offs. Samples in each family are first divided into subfamilies according to similarity with seed patterns, where a subfamily consists of a seed pattern and all samples within a 16-pixel difference as specified in dense sampling. Then, a 2-D embedding space is extracted from microstructure patterns in a subfamily using Isomap [6]. Regarding the distance metric for Isomap, we use the earth mover's distance (EMD) rather than the Euclidean distance. EMD, or the 1st Wasserstein distance [46], is a metric between probability distributions originally defined in optimal transportation theory [27]. As an analogy, it measures the minimal work (calculated by mass times distance)

required to redistribute a pile of earth into another. Compared with the Euclidean distance, EMD better reflects the shape difference between microstructure patterns by taking into account both pixel values and locations. Since EMD does not directly apply to microstructure patterns, they are converted into quarter-sized normalized histograms that represent discrete probability distributions. The conversion rules, illustrated in Figure 2-13, treat pixels of the soft base material as shape features and put non-zero values in the corresponding cells of the output histogram. Conversely, the other cells associated with the rigid material are set to zero. Therefore, the EMD between two microstructures x_a and x_b is given by the 1st Wasserstein distance between their normalized histograms h_a and h_b ($h_a, h_b \in [0,1]^{100}$).

$$\begin{aligned} \mathrm{EMD}(x_a, x_b) &= W_1(h_a, h_b) \\ &= \min_{\pi \in \Pi(h_a, h_b)} \pi \otimes D. \end{aligned} \tag{2.15}$$

In the equation above, $\Pi(h_a, h_b)$ is the collection of transportation plans, π, from h_a to h_b, given by

$$\Pi(h_a, h_b) = \left\{ \pi \in \mathbb{R}^{100\times 100} \,\middle|\, \pi \geq 0,\ \pi \mathbf{1} = h_a, \text{ and } \pi^{\mathrm{T}} \mathbf{1} = h_b \right\}. \tag{2.16}$$

Back to the pile-of-earth analogy, π tabulates the amount of earth to transfer from each cell to the others. D is the pairwise Euclidean distance matrix of histogram cells. $A \otimes B = \langle A, B \rangle_{\mathrm{F}} = \mathrm{tr}(A^{\mathrm{T}} B)$ is the Frobenius inner product of two identically shaped matrices.

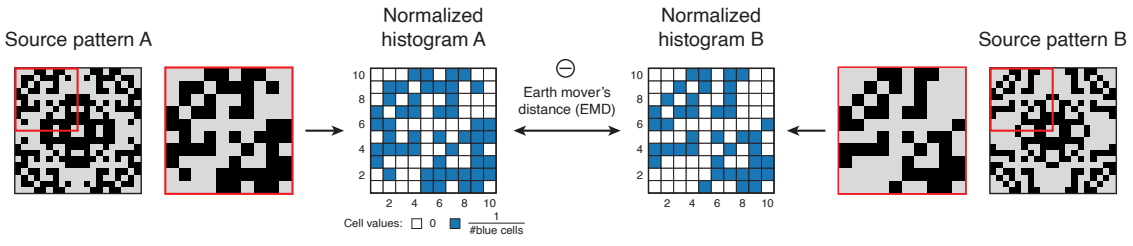


Figure 2-13: Calculation of the earth mover's distance (EMD) between two microstructure patterns.

The Isomap embedding space of a subfamily can be further refined by samples not covered in dense sampling. To generate these samples, we build an interpolation model for K neighboring microstructure patterns $x_1, x_2, \cdots, x_K$ in the embedding space based on Wasserstein barycenters [2] which is widely used in shape interpolation [45]. The workflow of interpolation is shown in Figure 2-14. Every source pattern is decomposed into a common part containing shared shape features and an exclusive part containing distinctive features. The exclusive parts are converted into quarter-sized normalized histograms denoted by $h_1, h_2, \cdots, h_K$. Our model interpolates among the histograms by computing their Wasserstein barycenter $\bar{h}$, weighted by $w_1, w_2, \cdots, w_K$, defined as

$$\bar{h} = \underset{h}{\operatorname{argmin}} \sum_{i=1}^{K} w_i W_2(h_i, h). \tag{2.17}$$

$W_2(\cdot, \cdot)$ refers to the 2nd Wasserstein distance,

$$W_2(h_a, h_b) = \min_{\pi \in \Pi(h_a, h_b)} \pi \otimes D_2, \tag{2.18}$$

where $D_2 = D \circ D$ is the pairwise squared distance matrix ($\circ$ refers to the Hadamard product). The minimization problem is tackled by solving the following equivalent linear programming problem [2] for optimal transportation plans π_i^* from each h_i to $\bar{h}$ $(i = 1, 2, \cdots, K)$.

$$\begin{aligned} \min_{\{\pi_i\}} \quad & w_i \left(\pi_i \otimes D_2\right) \\ s.t. \quad & \pi_i \geq 0, \quad \forall i = 1, 2, \cdots, K \\ & \pi_i \mathbf{1} = h_i, \quad \forall i = 1, 2, \cdots, K \\ & \pi_i^{\mathrm{T}} \mathbf{1} = \pi_1^{\mathrm{T}} \mathbf{1}, \quad \forall i = 2, 3, \cdots, K \end{aligned} \tag{2.19}$$

After that, $\bar{h}$ is obtained by $\bar{h} = \left(\pi_1^*\right)^{\mathrm{T}} \mathbf{1}$. To convert the interpolated histogram back to a microstructure pattern, $\bar{h}$ is quantized into a binary matrix that indicates the material assignments of the upper-left quarter of the resulting pattern. Let c_i be the

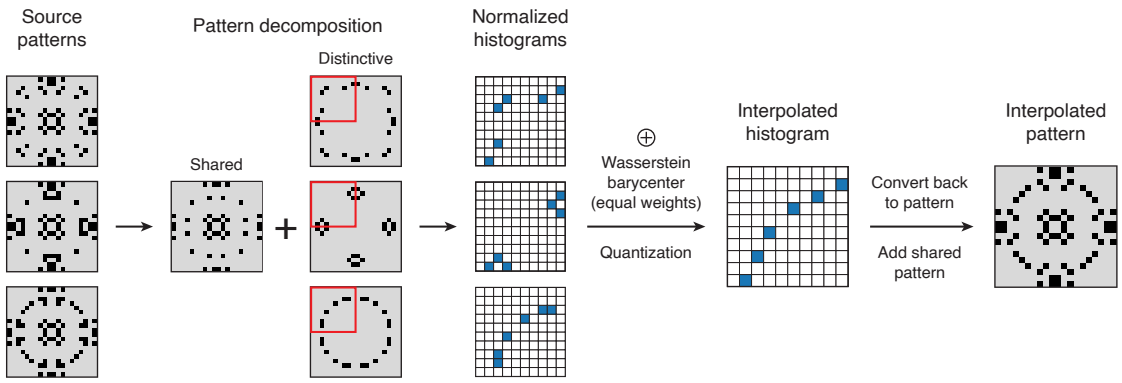


Figure 2-14: Interpolation of neighboring microstructure patterns in the Isomap embedding space based on Wasserstein barycenters.

unified value of all non-zero cells in h_i and $c = \left(\sum_{i=1}^{K} c_i\right)/K$. A cell in $\bar{h}$ translates into the soft base material if its value is no less than c; otherwise, it translates into the rigid material.

In practice, a Delaunay triangulation is computed in the 2D embedding space to determine neighboring samples, namely $K = 3$. Neighborhoods whose longest edge falls between 4% and 40% of the diameter of the family (the distance between two furthest points in the embedding space) are marked as candidates. New samples are generated from uniform grid interpolation in candidate neighborhoods and added to the subfamily if they are within a 16-pixel radius from the seed pattern while not violating any structural constraint. Finally, the refined embedding space is recomputed using Isomap.

Chapter 3

Results

In this chapter, we demonstrate experimental and computational results related to the proposed microstructure discovery approach. The content is arranged in three categories. As the foundation of physical measurements, we first evaluate the improvement in toughness from interface engineering. Then, moving to the computational pipeline, we show the entire competitive game in action using results from both participating agents and discovered Pareto-optimal microstructures. Also included are performances of key algorithms in the pipeline, i.e., simulation, system identification, and the learning-assisted exploration algorithm. Finally, we elaborate more on our findings after analyzing those Pareto-optimal microstructures, including microstructure families, representative toughening mechanisms, and embedding spaces of subfamilies.

Some common notations are used across the whole chapter. E and T denote Young's modulus and toughness, respectively. The default unit for them is Pa unless stated otherwise in figures. 'VW%' refers to the percentage of the rigid base material in a microstructure pattern.

3.1 Interface Engineering

We compare our method of engineering the interface between base materials (Section 2.2.3) against two baseline approaches. By default, the 3D printer applies prescribed

droplet configurations in interface areas which are invisible to the user. Conversely, the interface is completely user-specified in voxel printing mode. Therefore, it is possible to make a sharp transition between pixels of different base materials and perform no engineering. To test these interface options, we randomly drew 5 samples from microstructures containing 50% to 75% rigid base material (Figure 3-1) and measured their mechanical properties. The measurements are listed in Table 3.1.

Figure 3-1: Random microstructures used for testing the interface between base materials, numbered by 1-5 from left to right.

Table 3.1: Comparison of microstructure properties under three interface configurations: default to the printer (def.), not engineered (no eng.), and engineered (ours). The best values are marked in bold.

Sample	**VW%**	E **(GPa)**			T **(MPa)**		
		Def.	No eng.	Ours	Def.	No eng.	Ours
1	51	0.233	0.339	**0.488**	0.497	0.384	**0.504**
2	61	0.463	0.629	**0.784**	0.339	0.280	**0.420**
3	63	0.638	0.814	**0.979**	0.366	0.253	**0.457**
4	67	0.796	0.882	**1.074**	**0.330**	0.223	0.231
5	75	0.895	1.033	**1.294**	**0.293**	0.150	0.286

As seen in the results, microstructures with engineered interfaces have much higher Young's moduli than default interfaces. There is a stable increase from 35% to more than 100% over the latter. Non-engineered interfaces sit in between but still show a decent advantage over default. This is likely due to a change in droplet configurations outside interface areas as well. On the toughness side, engineered and default interfaces have their own strengths. Nonetheless, more than half of testing samples exhibit better properties after interface engineering. Therefore, interface engineering has better chances of improving sample properties overall.

We note that Microstructure 2 and 4 might have different physical measurements when they appear in later sections. This is mainly because experiments were con-

ducted at different times in a year and conditions of the 3D printer were subject to change, which can affect the properties of printed specimens. As such, results reported in the same table/figure were recorded consecutively so the influence is negligible.

3.2 Computational Pipeline

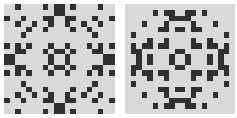
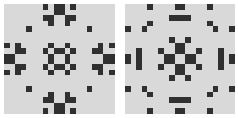
In practice, the competitive game was held between a proposal agent and a validation agent as follows.

In the initial round, the validation agent conducted physical measurements on two base materials plus the interface material and then matched simulation and experiment using system identification. Each material was involved in a separate system identification process that only estimated simulation parameters relevant to that material (Young's modulus, Poisson's ratio, energy density threshold, and damping constant). Among three estimates of the damping constant, the median was picked as the final guess. Using calibrated virtual testers provided by the validation agent, the proposal agent computed the gamut and the Pareto front to find samples that maximized the sim-to-real gap. We used NSGA2 equipped with the proposed mutation operator and hash table to compute gamuts here, and left the learning-assisted MOO algorithm to a separate comparison with other MOO algorithms. The NSGA2 was run in five different random seeds using a simulation budget of 2×10^4 samples. Results were merged into a gamut that had approximately 10^5 samples, after which 8 validation samples were selected near the Pareto front and transferred back to validation agent for the next round.

Starting from the second round, the validation agent performed system identification using a calibration dataset that only contained physically measured microstructures. The dataset had 10 initial random microstructures coming from stratified sampling based on VW% (Table 3.2). Microstructure patterns in each interval were generated by randomly selecting an integer VW% value and filling an empty matrix with a corresponding number of 1s. We note that these random microstructures help regularize system identification in early rounds of the game and do not put too much

burden on the simulation model. The proposal agent continued executing the same tasks, computing the Pareto front and sending 8 new validation samples back to the validation agent. Thus, the calibration dataset had $8n + 2$ samples in the n-th round ($n \geq 2$). The game roughly reached an equilibrium at $n = 4$.

Table 3.2: Initial microstructures in the calibration dataset categorized by VW%.

VW% interval	**Patterns**
50 - 59	
60 - 69	
70 - 79	
80 - 85	

For the rest of this section, we demonstrate the competitive game progress by visualizing the evolution of simulation and experiment data. Then, we show that the discovered microstructures of the computational pipeline have much better trade-offs in Young's modulus and toughness than homogeneous composites. We also present evaluation results of key algorithms in the pipeline to prove that they successfully address the target challenges.

3.2.1 Bridging the Sim-to-Real Gap

The competitive game process is visualized in Figure 3-2. Each subfigure is a snapshot of the game at the end of a round, where we show physical measurements of all discovered microstructures, the gamut found by the exploration algorithm, selected validation samples near the gamut boundary, and the benchmark line of homogeneous material properties. Game progress is primarily determined by the evolution of Pareto

fronts from simulation and experiment. The Pareto front of experiment data expands as the number of samples increases. Meanwhile, the Pareto front of the gamut shrinks, leading to a narrowing sim-to-real gap.

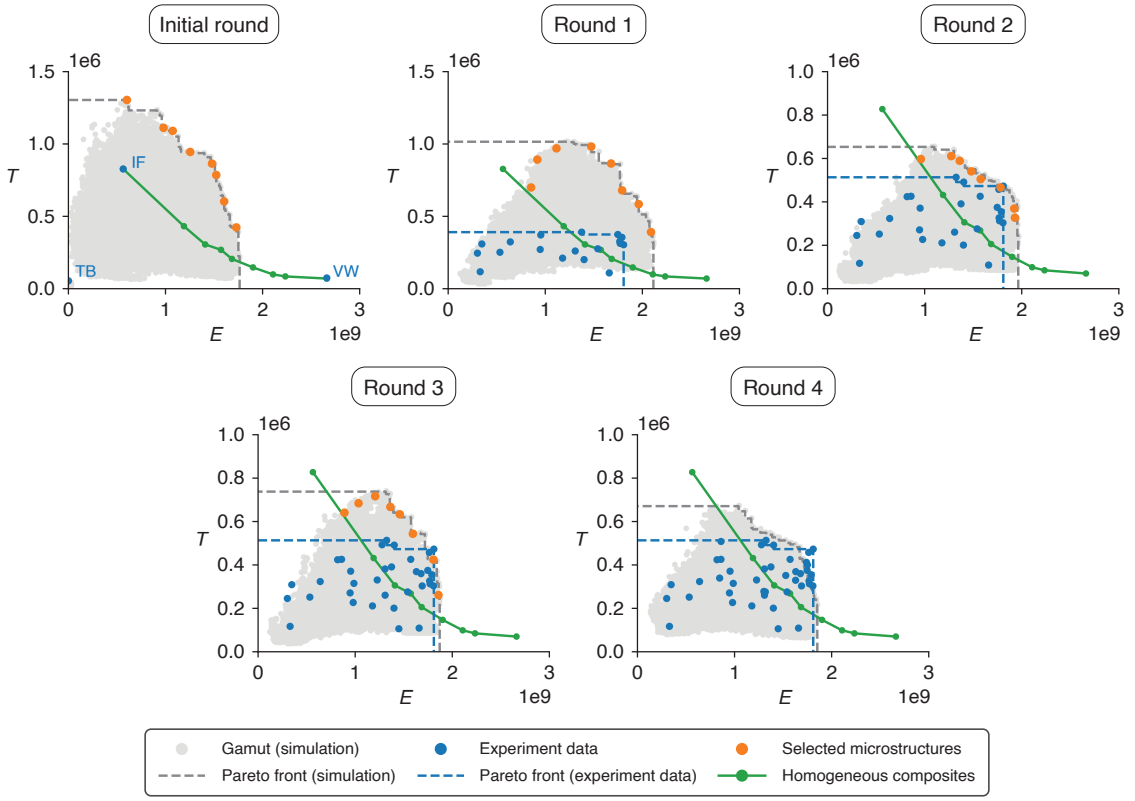


Figure 3-2: Evolution of simulation and experiment data in the competitive game.

The trends are reflected quantitatively in the evolution of Pareto hypervolume (Figure 3-3), i.e., the area enclosed by the Pareto front. The hypervolume of experiment data rises until Round 2 and becomes flat, implying that no better samples were discovered in later rounds. Nonetheless, the performance ranges covered are on par with what we can achieve from hundreds of physical measurements throughout the course of this work. The gamut hypervolume shows a statistic decline despite slight fluctuation. We measure the size of the sim-to-real gap using the area between simulation and experiment Pareto fronts. The sim-to-real gap diminished by an order of magnitude after the game finished. As suggested in Figure 3-2, the gap mostly resides in toughness simulation, which is normal since toughness simulation is

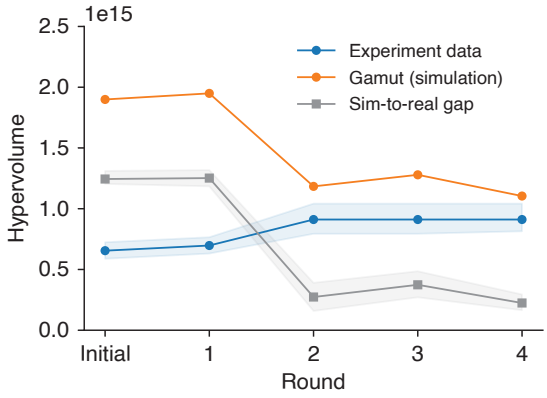


Figure 3-3: Evolution of Pareto hypervolumes of experiment data, the gamut, and the sim-to-real gap. The Pareto hypervolume of experiment data in the initial round is computed using 10 initial calibration samples. The colored region around a curve refers to error ranges estimated from standard deviations of physical measurements.

inherently more challenging than Young's modulus.

3.2.2 Discovered Microstructures

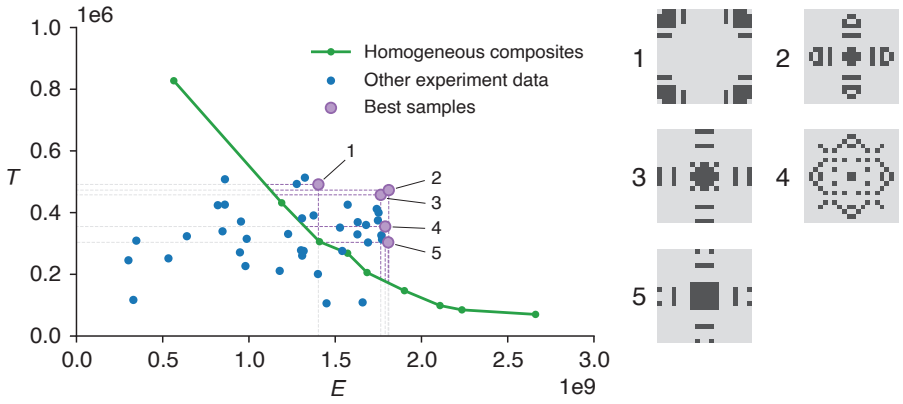


Figure 3-4: Five best microstructures found by the computational pipeline. Patterns of all discovered microstructures are tabulated in Figure B-1.

Microstructures discovered by the pipeline are compared against homogeneous composites in Figure 3-4. We consider a microstructure successful in defeating homogeneous counterparts if its physical measurements is above the benchmark line. Among the successful microstructures, we pick five best samples near the Pareto front and having more than 1.5× improvement in toughness from the benchmark line

(Table 3.3). Sample 2 shows the maximum improvement in both Young's modulus and toughness at 1.61× and 2.76×, respectively. Sample 1 and 3 also take a lead of more than 2× in toughness. These results are sufficient to prove that microstructured composites have a remarkable advantage over homogeneous composites. With computationally designed patterns, they are able to present much better trade-offs between stiffness and toughness. We also calculated the toughness improvement of best samples over base materials (VW has a higher toughness at 0.070 MPa than TB), where Sample 1 unsurprisingly outperforms VW by 7.01×. It is clear that homogeneous composites make a stronger baseline. Yet, they were barely mentioned in previous works.

Table 3.3: Properties of five best samples and their improvements relative to homogeneous composites and base materials. The highest values are marked in bold.

Sample	**VW%**	**Properties**		**Improvement (homogeneous)**		**Improv. (base)**
		E (GPa)	T (MPa)	E	T	T
1	79	1.402	0.491	1.35	2.00	**7.01**
2	85	1.810	0.473	**1.61**	**2.76**	6.75
3	85	1.764	0.458	1.54	2.48	6.53
4	84	1.790	0.355	1.27	1.76	5.07
5	85	1.807	0.304	1.28	1.59	4.33

3.2.3 Simulation Time

We evaluated simulation efficiency by timing the virtual testers under different batch sizes and strain limits. All performance data was collected on a machine with 8 CPU cores and an NVIDIA Tesla V100 GPU, averaged across 5 consecutive runs. The results are shown in Figure 3-5. For Young's modulus, simulation only becomes slower when at least 20 microstructures are batched together. While increasing the batch size beyond this point leads to a much longer simulation time, the throughput continues to improve, implying that GPU utilization is still approaching its maximum. A batch size of 100 is required to reach a peak throughput of less than 0.015 s per sample. For toughness, the highest throughput occurs at a batch size of 20, where

it takes less than 0.5 s to stretch a sample to $\varepsilon = 0.1$. Relative to the sequential setting, our batching mechanism brings about a 30× boost in throughput for Young's modulus and more than 3× for toughness regardless of the strain limit.

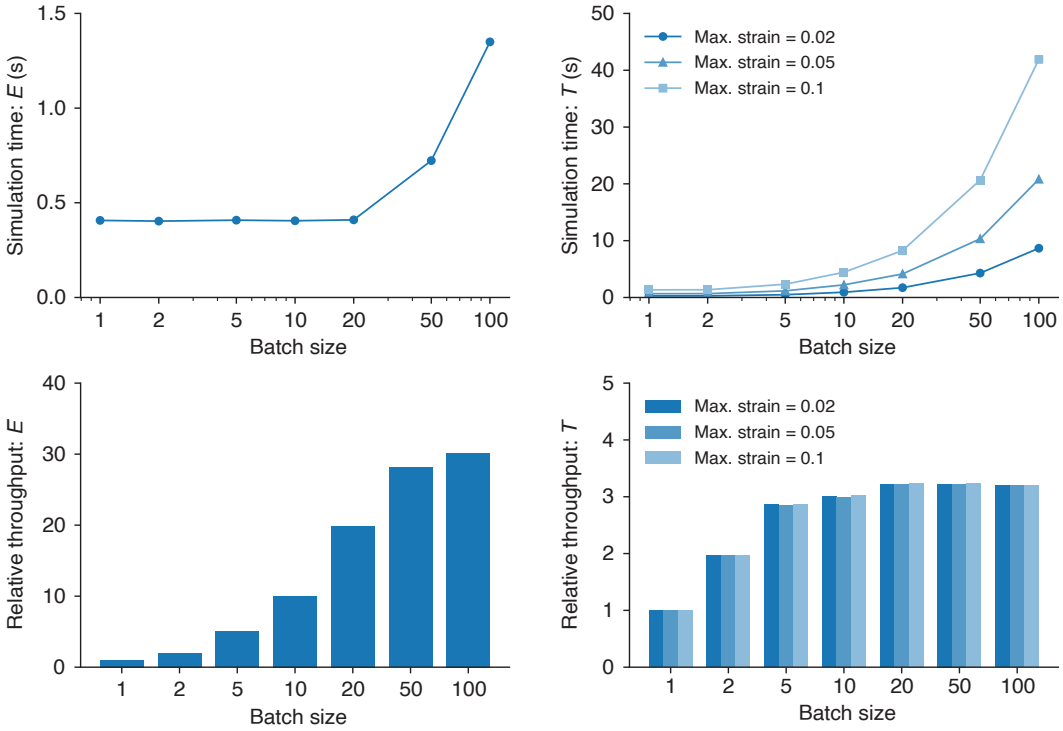


Figure 3-5: The time consumption and throughput of virtual testers under various input batch sizes.

3.2.4 System Identification

In the initial round of the competitive game, we fitted simulation to base materials and the interface material as a reference prediction model (Table 3.4). System identification handles this small case at ease and achieves almost perfect accuracy. Young's modulus predictions are within 0.1% off from measurements and toughness prediction errors stay below 1%. Besides, the simulated stress-strain curves align reasonably well with experimental ones (Figure 3-6). Although we uses fairly simplified virtual testers that trade off modeling capabilities for speed, system identification manages to match the dimensions of the curves and capture shape features to a decent amount by optimizing simulation parameters.

Table 3.4: Simulation predictions (Sim.) and relative errors (Err.) from physical measurements (Exp.) after system identification on each base material separately.

Sample	E **(GPa)**		**Err.**	T **(MPa)**		**Err.**
	Exp.	Sim.	(%)	Exp.	Sim.	(%)
TB	1.0094e-3	1.0091e-3	0.031	5.4335e-2	5.4271e-2	0.118
VW	2.6613	2.6613	0.0002	7.4217e-2	7.4484e-2	0.360
IF	0.5642	0.5642	0.003	0.8274	0.8193	0.984

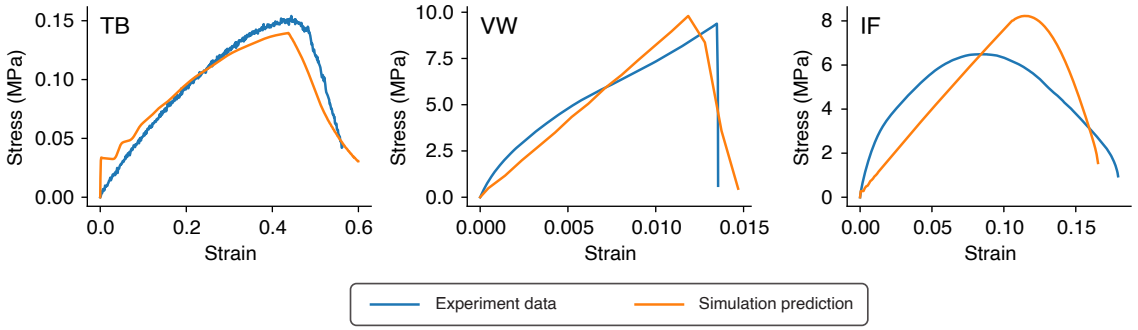


Figure 3-6: Comparison between stress-strain curves from simulation and experiment after system identification on each base material separately.

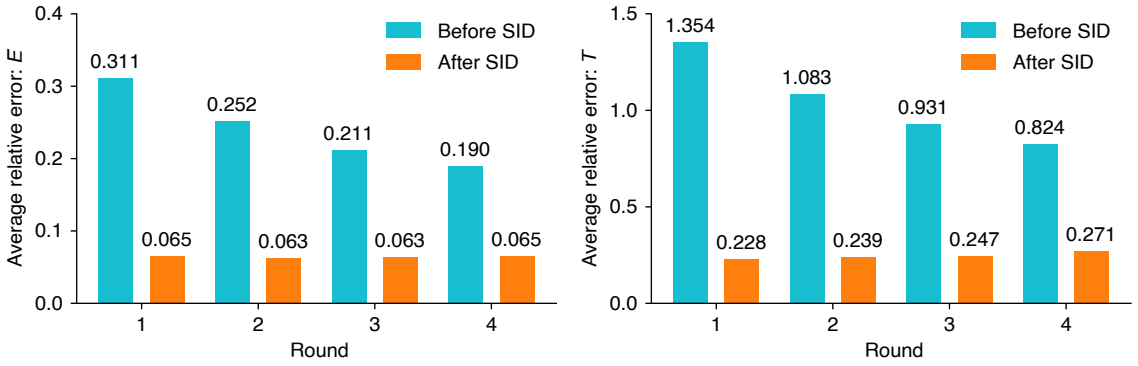


Figure 3-7: Simulation prediction errors before and after system identification (SID) in each competitive game round.

However, weaknesses of the reference model emerge when it comes to predicting microstructure properties. We compared between the reference model and the calibrated model in average relative prediction errors from Round 1 to Round 4. Results are demonstrated in Figure 3-7 and Table 3.5 (full data in Table A.3 and A.4). In Young's modulus predictions, the reference model has around 19% - 31% errors,

Table 3.5: Simulation prediction errors before and after system identification in each competitive game round. N denotes the number of calibration samples.

Round	N	Rel. error: E (%)				Rel. error: T (%)			
		Before		After		Before		After	
		Avg.	Max.	Avg.	Max.	Avg.	Max.	Avg.	Max.
1	18	31.09	93.15	**6.53**	22.95	135.59	767.44	**22.85**	62.23
2	26	25.19	93.15	**6.29**	20.40	108.26	767.44	**23.92**	100.53
3	34	21.11	93.15	**6.32**	22.75	93.09	767.44	**24.68**	84.30
4	42	18.95	93.15	**6.50**	25.40	82.38	767.44	**27.10**	88.87

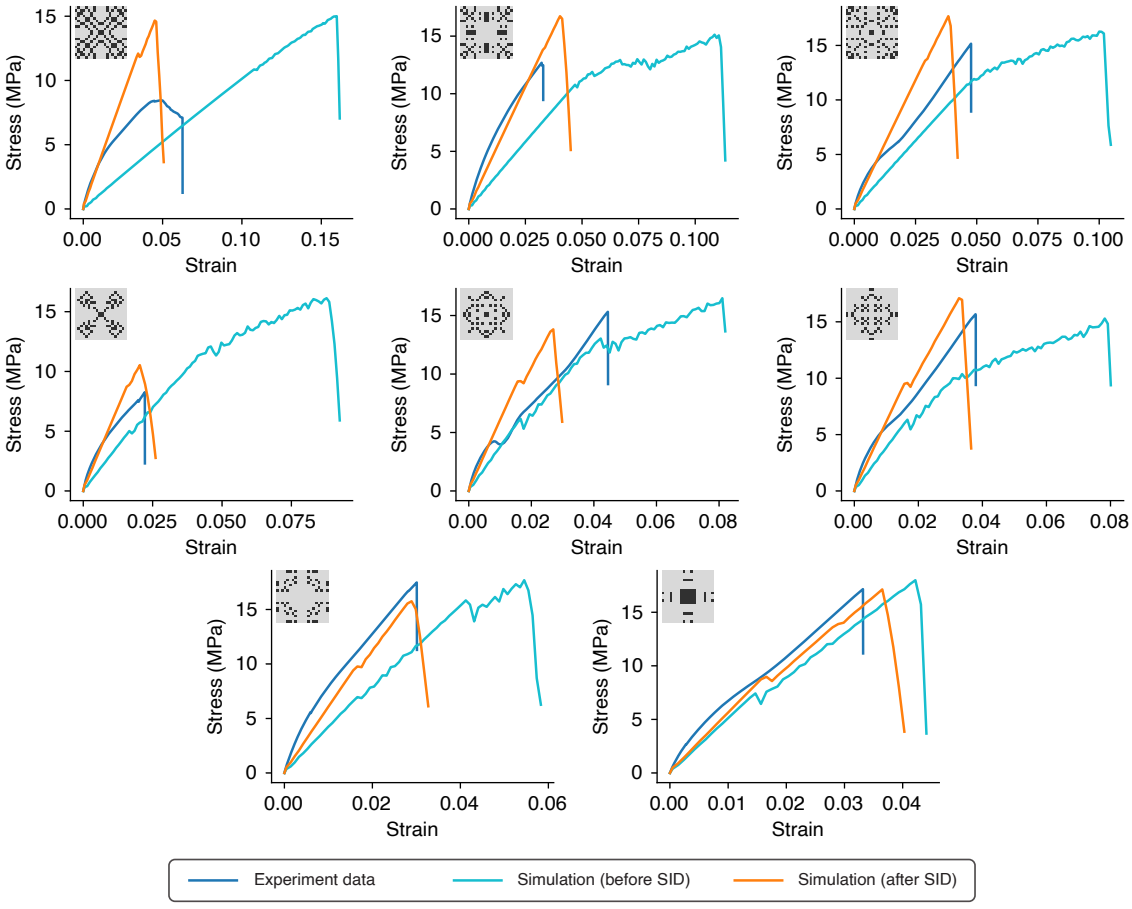


Figure 3-8: Simulated stress-strain curves before and after system identification (SID) compared against physical measurements in Round 1. Selected microstructures were proposed in the initial round.

but the numbers steadily reduce to about 6.5% after system identification. The improvement is even more obvious in toughness predictions. The reference model has

poor accuracy across the board with 82% - 135% errors in average. In contrast, the calibrated model is only 23% - 27% off. Such notable differences are also seen in stress-strain curves (an example is provided in Figure 3-8). Predictions from the calibrated model are qualitatively much closer to experiment data.

From a theoretical perspective, multiscale material characteristics is a primary reason why fitting virtual testers to base materials does not result in good predictions for microstructures. The base materials exhibit bulk properties as a homogeneous specimen but are confined within submicron voxels in microstructures. Thus, their properties can not be directly modeled using macroscale physical measurements. Material interfaces add more complication to the equation here now that a microstructure pixel contains merely a few droplets, and they are susceptible to manufacturing errors at this scale. All variables factored in, predicting microstructure properties proves to be a very challenging task especially for toughness. However, our data-driven system identification approach circumvents these difficulties using a black-box physics-based simulator. The simulation model achieves a 27% average relative error in 42 microstructure predictions despite a limited capacity, setting a state-of-the-art accuracy record.

3.2.5 Learning-Accelerated Exploration

To validate the exploration efficiency of our learning-accelerated evolutionary MOO algorithm, we observed the growth of the gamut in Pareto hypervolume with respect to the number of simulated samples (Figure 3-9). The same test was done on our NSGA-II implementation that generated results for the computational pipeline. It does not include CNN surrogate models but uses the customized mutation operator and the hash table. The baseline algorithms include two simpler versions of NSGA-II whose mutation operators only modify an individual pixel at a time, dubbed as 'baseline 1' and 'baseline 2'. The only difference between them is that the latter rolls back a mutation attempt when design constraints are violated before trying the next one. Hence, NSGA-II baseline 2 is more conservative in exploration. We added the hash table to them after finding that both went stuck quickly at local minima and

could not explore the design space properly otherwise. Also used for reference are two MOBO algorithms, DGEMO [32] and TSEMO [8]. DGEMO has state-of-the-art performance in synthetic MOO benchmarks, while TSEMO is an older but more popular alternative.

For data collection, the learning-accelerated algorithm was run for 500 iterations. NSGA-II algorithms did 2000 iterations. MOBO algorithms were capped by 200 iterations due to extremely slow fitting of Gaussian processes after a few hundred samples. 10 samples were simulated per iteration in all cases. NSGA-II algorithms were terminated if they failed to complete an iteration after proposing 3×10^5 mutated samples. Each algorithm was repeated on the same five random seeds which guarantees the same starting samples. The resulting Pareto hypervolume curves were averaged.

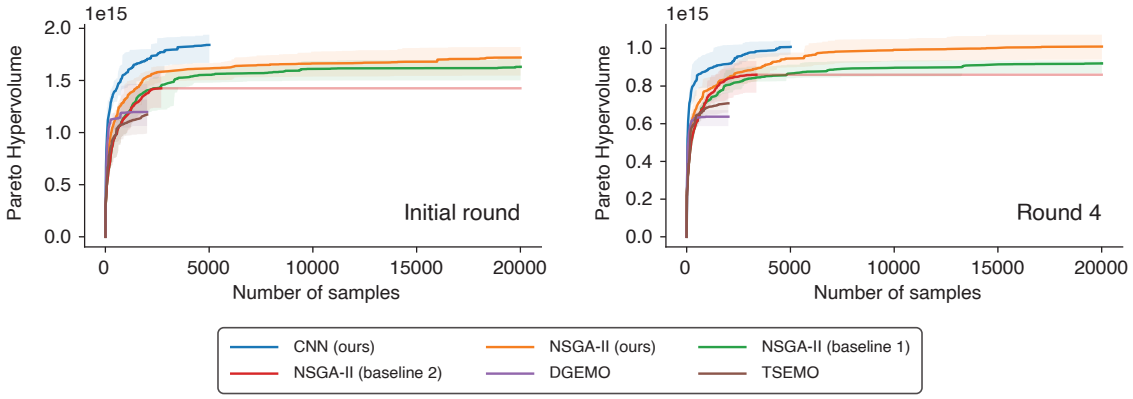


Figure 3-9: Pareto hypervolume growth of the gamut under various evolutionary MOO algorithms (NSGA-II, MOBO, and ours) in the initial round and Round 4.

Figure 3-9 demonstrates the Pareto hypervolume growth of each algorithm in two example test cases. Each solid curve is an average. The colored region around a curve indicates value ranges. Both cases see a clear victory from the learning-accelerated algorithm which yielded the fastest growth and covered the largest final hypervolume. It simulated less than 1500 samples (1320 and 1060, respectively) before reaching the final hypervolume of baseline NSGA-II algorithms, which used 20000 samples in contrast. This translates to a 15.16× and 18.88× improvement in sample efficiency. The other test cases are shown in Figure B-2 where the improvements stay around an order of magnitude, indicating a robust performance. Benefiting from the

customized mutation operator, our NSGA-II version also pulls ahead of baselines by a margin. Notably, the curve of NSGA-II baseline 2 is extended virtually because all five runs underwent early termination. This suggests that the algorithm got stuck at local minima without chances to escape using consecutive mutations. The MOBO algorithms severely underperformed due to weaknesses of Gaussian processes such as an incompetent modeling capability and high fitting cost.

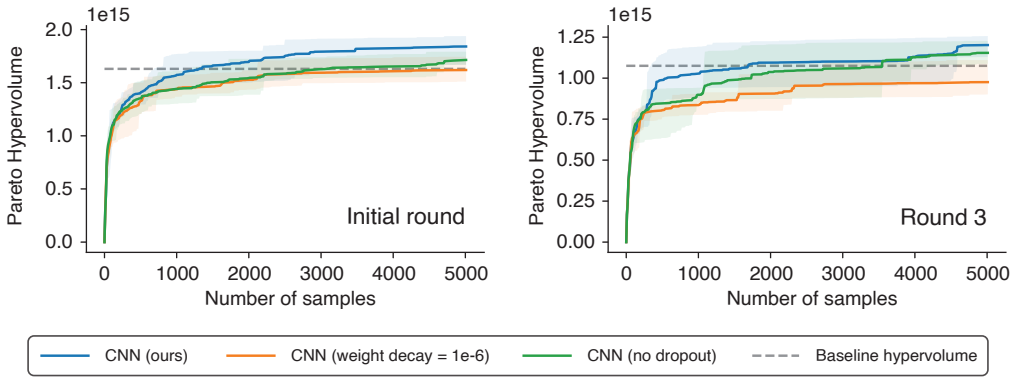


Figure 3-10: The impact of CNN regularization on Pareto hypervolume growth of the gamut, evaluated in the initial round and Round 3. The baseline hypervolume is the value covered by NSGA-II (baseline 1) after 20000 samples in Figure 3-9.

By tuning the CNNs' hyperparameters, we noticed that regularization plays an important role in accelerating hypervolume growth. In this experiment, we compared our CNNs with two less regularized variants, one using a negligible weight decay (1e-6) and the other coming with no dropout before the fully-connected layer. Our implementation outperformed less regularized versions for the examples in Figure 3-10. While the no-dropout variant secures narrow wins in some other test cases (Figure B-3), the small weight decay version keeps trailing behind. A reasonable explanation for more regularization is that it makes the networks less prone to overfitting, which can happen on small datasets.

Predictions from CNNs on training datasets were additionally checked against simulation results (Figure 3-11). The Young's modulus prediction network achieved very high accuracy with an average relative error of less than 2% and an r score of 0.999. The errors from toughness prediction network are larger, at 16.7% and 13.6%.

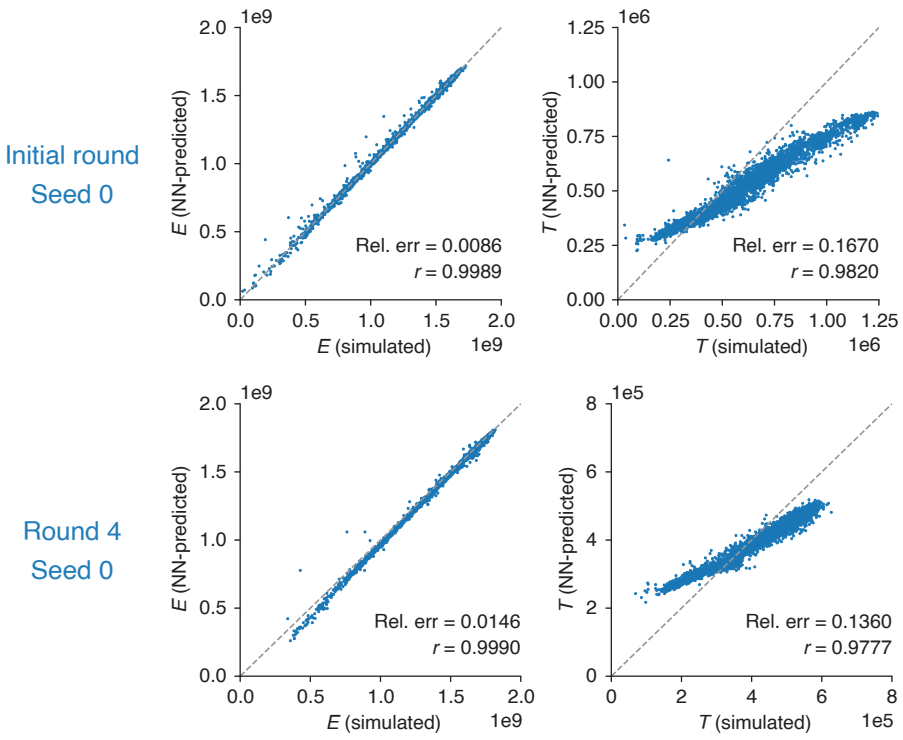


Figure 3-11: Two examples of CNN prediction accuracy in the last iteration of the learning-accelerated evoluationary algorithm. Simulation results are treated as the ground truth. r is correlation coefficient.

Nonetheless, accuracy is not the whole story since samples are picked for simulation in each iteration according to Pareto dominance instead of properties. Thus, predicting correct rankings will suffice. The CNN for toughness manages to get $r > 0.97$, which already leads to much faster hypervolume growth than other evolutionary algorithms. It is noteworthy that seeking the best network architecture or hyperparameter setting is not the focus of this work. We believe there is still room for even better performance from learning-based evolutionary algorithms.

3.3 Analysis on Pareto-Optimal Microstructures

From 13 microstructures near the Pareto front of physical measurements, we identified four families where samples bear similar shape features and have comparable properties (Figure 3-12). The first family mainly consists of high-toughness microstruc-

tures with moderate stiffness. A common feature in the seed patterns is that the rigid material pixel at the corner is surrounded by soft material that dampens crack propagation. The soft material also distributes along pattern edges in the form of perpendicular beams. The second family covers most samples that show the biggest improvement over homogeneous composites in both Young's modulus and toughness. Some traits are shared among four seed patterns, such as a chunk of soft material in the center and residual beam-like structures at boundaries. Unlike the previous two, microstructures in Family 3 and 4 tend to favor high stiffness over toughness. There are fewer beams or chunks of soft material but more sparse pixels scattered around the pattern. While the shape features of these families are slightly complex, they still help maintain a decent amount of toughness in samples. We will elaborate more about toughening mechanisms in Section 3.3.3.

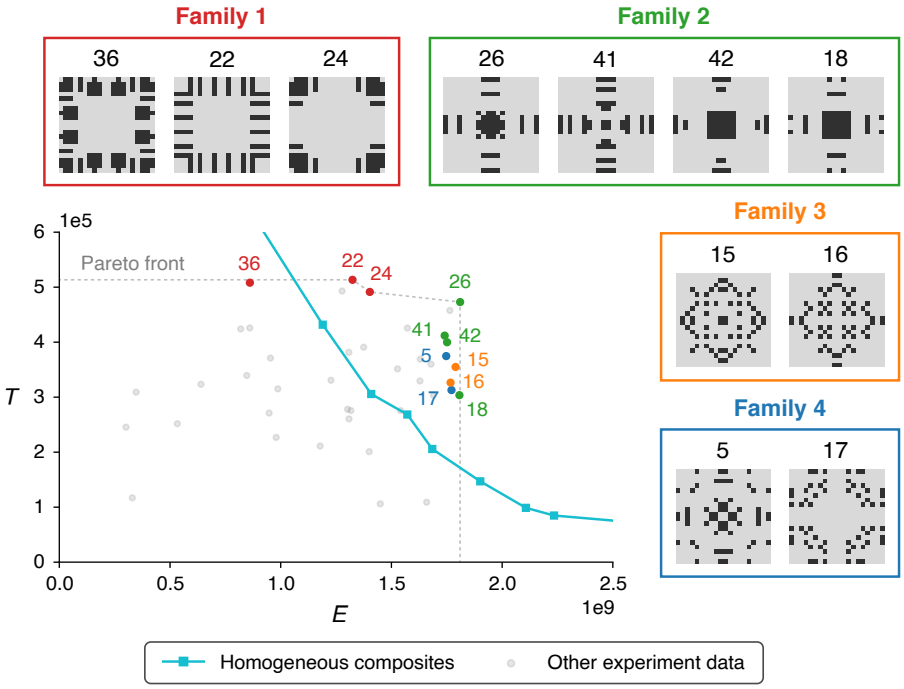


Figure 3-12: Four microstructure families near the Pareto front of physical measurements. Microstructure patterns are numbered as in Figure B-1.

Aside from geometric traits in patterns, most samples within a family also exhibit similar stress-strain responses (Figure 3-13) with an exception of Family 1. In this family, the difference in VW% between seed microstructures have a clear impact

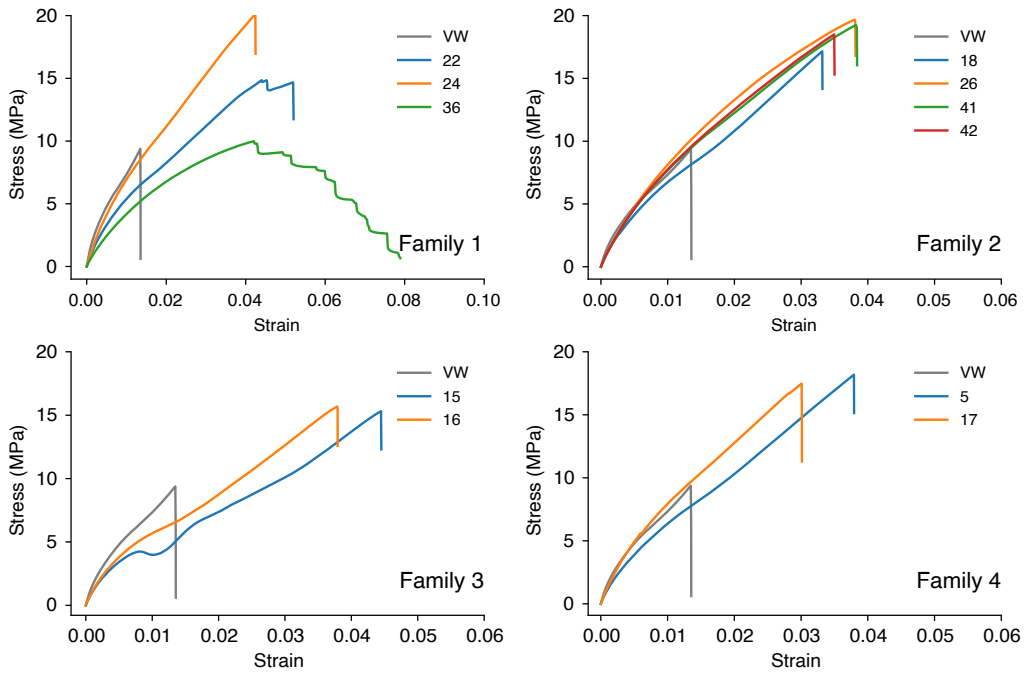


Figure 3-13: Stress-strain curves of seed microstructures in each family compared with VW. TB is omitted since its ultimate strength is two orders of magnitude smaller.

on their fracture behaviors. The stiffest one, Sample 22, has the largest ultimate strength and undergoes brittle fracture. Conversely, the softest one, Sample 36, is able to reach the same level of toughness by slowing down crack propagation as much as possible, which is seen from the long descending tail of the stress-strain curve. All microstructures shown in the figure have more than 2× the failure strain of VW, a fundamental cause for superior toughnesses.

For the remaining of this section, we demonstrate in-depth analysis results of microstructure families and their intrinsic toughening mechanisms.

3.3.1 Family Expansion and Validation

Before expanding microstructure families into gamuts, we first obtained an exclusive simulation model for each of them from system identification. This is for a better fitting quality within the family as illustrated in Figure 3-14 and Table 3.6. Since family-specific system identification only uses seed patterns and other Pareto-optimal microstructures for calibration, the accuracy is much better than the model in the

main pipeline. The contrast is particularly obvious in toughness predictions where the family-specific (FS) simulation model has one order of magnitude smaller relative errors in three families.

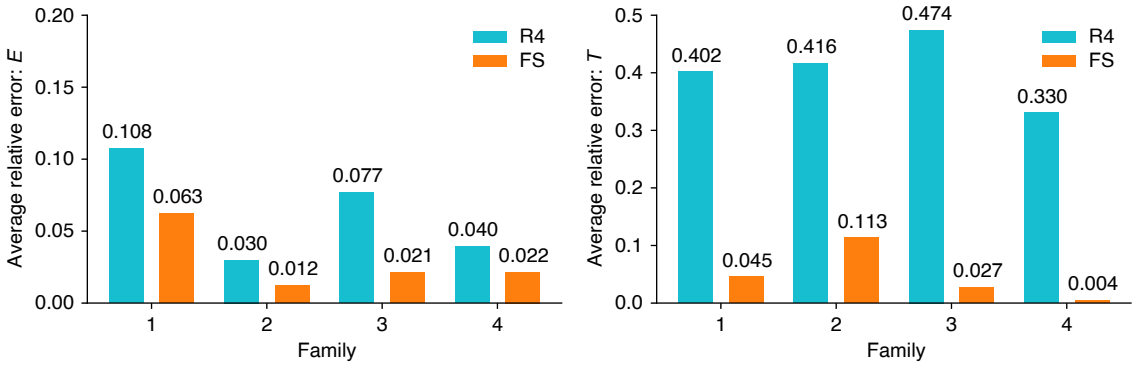


Figure 3-14: Simulation prediction errors from the calibrated model in Round 4 (R4) and family-specific models (FS) on seed microstructures.

Table 3.6: Simulation prediction errors from the calibrated model in Round 4 (R4) and family-specific models (FS) on seed microstructures. N denotes the number of samples. Better values are marked in bold.

Family	N	**Rel. error: E (%)**				**Rel. error: T (%)**			
		R4		**FS**		**R4**		**FS**	
		Avg.	Max.	Avg.	Max.	Avg.	Max.	Avg.	Max.
1	3	10.77	19.98	**6.25**	12.56	40.17	48.25	**4.55**	8.28
2	4	2.99	5.45	**1.22**	3.45	41.61	56.81	**11.32**	21.29
3	2	7.69	9.64	**2.13**	4.22	47.42	62.99	**2.69**	5.35
4	2	3.96	5.76	**2.16**	4.29	33.03	43.64	**0.45**	0.73

Using family-exclusive simulation models, we computed the gamut of each family that comprises samples within a 16-pixel radius from seed patterns (Figure 3-15). Around 7000 to 9000 samples were generated from dense sampling. Then, three validation microstructures were selected per family near the Pareto front. We measured their properties and compared the prediction quality again between the universal simulation model (R4) and the family-specific one (Table 3.7). However, the latter did not maintain a good accuracy and only managed to take one win from its opponent. Our explanation is that the universal model is better regularized against overfitting

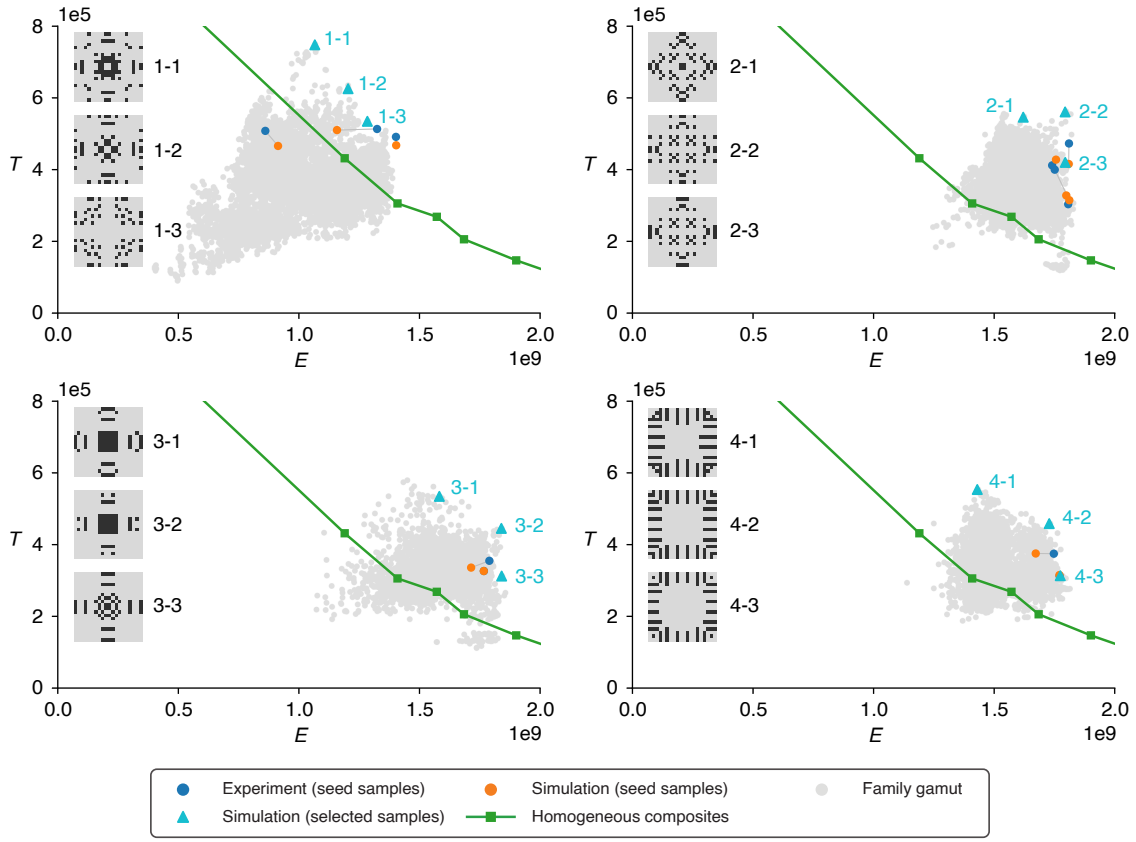


Figure 3-15: Gamut of each microstructure family after dense sampling along with selected samples near the Pareto front for validation.

Table 3.7: Simulation prediction errors from the calibrated model in Round 4 (R4) and family-specific models (FS) on validation microstructures. N denotes the number of samples. Better values are marked in bold.

Family	N	Rel. error: E (%)				Rel. error: T (%)			
		R4		FS		R4		FS	
		Avg.	Max.	Avg.	Max.	Avg.	Max.	Avg.	Max.
1	3	**2.30**	2.43	12.62	14.91	**14.24**	27.45	56.63	102.79
2	3	5.75	7.33	**3.82**	5.42	48.37	70.34	**37.37**	59.27
3	3	**2.02**	2.68	8.22	8.98	**31.13**	51.62	46.05	65.26
4	3	**3.25**	7.08	4.53	9.44	**34.65**	44.44	35.14	45.50

with a larger dataset. It seems to have 'learned' to make conservative toughness predictions that are often below physical measurements (Table A.5 and A.6), which is still suboptimal. Essentially, we view this as a trade-off between fitting quality

and generalization that originates from the simulator. A better simulation model will likely ease the conflict but at the cost of more expensive computation.

3.3.2 Dimensionality Reduction in Subfamilies

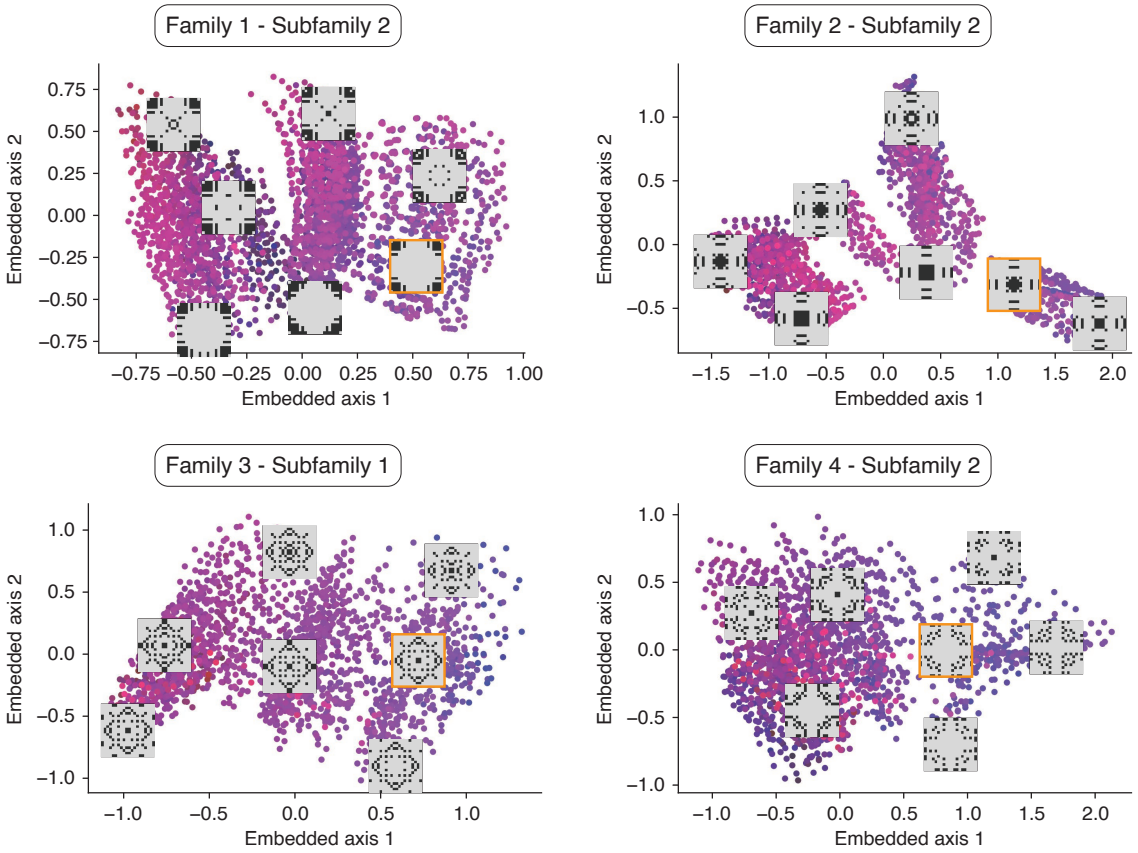


Figure 3-16: Isomap embedding spaces of four subfamilies annotated with pattern examples, eaching containing about 2000 to 3000 samples. Microstructure properties are encoded in colors (E: the blue channel; T: the red channel). Seed patterns are highlighted by orange boxes.

We ran Isomap to embed 400D microstructure patterns into 2D reduced spaces (using a neighborhood size of 10) inside each subfamily, shown in Figure 3-16 and B-4. Only a few example patterns are visualized due to space limitations. Here, we add information about microstructure properties by color-coding them in the scatter plots. A higher intensity of the red channel indicates better toughness. The blue channel represents Young's modulus in the same way. Starting from seed patterns, our dense

sampling algorithm created a multitude of variations within a 16-pixel radius. New shape features were introduced such as individual pixels, beams, arcs, small chunks of soft material, etc.

For an intuitive exploration of pattern variations, we developed an interactive tool that allows a user to navigate inside Isomap embedding spaces freely. The user can also play the simulation video of any microstructure simply at a mouse click. We found the tool very helpful in boosting the efficiency of human analysis.

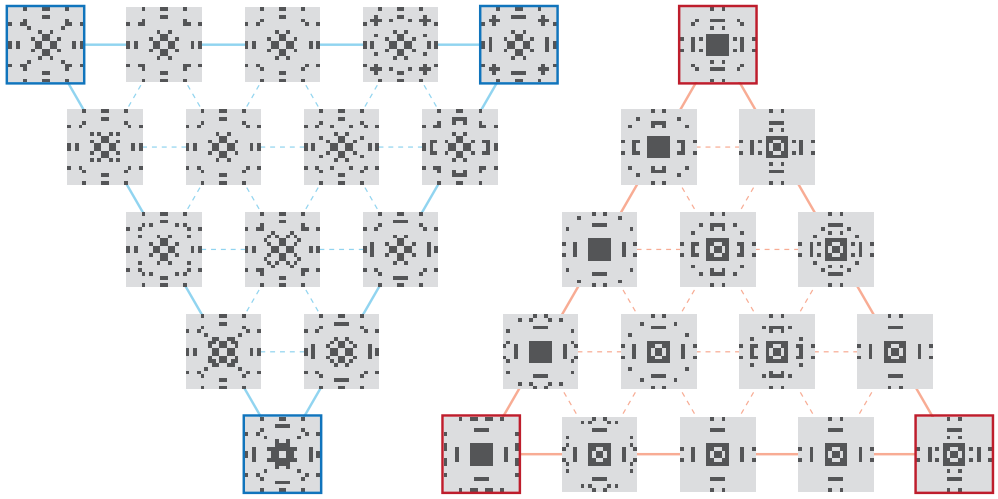


Figure 3-17: Examples of Wasserstein barycenter interpolation between 3 microstructure patterns.

As described in Section 2.4.2, Wasserstein barycenter interpolation (Figure 3-17) was applied to generating patterns not covered in dense sampling. In Figure 3-18 and B-5, we make a side-by-side comparison between Isomap embedding spaces before and after interpolation. Subfamilies became roughly 5% larger in size on average. This implies that the dense sampling algorithm has already discovered a majority of possible variants.

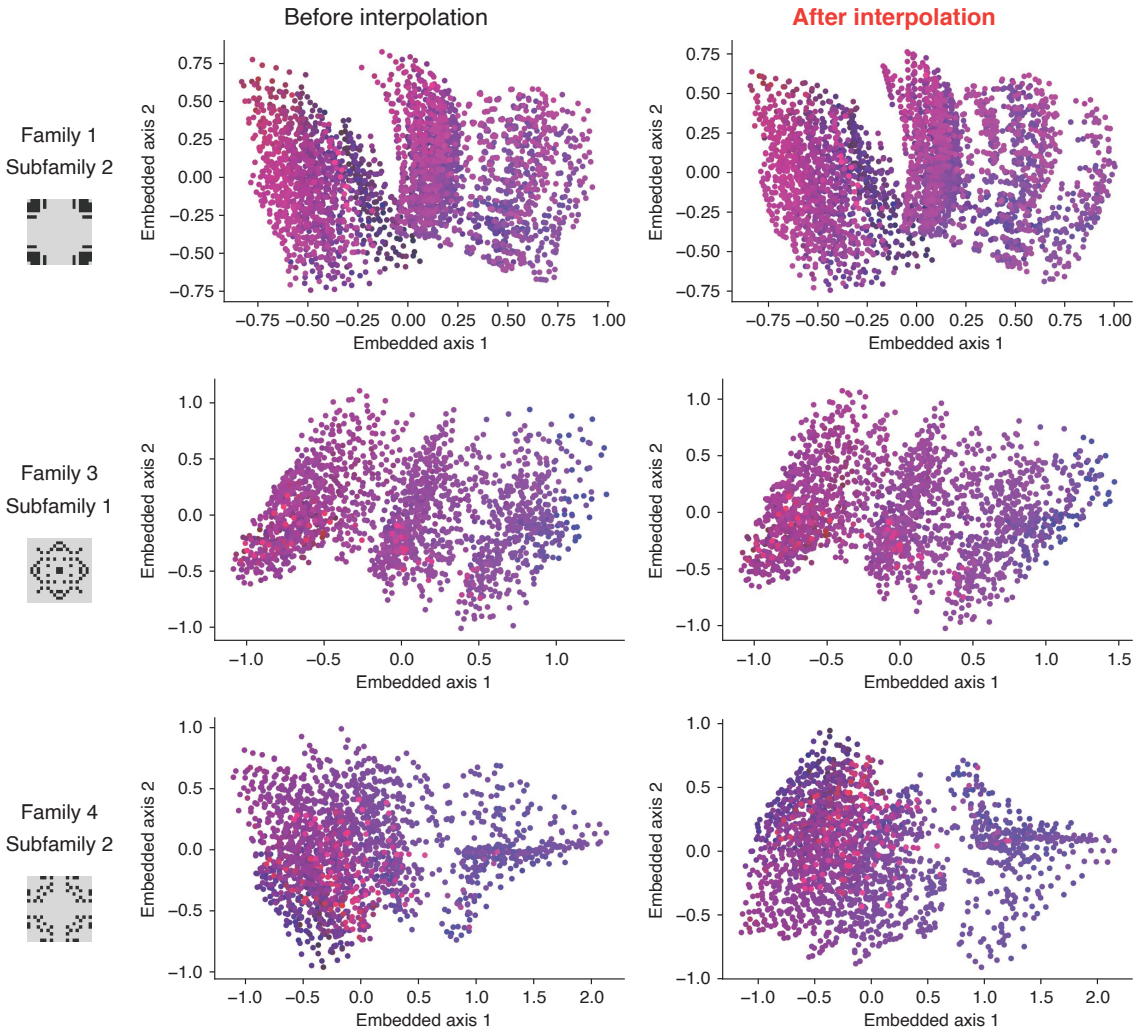


Figure 3-18: Isomap embedding spaces of three subfamilies before and after refinement using interpolation. Microstructure properties are encoded in colors.

3.3.3 Toughening Mechanisms

In this section, we report our in-depth analysis on the intrinsic toughening mechanisms of each microstructure family.

Family 1 maintains a relatively high fraction of soft pixels in the shape of bars or chunks as exemplified in Figure 3-19A. This family is located on the left side of the performance gamut with a relatively small Young's modulus. To maintain a proper strength, continuous areas of rigid material form horizontal and vertical grids (stiff grids in Figure 3-19A). A distinctive feature appearing in this family is parallel bars made of the soft base material. Their benefits are twofold. When

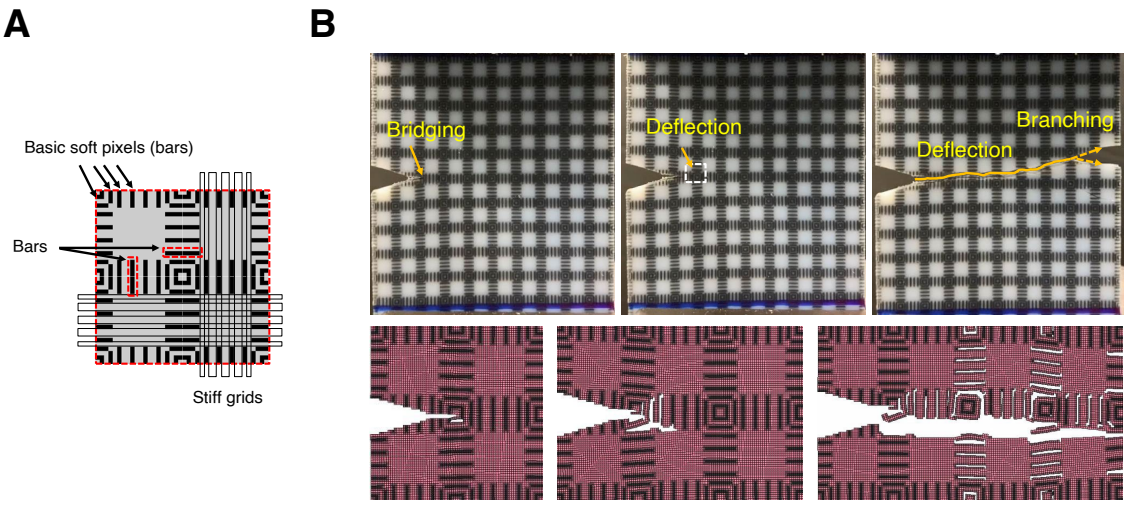


Figure 3-19: Major shape features (A) and toughening mechanisms (B) of microstructures in Family 1.

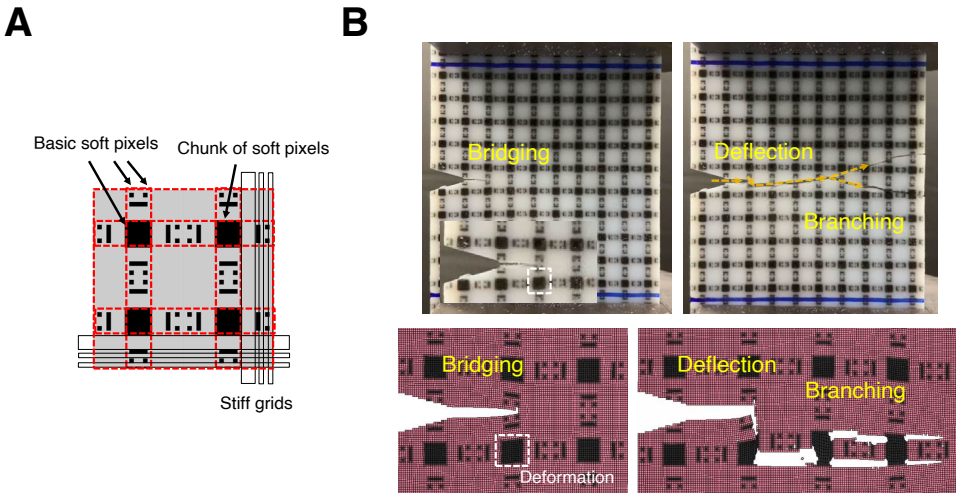


Figure 3-20: Major shape features (A) and toughening mechanisms (B) of microstructures in Family 2.

perpendicular to the pulling direction, they introduce additional resistance to stress through elastic deformation. Alternatively, when parallel to the pulling direction (or directly facing crack propgation), they effectively absorb energy and slow down the propagation. Bridging and deflection effects are the most prominent toughening mechanisms (Figure 3-19B). A zig-zag crack path is observed during fracture which results in high toughness. The propagation energy accumulates under the trapping effect of the soft material and dissipates via branched cracks [39].

Different from Family 1, Family 2 concentrates soft pixels in a cross-like band in the center of the unit (Figure 3-20A). Samples withstand a larger strain via the elastic deformation of soft material chunks, as indicated by a dashed white square (Figure 3-20B). It is noteworthy that such deformation is accurately reflected in simulation. Stiff grids are also present as a means of improving strength. Bridging, deflection, and branching (Figure 3-20B) are all essential anti-cracking mechanisms in the family.

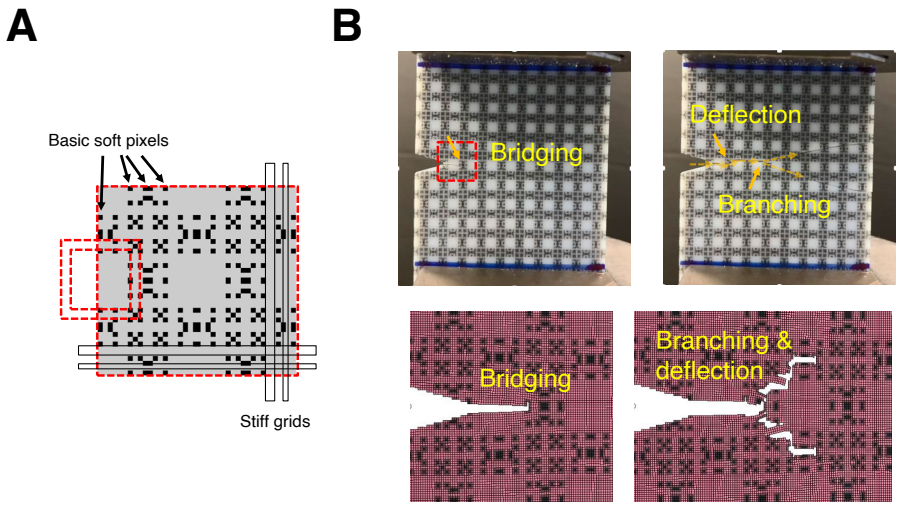


Figure 3-21: Major shape features (A) and toughening mechanisms (B) of microstructures in Family 3.

While the above two families have more concentrated soft pixels, they are mostly dispersed in Family 3. Discrete soft pixels form a quarter of a diamond-like structure at corners and another diamond structure in the center (Figure 3-21A). Meanwhile, some soft pixels are assigned at edges and located ahead of the notch tip. These pixels are typically on the path of crack propagation and act as an energy dissipation zone that delays crack propagation. Since the interface is stronger than the soft material, the soft pixels work as defects to guide crack propagation. Such an deflection effect can increase the crack length dramatically and dissipate much more energy. Additional soft pixels introduced during dense sampling not only enrich the diversity of the family but also increase the overall stress resistance.

As for Family 4, discrete soft pixels at corners form a quarter of circular outline (basic soft pixels in Figure 3-22A). Similar to Family 3, some soft pixels are placed

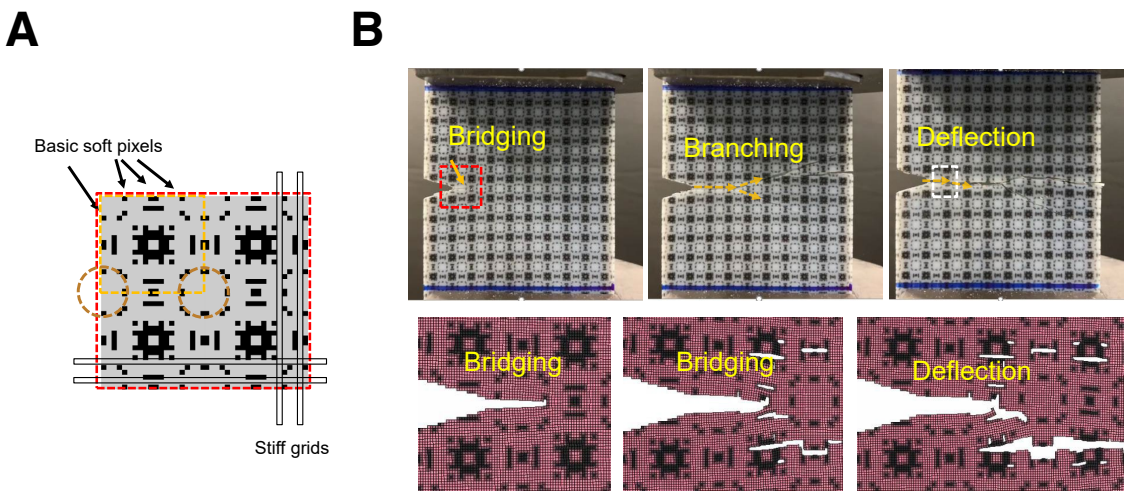


Figure 3-22: Major shape features (A) and toughening mechanisms (B) of microstructures in Family 4.

around edges, which retards crack propagation by dampening the cracking velocity and dissipating energy through elastic deformation. Basic features like stiff grids are preserved to maintain strength. The same anti-cracking mechanisms, such as bridging, deflection, and branching, are observed in this family (Figure 3-22B). Since Family 3 and 4 bear some resemblance in structure, they show comparable mechanical performance. These two families are dominated by the rigid base material where VW% lies between 80% and 85%. Thus, they tend to have higher strengths but breaking at relatively smaller strains.

Chapter 4

Summary

We propose a computational pipeline for automatic discovery of microstructures with optimal Young's modulus and toughness trade-offs for additively manufactured heterogeneous composites. It features a data-driven competitive game approach to bridging the gap between simulation and real experiment, incorporating a fast customized FEM simulator, a system identification step for simulation calibration, and a sample-efficient machine learning-assisted exploration algorithm for finding Pareto-optimal samples in a combinatorial design space. The pipeline is adaptive to sim-to-real problems in many other research fields where simulation is feasible but inaccurate.

Another key contribution of our work is the analysis of Pareto-optimal microstructure families, from which we extract and interpret several representative mechanisms that contribute to extremal toughness. Our pipeline opens the door for reversing the traditional scientific discovery process through analysis-by-synthesis: first computing the optimal samples in a large design space using calibrated simulation, then uncovering governing patterns and rules in these samples. It will potentially give rise to new designs and theories in a wide range of applications including material science, chemistry, pharmaceutics, engineering mechanics, and robotics [14].

Finally, this work lays the foundation of many future improvements and extensions in computational discovery, which are not limited to the following.

- The primary focus of this work is microstructured composites fabricated using

inkjet 3D printing. That said, the computational pipeline can be adapted to hollow structures and lattice structures manufactured by laser cut. They add an extra degree of freedom to the problem by introducing void material as an option, which might lead to more sophisticated but appealing designs for engineering materials. An equally straightforward extension of this work is studying 2.5D or 3D microstructures. For example, material distribution in the z direction can be non-uniform in a predetermined or configurable manner.

- The physics-based FEM simulator faces a multilateral trade-off between speed, expressibility, and generalization. It is considered as a bottleneck of the pipeline. Though extremely challenging, advancement in any direction will likely increase the potent of computational microstructure discovery by a fair margin. We look forward to new scientific research breakthroughs in the modeling of heterogeneous composites and relevant simulation approaches. Meanwhile, given the current computation power, it is a viable choice to narrow down the design space and dedicate to, for example, a small range of VW%. This grants more opportunities for discovering high performance microstructures as less exploration will be needed.

- Limited by printer resolution, the toughness of a heterogeneous composite is spatially varying contingent on the position of the notch tip and sensitive to manufacturing errors. Materials in printed specimens also show multiscale characteristics that complicate modeling dramatically. We are curious about an ideal combination of manufacturing technique and feature size for microstructures that results in consistent toughness measurements within a specimen.

- The machine learning community continues to thrive today as new theories, datasets, network architectures, and training methods are proposed for an ever-growing number of applications like daily routines. Provided proper ways to bridge the sim-to-real gap, learning-assisted microstructure discovery will remain a promising direction that calls for more innovations in methodology.

Appendix A

Tables

Table A.1: Value ranges of simulation parameters used in system identification (fitting to the experiment results of base materials when the competitive game starts).

Young's modulus			**Toughness**		
Name	Min.	Max.	Name	Min.	Max.
E_{TB}	8.08e5	1.21e6	E_{TB}	1e5	2e6
E_{VW}	2.13e9	3.19e9	E_{VW}	1e7	1e9
E_{IF}	4.51e8	6.77e8	E_{IF}	1e7	1e9
ν_{TB}	0.4	0.5	ν_{TB}	0.4	0.5
ν_{VW}	0.2	0.4	ν_{VW}	0.2	0.4
ν_{IF}	0.3	0.4	ν_{IF}	0.3	0.4
-	-	-	W_{TB}	5e4	1e6
-	-	-	W_{VW}	5e4	1e7
-	-	-	W_{IF}	1e6	1e8
γ	2e4	2e6	γ	2e4	4e6

Table A.2: Value ranges of simulation parameters used in system identification (fitting to the experiment results of microstructures in the competitive game).

Young's modulus			**Toughness**		
Name	Min.	Max.	Name	Min.	Max.
E_{TB}	8e5	1.2e6	E_{TB}	5e5	5e8
E_{VW}	2.08e9	3.12e9	E_{VW}	5e7	1e9
E_{IF}	4.48e8	6.72e8	E_{IF}	1e7	5e8
ν_{TB}	0.4	0.5	ν_{TB}	0.4	0.5
ν_{VW}	0.2	0.4	ν_{VW}	0.2	0.4
ν_{IF}	0.3	0.4	ν_{IF}	0.3	0.4
-	-	-	W_{TB}	5e4	1e6
-	-	-	W_{VW}	5e5	1e7
-	-	-	W_{IF}	1e6	1e8
γ	2e4	2e6	γ	2e4	4e6

Table A.3: Physical measurements, simulation predictions, and relative errors of all microstructures discovered by the pipeline (E/GPa). IR = initial round, Ri = Round i. Text in blue shows predictions for the next round.

R	No.	VW %	Exp.	Sim. (IR)	Rel. err.	Sim. (R1)	Rel. err.	Sim. (R2)	Rel. err.	Sim. (R3)	Rel. err.	Sim. (R4)	Rel. err.
1	1	61	0.640	0.402	0.372	0.653	0.020	0.633	0.011	0.632	0.013	0.637	0.004
	2	76	1.400	0.980	0.300	1.281	0.085	1.216	0.131	1.185	0.153	1.185	0.154
	3	67	0.948	0.663	0.301	0.931	0.018	0.889	0.062	0.878	0.074	0.881	0.070
	4	83	1.541	1.333	0.135	1.646	0.068	1.559	0.012	1.510	0.020	1.505	0.023
	5	84	1.747	1.509	0.136	1.826	0.045	1.718	0.017	1.654	0.053	1.647	0.058
	6	51	0.347	0.118	0.659	0.346	0.002	0.340	0.020	0.349	0.005	0.357	0.029
	7	51	0.301	0.133	0.560	0.370	0.229	0.363	0.204	0.370	0.228	0.378	0.254
	8	54	0.330	0.023	0.932	0.347	0.053	0.342	0.037	0.356	0.080	0.368	0.115
	9	60	0.533	0.261	0.511	0.526	0.013	0.511	0.041	0.515	0.034	0.523	0.020
	10	75	1.178	0.986	0.163	1.231	0.045	1.180	0.002	1.155	0.020	1.153	0.021
	11	66	0.954	0.603	0.368	0.801	0.160	0.784	0.178	0.778	0.185	0.780	0.182
	12	74	1.308	0.980	0.251	1.232	0.059	1.171	0.105	1.142	0.127	1.140	0.128
	13	76	1.375	1.075	0.218	1.330	0.033	1.264	0.080	1.231	0.104	1.228	0.107
	14	82	1.659	1.255	0.243	1.539	0.072	1.470	0.114	1.424	0.141	1.420	0.144
	15	84	1.790	1.479	0.174	1.775	0.008	1.681	0.061	1.625	0.092	1.617	0.096
	16	85	1.767	1.523	0.138	1.827	0.034	1.729	0.021	1.674	0.053	1.666	0.057
	17	85	1.771	1.604	0.094	1.915	0.081	1.807	0.020	1.742	0.016	1.733	0.022
	18	85	1.807	1.729	0.043	2.080	0.151	1.934	0.070	1.844	0.020	1.833	0.014
2	19	64	0.819	0.605	0.261	0.854	0.042	0.810	0.012	0.793	0.032	0.796	0.028
	20	58	0.860	0.736	0.144	0.919	0.068	0.876	0.018	0.854	0.007	0.853	0.008
	21	67	0.980	0.874	0.108	1.115	0.139	1.054	0.076	1.021	0.042	1.019	0.041
	22	77	1.324	1.162	0.123	1.474	0.113	1.378	0.041	1.328	0.003	1.325	0.001
	23	79	1.572	1.356	0.138	1.678	0.067	1.566	0.004	1.502	0.045	1.496	0.049
	24	79	1.402	1.479	0.054	1.792	0.278	1.661	0.185	1.583	0.129	1.574	0.123
	25	85	1.764	1.613	0.086	1.963	0.113	1.838	0.042	1.760	0.002	1.751	0.007
	26	85	1.810	1.738	0.040	2.089	0.154	1.945	0.074	1.856	0.025	1.845	0.019
3	27	68	0.987	0.774	0.215			0.961	0.026	0.944	0.044	0.944	0.043
	28	74	1.227	1.056	0.140			1.273	0.038	1.232	0.004	1.230	0.002
	29	75	1.277	1.166	0.087			1.361	0.066	1.313	0.028	1.309	0.025
	30	79	1.308	1.281	0.021			1.480	0.132	1.432	0.095	1.427	0.091
	31	80	1.450	1.355	0.065			1.577	0.088	1.516	0.046	1.510	0.042
	32	85	1.630	1.541	0.055			1.782	0.093	1.709	0.048	1.702	0.044
	33	85	1.690	1.723	0.019			1.923	0.138	1.833	0.085	1.822	0.078
	34	85	1.679	1.724	0.027			1.930	0.150	1.841	0.097	1.831	0.091
4	35	68	0.847	0.724	0.145					0.890	0.051	0.891	0.052
	36	66	0.861	0.904	0.050					1.035	0.203	1.033	0.200
	37	74	1.317	1.033	0.216					1.206	0.084	1.204	0.086
	38	76	1.302	1.207	0.073					1.362	0.047	1.358	0.043
	39	79	1.528	1.305	0.146					1.459	0.045	1.454	0.048
	40	82	1.629	1.443	0.114					1.594	0.021	1.587	0.025
	41	85	1.741	1.673	0.039					1.805	0.037	1.796	0.032
	42	85	1.751	1.749	0.001					1.858	0.061	1.847	0.054

Table A.4: Physical measurements, simulation predictions, and relative errors of all microstructures discovered by the pipeline (T/MPa). IR = initial round, Ri = Round i. Text in blue shows predictions for the next round.

R	No.	VW %	Exp.	Sim. (IR)	Rel. err.	Sim. (R1)	Rel. err.	Sim. (R2)	Rel. err.	Sim. (R3)	Rel. err.	Sim. (R4)	Rel. err.
1	1	61	0.323	0.528	0.633	0.361	0.116	0.254	0.214	0.239	0.261	0.309	0.045
	2	76	0.201	0.329	0.637	0.227	0.131	0.236	0.173	0.143	0.290	0.221	0.102
	3	67	0.271	0.398	0.470	0.395	0.458	0.327	0.207	0.286	0.056	0.273	0.009
	4	83	0.276	0.307	0.114	0.274	0.007	0.246	0.109	0.190	0.310	0.203	0.264
	5	84	0.375	0.277	0.261	0.310	0.171	0.282	0.247	0.214	0.429	0.211	0.436
	6	51	0.309	0.487	0.574	0.255	0.177	0.275	0.110	0.190	0.387	0.275	0.110
	7	51	0.245	0.231	0.057	0.270	0.099	0.132	0.461	0.141	0.425	0.221	0.098
	8	54	0.117	0.022	0.809	0.102	0.126	0.078	0.329	0.102	0.124	0.143	0.226
	9	60	0.252	0.222	0.119	0.344	0.365	0.194	0.227	0.204	0.188	0.244	0.029
	10	75	0.211	0.482	1.283	0.343	0.622	0.263	0.248	0.227	0.073	0.343	0.624
	11	66	0.371	1.304	2.515	0.410	0.105	0.359	0.031	0.310	0.165	0.386	0.039
	12	74	0.261	1.112	3.268	0.407	0.564	0.413	0.586	0.218	0.162	0.394	0.514
	13	76	0.391	1.091	1.791	0.398	0.018	0.294	0.247	0.303	0.225	0.420	0.074
	14	82	0.109	0.945	7.674	0.159	0.457	0.120	0.104	0.110	0.012	0.099	0.095
	15	84	0.355	0.865	1.436	0.236	0.335	0.278	0.218	0.179	0.495	0.131	0.630
	16	85	0.326	0.786	1.407	0.342	0.048	0.315	0.035	0.276	0.154	0.222	0.319
	17	85	0.313	0.603	0.927	0.291	0.069	0.222	0.292	0.223	0.289	0.243	0.224
	18	85	0.304	0.424	0.395	0.378	0.244	0.324	0.067	0.305	0.005	0.271	0.108
2	19	64	0.424	0.295	0.304	0.700	0.650	0.295	0.304	0.540	0.275	0.448	0.056
	20	58	0.426	0.553	0.300	0.892	1.095	0.355	0.166	0.345	0.189	0.458	0.075
	21	67	0.227	0.669	1.950	0.970	3.277	0.455	1.005	0.418	0.843	0.428	0.889
	22	77	0.513	0.554	0.079	0.982	0.913	0.449	0.125	0.526	0.024	0.346	0.326
	23	79	0.426	0.452	0.062	0.865	1.033	0.452	0.063	0.494	0.161	0.343	0.193
	24	79	0.491	0.315	0.359	0.680	0.383	0.371	0.245	0.447	0.089	0.254	0.482
	25	85	0.458	0.280	0.389	0.584	0.276	0.431	0.058	0.374	0.182	0.266	0.418
	26	85	0.473	0.314	0.336	0.391	0.172	0.309	0.346	0.315	0.333	0.206	0.564
3	27	68	0.315	0.612	0.943			0.598	0.897	0.338	0.074	0.467	0.483
	28	74	0.331	0.395	0.195			0.611	0.847	0.340	0.029	0.304	0.081
	29	75	0.493	0.423	0.141			0.589	0.196	0.530	0.075	0.390	0.209
	30	79	0.381	0.364	0.045			0.541	0.417	0.269	0.295	0.394	0.034
	31	80	0.106	0.304	1.865			0.505	3.761	0.175	0.648	0.172	0.620
	32	85	0.369	0.349	0.053			0.466	0.263	0.330	0.105	0.289	0.216
	33	85	0.303	0.266	0.122			0.369	0.218	0.151	0.501	0.260	0.143
	34	85	0.360	0.310	0.139			0.327	0.092	0.172	0.521	0.218	0.394
4	35	68	0.339	0.504	0.485					0.641	0.888	0.291	0.142
	36	66	0.508	0.363	0.286					0.684	0.347	0.306	0.397
	37	74	0.276	0.389	0.412					0.716	1.598	0.360	0.307
	38	76	0.278	0.386	0.388					0.667	1.401	0.308	0.108
	39	79	0.351	0.434	0.235					0.633	0.802	0.312	0.111
	40	82	0.329	0.479	0.453					0.544	0.650	0.265	0.196
	41	85	0.412	0.294	0.287					0.424	0.030	0.237	0.424
	42	85	0.400	0.239	0.403					0.261	0.346	0.173	0.568

Table A.5: Physical measurements, simulation predictions, and relative errors of microstructures in the family analysis stage (E/GPa). R4 = Round 4, FS = family-specific model.

Seed microstructures

Family	No	VW%	Exp.	Sim. (R4)	Rel. err.	Sim. (FS)	Rel. err.
1	22	77	1.324	1.325	0.001	1.158	0.126
	24	79	1.402	1.574	0.123	1.403	0.001
	36	66	0.861	1.033	0.200	0.913	0.061
2	18	85	1.807	1.833	0.014	1.799	0.004
	26	85	1.810	1.845	0.019	1.809	0.001
	41	85	1.741	1.796	0.032	1.756	0.009
	42	85	1.751	1.847	0.054	1.812	0.034
3	15	84	1.790	1.617	0.096	1.714	0.042
	16	85	1.767	1.666	0.057	1.767	0.000
4	5	84	1.747	1.647	0.058	1.672	0.043
	17	85	1.771	1.733	0.022	1.771	0.000

Validation microstructures

Family	No	VW%	Exp.	Sim. (R4)	Rel. err.	Sim. (FS)	Rel. err.
1	F1-1	73	1.253	1.222	0.024	1.066	0.149
	F1-2	75	1.329	1.359	0.022	1.204	0.094
	F1-3	79	1.485	1.451	0.022	1.284	0.135
2	F2-1	81	1.541	1.654	0.073	1.621	0.052
	F2-2	85	1.779	1.825	0.026	1.794	0.009
	F2-3	85	1.703	1.827	0.073	1.795	0.054
3	F3-1	83	1.470	1.510	0.027	1.582	0.076
	F3-2	85	1.703	1.725	0.013	1.840	0.081
	F3-3	85	1.689	1.724	0.021	1.841	0.090
4	F4-1	80	1.447	1.426	0.015	1.430	0.012
	F4-2	84	1.679	1.699	0.012	1.728	0.030
	F4-3	85	1.621	1.736	0.071	1.774	0.094

Table A.6: Physical measurements, simulation predictions, and relative errors of microstructures in the family analysis stage (T/MPa). R4 = Round 4, FS = family-specific model.

Seed microstructures

Family	No	VW%	Exp.	Sim. (R4)	Rel. err.	Sim. (FS)	Rel. err.
1	22	77	0.513	0.346	0.326	0.510	0.006
	24	79	0.491	0.254	0.482	0.468	0.048
	36	66	0.508	0.306	0.397	0.466	0.083
2	18	85	0.304	0.271	0.108	0.328	0.080
	26	85	0.473	0.206	0.564	0.416	0.121
	41	85	0.412	0.237	0.424	0.428	0.039
	42	85	0.400	0.173	0.568	0.315	0.213
3	15	84	0.355	0.131	0.630	0.336	0.054
	16	85	0.326	0.222	0.319	0.326	0.000
4	5	84	0.375	0.211	0.436	0.375	0.002
	17	85	0.313	0.243	0.224	0.315	0.007

Validation microstructures

Family	No	VW%	Exp.	Sim. (R4)	Rel. err.	Sim. (FS)	Rel. err.
1	F1-1	73	0.369	0.375	0.017	0.748	1.028
	F1-2	75	0.447	0.386	0.136	0.626	0.399
	F1-3	79	0.420	0.305	0.275	0.535	0.272
2	F2-1	81	0.343	0.273	0.204	0.546	0.593
	F2-2	85	0.388	0.115	0.703	0.561	0.446
	F2-3	85	0.458	0.209	0.544	0.420	0.083
3	F3-1	83	0.330	0.281	0.147	0.535	0.623
	F3-2	85	0.269	0.197	0.271	0.445	0.653
	F3-3	85	0.350	0.169	0.516	0.312	0.106
4	F4-1	80	0.381	0.258	0.321	0.554	0.455
	F4-2	84	0.347	0.252	0.274	0.459	0.322
	F4-3	85	0.433	0.241	0.444	0.313	0.277

Table A.7: Increases in the number of samples in subfamilies from interpolation.

Family	**Subfamily**	**#Samples** Before	After	**Inc.** (%)
1	1	3251	3328	2.37
	2	2959	3164	6.97
	3	3313	3371	1.75
2	1	1862	1939	4.14
	2	1861	1904	2.31
	3	1917	2054	7.15
	4	1859	1903	2.37
3	1	1860	1988	6.88
	2	1862	1964	5.48
4	1	1997	2112	5.76
	2	1915	2044	6.74
Total		24656	25771	4.52

Appendix B

Figures

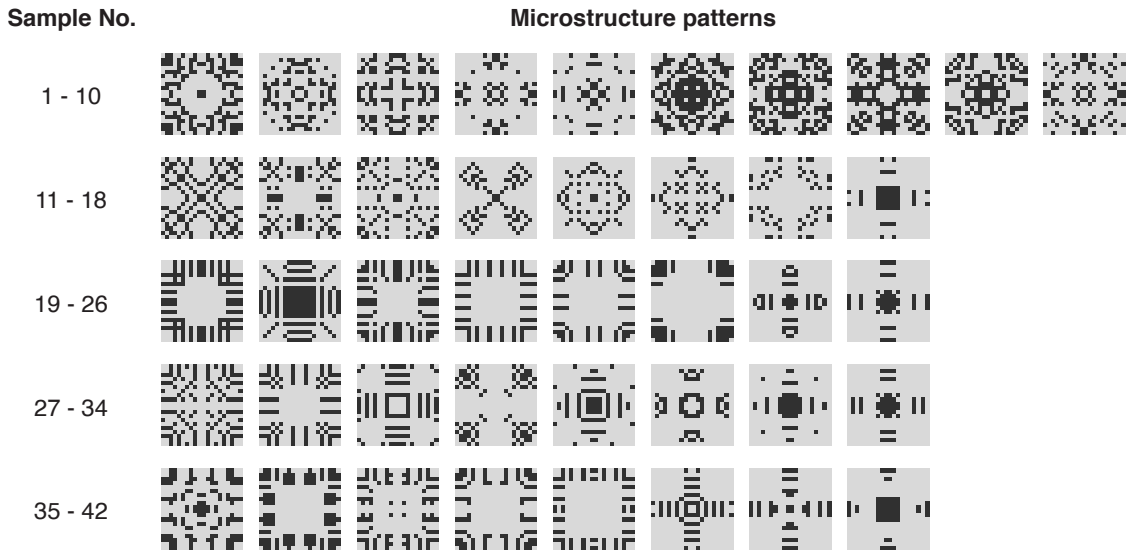


Figure B-1: Microstructures discovered by the computational pipeline. The first row comprises 10 random patterns in the calibration dataset. The following rows are patterns proposed in 4 competitive game rounds.

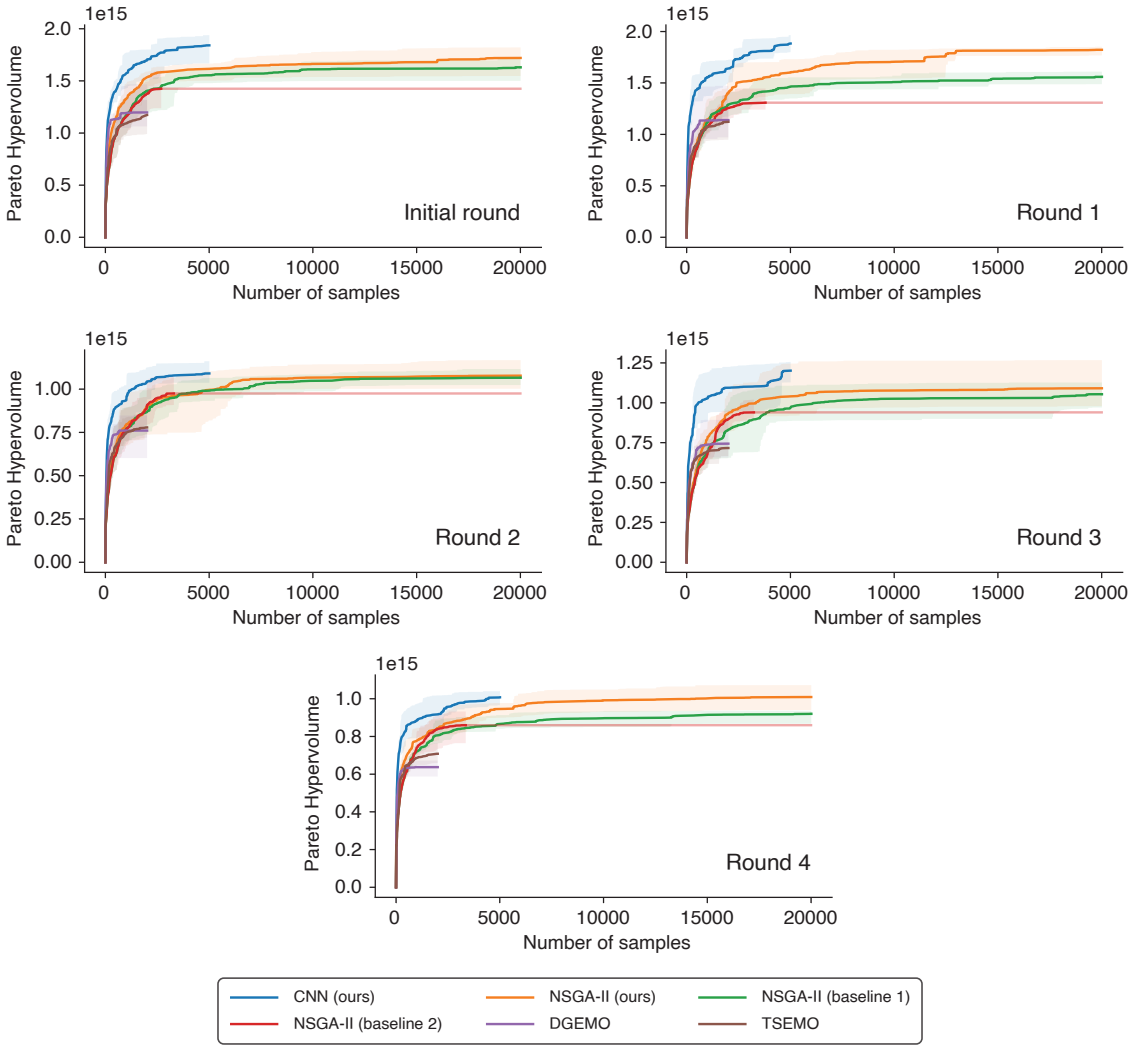


Figure B-2: Pareto hypervolume growth of the gamut under various evolutionary MOO algorithms (NSGA-II, MOBO, and ours) in each competitive game round.

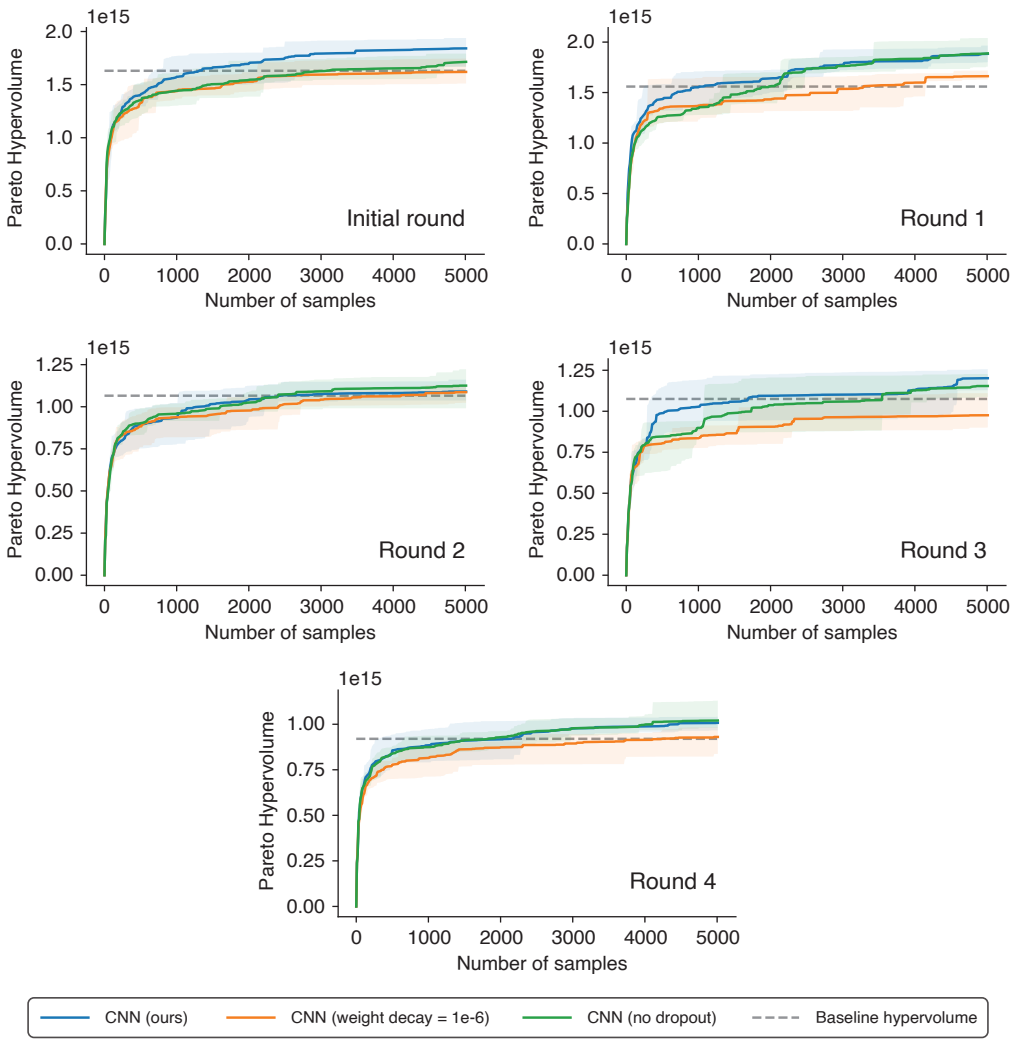


Figure B-3: The impact of CNN regularization on Pareto hypervolume growth of the gamut, evaluated each game round. The baseline hypervolume is the value covered by NSGA-II (baseline 1) after 20000 samples in Figure B-2.

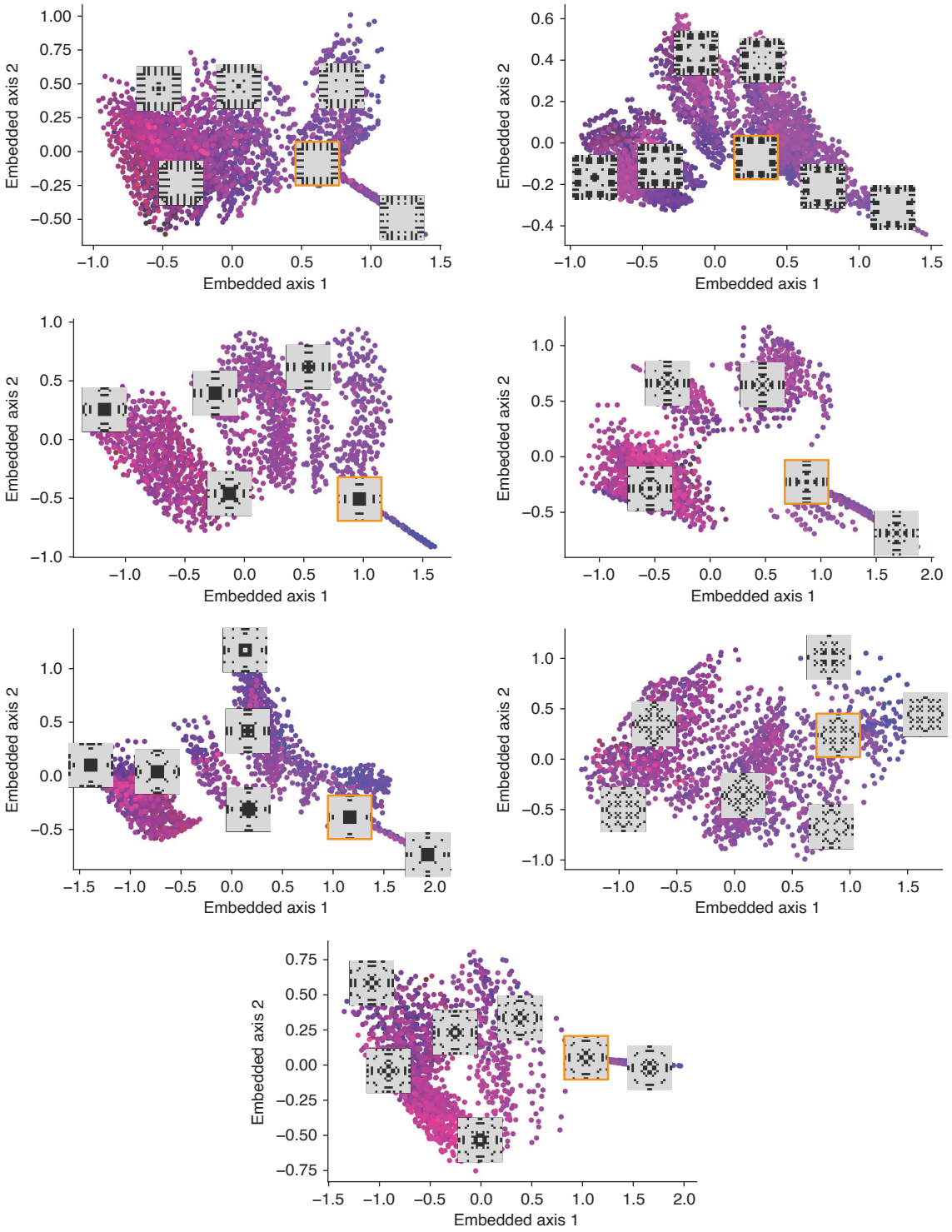
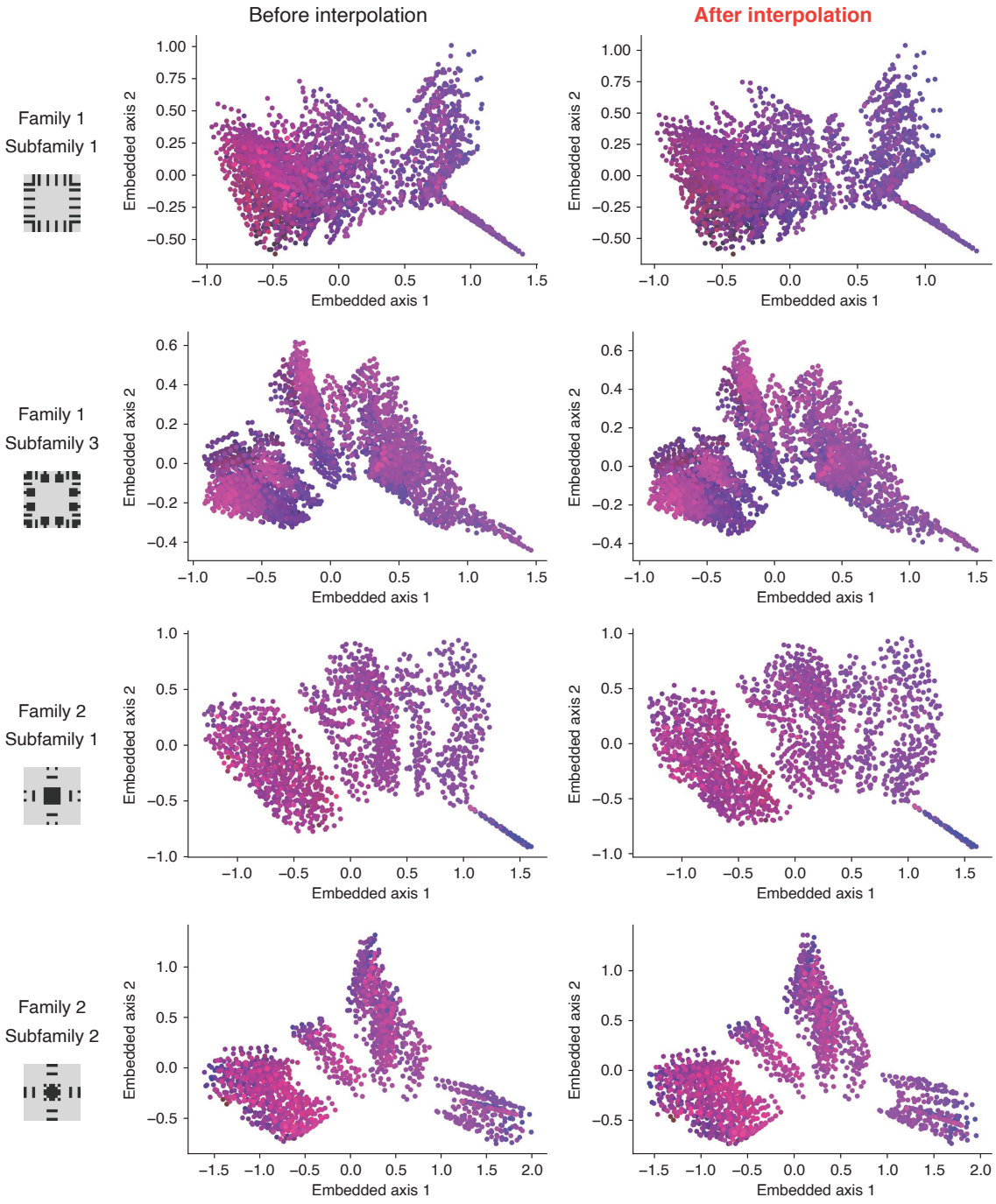


Figure B-4: Isomap embedding spaces of subfamilies not displayed in Figure 3-16, each containing around 2000 to 3000 samples. Microstructure properties are encoded in colors (E: the blue channel; T: the red channel). Seed patterns are marked in orange boxes.


Before interpolation
After interpolation
Family 1
Subfamily 1
Family 1
Subfamily 3
Family 2
Subfamily 1
Family 2
Subfamily 2
Embedded axis 1
Embedded axis 2

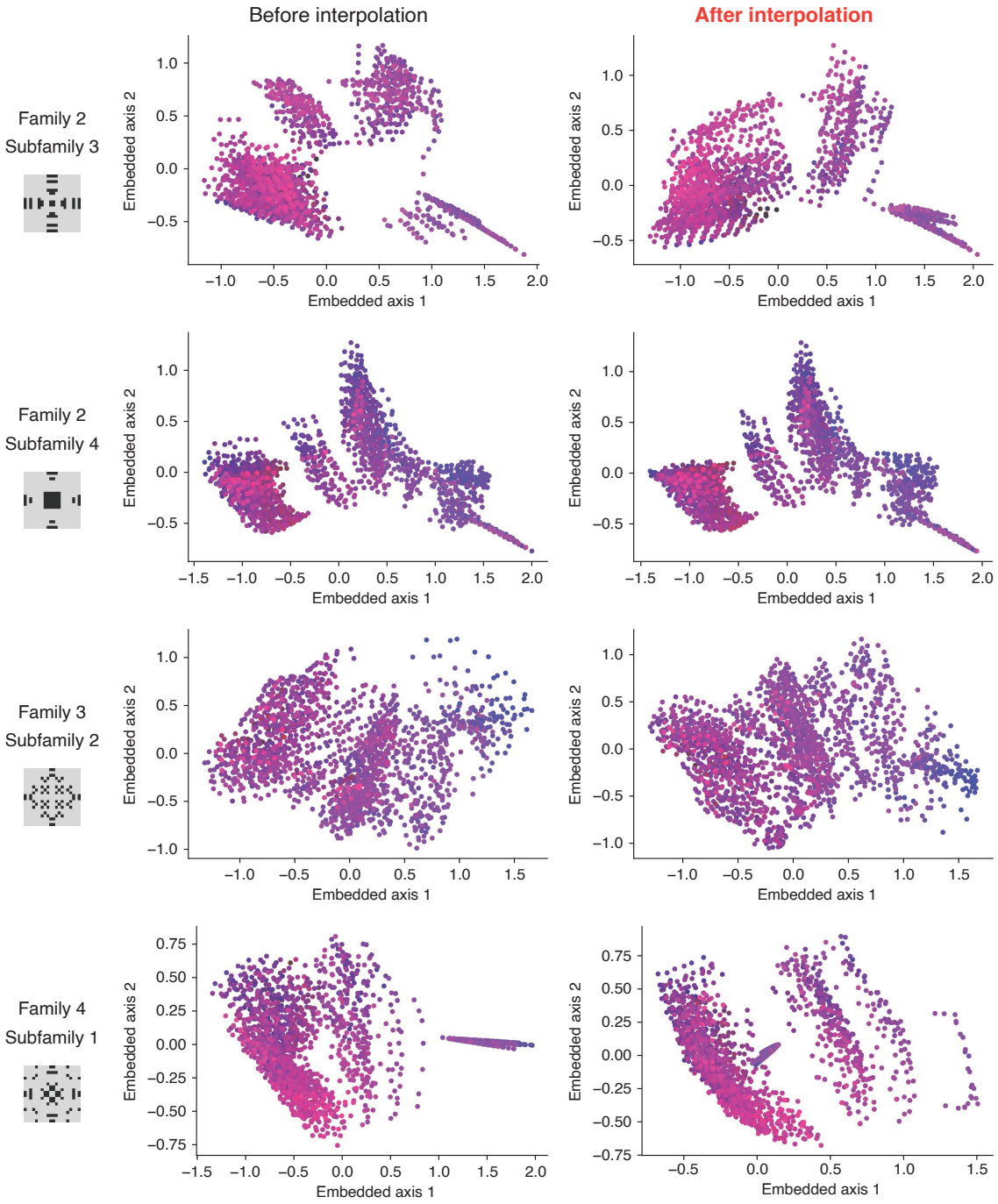


Figure B-5: Isomap embedding spaces of more subfamilies before and after refinement using interpolation. Microstructure properties are encoded in colors.

Bibliography

[1] Niels Aage, Erik Andreassen, Boyan S Lazarov, and Ole Sigmund. Giga-voxel computational morphogenesis for structural design. *Nature*, 550(7674):84–86, 2017.

[2] Martial Agueh and Guillaume Carlier. Barycenters in the wasserstein space. *SIAM Journal on Mathematical Analysis*, 43(2):904–924, 2011.

[3] MF Ashby. Multi-objective optimization in material design and selection. *Acta materialia*, 48(1):359–369, 2000.

[4] Peter M Attia, Aditya Grover, Norman Jin, Kristen A Severson, Todor M Markov, Yang-Hung Liao, Michael H Chen, Bryan Cheong, Nicholas Perkins, Zi Yang, et al. Closed-loop optimization of fast-charging protocols for batteries with machine learning. *Nature*, 578(7795):397–402, 2020.

[5] Christoph Bader, Dominik Kolb, James C Weaver, Sunanda Sharma, Ahmed Hosny, João Costa, and Neri Oxman. Making data matter: Voxel printing for the digital fabrication of data across scales and domains. *Science advances*, 4(5):eaas8652, 2018.

[6] Mukund Balasubramanian, Eric L Schwartz, Joshua B Tenenbaum, Vin de Silva, and John C Langford. The isomap algorithm and topological stability. *Science*, 295(5552):7–7, 2002.

[7] Francois Barthelat, Zhen Yin, and Markus J Buehler. Structure and mechanics of interfaces in biological materials. *Nature Reviews Materials*, 1(4):1–16, 2016.

[8] Eric Bradford, Artur M Schweidtmann, and Alexei Lapkin. Efficient multiobjective optimization employing gaussian processes, spectral sampling and a genetic algorithm. *Journal of global optimization*, 71(2):407–438, 2018.

[9] Keith T Butler, Daniel W Davies, Hugh Cartwright, Olexandr Isayev, and Aron Walsh. Machine learning for molecular and materials science. *Nature*, 559(7715):547–555, 2018.

[10] Desai Chen, Mélina Skouras, Bo Zhu, and Wojciech Matusik. Computational discovery of extremal microstructure families. *Science advances*, 4(1):eaao7005, 2018.

[11] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.

[12] Leon S Dimas, Graham H Bratzel, Ido Eylon, and Markus J Buehler. Tough composites inspired by mineralized natural materials: computation, 3d printing, and testing. *Advanced Functional Materials*, 23(36):4629–4638, 2013.

[13] Leon S Dimas and Markus J Buehler. Modeling and additive manufacturing of bio-inspired composites with tunable fracture mechanical properties. *Soft Matter*, 10(25):4436–4442, 2014.

[14] Tao Du, Josie Hughes, Sebastien Wah, Wojciech Matusik, and Daniela Rus. Underwater soft robot modeling and control with differentiable simulation. *IEEE Robotics and Automation Letters*, 6(3):4994–5001, 2021.

[15] Randall M Erb, Rafael Libanori, Nuria Rothfuchs, and André R Studart. Composites reinforced in three dimensions by using low magnetic fields. *Science*, 335(6065):199–204, 2012.

[16] Javier González, Zhenwen Dai, Philipp Hennig, and Neil Lawrence. Batch bayesian optimization via local penalization. In *Artificial intelligence and statistics*, pages 648–657. PMLR, 2016.

[17] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014.

[18] Alan Arnold Griffith. Vi. the phenomena of rupture and flow in solids. *Philosophical transactions of the royal society of london. Series A, containing papers of a mathematical or physical character*, 221(582-593):163–198, 1921.

[19] Grace X Gu, Chun-Teh Chen, and Markus J Buehler. De novo composite design based on machine learning algorithm. *Extreme Mechanics Letters*, 18:19–28, 2018.

[20] Grace X Gu, Chun-Teh Chen, Deon J Richmond, and Markus J Buehler. Bioinspired hierarchical composite design using machine learning: simulation, additive manufacturing, and experiment. *Materials Horizons*, 5(5):939–945, 2018.

[21] Grace X Gu, Flavia Libonati, Susan D Wettermark, and Markus J Buehler. Printing nature: Unraveling the role of nacre's mineral bridges. *Journal of the mechanical behavior of biomedical materials*, 76:135–144, 2017.

[22] Grace X Gu, Susan Wettermark, and Markus J Buehler. Algorithm-driven design of fracture resistant composite materials realized through additive manufacturing. *Additive Manufacturing*, 17:47–54, 2017.

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[24] MZ Hossain, C-J Hsueh, B Bourdin, and K Bhattacharya. Effective toughness of heterogeneous media. *Journal of the Mechanics and Physics of Solids*, 71:15–32, 2014.

[25] Yu-Chuan Hsu, Chi-Hua Yu, and Markus J Buehler. Using deep learning to predict fracture patterns in crystalline solids. *Matter*, 3(1):197–211, 2020.

[26] Yuanming Hu, Tzu-Mao Li, Luke Anderson, Jonathan Ragan-Kelley, and Frédo Durand. Taichi: a language for high-performance computation on spatially sparse data structures. *ACM Transactions on Graphics (TOG)*, 38(6):1–16, 2019.

[27] Leonid V Kantorovich. On the translocation of masses. In *Dokl. Akad. Nauk. USSR (NS)*, volume 37, pages 199–201, 1942.

[28] Piaras Kelly. Solid mechanics part i: An introduction to solid mechanics. *Solid mechanics lecture notes*, pages 241–324, 2013.

[29] Yongtae Kim, Youngsoo Kim, Flavia Libonati, and Seunghwa Ryu. Designing tough isotropic structural composite using computation, 3d printing and testing. *Composites Part B: Engineering*, 167:736–745, 2019.

[30] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[31] Joshua Knowles. Parego: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, 10(1):50–66, 2006.

[32] Mina Konakovic Lukovic, Yunsheng Tian, and Wojciech Matusik. Diversity-guided multi-objective bayesian optimization with batch evaluations. *34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, 2020.

[33] Li-Bo Mao, Huai-Ling Gao, Hong-Bin Yao, Lei Liu, Helmut Cölfen, Gang Liu, Si-Ming Chen, Shi-Kuo Li, You-Xian Yan, Yang-Yi Liu, et al. Synthetic nacre by predesigned matrix-directed mineralization. *Science*, 354(6308):107–110, 2016.

[34] R Timothy Marler and Jasbir S Arora. Survey of multi-objective optimization methods for engineering. *Structural and multidisciplinary optimization*, 26(6):369–395, 2004.

[35] George Mayer. Rigid biological systems as models for synthetic composites. *Science*, 310(5751):1144–1147, 2005.

[36] Marc André Meyers, Po-Yu Chen, Albert Yu-Min Lin, and Yasuaki Seki. Biological materials: structure and mechanical properties. *Progress in Materials Science*, 53(1):1–206, 2008.

[37] MJ Mirzaali, A Herranz de la Nava, D Gunashekar, M Nouri-Goushki, RPE Veeger, Quentin Grossman, L Angeloni, MK Ghatkesar, LE Fratila-Apachitei, Davide Ruffoni, et al. Mechanics of bioinspired functionally graded soft-hard composites made by multi-material 3d printing. *Composite Structures*, 237:111867, 2020.

[38] Kevin N Otto and Erik K Antonsson. Trade-off strategies in engineering design. *Research in Engineering Design*, 3(2):87–103, 1991.

[39] M Ramulu and AS Kobayashi. Mechanics of crack curving and branching—a dynamic fracture analysis. *Dynamic fracture*, pages 61–75, 1985.

[40] Robert O Ritchie. The conflicts between strength and toughness. *Nature materials*, 10(11):817–822, 2011.

[41] Artur M Schweidtmann, Adam D Clayton, Nicholas Holmes, Eric Bradford, Richard A Bourne, and Alexei A Lapkin. Machine learning meets continuous flow chemistry: Automated optimization towards the pareto front of multiple objectives. *Chemical Engineering Journal*, 352:277–282, 2018.

[42] Eftychios Sifakis and Jernej Barbic. Fem simulation of 3d deformable solids: a practitioner's guide to theory, discretization and model reduction. In *Acm siggraph 2012 courses*, pages 1–50. 2012.

[43] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

[44] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *nature*, 550(7676):354–359, 2017.

[45] Justin Solomon, Fernando De Goes, Gabriel Peyré, Marco Cuturi, Adrian Butscher, Andy Nguyen, Tao Du, and Leonidas Guibas. Convolutional wasserstein distances: Efficient optimal transportation on geometric domains. *ACM Transactions on Graphics (TOG)*, 34(4):1–11, 2015.

[46] Cédric Villani. *Topics in optimal transportation*. Number 58. American Mathematical Soc., 2003.

[47] Ulrike GK Wegst, Hao Bai, Eduardo Saiz, Antoni P Tomsia, and Robert O Ritchie. Bioinspired structural materials. *Nature materials*, 14(1):23–36, 2015.

[48] Zhen Yin, Florent Hannard, and Francois Barthelat. Impact-resistant nacre-like transparent materials. *Science*, 364(6447):1260–1263, 2019.

[49] Chuangqi Zhao, Pengchao Zhang, Jiajia Zhou, Shuanhu Qi, Yoshihiro Yamauchi, Ruirui Shi, Ruochen Fang, Yasuhiro Ishida, Shutao Wang, Antoni P Tomsia, et al. Layered nanocomposites by shear-flow-induced alignment of nanosheets. *Nature*, 580(7802):210–215, 2020.