A High-Throughput Statistical Homogenization Technique to Convert Realistic Microstructures into Idealized Periodic Unit Cells

S.C. Foster[a], Justin W. Wilkerson[a,b,*]

[a] *J. Mike Walker '66 Department of Mechanical Engineering, Texas A&M University, College Station, TX 77843*
[b] *Department of Materials Science & Engineering, Texas A&M University, College Station, TX 77843*

Abstract

Metal alloys frequently contain distributions of second-phase particles that deleteriously affect the material behavior by acting as sites for void nucleation. These distributions are often extremely complex and processing can induce high levels of anisotropy. The particle length-scale precludes high-fidelity microstructure modeling in macroscale simulations, so computational homogenization methods are often employed. These, however, involve simplifying assumptions to make the problem tractable and many rely on periodic microstructures. Here we propose a methodology to bridge the gap between realistic microstructures composed of anisotropic, spatially varying second-phase void morphologies and an idealized periodic microstructure with roughly equivalent mechanical response. We create a high-throughput, parametric study to investigate 96 unique bridging methods. We apply our proposed solution to rolled AZ31B magnesium alloy, for which we have a rich dataset of microstructure morphology and mechanical behavior. Our methodology converts a μ-CT scan of the realistic microstructure to idealized periodic unit cell microstructures that are specific to the loading orientation. We recreate the unit cells for each parameter set in a commercial finite element software, subject them to macroscopic uniaxial loading conditions, and compare our results to the datasets for the various loading orientations. We find that certain combinations of our parameters capture the overall stress-strain response, including anisotropy effects with some degree of success. The effect of different parameter options are explored in detail and we find that excluding certain particle populations from the analysis can give improved results.

Keywords: second-phase particles, high-throughput multiscale modeling, magnesium alloy, realistic microstructures, material anisotropy

1. Introduction and Background

Predictive models that capture the damage mechanics of metals largely fail to capture the heterogeneous effects due to fluctuations of the microstructure distributions [1]. Such fluctuations play a key role in material behavior as realistic microstructures are often random and damage is an inherently local process [2]. Many of the models developed in the past 50 years involved simplifying assumptions to make the problem tractable, e.g., assumptions of periodicity, particle/void shape simplifications, etc. [3, 4]. Bridging the gap between realistic microstructures and these simplified damage models with periodic microstructures is quite challenging. In this paper, we assume the existence of a periodic microstructure that provides an equivalent response to a realistic microstructure and propose a methodology that creates a statistically-informed bridge between these two microstructures.

1.1. Computational Homogenization

We can achieve meaningful advances in material performance through a two-step process: (i) acquiring a thorough knowledge of the complex physics underlying material behavior and (ii) coupling this knowledge

*wilkerson@tamu.edu

with advanced material processing techniques to produce engineered materials [5–7]. The first step of this process requires a robust modeling approach as the mechanics that govern material behavior are myriad, intricately coupled, and operate on various time- and length-scales, i.e., the atomistic scale, mesoscale, and macroscale [8–11]. Several of these underlying physical damage mechanisms include the initial distribution and evolution of crystallographic defects, grain sizes/orientations, precipitates/inclusions, voids, etc. [12, 13]. These features affect the damage initiation, propagation, and eventual fracture [14, 15]. Since individual defects cannot be feasibly resolved in large-scale simulations, we instead rely on computational homogenization methods to represent the evolution of the physics in a continuum formulation, i.e., hierarchical multiscale modeling [7–9, 16, 17].

Computational homogenization schemes often use averaging theorems to capture the microstructure evolution and rely on the separation of scales principle [7, 9]. Generally, the microstructure is explicitly resolved on a single length-scale in a statistically-representative domain referred to as a representative volume element (RVE) [9, 11]. A bridging method is then employed to extract the pertinent features and resolve them in an averaged sense at a larger length-scale. This process is conducted in either a coupled or decoupled procedure [11, 18]. In a coupled method, the RVE is solved at each timestep for the given boundary conditions of the macroscale problem and the response is taken as the constitutive response of the material. In a decoupled method, the RVE is solved *a priori* and the constitutive response is stored in the material description for the macroscale problem. Although this recursive solution for coupled problems can lead to higher accuracy, it also requires much larger computational resources than are necessary for decoupled methods.

Decoupled, hierarchical modeling methods are therefore desirable when seeking to minimize computational resources. These continuum models are limited, however, as they generally rely on idealized, periodic microstructures with defects represented as simple shapes, e.g., spheres or ellipsoids [3, 19–21]. Since most real-world materials are infinitely more complex than these idealized representations, the modeler's quest is to construct a methodology that bridges the gap between the realistic defects and the idealized damage model microstructure [4, 8]. We focus this paper on creating a bridge between the microscale ($\mathcal{O}(10^{-6}\text{m})$ and macroscale ($\mathcal{O}(\text{m})$). We use a decoupled approach that utilizes microstructural simulation data from Lloyd et al. for a magnesium alloy, AZ31B. That said, we believe that this methodology could be easily adapted to other microstructural features (e.g., voids, structural porosity, inclusions in metal matrix composites, etc.) with similar success. [22].

1.2. Second-Phase Particles

Several industries (e.g., space, defense, transportation) widely use alloys comprised of a metallic matrix with intermetallic particles (often referred to as second-phase particles, inclusions, or precipitates) [5, 14]. Second-phase particles usually occur as a by-product of the alloying process and are generally harder and more brittle than the matrix phase in which they are embedded [23]. This hardens the bulk material as the particles impede dislocation movement, which is an important component of plastic flow [24]. Additionally, second-phase particles are often non-uniformly distributed and consist of various sizes, orientations, and shapes [19, 22]. Processing methods, i.e., cold-rolling and extrusion processes, also introduce anisotropy into the distribution by flattening, elongating, and/or reorienting the second-phase particles [19, 22, 25].

The deleterious effects of second-phase particles on alloy behavior is a well-known and widely studied phenomenon [26]. The particles play a crucial role in ductile damage wherein the material fails through a process of void nucleation, growth, and coalescence [13, 21, 27–29]. Voids often nucleate at the second-phase particles, either through debonding of the particle-matrix interface and/or particle cracking [13, 25, 28, 30]. Additionally, in disordered second-phase particle distributions, clustering can have pronounced effects on void nucleation and coalescence [19]. However, attempts to incorporate this knowledge into predictive failure models is obfuscated by the convolution of rapid void growth/coalescence regions with local stress-strain states [13].

Historically, characterization of realistic second-phase particle distributions presented a non-trivial problem. Material properties are difficult to measure and characterization methods such as serial sectioning result in the destruction of the sample. Micro-computed tomography (µ-CT) has emerged in recent years as an attractive tool to non-destructively quantify the second-phase particle morphology, often with a high

degree of fidelity [13, 31]. This scanning technology allows us to digitally reconstruct the microstructure and incorporate the particle geometry into small-scale simulations ($\mathcal{O}(\mu m)$). We can then compare these to experiments with miniaturized specimens or use them in RVE studies [22, 32]. In the next section, we provide a brief overview of relevant literature that grapples with realistic characterizations of non-uniform particle morphology.

1.3. Multiscale Modeling of Realistic Second-Phase Particle Morphologies

The past several decades have seen an increase in numerical solutions to heterogeneous materials enabled by increases in computational capabilities [12]. In the 1990s, Ghosh et al. developed a coupled multiscale method known as the Voronoi Cell Finite Element Method (VCFEM) to represent non-periodic microstructures [33]. This methodology uses Dirichlet tessellation (i.e., Voronoi polygons) to form polygonal elements that each contain a single heterogeneity. A coupled finite element (FE) scheme is then used to resolve the microscale and macroscale material response. These elements were cumulatively modified over the next decade to include elasticity [33], plasticity [34], and void nucleation (due to particle cracking) [35]. Although more robust than homogenized solutions, the application of this method to large-scale problems was limited by the mesh size, the smallest element of which is determined by the scale of inter-particle distances. To mitigate this, Ghosh et al. proposed an adaptive meshing algorithm which transitions from the macroscale to the inclusion scale in regions of high stress or strain gradients. Despite this reduction in computational cost, the VCFEM methodology is still a fairly complex approach and is not readily available in commercial FE solvers.

Other studies have attempted to quantify the anisotropy of second-phase particle distributions and model microstructural phenomena. Hannard et al. studied the anisotropy induced by the second-phase particle morphology on the ductility of rolled Al 6000-series alloys in 2018 [1]. They found that tensile tests of Al 6065 in the rolled direction and the transverse direction exhibited different fracture strains which were explained through particle clustering effects. They also emphasized the need for anistropic and clustering effects in current damage models and proposed the Pair Correlation Function (PCF) as an effective method of quantifying the degree of particle distribution anisotropy. Abedini et al. investigated the effects of particle clustering by creating a periodic microstructure that consisted of multi-particle clustered unit cells with spherical particles [36]. Their analysis, however, did not investigate the effect of this clustering on the damage behavior. A 2019 study by Pinz et al. focused on creating statistically equivalent RVEs (SERVEs) from realistic particle morphologies in a nickel-based superalloy [37]. They approximated µ-CT scans of nano-scale particles as a distribution of generalized super ellipsoids and used various methods to generate statistical descriptors of the spacing between particles to find the smallest possible SERVE. Sarmah and Jain recently investigated interfacial debonding between second-phase particles and the matrix by conducting 2D FE simulations of a realistic Al alloy microstructure [38]. They used cohesive zones at the particle-matrix interface with properties informed by molecular dynamics (MD) simulations and modeled both realistic and idealized second-phase particle morphology (particles were represented as circles in the idealized case). Although they reported the effects of particle morphology on local decohesion behavior and the local stress fields, they did not report observations on the effective stress-strain behavior.

Several homogenized damage models with corresponding homogenization procedures have been proposed. In 1999, Wilkinson et al. proposed an approach to model non-uniform distributions of second-phase particles using a plastic extension of the self-consistent method [39]. They partition the continuum response into phases, each of which has a particle volume fraction and associated constitutive response (which includes a damage parameter based on the particle size distribution). Gammage et al. further developed this model in 2004 to statistically account for void coalescence effects which helps the model better capture the ductility [40]. Both of these models, however, only account for local variation in particle volume fraction, do not account for particle shapes, and only include realistic particle spacings as statistical probabilities. Ghosh et al. developed a homogenization-based continuum plasticity-damage (HCPD) model in 2009 which was based on based on the well-known Gurson-Tvergaard-Needleman (GTN) model and used the aforementioned VCFEM for the micromechanical analysis [14]. They modified the void nucleation criterion to account for observed inclusion distributions and included sufficient degrees-of-freedom to capture anisotropy in inclusion cracking. The developed HCPD, however, still requires micromechanical analyses of RVEs. Tekoğlu and

Pardoen sought to bypass this expensive RVE analysis by coupling a Mori-Tanka mean field homogenization method with a Gologanu-Leblond-Devaux (GLD) damage model [41]. They validated the model with a distribution of silicon particles in an aluminum alloy. This model assumes that the particles are all ellipsoids and only accounts for non-uniform distributions by adjusting the GLD void spacing aspect ratio. Ismail et al. also proposed a modeling strategy that precludes explicit resolution of the microstructure by dividing the material into "pseudo-grains" that each have a dominant second-phase particle orientation [42]. Each pseudo-grain is then modeled as a periodic distribution of aligned ellipsoids. This study, however, only accounts for damage predictions by probing the intensity of the stress at particle-matrix interfaces in the unit cells. In 2020, Olinger et al. fit a GTN model to data from direct numerical simulations (DNS) that explicitly resolved the anisotropic second-phase particle morphology of an AZ31B alloy [5]. However, the GTN model parameters were phenomenologically derived from the DNS stress-strain data and do not incorporate more detailed aspects of the second-phase particle morphology. A recent study by Xie et al. modeled realistic second-phase particle distributions in an Al alloy and used a GTN model in the matrix-phase of their RVEs to capture the damage phenomena [43]. They compared the realistic RVEs with various simplified microstructures to quantify how particle size, distribution, and shape affect the material properties.

1.4. Study Motviation and Overview

Despite these advances, we are unaware of any effort that seeks to investigate the existence of a simple, periodic microstructure that is "equivalent" to a realistic microstructure[1]. Many homogenized models, such as Gurson-type models, rely on analytic solutions to spheroidal or cylindrical-shaped voids in an elastoplastic medium. If an equivalent idealized, periodic microstructure exists, then linking the realistic microstructure to the Gurson-type microstructure has tangible benefits that would enable a simple, efficient bridge between a microstructural dataset and a homogenized damage model.

To further motivate our approach, imagine an entity such as a company or government laboratory that wishes to run microstructure-based FE simulations using µ-CT data they acquired. They have access to commercial FE software, e.g., Abaqus, ANSYS, etc., but do not have a robust set of custom user material subroutines (UMATs) and sophisticated scripts. Firstly, they would have to spend weeks converting the µ-CT data into a CAD model that they can import into the FE solver. This is labor intensive without expensive additional software. Once the geometry is created, the modeler must decide whether to model the particles and matrix as a single, multi-phase part or as separate parts and must define a material model for each region. This is a non-trivial step that involves choosing and fitting a material model for both the matrix and particles that must include damage in at least one of the phases. An additional technique is needed to model the interactions between the two phases, assuming they are not modeled as a single part. Perhaps cohesive elements are used which introduces an extra geometry and mesh generation challenge. The choices made in the construction of this FE model have implications for the accuracy of the modeler's RVE solution. Secondly, the modeler must choose how many loading paths (proportional and/or non-proportional) they believe are necessary to characterize the second-phase particle anisotropy. This can quickly reach hundreds of simulations to fully characterize the homogenized response. The modeler does not possess infinite time so they now have to grapple with the minimum amount of material necessary for representativity and might have to ignore a large portion of their µ-CT data. Finally, they face the problem of creating a constitutive model from their high-throughput suite of RVE studies. They have presumably generated a large amount of anisotropic, tabulated tensorial data. Commercial FE solvers do not possess the infrastructure to directly use this data, so the modeler must now write a UMAT or lose a high amount of fidelity by fitting their RVE data to a readily available homogenized damage model (see Figure 1).

Conversely, finding a conversion process between a realistic microstructure and a Gurson-type microstructure would preclude the months of work required for the above study. Once the process is established, a

[1]For the sake of simplicity, we define "equivalency" in this paper as a microstructure that exhibits roughly similar uniaxial stress-strain behavior. However, an ideal "equivalent" periodic microstructure would capture the general stress tensor-strain tensor response for a broad range of proportional and non-proportional loadings. Such multiaxial loading is beyond the scope of this paper.

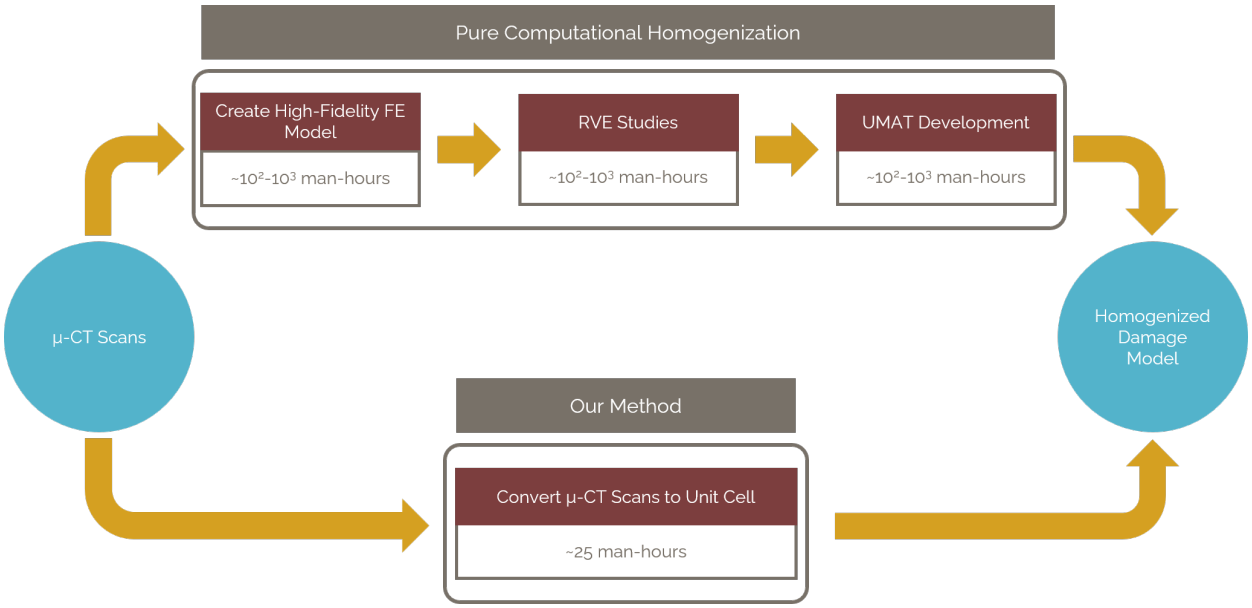


Figure 1: Comparison of the costs between pure computational homogenization techniques and our proposed method to create a homogenized damage model from a second-phase particle morphology.

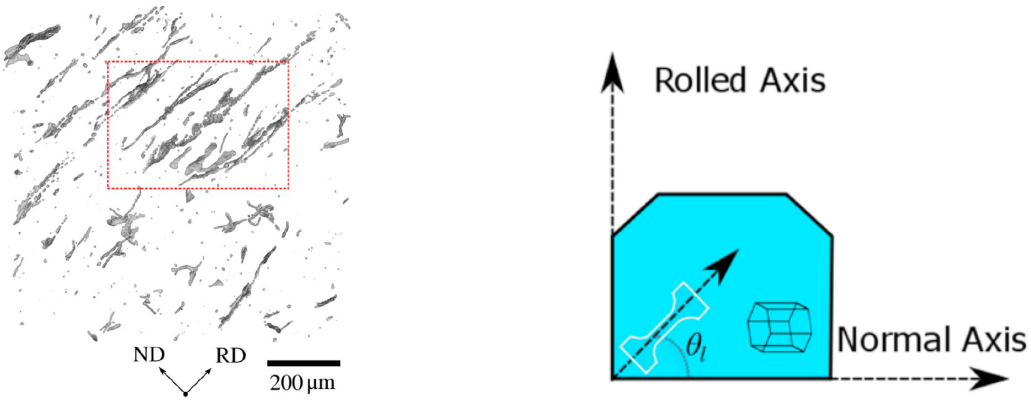
modeler could quickly analyze a large μ-CT dataset without the need for CAD modeling. They could conduct a high-throughput suite of FE simulations that they could run at very little cost by using the simplicity of a unit cell coupled with periodic boundary conditions. If they wish to use Gurson-type models that are built-in to commercial FE software then they could fit those parameters to the unit cell data. Better yet, if they have access to anisotropic extensions of the Gurson model (such as the Gologanu-Leblond-Devaux model), then they can input the geometric parameters from the unit cell without ever running an FE unit cell study. The modeler has now arrived at a homogenized damage model that they can immediately use to simulate structural level deformation and localization with an approximate cost savings of months as shown in Figure 1.

In this paper, we propose a method to convert a realistic second-phase particle morphology to an idealized, periodic microstructure consisting of a unit cell with a single void. We take μ-CT scans of rolled AZ31B and quantify the volumes of matrix and second-phase particle material. Since damage nucleates at second-phase particles via particle cracking and/or interfacial decohesion, we approximate the particles as voids with no stress-carrying capacity [29, 44]. We then use a MATLAB algorithm to fit ellipsoids to the particle shapes and extract the spatial and size distributions. Finally, we create a periodic microstructure consisting of a single-void unit cell using the average size and spacing values. Assuming a hexagonal repeating unit cell allows us to use axisymmetry in FE calculations and reduces the averaged parameters to four dimensions, L_z, L_r, R_r, and R_z [21]. We repeat this process for varying loading orientations in the plane formed by the normal- and rolled-directions (ND-RD plane) and compare to the data generated from [22]. Additionally, we consider various means of creating ellipsoids from particle shapes, measuring the spacing between particles, and generating statistical averages. This results in a parametric, high-throughput study that allows us to compare the effectiveness of such choices.

2. Methodology

2.1. Experimental and Numerical Data

Our work uses the experimental and numerical data published by Lloyd et al. [22]. Their study investigated the effects of the realistic particle morphology on the uniaxial tensile behavior of rolled AZ31B



(a) (b)

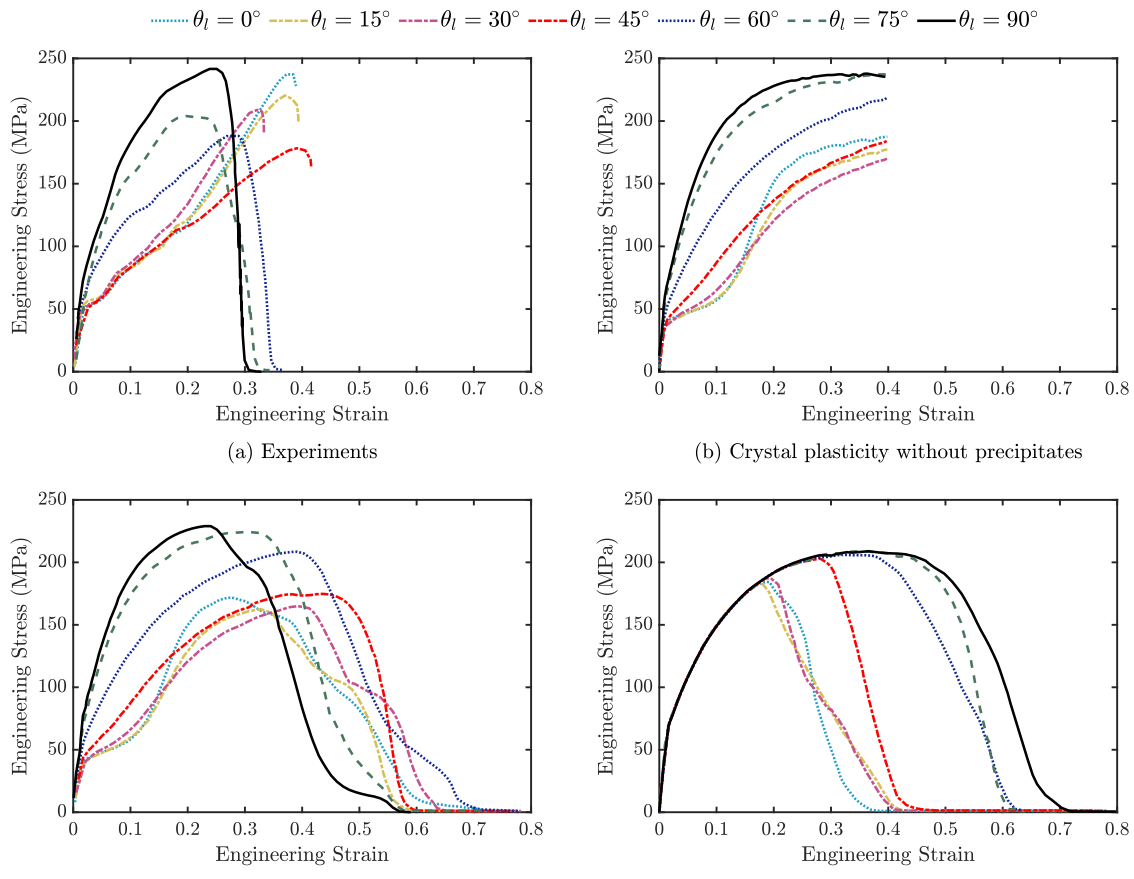
Figure 2: (a) Particle morphology of rolled AZ31B. "Stringer" particles are clearly seen oriented along the RD. (b) Definition of the loading vector defined by the cutting angle, θ, with $\theta = 0°$ corresponding to the tensile direction along the ND and $\theta = 90°$ corresponding to the tensile direction along the RD.

magnesium. The cold-rolling processing of the material leads to particle morphology anisotropy with long "stringer" particles preferentially aligned along the rolled direction (Figure 2a). These particles range in size, with the smallest particles presumably being $Mg_{17}Al_{12}$ and the largest presumably being Al_8Mn_5, c.f. [45]. Mg alloys also possess high levels of anisotropy due to their hexagonal close packed (HCP) crystal structure and twinning deformation behavior which manifests as tension-compression asymmetry [5]. Therefore, the anisotropy observed in the experimental data generally stems from a complex interplay of physical mechanisms including the crystal structure, twinning deformation, grain texture, and second-phase particles.

Lloyd et al. studied this anisotropy by cutting miniature dog-bone tensile specimens from an AZ31B sheet at various orientations in the plane formed by the normal- and rolled-directions (ND and RD, respectively). The orientations were denoted by the angle θ measured from the ND as shown in Figure 2b. They obtained seven specimens by varying the loading vector, $\boldsymbol{e_z}$, in 15° increments from 0° ($\boldsymbol{e_z}$ aligned along the ND) to 90° ($\boldsymbol{e_z}$ aligned along the RD). The specimens were cut from a 6in. $\times$ 6in. plate and milled down to dogbone specimens with gauge dimensions 1.0mm $\times$ 0.5mm $\times$ 0.2mm. Tensile tests were conducted on the specimens using a desktop miniaturized Kolsky bar setup. Three tests were run for each orientation and the engineering stress-strain response was measured by analyzing the strain gauges on the incident and transmitted bars. Representative stress-strain responses are shown in Figure 3a. They characterized the particle morphology in the gauge sections using µ-CT scans of the dog-bone specimens with a spatial resolution of 1 µm. We refer the reader to [22] for a complete description of the experimental setup and procedure.

In the second portion of the study, they conducted a set of direct numerical simulations (DNS), i.e., high-fidelity FE simulations, using the µ-CT scans of the gauge sections. The DNS differ from an RVE since the dimensions of the resolved microstructure match the dimensions of the gauge sections in the miniaturized Kolsky bar tests. However, the DNS and experiments may not be representative of bulk AZ31B behavior due to the small sample size. The particle morphology was recreated for seven simulations, each corresponding to a distinct loading orientation, $\boldsymbol{e_z}$. They assumed elastic-plastic material properties for the matrix and particles and formulated a critical-stress failure criterion, σ_{fail}, for the particle regions (no failure criterion existed for the matrix phase). The FE simulations used an arbitrary Lagrangian-Eularian formulation to handle the void nucleation, and growth process while conserving mass. This portion of the study used four simulation sets to investigate various features of AZ31B:

1. A uniform distribution of spherical particles embedded in a matrix. Both the particles and the matrix were modeled with an identical, isotropic elastic-plastic material model.(Results not shown.)
2. Experimentally measured particles embedded in a matrix. Both the particles and the matrix were modeled with an identical, isotropic elastic-plastic material model. (Results shown in Figure 3c.)



(a) Experiments

(b) Crystal plasticity without precipitates

(c) Crystal plasticity with realistic precipitates

(d) Isotropic plasticity with realistic precipitates

Figure 3: Stress-strain data from (a) tensile Kolsky bar experiments, (b) FE crystal-plasticity model without precipitates, (c) FE DNS with crystal plasticity and realistic precipitates, and (d) FE DNS with isotropic plasticity and realistic precipitates. Reproduced from [22].

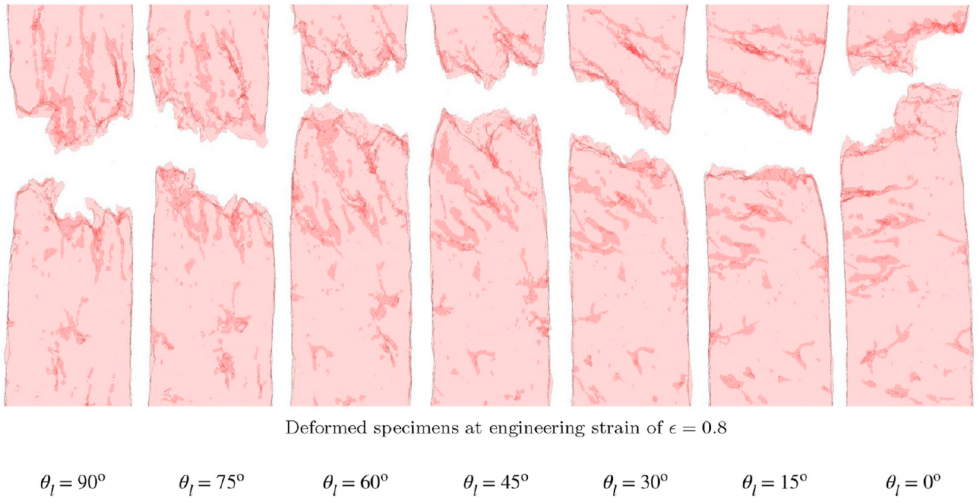


Figure 4: Fracture surfaces as predicted by direct numerical simulations (DNS) with the realistic second-phase particle morphology. The particle and matrix material are modeled as elastic-plastic with isotropic hardening. This isolates the effects of the second-phase particles on the damage behavior. Reproduced from [22].

3. A uniform distribution of spherical particles embedded in a matrix. The particles were modeled with an isotropic elastic-plastic material model and the matrix was modeled with an anisotropic crystal plasticity model. (Results not shown.)
4. Experimentally measured particles embedded in a matrix. The particles were modeled with an isotropic elastic-plastic material model and the matrix was modeled with an anisotropic crystal plasticity model. (Results shown in Figure 3c.)

The details of the crystal plasticity model used in the second two simulation sets are given in [22]. The crystal plasticity uniaxial stress-strain responses for each loading orientation are reproduced in Figure 3b. The other two simulation sets make use of J_2 plasticity with a Voce-type isotropic hardening law, i.e.

$$\sigma_y = \sigma_{0,s} + \sigma_s^{\infty}\left[1 - \exp\left(-\frac{h_{1,s}}{\sigma_s^{\infty}}\bar{\varepsilon}^{pl}\right)\right] \tag{1}$$

Here, σ_y is the flow stress, $\sigma_{0,s}$ is the initial yield strength, σ_s^{∞} is a parameter that controls the saturation strength, $h_{1,s}$ is a hardening modulus, and $\bar{\varepsilon}^{pl}$ is the equivalent von Mises plastic strain. This results in a hardening curve that asymptotes to a flow stress equal to $\sigma_{0,s} + \sigma_s^{\infty}$. One should note that, in reality, the material properties for the Mg matrix phase, the $Mg_{17}Al_{12}$ particles, and the Al_8Mn_5 particles are not equivalent [23].

The conditions in the third simulation type (Figure 3d) illuminate the anisotropy stemming from the particle morphology quite nicely. The fracture surfaces for this case are shown in Figure 4. We choose to validate our methodology against this numerical data as these particular simulations isolate the role of second-phase particles on the damage response. We also extend our approach using an approximation of the crystal plasticity data in Figure 3b and compare to the experimental data in Figure 3a.

2.2. Statistical Analysis of μ-CT Data

For our study, we digitally reconstruct the AZ31B microstructure used in [22] as a 3D array from μ-CT scans. Prior to our procurement, the data was converted to a binary array where a value of 0 corresponds to matrix material and a value of 1 corresponds to particle material. Plotting the 3D array yields a digital approximation of the microstructure which is shown in Figure 5. The resolution of the resultant image

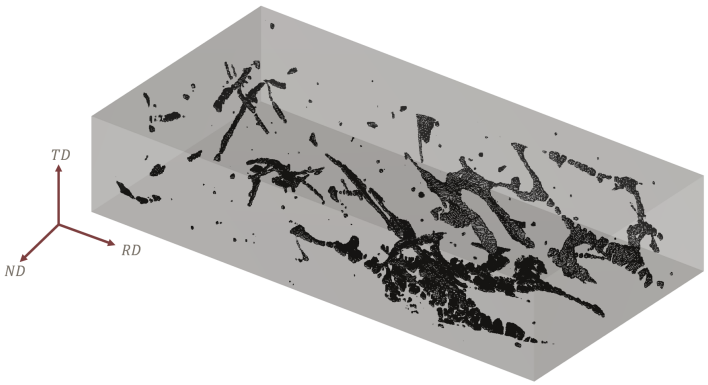


Figure 5: 3D reconstruction of rolled AZ31B microstructure from µ-CT scans showing second-phase particles. A non-uniform distribution can be seen with regions of various particle volume fractions. The digital reconstruction also clearly shows the particle anisotropy induced by the rolling process.

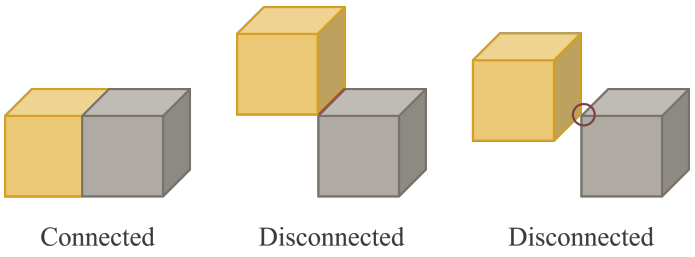


Figure 6: Voxel connectivity conditions for voxels that correspond to particle material. The particle voxels are only considered connected if they share a face.

is limited by the µ-CT resolution of 1 µm. Therefore, each element (or voxel) of the 3D binary array corresponds to 1 µm^3 of material. Our sample size is 701µm × 350µm × 139µm.

Once the microstructure is reconstructed from the µ-CT data, we extract the particle morphology. We first define a connectivity condition that determines the set of voxels that belong to each particle. Three possibilities exist for defining connectivity, with connected voxels defined as voxels i) sharing a face, ii) sharing an edge, or iii) sharing a vertex. For this project, we define connectivity using the first criterion as shown in Figure 6. We use the MATLAB function `bwconncomp` with the selected connectivity condition to determine the number, size, and position of the second-phase particles [46]. For our sample, the analysis resulted in a particle count of 479, with 134 of those particles comprised of a single voxel, i.e., $V_p = 1$ µm^3, where V_p is the particle volume. The particle volumes are shown in Figure 7a and range from 1 µm^3 to $\sim 2.86 \times 10^4$ µm^3.

Many of the particles have highly complex shapes. The rolling process flattens many particles, as expected. Additionally, there are several "stringer" type particles that are highly elongated and generally oriented along the RD. The smaller particles are generally ellipsoidal or flat, with varying levels of irregular features. Additionally, several of the larger particles consist of a collection of "finger-like" protrusions that splay out in various directions and are joined at a central hub. The variety of shapes, sizes, and spacings in this complex, realistic microstructure presents several unique challenges when trying to convert to an idealized, periodic representation. We therefore conduct a parametric study where we define various options for several pertinent parameters, i.e., number of particles accounted for, ellipsoidal approximation method of the particles, spacing measurement, and ellipsoidal/spacing averages. We place these parameters with their associated values into a test matrix (Table 1) and use all possible combinations to provide ∼100 unique unit cell constructions. Although we provide a comprehensive assessment of these parameters, we also acknowledge the existence of hundreds of hypothetical schemes that could be used to weight various aspects

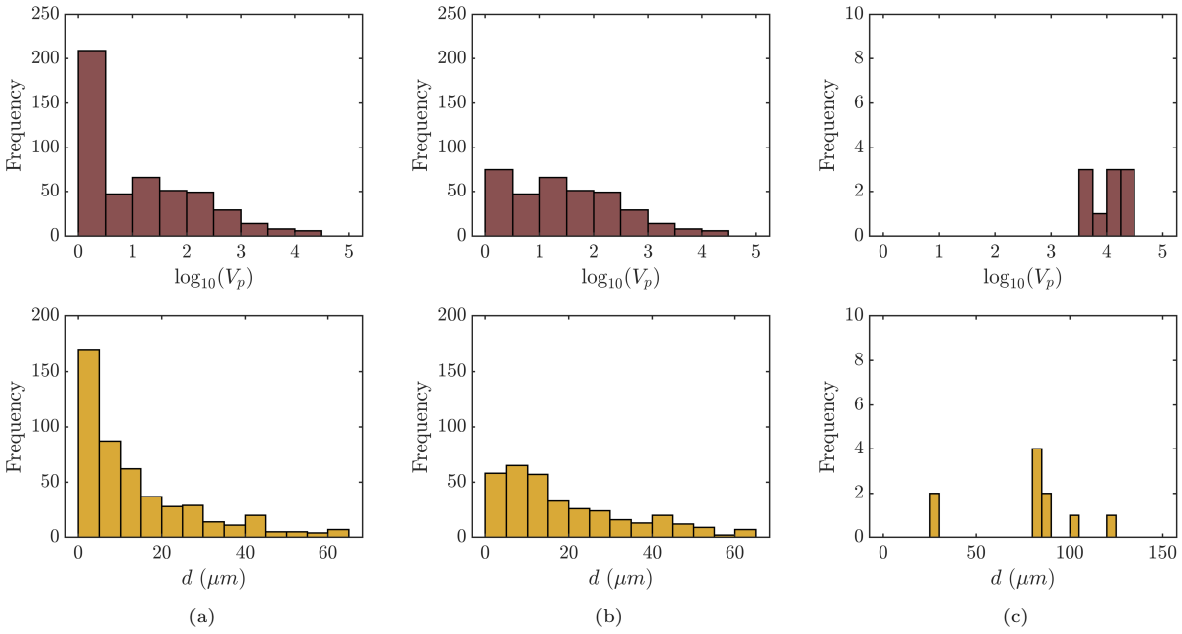


Figure 7: Histograms showing the distributions of particle volumes, V_p, and spacings, d, measured from the µ-CT data for (a) all particles, (b) all except single-voxel particles, and (c) only the ten largest particles.

of the full µ-CT statistical data. We leave these unexplored as they are outside the scope of this paper. Our study parameters and associated options are detailed below.

2.2.1. Parameter 1: Particle Inclusion

The first challenge in homogenization is deciding which particles observed in the µ-CT data most affect the damage process. For instance, studies show that larger particles tend to fail first in the absence of small-particle clustering [47] while small particles, i.e., < 3 µm, tend to nucleate a population of smaller voids that accelerate void coalescence [44]. Although we can include all of our available data in our analysis, selecting a subset of "important" particles might yield more accurate responses. One obvious choice is to forgo such a decision and include all measured particles in the analysis. However, due to resolution effects and image manipulation, there are questions regarding the validity of counting single-voxel particles. Additionally, assuming that all measured single-voxel particles are valid, they might be better categorized as a secondary population of particles that should be modeled on a separate length-scale, similar to [48]. Therefore, one might reasonably exclude all the single-voxel particles from the analysis. Finally, although we have high-fidelity microstructural data, the damage might depend almost entirely on the largest particles and the smaller, more dispersed particles might not significantly affect the results. An accurate model might achieve better results by only including some subset of the largest particles and ignoring the hundreds of smaller particles. In our analysis we account for all three of these choices by 1) including all particles in our analysis, 2) excluding all single-voxel particles in our analysis (which reduces the number of particles from 479 to 345), and 3) including only the ten largest particles (by volume). The V_p distributions for these particle populations are given in Figure 7. We place these choices into the first rows of our test matrix in Table 1, where they are each assigned a numeric value.

2.2.2. Parameter 2: Ellipsoid Construction

Our second challenge is to approximate each particle as a simple geometrical shape. Since we wish to avoid more costly 3D simulations and want the periodic microstructure to work well with homogenized damage models, we choose an ellipsoid of revolution i.e., a spheroid. This allows us to introduce void shape anisotropy into our unit cells and allows us to utilize axisymmetry in our FE simulations. Fitting a spheroid

Variable	Option	Numeric Code
Number of Particles (NP)	All Particles	1
	Exclude Single-Voxel Particles	2
	10 Largest Particles	3
Ellipsoid Construction (EC)	Double-Centroid Method	1
	Projected Area Method	2
	Combined Method	3
	Combined Method (Match Volume)	4
Spacing Construction (SC)	Centroid-to-Centroid Nearest Neighbor Distance	1
	Centroid-to-Centroid Nearest Neighbor Distance with Ellipsoid Projections Subtracted	2
Ellipsoid Averaging Scheme (EA)	Simple Average	1
	Volume-Weighted Average	2
Spacing Averaging Scheme (SA)	Simple Average	1
	Volume-Weighted Average (Using Volume of Ellipsoid Pair for Each Spacing)	2

Table 1: Parameter test matrix. Each parameter is given a set of numerical values that correspond to an option for that parameter. The numerical identifiers are used to create a five-number unique I.D. for each unit cell set.

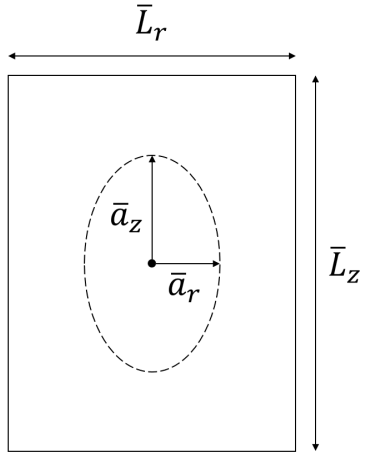


Figure 8: Axisymmetric unit cell geometry. The spacing between void centroids in the z- and r-directions are given by $\bar{L}_z$ and $\bar{L}_r$ respectively. The void radii are given by $\bar{a}_z$ and $\bar{a}_r$.

to each particle gives us two tunable parameters, a_r and a_z, where a_r is the semi-axis length of revolution and a_z is the semi-axis length along the axis of revolution. A representative spheroid is shown in our unit cell geometry in Figure 8.

Fitting an ellipsoid to a given particle is a non-straightforward process. The easiest method is to construct an error ellipsoid using the second central moments of the particle voxel distribution. The error ellipsoid dimensions are set by defining the chi-squared value corresponding to a desired probability value. The resultant ellipsoid is oriented with its axes aligned along the directions defined by the eigenvectors of the covariance matrix. Although this generally gives the best fit for a particle, the resulting ellipsoid distributions have varying orientations which inhibits finding an average ellipsoid size and shape such that it is oriented along the $z-$ and $r-$directions of our unit cell. Additionally, since we want to generate a unit cell for each loading direction, we require a distribution of ellipsoids with their a_z semi-axes aligned along the loading vector, $\boldsymbol{e_z}$. This precludes using the error ellipsoid method and forces us to formulate other ellipsoid generation methods.

We identify three main aspects of the particle geometry that we believe should be captured by any simple-shape fit:

i. the length of the particle along the loading direction $\boldsymbol{e_z}$,
ii. the projected area, A_p, of the particle on a plane with normal $\boldsymbol{e_z}$,
iii. the particle volume, V_p.

It is generally impossible to simultaneously fit all three of these metrics, except for nicely-shaped particles, and therefore we create four ellipsoid generation methods that weight each of these aspects differently.

The first method places the most weight on aspects (i) and (iii). To construct an ellipsoid, a particle is first sliced by a cutting plane formed by the particle centroid, G_p, and the loading vector, $\boldsymbol{e_z}$. We then compute the centroid of the upper half and lower half, denoted as G_p^u and G_p^l, respectively. We do this to mitigate any particular geometry quirks such as a u-shaped particle where counting voxels along a load vector that passes through G_p would not give an accurate representation of the particle length along that vector. Once we compute G_p^u and G_p^l, the vector length between the two, $\boldsymbol{l_G}$, is calculated. The analytic formula for the location of the centroid of a half-ellipsoid is $h = \frac{3}{8}a_z$. Substituting $\frac{1}{2}(\boldsymbol{l_g} \cdot \boldsymbol{e_z})$ for h, we can solve for a_z, i.e., $a_z = \frac{4}{3}(\boldsymbol{l_g} \cdot \boldsymbol{e_z})$. We then calculate a_r by setting the volume of the ellipsoid, V_e, equal to V_p and solving for a_r, i.e., $V_p = V_e = \frac{4}{3}\pi a_z a_r$. This method results in ellipsoids that capture some "effective length" along $\boldsymbol{e_z}$ and also capture V_p. However, the method does not capture A_p, except by coincidence.

The second method places the most weight on aspects (ii) and (iii). Here, the voxels of a given particle are projected onto the plane formed by G_p and $\boldsymbol{e_z}$. An error ellipse is then fit to the resultant 2-D projected-voxel distribution using the 2D version of the second central moments technique described above. We chose a chi-squared value of 6.251 which gives a probability value (p-value) between 0.95 and 0.975, i.e., the ellipse encompasses $95\% - 97.5\%$ of the in-plane points. The projected area must be a circle for a spheroid, so the resultant error ellipse is converted to a circle by setting the area of the ellipse, A_e equal to the area of a circle, A_c. The circle radius, a_r, is then found by the following expression, $a_r = \sqrt{a_1 a_2}$, where a_1 and a_2 are the semi-axes of the ellipse. We then solve for a_z by setting V_e equal to V_p, similarly to the first method. The resulting ellipsoids from this method approximately capture the projected area, A_p, and accurately capture V_p. They do not capture the particle dimensions along $\boldsymbol{e_z}$ except by coincidence.

The third method places the most weight on aspects (i) and (ii). First, we generate a_z using the strategy from the first method. Then we generate a_r using the strategy from the second method. V_p is ignored, and therefore only captured incidentally. The fourth and final method attempts to account for all three aspects by constructing an ellipsoid by way of the third method, and then scaling the resultant ellipsoid to match V_p, while keeping the aspect ratio $a_r : a_z$ constant. We accomplish this by introducing a scaling factor, β, into the expression for V_e and solving for β by setting V_e equal to V_p, i.e., $V_p = V_e = \frac{4}{3}\beta^3 \pi a_z a_r^2$. We then multiply a_z and a_r by β to obtain the new spheroid semi-axis lengths.

These methods provide four different ways to approximate a particle with a spheroid, taking into account three metrics of particle geometry. Each method provides a unique ellipsoid distribution for a given $\boldsymbol{e_z}$ and it is not immediately clear which method will yield better results. Our project therefore incorporates all four methods into our test matrix in Table 1 and assigns each method a numeric value, similarly to the particle size inclusion parameters. The effect of each method on the shape of the ellipsoid for a randomly chosen particle is given in Figure 9.

2.2.3. Parameter 3: Particle Spacing

Second-phase particles exist as discrete entities interspersed in a matrix phase. Therefore, a measure of the spacing between particles is necessary for conversion to a periodic unit cell. The nearest-neighbor distance provides a simple inter-particle spacing metric which measures the vector distance between a given particle centroid and the centroid of its nearest neighboring particle. Although this provides a rough estimate of the spatial distribution of particles, it is less useful in describing the amount of matrix material in between two particles as it does not account for particle volume. Measuring the amount of matrix-phase material between a particle and all its neighbors, however, is computationally prohibitive. To provide an approximation of this length, which is an arguably more influential aspect of the particle morphology than the centroid-to-centroid distance, we implement a second method that uses the fitted ellipsoid geometries of the particles to define an estimate of matrix phase material between a particle and its nearest neighbor.

Figure 9: Effect of ellipsoid generation method on the resulting shape of the ellipsoid. A single particle is reconstructed and shown from two directions: the loading direction along the z-axis (top row) and perpendicular to the loading direction (bottom row). The ellipsoid that results from each of the four options is shown for the particle, where the numeric identifier corresponds to the options as listed in Table 1.

The first method employs the nearest-neighbor algorithm to find the vector distance between a given particle's centroid and the centroids of all other particles. For a particle, i, the magnitude of the distances between its centroid and the centroids of all other particles are calculated and the minimum distance is stored as d, such that

$$d^i = \min \sqrt{\left(G_R^i - G_R^j\right)^2 + \left(G_N^i - G_N^j\right)^2 + \left(G_T^i - G_T^j\right)^2} \tag{2}$$

where $\{G_R, G_N, G_T\}$ are the centroid coordinates of a given particle in the material coordinate system (rolled-, normal-, and transverse-directions, respectively), i is the particle of interest, and j is an index that loops over every other particle. Once d is found for each particle, that distance is stored in vector form and the absolute value is taken of all the components such that, for the i^{th} particle,

$$\boldsymbol{L^i} = \left\{L_R^i, L_N^i, L_T^i\right\} = \left\{\left|G_R^i - G_R^{j,\min}\right|, \left|G_N^i - G_N^{j,\min}\right|, \left|G_T^i - G_T^{j,\min}\right|\right\} \tag{3}$$

This process is repeated for every particle and results in an associated nearest-neighbor distance, $\boldsymbol{L}$, for each particle. A 2D representation of this centroid-to-centroid distance, $\boldsymbol{L}$, is shown in Figure 10 for two particles that have been approximated by ellipsoids. The resulting distribution of the Euclidean norm of spacings, d is shown in Figure 7 for all three particle populations considered.

The second method indirectly measures the amount of matrix-phase material between the two nearest particles instead of $\boldsymbol{L}$. As previously mentioned, it is not computationally tractable to measure the true smallest distance of matrix material between two particles. Our solution is to use fitted ellipsoids to create an artificial "ligament length" between a given particle and its nearest neighbor. We do this by first taking the distance $\boldsymbol{L}$ (as defined above) between the particle of interest and its nearest neighbor. We then take the ellipsoids that correspond to each particle, project them onto the vector $\boldsymbol{L}$, and subtract off the projected lengths to determine the ligament length, $\boldsymbol{L'}$, as seen in Figure 10. The ellipsoid projection onto the vector $\boldsymbol{L}$ is as follows [49].

Assume a matrix, $\mathbf{A}$, of the form

$$\mathbf{A} \equiv \mathbf{U}\boldsymbol{\Sigma}^2\mathbf{U}^T \tag{4}$$

where $\boldsymbol{\Sigma}^2$ is the matrix of eigenvalues of an ellipsoid and $\mathbf{U}$ is the matrix whose columns are the corresponding eigenvectors. Let $\mathbf{C}$ be the Cholesky factorization of $\mathbf{A}$ such that

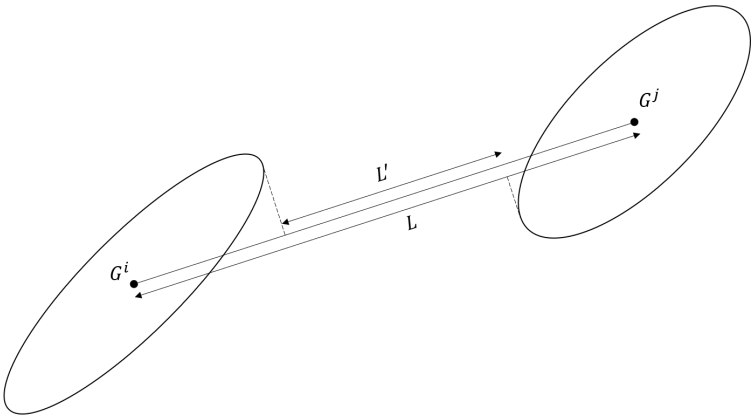


Figure 10: Two methods for generating spacings between neighboring ellipsoids, i and j with centroids G^i and G^j. The first method gives a centroid-to-centroid length $\boldsymbol{L}$ and the second method gives an approximation of the ligament length between the ellipsoids $\boldsymbol{L'}$.

$$\mathbf{A} = \mathbf{C}\mathbf{C}^T \tag{5}$$

Then the projection, $\boldsymbol{w}$, of the ellipsoid from Equation 4 onto a vector, $\boldsymbol{v}$, is obtained through

$$\boldsymbol{w} \equiv \frac{\mathbf{C}^{-1}\boldsymbol{v}}{\boldsymbol{v}^T\boldsymbol{v}} \tag{6}$$

By substituting $\boldsymbol{L}$ for $\boldsymbol{v}$, the projections ($\boldsymbol{w^i}$ and $\boldsymbol{w^j}$) of an ellipsoid pair (with i and j corresponding to the particle pair) are made onto $\boldsymbol{L}$. The magnitude of $\boldsymbol{L'}$ is then calculated by $\|\boldsymbol{L'}\| = \|\boldsymbol{L}\| - \|\boldsymbol{w^i}\| - \|\boldsymbol{w^j}\|$ and is in the same direction as $\boldsymbol{L}$.

Both of these spatial metrics, $\boldsymbol{L}$ and $\boldsymbol{L'}$ are calculated in material coordinates. Since our loading direction varies, so does our $\{x, y, z\}$ loading coordinate system. For each rotation, θ, about the TD-axis, we use a rotation transformation to convert $\boldsymbol{L}$ and $\boldsymbol{L'}$ from the material coordinates into the correct loading coordinates.

2.2.4. Parameters 4 & 5: Averaging Schemes

We now have to convert the information from the ellipsoid and spacing distributions into a periodic unit cell. Our constraints on the ellipsoid generation, i.e., ellipsoids forced to align along $\boldsymbol{e_z}$, enables us to simply average the ellipsoid axes. Averaging the spacing vectors is also straightforward. For both distributions we calculate either an arithmetic average or a volume-weighted average. The volume-weighted average gives more influence to the larger particles and associated spacings in the unit cell construction. To weight the spacing distribution by volume, the combined volumes of the two particles associated with that spacing vector are used as the weight. This provides two extra parameters in our parametric study - the average of the ellipsoids and the average of the spacing vectors. Each of these parameters has two options and we give each a numeric value and add them to the test matrix in Table 1.

2.2.5. Unit Cell Construction

Combining all of the parameters from Table 1 gives a total of 96 unique permutations. A given combination of the parameters results in a unit cell that consists of an average ellipsoid of revolution embedded in a rectangular prism with dimensions $\bar{L}_x$, $\bar{L}_y$, and $\bar{L}_z$. To convert the rectangular prism to a cylinder (which enables axisymmetry), the radial dimension, $\bar{L}_r$, is defined as the average of the $\bar{L}_x$ and $\bar{L}_y$ dimensions, e.g. $\bar{L}_r = \frac{1}{2}\left(\bar{L}_x + \bar{L}_y\right)$. The geometry of the periodic unit cell is shown in Figure 8.

To capture the effect of the loading orientation, $\boldsymbol{e_z}$ is varied between $\theta = 0°$ and $\theta = 90°$ in 15° increments. The conversion from a particle and spacing distribution to a periodic unit cell is repeated for each orientation

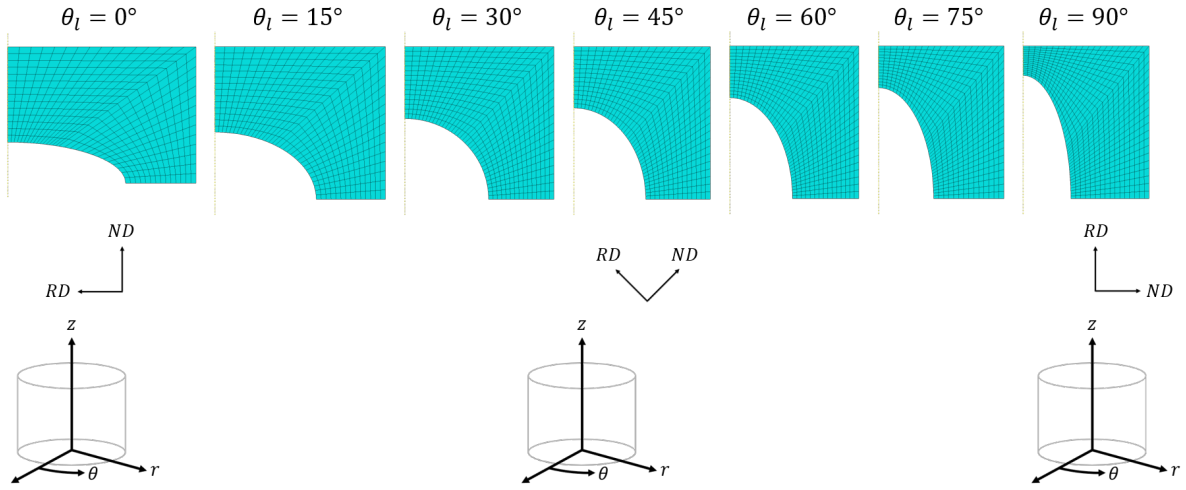


Figure 11: A representative set of unit cells for a given combination of parameters. The unit cell geometry varies for each loading orientation even though the parameter options used to construct it were held constant. The loading vector is aligned along the z-direction for the FE simulations. Axisymmetry and half-symmetry are utilized to reduce the computational load.

for a total of seven unit cells for each test matrix permutation. This results in 96 sets of periodic unit cells, where each set contains seven unit cells - one for each orientation. The various options shown in Table 1 are given alphanumeric identifiers, where the two-letter acronym defines the parameter and the number corresponds to the value of that parameter. Each set of unit cells is identified by a set of five numbers that correspond to the parameters' numeric value for that set. The I.D. number is translated as follows. For I.D. $ijklm$, i denotes the value for the number of particles included in the analysis (NP), j denotes the value for the ellipsoid construction method (EC), k denotes the value for the spacing construction method (SC), and l and m denote the values for the ellipsoid averaging method (EA) and the spacing averaging method (SA), respectively.

A representative set of unit cell geometries are shown in Figure 11. Note that the geometry is different for each unit cell even though the same parameter set generates all seven unit cells. Half-symmetry and axisymmetry are both appropriate assumptions for our boundary conditions and reduce the computational load. We generate the models using Abaqus 2018 and mesh the unit cells with linear axisymmetric stress quadrilateral elements (CAX4 elements). A linear elastic-plastic model is used for the material with an isotropically hardening J_2 yield surface. The Young's modulus and Poisson's Ratio are $E = 45$ GPa and $\nu = 0.35$ respectively. The power-hardening law is the same as that used by Lloyd et al. in Equation 1 with the following values: $\sigma_{0,s} = 40$ MPa, $\sigma_s^{\infty} = 180$ MPa, and $h_{1,s} = 2500$ MPa. The particle is represented as a void, which essentially assumes that void nucleation due to either particle cracking or interfacial debonding has already occurred at the beginning of loading. We model the particle as a void for three reasons. First, we do not have data for the material properties of the particles. Second, for most structural metals and most loading conditions, second-phase particle fracture and/or particle-matrix decohesion occurs early in the deformation process prior to the accumulation of damage. Often, void growth is the limiting process requiring the greater stress. For this reason, there is relatively little difference between modeling the finite stiffness and strength of second-phase particles in comparison to the zero stiffness and strength case, i.e. a pore. Finally, as previously stated, we want to provide a direct link to a porous microstructure that we could fit Gurson-type parameters to. If we assume a spheroidal void, then the Gologanu-Leblond-Devaux (GLD) model correlates directly with our proposed microstructure.

Periodic boundary conditions are applied for all unit cells at $r = L_r$, such that the outer edge remains perpendicular [21, 29]. The top face of the model is displaced in the z-direction to a distance twice the initial height (e.g., $\varepsilon_{eng} = 2.0$). The boundary conditions are applied by using a primary node connected to a set of secondary nodes on the boundary using tie constraints as shown in Figure 12. Appropriate symmetry

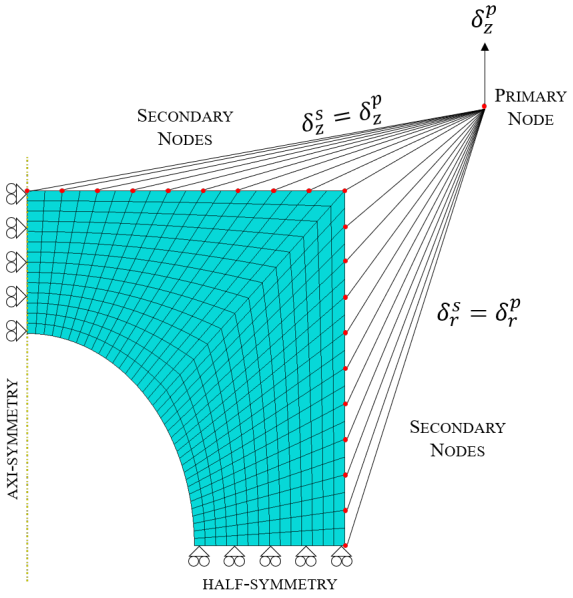


Figure 12: Boundary conditions for the FE simulations. Periodic boundary conditions are applied on the lateral surface, while a displacement is applied to the top edge in the z-direction. The boundary conditions are applied to a primary node and imposed on the unit cell using a tie constraint. The D.O.F. in the z-direction for the nodes at $z = \bar{L}_z$ are constrained to the primary node and the D.O.F. in the r-direction for the nodes at $r = \bar{L}_r$ are constrained to the primary node. Roller boundary conditions are applied to the line of axisymmetry and the half-symmetry plane.

boundary conditions are used on the half-symmetry plane such that the displacement in the z-direction is zero with no constraints on the displacement in the r-direction. This loading imposes a stress triaxiality of $T = 1/3$ and the simulations are conducted using Abaqus/Standard which uses the implicit formulation. Since the imposed boundary conditions cause large displacements/strains, we include the effects of nonlinear geometry, i.e., the finite-strain formulation.

The resulting force-displacement data is extracted from the output database files for the primary node. We then calculate the engineering stress and strain and plot the stress-strain data. Since the FE simulations are rate-independent, we correct our data to match the DNS (which were high strain-rate simulations) by multiplying our resulting stress values by a rate correction factor of $(\dot{\bar{\varepsilon}}_s/\dot{\bar{\varepsilon}}_0)^m$ where $\dot{\bar{\varepsilon}}_s = 10^4$, $\dot{\bar{\varepsilon}}_0 = 1$, and $m = 0.02$.

The resultant stress-strain responses are compared to the DNS from Figure 3d. To quantitatively measure the performance of each parametric response, we choose three metrics for comparison: i) the ultimate tensile strength, σ_{UTS}, ii) the failure strain, ε_f, and iii) the order in which the failure strains occur. A given parameter set should ideally minimize the error of all three metrics. We easily calculate the σ_{UTS} error by comparing the maximum observed engineering stress from the unit cells to the maximum observed stress in the DNS. The other two metrics are less straightforward to measure and we therefore define criteria to provide quantitative comparisons. We define the failure strain (ε_f) for our unit cell FE models as follows. Our matrix material consists of a linear elastic-plastic response with a true stress σ_{true} that asymptotes to $\sigma_{0,s} + \sigma_s^\infty = 220$ MPa (using Equation 1 with the specified values). Assuming plastic incompressibility, the slope of the true stress-strain curve will never be < 0. We define a failure strain as the strain at which the slope of the macroscopic true stress of the unit cell begins to decrease (thereby breaking the plastic incompressibility assumption). Since we are measuring our response in terms of engineering stress and strain, we use Equation 7 to calculate an upper-bound criteria in terms of engineering stress and strain.

$$\frac{\partial \sigma_{eng}}{\partial \varepsilon_{eng}} < \min\left(-\frac{\sigma_{true}}{(1+\varepsilon_{eng})^2}\right) = -220 \text{ MPa} \tag{7}$$

We define −220 MPa as our limit for the slope by setting $\varepsilon_{eng} = 0$ and setting σ_{true} to its maximum theoretical value of 220 MPa. Since σ_{true} reaches its maximum at $\varepsilon_{eng} > 0$, the engineering stress-strain slope will be ≥ -220 MPa before significant strain localization and geometric softening. Defining the failure strain, ε_f, as the point where $\partial\sigma_{eng}/\partial\varepsilon_{eng} < -220$ MPa provides a quantitative, objective measure of performance between the unit cells and the DNS.

Determining how well the unit cells reproduce the order of failure strains is a more nebulous problem. In the absence of a clear solution, we propose a 23-point rating system where a high numeric value corresponds to little or no agreement between the unit cell and the DNS and a low numeric value corresponds to good agreement. Visual inspection of Figure 3d along with the recorded failure strains in Table A.2 show that the 0°-30° orientations fail within ±2% engineering strain of each other. The 45° and 60° orientations then fail sequentially, and finally the 75° and 90° orientations fail within 1% engineering strain of each other. Therefore, we develop a rating system and express it as the following function:

$$\varepsilon_f^{Order} = 23 - H\left(\min\left(\varepsilon_f\left(\theta\right)\right) - \varepsilon_f\left(\theta_1\right) + 0.02\right) - \sum_{n=2}^{7}\sum_{m=1}^{n-1} H\left(\varepsilon_f\left(\theta_n\right) - \varepsilon_f\left(\theta_m\right) + tol\right) \tag{8}$$

where $H\left(x\right)$ indicates the Heaviside step function such that it returns a value of 1 for $x \geq 0$ and 0 for $x < 0$. $\varepsilon_f\left(\theta\right)$ is the failure strain for each orientation defined by the angle θ. The θ values are stored in a vector and sorted in the order of the DNS ε_f magnitudes, i.e., $\theta = [15°, 0°, 30°, 45°, 60°, 75°, 90°]$. A tolerance, tol, of ±2% engineering strain is applied to each inequality in the equation as defined below

$$tol = \begin{cases} +0.02 & \text{if } \varepsilon_f^{dns}\left(\theta_n\right) \geq \varepsilon_f^{dns}\left(\theta_m\right) \\ -0.02 & \text{if } \varepsilon_f^{dns}\left(\theta_n\right) < \varepsilon_f^{dns}\left(\theta_m\right) \end{cases} \tag{9}$$

where ε_f^{dns} are the DNS failure strains. The Heaviside functions total the number of points that are subtracted off the starting "error" of 23. We then take the final count of remaining points as an error metric with a lowest possible value of 1. This results in an objective, quantitative metric that measures how well each set of unit cells reproduces the ε_f order. The absolute ε_f error and absolute σ_{UTS} error are both recorded for all seven unit cells belonging to a single set and then averaged across the set to provide a single ε_f error and single σ_{UTS} error for each set of parameters.

The plastic behavior of Mg alloys is anisotropic, as previously noted, and therefore our methodology (which seeks to isolate the second-phase particle anisotropy by ignoring other anisotropy contributors) is not suited to capture real-world AZ31B behavior. However, after validating our approach against the isotropic DNS (Figure 3d) we attempt to validate our methodology against experiments (Figure 3a). We accomplish this by roughly incorporating plastic anisotropy into the matrix phase of our unit cells by fitting isotropic hardening curves to the data from Figure 3b. Each hardening curve is then imported into the unit cell with the corresponding load angle for a subset of our unit cells. This subset of unit cells is chosen based on which ones best approximate the isotropic DNS solution. Ideally, one would implement either crystal plasticity or Hill's plasticity to capture the plastic anisotropy but this is outside the scope of the current work. These unit cells are compared to the experimental data from Figure 3a instead of the isotropic DNS data.

3. Results

3.1. Viability

Each parameter set of the test matrix gives a unique set of four dimensions that fully define a unit cell for each loading orientation. Some parameter sets, however, yield ellipsoidal voids that are larger than the cylindrical domain (either in the z-direction, r-direction, or both), which results in a non-viable unit cell. We only include in this study the parameter sets that produce seven viable unit cells, i.e., seven unit cells with voids fully enclosed within a cylindrical domain. If one or more of the unit cells generated by a parameter set did not meet that criterion, then the entire set was classified as "non-viable" and excluded from our analysis. For our specific microstructure, approximately 40% of the combinations are not able to construct

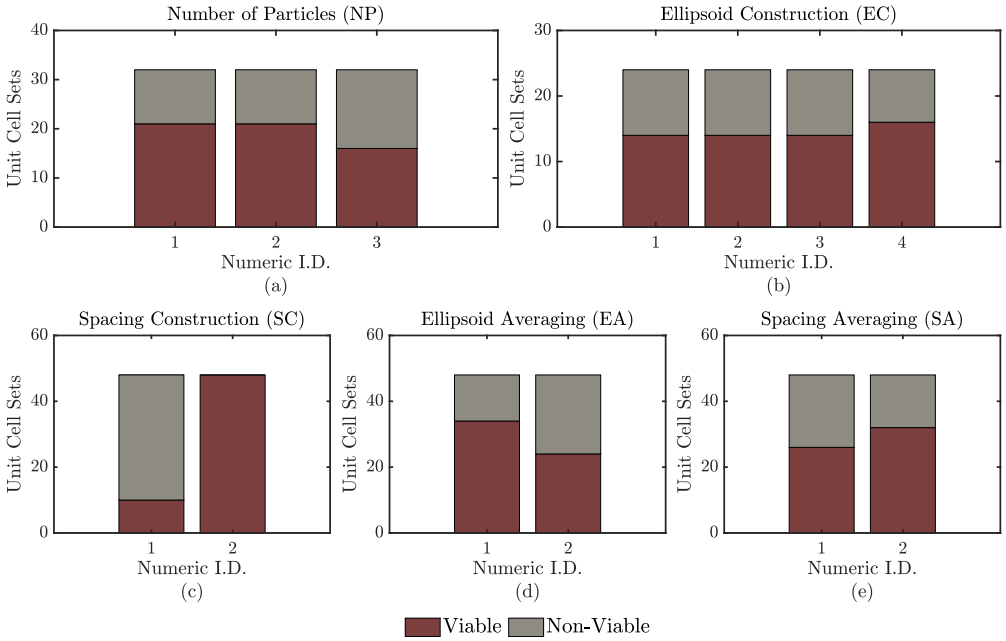


Figure 13: Effect of each parameter on the viability of the unit cell sets.

a unit cell of the desired geometry, i.e. a spheroidal void enclosed in a cylinder. We observed some trends in the effect of the individual parameters on the viability and these are shown in Figure 13.

Figure 13a shows that, when choosing the number of particles to include in the analysis, including all the particles or excluding the single-voxel particles does not affect the viability. Only including the largest particles, however, results in slightly fewer viable sets. The method of ellipsoid construction does not meaningfully affect the viability, as all four methods show similar ratios of viable to non-viable unit cells in Figure 13b. The choice of spacing construction, shown in Figure 13c, demonstrates a marked influence on the unit cell viability. When only accounting for the centroid-to-centroid nearest-neighbor distance, the ellipsoid is often not contained in the cylindrical domain. Conversely, approximating the ligament length results in 100% viability. This is due to the method of creating the unit cell geometry. The first spacing method gives the dimensions $\bar{L}_z$ and $\bar{L}_r$ which correspond to the length and diameter of the cylinder, respectively. The second spacing method yields $\bar{L}'_z$ and $\bar{L}'_r$ which are added to the ellipsoid dimensions $\bar{a}_z$ and $\bar{a}_r$, such that $\bar{L}_z = \bar{L}'_z + \bar{a}_z$ and $\bar{L}_r = \bar{L}'_r + \bar{a}_r$. This additive method of constructing the unit cell ensures that $\bar{a}_z$ and $\bar{a}_r$ can never be greater than $\bar{L}_z$ and $\bar{L}_r$ respectively. Finally, Figs. 13d and 13e show a slight dependence on the averaging method chosen, with a volume-weighted ellipsoid average resulting in slightly less viable sets and a volume-weighted spacing average resulting in slightly more viable sets.

3.2. Comparison to Numerical Data

We are left with 58 viable unit cell sets for our analysis (out of a possible 96) after removing the methods that yielded non-viable unit cells. We calculate ε_f for the DNS in Figure 3d using the definition developed in Equation 7. Many of the unit cells fail to reproduce failure strains for deformations up to $\varepsilon_{eng} = 2.0$. Additionally, of the 58 viable unit cell sets, only 31 sets generate 5+ failure strains. The remaining 17 sets either capture none of the failure strains or between 1 and 4 failure strains. We observe an interesting phenomenon wherein a parameter set tends to yield either a high number of failure strains (i.e., 6-7) or a low number (i.e., 0-1). Rarely does a set of unit cells have comparable numbers of failure strains and non-failure. This observation is illustrated in Figure 14 which shows the number of failure strains recorded for all viable unit cell sets. A complete tabulation of ε_f for all unit cells is included in the appendix along with a rating scale that categorizes the performance of each set.

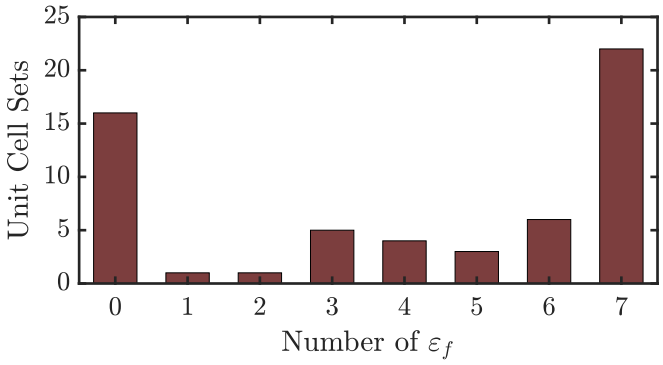
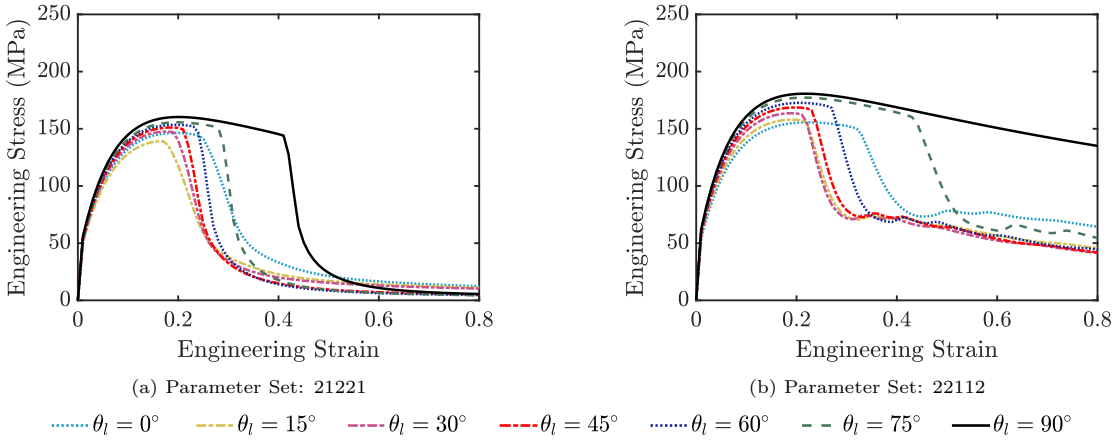


Figure 14: Histogram classifying each method by the number of ε_f that results from the parameter combinations.



(a) Parameter Set: 21221

(b) Parameter Set: 22112

Figure 15: Stress-strain responses for unit cell sets that minimized all three error metrics. The numeric identifiers are shown below each chart and correspond to the combination of parameter options. The σ_{UTS}, failure strain magnitude, and the order of the failure strains are all well-captured, with the exception of the 0° case.

Two parametric sets emerge as "best performers" when comparing our results to the DNS according to the three error metrics (σ_{UTS}, ε_f magnitude, and ε_f order). The stress-strain responses for these two sets are given in Figure 15. Not only is the order of the failure strains generally reproduced (with the exception of the 0° orientation which exhibits unusual behavior, most likely due to its extreme oblate geometry), but the magnitudes of the failure strains are largely on the same order of magnitude of the DNS (again with the exception of the 0° case and, in Figure 15b, the 90° case). Many of the cases that exhibited no failure contain smaller voids that have a negligible impact on the macroscopic response of the unit cell. Others contain an extremely high void volume fraction and exhibit failure far before the peak stress of ∼220 MPa is reached.

The average values for all three error metrics are shown in Figure 16. Parameter sets that minimize all three errors result in green-colored data points in the bottom-left quadrant of Figure 16a. Some of the unit cell sets match the σ_{UTS} fairly well, but fail to capture the failure trends. Alternatively, some schemes capture the failure trends fairly well but have very high σ_{UTS} errors. Additionally, some of the unit cell sets fail to capture any of the pertinent behavior. Comparing the ε_f error with the ε_f-order error shows that the ε_f-order error generally correlates with low ε_f errors. Additionally, high errors in the σ_{UTS} tend to occur only at low ε_f errors as evidenced by the color legend in Figure 16a. Figure 16b shows this trend in more detail. The error measures displayed in these charts are absolute error measures, i.e., the σ_{UTS} error is given in MPa and the ε_f error is unitless. One should note that increasing σ_{UTS} error actually means that

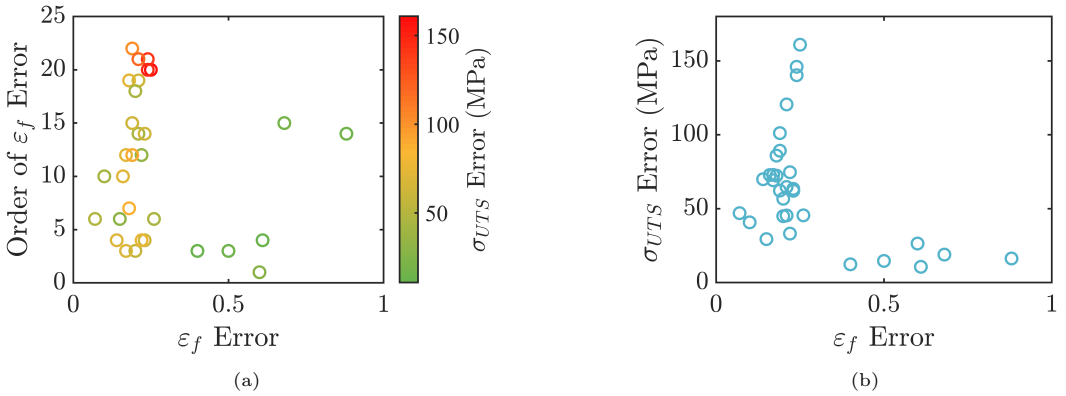


Figure 16: Comparison of the error between unit cell results and DNS. (a) ε_f-order error plotted against the mean ε_f error with the mean σ_{UTS} error plotted as a colormap. (b) mean absolute σ_{UTS} error plotted against mean absolute ε_f error.

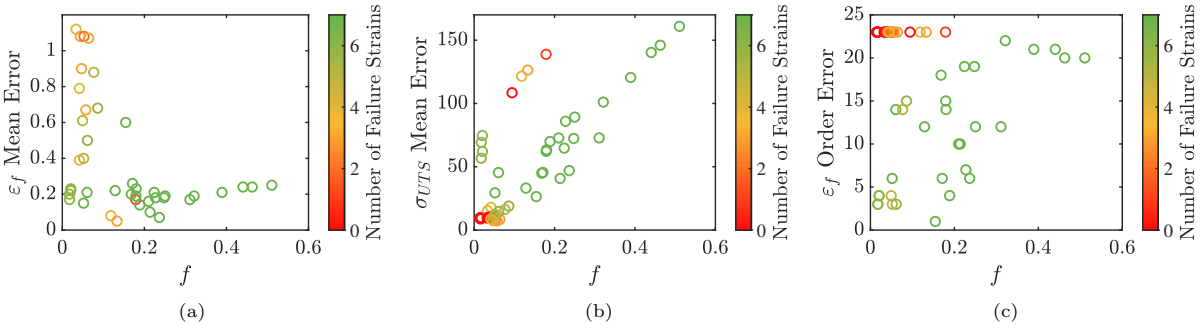


Figure 17: Effect of porosity, f, on (a) ε_f, (b) σ_{UTS}, and (c) ε_f-order errors. The number of ε_f for each set of unit cells is shown with a colormap.

we are increasingly under-predicting the σ_{UTS} of the DNS. Introducing a geometrical void into the domain ensures that the structural response is always softer than the continuum response.

3.3. Effect of Porosity

In the following sections we parse through various parameters and features of our unit cells to elucidate the effect of each one on the overall error of our simulations. We first focus on the void volume fraction or porosity, f, which has a significant effect on the behavior of the unit cells. Each error metric is plotted against porosity in Figure 17. The color bar corresponds to the number of failure strains reproduced in each set. As seen in Figure 17a, higher porosity generally corresponds to lower ε_f errors while low porosities often correspond with high ε_f errors. When looking at the correlation between porosity and σ_{UTS} error in Figure 17b, the σ_{UTS} error generally increases with increasing porosity in a near-linear relationship. There does not seem to be any clear trend between f and the ε_f order error in Figure 17c, but there is some correlation between increasing porosity and higher ε_f order errors. One should not that Figs. 17b and 17c contain data points from all 58 simulation sets, including the sets with zero failure strains. Figure 17a only includes the sets with at least one ε_f as the error of a set with no failure strains is undefined. These observations indicate that i) lower porosity correlates strongly with higher ε_f error and lower σ_{UTS} error (an intuitive conclusion) and ii) the porosity does not seem to be the determining factor on the ε_f-order response.

3.4. Effect of Aspect Ratios

The unit cells contain four distinct aspect ratios: $\bar{L}_r : \bar{L}_z$, $\bar{a}_r : \bar{a}_z$, $\bar{a}_z : \bar{L}_z$, and $\bar{a}_r : \bar{L}_r$. We do not expect any trends to be observed between the error and the first three aspect ratios, and this is borne out

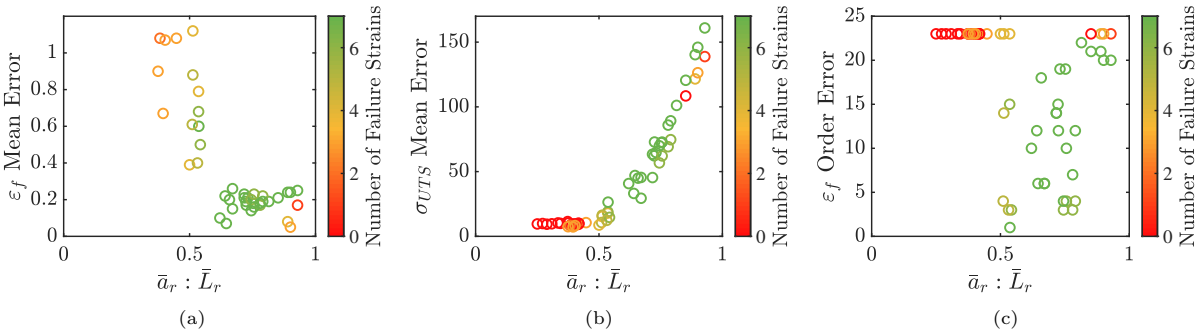


Figure 18: Effect of radial aspect ratio, $\bar{a}_r : \bar{L}_r$, on (a) ε_f, (b) σ_{UTS}, and (c) ε_f-order errors. The number of ε_f for each set of unit cells is shown with a colormap.

in the data. We do observe trends, however, between the error and the $\bar{a}_r : \bar{L}_r$ aspect ratio which are shown in Figure 18. First, no average aspect ratios < 0.2 occur in our data and aspect ratios lower than 0.5 tend to perform poorly by all metrics. An extremely strong, monotonic correlation is observed between the σ_{UTS} error and the average $\bar{a}_r : \bar{L}_r$ aspect ratio. Additionally, the ε_f error significantly decreases for aspect ratios above 0.5. These trends show a clear correlation between the unit cell dimension ratios in the radial direction and the average performance of a unit cell set.

3.5. Effect of the Number of Particles (NP)

Here, we investigate trends in the data when sorted by the number of particles (NP) parameter. Figure 19a shows the error metrics with various symbols representing the parameter value. Note that this is the same graph from Figure 16a with the scatter points sorted into categories. Figure 19b shows the ε_f mean error plotted against f, again with various symbols representing the parameter value. The data plotted here is the same as Figure 17a.

Figure 19a shows that only including the largest particles seems to place a tighter bound on all three error metrics than either of the more inclusive particle methods. The data shows no meaningful variation between including or excluding the single-voxel particles. Additionally, the schemes that only include the largest particles corresponds to a lower f, as none of the porosities for this choice are higher than 0.3. This may be due less to the number of particles included, and more a facet of the spacing measure. Essentially, when only the largest particles are included, the centroid-to-centroid spacing distance does not yield viable unit cells. Therefore, the second spacing method must be used which depresses the void size relative to the cylinder size. Further work is needed to determine if these observations are artifacts of the AZ31B morphology or if they hold true across various microstructures.

3.6. Effect of Ellipsoid Construction Method (EC)

Here, we represent the same data in the previous section, but sort it instead by the value of ellipsoid construction parameter (EC). The modified data is plotted in Figure 20 (again this is the same data as plotted in Figures 19 and 17b). We observe several interesting trends when viewing the data through this paradigm. First, there are only four data points for the first ellipsoid construction method (which prioritizes length along the loading direction, $\boldsymbol{e_z}$). Therefore, most of the parameter sets that use that method generated less than two failure strains (since 2+ failure strains is the requirement for representation in this chart). Secondly, an interesting trend is observed between the second method (which prioritizes the projected area and volume) and the ε_f mean error. As seen in Figure 20a, parameter sets using this method result in an average ε_f error of ~ 0.2. However, this is only true for data points generated with this method that produce 5+ failure strains. Figure 20b, which contains cases that only have 1-2 failure strains shows a more scattered distribution of ε_f for the same method. The second method also results in a seemingly non-linear correlation between the σ_{UTS} error and f that deviates significantly from the linear relationship exhibited by the three other choices (as shown in Figure 20c). Thirdly, the method that prioritizes both length and

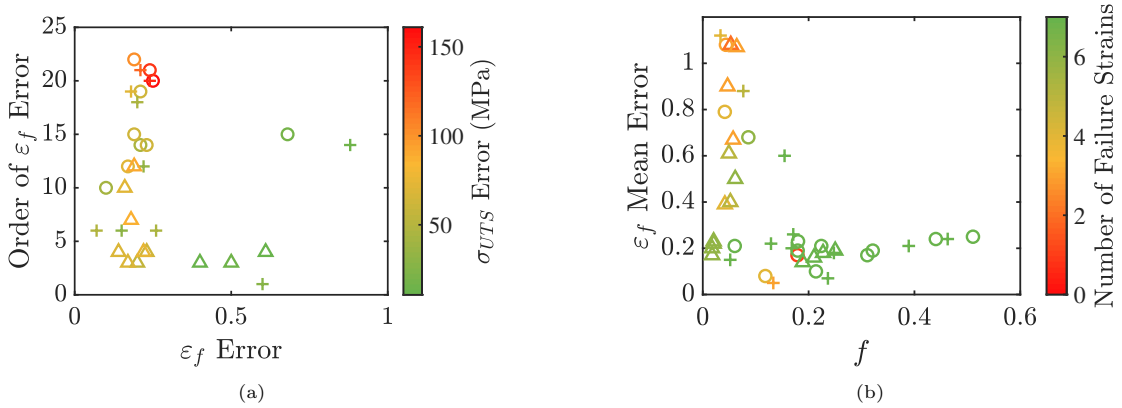


Figure 19: ◯ = **All Particles Included,** + = **Single-Voxel Particle Excluded,** △ = **10 Largest Particles Included.** (a) ε_f-order error plotted against the mean ε_f error with the mean σ_{UTS} error plotted as a colormap. (b) ε_f mean error vs. porosity. The data is plotted in both graphs with symbols corresponding to the number of particles included in the analysis.

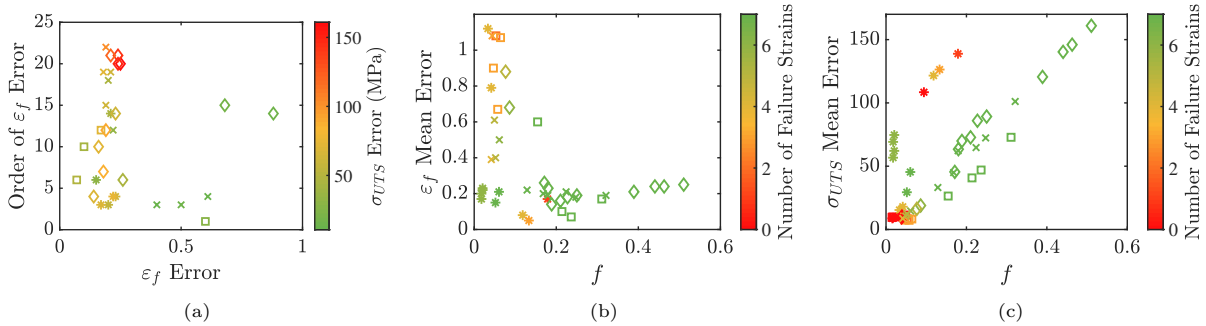


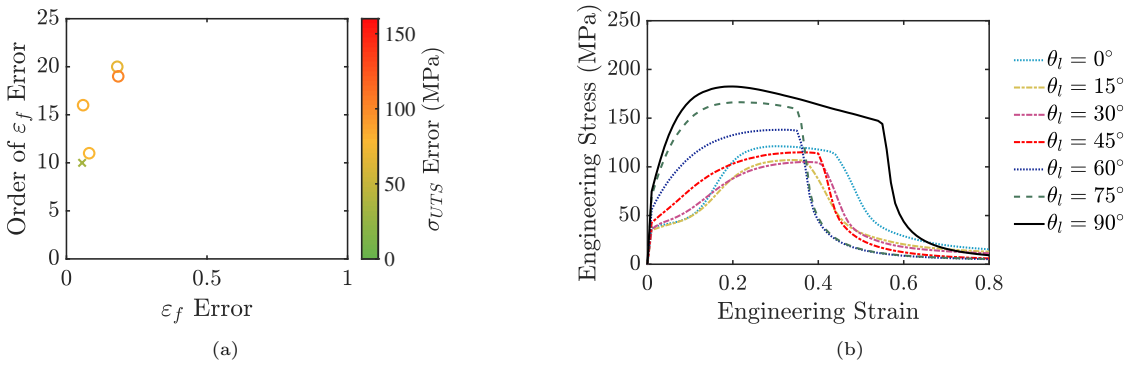
Figure 20: □ = **Length,** ∗ = **Area,** ♢ = **Length and Area,** × = **Length, Area, and Volume.** (a) ε_f-order error plotted against the mean ε_f error with the mean σ_{UTS} error plotted as a colormap, (b) ε_f mean error vs. porosity, and (c) σ_{UTS} error vs. porosity. The data is plotted in all graphs with symbols corresponding to the ellipsoid construction method used.

area gives the widest range of errors and porosities, as shown in Figure 20a and 20b. This is due to the non-volume-preserving nature of the ellipsoid construction scheme that can result in ellipsoids that have much higher or much smaller volumes than the corresponding particles. Finally, the fourth method, which scales method three to match the volume, follows the same trends as method three, but shrinks the upper bounds of the σ_{UTS} and ε_f error. The ε_f-order error, however, remains fairly similar between methods three and four.

Both the second and third methods (EC 2 and EC 3) minimize all three errors and generally result in unit cell sets with failure strains for this study. Either of these are therefore reasonable choices for this microstructure. EC 1 and EC 4 are both ill-suited as EC 1 generally does not result in unit cells with failure strains and EC 4 does not minimize the error well.

3.7. Comparison to Experimental Data

After quantifying the performance of the various unit cell sets in the preceding sections, we identify four "best performers" and implement the orientation-dependent hardening curves from the crystal plasticity model. The parameter sets chosen for this section are given in the appendix, along with their failure strains. The results are compared to the experimental results from Figure 3a instead of the isotropic DNS from Figure 3d. The errors are plotted in Figure 21a and the stress strain results from the best performer in Figure 21b. We find that using a different material model for each simulation leads to fairly low ε_f errors but does not accurately capture the σ_{UTS} or ε_f-order error. When compared against the error exhibited



(a) (b)

Figure 21: Comparison of the error between unit cell results with the orientation-associated hardening curves and experiments. (a) ε_f-order error plotted against the mean ε_f error with the mean σ_{UTS} error plotted as a colormap. ◯ = Unit cells from this study, × = Crystal plasticity DNS from [22]. (b) Stress-strain results for the unit cell results that minimize all three error metrics (left-bottom-most point from the plot in (a)). The parameter I.D. for the plot in (b) is 11222.

by the anisotropic crystal plasticity DNS (from Figure 3c), our unit cells have errors on the same order of magnitude. Therefore, it is not surprising that our rudimentary "anisotropic" plasticity coupled with our simple geometry was not able to out-perform the expensive high-fidelity simulations. This inability to capture the experimentally observed behavior is most likely due to the simplicity of the model that does not account for dominant mechanisms (such as twinning behavior) that drive the deformation of Mg alloys. Effectively modeling such a complex material remains an open research question, as even coupling a crystal-plasticity model with a realistic particle morphology as performed in [22] struggles to capture the behavior.

4. Summary & Conclusions

Homogenization of a microstructure inherently involves a loss of fidelity in the macroscale simulations. Fitting a damage model to RVE studies of realistic microstructural data remains a cost- and time-intensive process. Here, we attempt to create a method that precludes multiple time-consuming steps by constructing an idealized *equivalent* microstructure directly from the statistical data of the realistic microstructural. Due to the parametric nature of the study, our 672 unit cell sets exhibit a wide range of responses (including non-viability). We discovered that several of our 96 proposed methodologies were able to generate *equivalent* idealized unit cells that roughly capture the effective response predicted by the full direct numerical simulation (DNS) study. Further refinement is needed to tune our process for even better agreement. This could include fine-tuning the parameters to get better agreement in uniaxial tension and comparing unit cell responses to DNS under more general loading paths.

We also investigated the effects of several parameters on our methodology and summarize our findings as follows:

- Number of particles included (NP): Only including the largest particles in our method shrinks the error bounds between our unit cells and the isotropic DNS.
- Method of ellipsoid construction (EC): The ellipsoid construction methods exhibit unclear trends that warrant further investigation. A forthcoming paper studies these methods in greater detail.
- Methods of spacing construction (SC), ellipsoid averaging (EA), and spacing averaging (SA): All other parameters do not exhibit strong correlations or trends in regard to the error between the unit cells and the isotropic DNS.

In a final validation step, we implemented anisotropic hardening curves and compared the response of select unit cell sets to AZ31B experimental data. As expected, we found that we were largely unable to

capture the real-world behavior of AZ31B. However, our unit cell method performed comparably to a high-fidelity DNS with a crystal-plasticity model. This indicates a highly-complex interplay of phenomena in AZ31B that warrant future efforts to resolve, both in our methodology and in high-fidelity DNS modeling.

Further work should be conducted on best practices for fitting simple shapes to individual particles, e.g., which aspect(s) of the particle are most important to capture? Using morphological erosion and dilation operations to separate particles that are weakly connected might also yield improved results. Additionally, this methodology should be extended to other microstructures. The conclusions of this effort are limited as the particular microstructural effects are not easily distinguishable from methodology effects. Applying this procedure to other data and/or alloys can elucidate such distinctions. Once our methodology is sufficiently validated and refined, the unit cell geometry can be directly implemented into an ellipsoidal-based Gurson-type model, e.g., the GLD model, and an assessment of the performance can take place. Despite these limitations, this study is an important first-step in creating a standardized process that can convert any complex microstructure to an *equivalent* periodic unit cell and expand the state-of-the-art in multi-scale modeling of heterogeneous materials.

5. Acknowledgments

We would like to thank Angela Olinger, Jeff Lloyd, Dan Magagnosc, and Beñat Gurrutxaga-Lerma for helpful discussions and mentorship. Caleb Foster was supported by the Department of Defense (DoD) through the National Defense Science & Engineering Graduate (NDSEG) Fellowship Program. This material is also partially based upon work supported by the National Science Foundation under Grant No. 2239678 and by the Army Research Laboratory under Cooperative Agreement Nos. W911NF-12-2-0022, W911NF-22-2-0105, and W911NF-22-2-0106. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory, the National Science Foundation, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein. Portions of this research were conducted with high performance research computing resources provided by Texas A&M University (https://hprc.tamu.edu).

Appendix A. Failure Strain Data

The ε_f for each unit cell is recorded in Table A.2 for the main suite of unit cell calculations along with the failure strains from the DNS. Table A.3 presents the failure strains for the experimental data along with the four sets of unit cells that incorporated anisotropic plasticity. The performance of each parameter set in both tables is generally classified as poor, fair, or good performance (denoted by red squares, yellow diamonds, and green circles, respectively). Good performance is defined as a score of < 10 in the ε_f-order score from Equation 8. Fair performance is defined as reproducing most or all of the failure strains, but failing to capture the order of the failure strains by orientation (score of 10+ from Equation 8). Poor performance is defined as reproducing less than half the failure strains. The degree to which the ε_f magnitudes and σ_{UTS} match between the unit cell calculations and DNS/Experiments is not captured in this rating scheme. One should note that the experiments and DNS do not predict the same failure strains. Therefore, the unit cell ε_f is compared to the DNS in Table A.2 and to the experiments in Table A.3.

Simulation I.D.	0°	15°	30°	45°	60°	75°	90°	Rating
DNS	0.19	0.18	0.20	0.29	0.38	0.45	0.46	
11112								■
11211								■
11212								■
11221	0.14	0.11	0.11	0.12	0.13	0.14	0.18	♦
11222	0.34	0.24	0.25	0.25	0.24	0.32	0.58	♦
12112	0.57	0.26	0.14	0.09	0.06	0.14	0.61	♦
12211			1.24	1.00	0.99	1.24		■
12212								■
12221							0.29	■
12222				0.51	0.41	0.39	0.44	■
13112	0.62	0.21	0.11	0.08	0.06	0.12	0.68	♦
13211		1.44	0.93	0.68	0.66	0.84	1.49	♦
13212								■
13221	0.07	0.07	0.06	0.05	0.04	0.04	0.05	♦
13222	0.10	0.09	0.07	0.06	0.05	0.06	0.07	♦
14111	0.43	0.14	0.10	0.08	0.09	0.12	0.36	♦
14112				1.62	0.98	1.75		■
14211								■
14212								■
14221	0.19	0.15	0.11	0.09	0.09	0.09	0.11	♦
14222	0.51	0.34	0.22	0.16	0.13	0.14	0.21	♦
21112								■
21211								■
21212								■
21221	0.24	0.17	0.19	0.21	0.23	0.28	0.41	●
21222	0.56	0.42	0.59	0.77	0.95	1.20	1.85	●
22112	0.32	0.22	0.21	0.23	0.27	0.43	1.15	●
22211			1.50	1.29	1.35	1.66		■
22212								■
22221					0.44	0.42	0.41	■
22222								■
23112	0.29	0.17	0.16	0.18	0.22	0.45	1.85	●
23211		1.65	1.16	0.94	0.95	1.18		♦
23212								■
23221	0.09	0.08	0.07	0.06	0.05	0.06	0.07	♦
23222	0.12	0.11	0.09	0.08	0.08	0.08	0.09	♦
24111	0.70	0.25	0.20	0.18	0.20	0.29	0.94	♦
24112								■
24211								■
24212								■
24221	0.35	0.24	0.17	0.14	0.13	0.13	0.17	♦
24222	0.69	0.53	0.45	0.38	0.34	0.35	0.51	♦
31211		0.82	1.73					■
31212	1.67	0.44	1.17					■
31221	1.97	0.58	1.22					■
31222	1.43	0.32	0.83					■
32211	0.66	0.38	0.37	0.42	0.51	0.74		●
32212	0.43	0.23	0.26	0.38	0.58	0.99		●
32221	0.64	0.38	0.36	0.42	0.50	0.70		●
32222	0.42	0.23	0.26	0.37	0.55	0.90		●
33211	0.18	0.11	0.10	0.10	0.11	0.14	0.31	♦
33212	0.14	0.08	0.08	0.11	0.14	0.20	0.51	●
33221	0.13	0.10	0.09	0.09	0.10	0.12	0.20	♦
33222	0.11	0.07	0.07	0.09	0.11	0.15	0.30	●
34211	0.98	0.47	0.60	0.90	1.32			●
34212	0.72	0.29	0.43	1.00				■
34221	0.80	0.36	0.45	0.65	0.92	1.53		●
34222	0.58	0.23	0.32	0.71	1.42			●

Table A.2: Failure strains of plastically isotropic AZ31B for all parameter sets studied. The numeric identifier corresponds with the parameter values from Table 1 ordered as described in Section 2.2.5. Blank cells represent no failure strain detected. The unit cell performance are sorted into three categories: good, fair, and poor, represented by a ●,♦, and ■ respectively. The criteria are as follows: ● = generally correct order of failure strains by orientation, ♦ = incorrect order, but most or all failure strains reproduced, ■ = missing failure strains, order not reproduced.

Simulation I.D.	0°	15°	30°	45°	60°	75°	90°	Rating
Experiments	0.38	0.38	0.33	0.41	0.30	0.25	0.26	
11222	0.43	0.37	0.40	0.40	0.35	0.35	0.54	♦
21221	0.34	0.31	0.35	0.35	0.35	0.31	0.37	♦
22112	0.39	0.32	0.32	0.31	0.32	0.44	1.14	♦
32222	0.48	0.32	0.34	0.42	0.57	0.90		♦
Crystal Plasticity DNS	0.39	0.35	0.41	0.47	0.40	0.34	0.25	♦

Table A.3: Failure strains of plastically anisotropic AZ31B for the four "best performer" unit cell construction methodologies. The numeric identifier corresponds with the parameter values from Table 1 ordered as described in Section 2.2.5. Blank cells represent no failure strain detected. The unit cell performance are sorted into three categories: good, fair, and poor, represented by a ●,♦, and ■ respectively. The criteria are as follows: ● = generally correct order of failure strains by orientation, ♦ = incorrect order, but most or all failure strains reproduced, ■ = missing failure strains, order not reproduced.

References

[1] F. Hannard, A. Simar, E. Maire, T. Pardoen, Quantitative assessment of the impact of second phase particle arrangement on damage and fracture anisotropy, Acta materialia 148 (2018) 456–466.

[2] T. S. Cao, Models for ductile damage and fracture prediction in cold bulk metal forming processes: a review, International Journal of Material Forming 10 (2) (2017) 139–171.

[3] A. L. Gurson, Continuum Theory of Ductile Rupture by Void Nucleation and Growth: Part I—Yield Criteria and Flow Rules for Porous Ductile Media, Journal of Engineering Materials and Technology 99 (1) (1977) 2–15. arXiv:https://asmedigitalcollection.asme.org/materialstechnology/article-pdf/99/1/2/5788715/2_1.pdf.

[4] K. Terada, M. Hori, T. Kyoya, N. Kikuchi, Simulation of the multi-scale convergence in computational homogenization approaches, International Journal of Solids and Structures 37 (16) (2000) 2285–2311. doi:10.1016/S0020-7683(98)00341-2. URL `https://www.sciencedirect.com/science/article/pii/S0020768398003412`

[5] A. Olinger, C. Foster, J. Wilkerson, Homogenized Modeling of Anisotropic Impact Damage in Rolled AZ31B with Aligned Second-Phase Particles, Journal of Dynamic Behavior of Materials 6 (4) (2020) 445–458. doi:10.1007/s40870-020-00267-3.

[6] D. L. McDowell, A perspective on trends in multiscale plasticity, International Journal of Plasticity 26 (9) (2010) 1280–1309. doi:10.1016/j.ijplas.2010.02.008.

[7] E. v. d. Giessen, P. A. Schultz, N. Bertin, V. V. Bulatov, W. Cai, G. Csányi, S. M. Foiles, M. G. D. Geers, C. González, M. Hütter, W. K. Kim, D. M. Kochmann, J. LLorca, A. E. Mattsson, J. Rottler, A. Shluger, R. B. Sills, I. Steinbach, A. Strachan, E. B. Tadmor, Roadmap on multiscale materials modeling, Modelling and Simulation in Materials Science and Engineering 28 (4) (2020) 043001, publisher: IOP Publishing. doi:10.1088/1361-651X/ab7150.

[8] M. F. Horstemeyer, Multiscale modeling: A review, in: J. Leszczynski, M. K. Shukla (Eds.), Practical Aspects of Computational Chemistry: Methods, Concepts and Applications, Springer Netherlands, Dordrecht, 2010, pp. 87–135.

[9] M. G. D. Geers, V. G. Kouznetsova, K. Matouš, J. Yvonnet, Homogenization Methods and Multiscale Modeling: Nonlinear Problems, in: Encyclopedia of Computational Mechanics Second Edition, John Wiley & Sons, Ltd, 2017, pp. 1–34. doi:10.1002/9781119176817.ecm2107.

[10] Y. L. Bai, H. Y. Wang, M. F. Xia, F. J. Ke, Statistical Mesomechanics of Solid, Linking Coupled Multiple Space and Time Scales, Applied Mechanics Reviews 58 (6) (2005) 372–388. doi:10.1115/1.2048654.

[11] K. Matouš, M. G. D. Geers, V. G. Kouznetsova, A. Gillman, A review of predictive nonlinear theories for multiscale modeling of heterogeneous materials, Journal of Computational Physics 330 (2017) 192–220. doi:10.1016/j.jcp.2016.10.070.

[12] M. F. Horstemeyer, Integrated computational materials engineering (ICME) for metals: using multiscale modeling to invigorate engineering design with science, WILEY [u.a.], Hoboken, NJ, 2012.

[13] A. Pineau, A. A. Benzerga, T. Pardoen, Failure of metals I: Brittle and ductile fracture, Acta Materialia 107 (2016) 424–483. doi:10.1016/j.actamat.2015.12.034.

[14] S. Ghosh, J. Bai, D. Paquet, Homogenization-based continuum plasticity-damage model for ductile failure of materials containing heterogeneities, Journal of the Mechanics and Physics of Solids 57 (7) (2009) 1017–1044. doi:10.1016/j.jmps.2009.04.002.

[15] J. Caulkins, C. Fauver, S. Adibi, J. Wilkerson, Effect of Grain Boundary Misorientation on Spall Strength in Ta via Shock-Free Simulations with Relatively Few Atoms, Metals 12 (10) (2022) 1586. doi:10.3390/met12101586.

[16] T. I. Zohdi, Homogenization methods and multiscale modeling, Encyclopedia of computational mechanics. Solids and structures 2 (2004) 357–383.

[17] M. Jebahi, F. Dau, J.-L. Charles, I. Iordanoff, Multiscale Modeling of Complex Dynamic Problems: An Overview and Recent Developments, Archives of Computational Methods in Engineering 23 (1) (2016) 101–138. doi:10.1007/s11831-014-9136-6.

[18] M. G. D. Geers, J. Yvonnet, Multiscale modeling of microstructure–property relations, MRS Bulletin 41 (8) (2016) 610–616, publisher: Cambridge University Press. doi:10.1557/mrs.2016.165.

[19] Z. Chen, C. Butcher, Micromechanics Modelling of Ductile Fracture, Vol. 195 of Solid Mechanics and Its Applications, Springer Netherlands, Dordrecht, 2013. doi:10.1007/978-94-007-6098-1.

[20] P. Suquet (Ed.), Continuum micromechanics, no. 377 in Courses and lectures / International Centre for Mechanical Sciences, Springer, Wien, 1997.

[21] A. A. Benzerga, J.-B. Leblond, Ductile fracture by void growth to coalescence, in: H. Aref, E. van der Giessen (Eds.), Advances in Applied Mechanics, Vol. 44 of Advances in Applied Mechanics, Elsevier, 2010, pp. 169–305.
[22] J. Lloyd, A. Matejunas, R. Becker, T. Walter, M. Priddy, J. Kimberley, Dynamic tensile failure of rolled magnesium: Simulations and experiments quantifying the role of texture and second-phase particles, International Journal of Plasticity 114 (2019) 174–195.
[23] R. Sarvesha, W. Alam, A. Gokhale, T. Guruprasad, S. Bhagavath, S. Karagadde, J. Jain, S. Singh, Quantitative assessment of second phase particles characteristics and its role on the deformation response of a mg-8al-0.5 zn alloy, Materials Science and Engineering: A 759 (2019) 368–379.
[24] A. Sarmah, M. K. Jain, S. Asqardoust, P. Mohammadpour, Multiscale modeling of particle-induced damage in aa7075 aluminum sheet at large plastic strains, International Journal of Plasticity 169 (2023) 103741.
[25] D. Magagnosc, P. Jannotti, J. Ligda, J. Lloyd, Pre-twinned magnesium for improved ballistic performance, Mechanics of Materials 161 (2021) 104005. doi:https://doi.org/10.1016/j.mechmat.2021.104005.
URL https://www.sciencedirect.com/science/article/pii/S0167663621002350
[26] P. J. Noell, R. B. Sills, A. A. Benzerga, B. L. Boyce, Void nucleation during ductile rupture of metals: A review, Progress in Materials Science (2023) 101085.
[27] B. Kondori, A. A. Benzerga, On the notch ductility of a magnesium-rare earth alloy, Materials Science and Engineering: A 647 (2015) 74–83. doi:10.1016/j.msea.2015.08.077.
URL https://www.sciencedirect.com/science/article/pii/S0921509315303257
[28] A. A. Benzerga, J.-B. Leblond, A. Needleman, V. Tvergaard, Ductile failure modeling, International Journal of Fracture 201 (1) (2016) 29–80. doi:10.1007/s10704-016-0142-6.
[29] J. Besson, Continuum models of ductile fracture: A review, International Journal of Damage Mechanics 19 (1) (2010) 3–52. arXiv:https://doi.org/10.1177/1056789509103482.
[30] T. S. Srivatsan, S. Vasudevan, M. Petraroli, The tensile deformation and fracture behavior of a magnesium alloy, Journal of alloys and compounds 461 (1-2) (2008) 154–159.
[31] Z. Shang, T. Li, S. Yang, J. Yan, H. Guo, Three-dimensional characterization of typical inclusions in steel by x-ray micro-ct, Journal of Materials Research and Technology 9 (3) (2020) 3686–3698. doi:https://doi.org/10.1016/j.jmrt.2020.01.106.
[32] S. Bargmann, B. Klusemann, J. Markmann, J. E. Schnabel, K. Schneider, C. Soyarslan, J. Wilmers, Generation of 3d representative volume elements for heterogeneous materials: A review, Progress in Materials Science 96 (2018) 322–384.
[33] S. Ghosh, K. Lee, S. Moorthy, Multiple scale analysis of heterogeneous elastic structures using homogenization theory and voronoi cell finite element method, International Journal of Solids and Structures 32 (1) (1995) 27–62. doi:https://doi.org/10.1016/0020-7683(94)00097-G.
URL https://www.sciencedirect.com/science/article/pii/002076839400097G
[34] S. Ghosh, K. Lee, S. Moorthy, Two scale analysis of heterogeneous elastic-plastic materials with asymptotic homogenization and voronoi cell finite element model, Computer methods in applied mechanics and engineering 132 (1-2) (1996) 63–116.
[35] S. Ghosh, K. Lee, P. Raghavan, A multi-level computational model for multi-scale damage analysis in composite and porous materials, International journal of solids and structures 38 (14) (2001) 2335–2385.
[36] A. Abedini, C. Butcher, Z. Chen, Numerical simulation of the influence of particle clustering on tensile behavior of particle-reinforced composites, Computational materials science 73 (2013) 15–23.
[37] M. Pinz, G. Weber, S. Ghosh, Generating 3d virtual microstructures and statistically equivalent rves for subgranular gamma-gamma'microstructures of nickel-based superalloys, Computational Materials Science 167 (2019) 198–214.
[38] A. Sarmah, M. K. Jain, Multi-scale modeling of decohesion characteristics of second phase particles from the matrix in uniaxial tension in a high strength aluminum alloy, Engineering Fracture Mechanics 301 (2024) 110013.
[39] D. Wilkinson, E. Maire, R. Fougeres, A model for damage in a clustered particulate composite, Materials Science and Engineering: A 262 (1) (1999) 264–270. doi:https://doi.org/10.1016/S0921-5093(98)01011-9.
URL https://www.sciencedirect.com/science/article/pii/S0921509398010119
[40] J. Gammage, D. Wilkinson, Y. Brechet, D. Embury, A model for damage coalescence in heterogeneous multi-phase materials, Acta Materialia 52 (18) (2004) 5255–5263. doi:https://doi.org/10.1016/j.actamat.2004.07.009.
URL https://www.sciencedirect.com/science/article/pii/S1359645404004185
[41] C. Tekoğlu, T. Pardoen, A micromechanics based damage model for composite materials, International Journal of Plasticity 26 (4) (2010) 549–569. doi:https://doi.org/10.1016/j.ijplas.2009.09.002.
URL https://www.sciencedirect.com/science/article/pii/S0749641909001107
[42] K. Ismail, A. Perlade, P. J. Jacques, T. Pardoen, L. Brassart, Impact of second phase morphology and orientation on the plastic behavior of dual-phase steels, International Journal of Plasticity 118 (2019) 130–146.
[43] D. Xie, C. Liu, Y. Wang, Q. Ouyang, Y. Gao, W. Xu, Effect of coarse second-phase particles on mechanical properties of large-scale 2219 al alloy rings, Engineering Fracture Mechanics 301 (2024) 110030.
[44] D. Fabrègue, T. Pardoen, A constitutive model for elastoplastic solids containing primary and secondary voids, Journal of the Mechanics and Physics of Solids 56 (3) (2008) 719–741.
[45] B. Kondori, A. A. Benzerga, Effect of stress triaxiality on the flow and fracture of mg alloy az31, Metallurgical and Materials Transactions A 45 (2014) 3292–3307.
[46] T. M. Inc., Bwconncomp (2020b).
URL https://www.mathworks.com
[47] M. Horstemeyer, S. Ramaswamy, M. Negrete, Using a micromechanical finite element parametric study to motivate a phenomenological macroscale model for void/crack nucleation in aluminum with a hard second phase, Mechanics of Materials 35 (7) (2003) 675–687.
[48] Y. Liu, X. Zheng, S. Osovski, A. Srivastava, On the micromechanism of inclusion driven ductile fracture and its implications

on fracture toughness, Journal of the Mechanics and Physics of Solids 130 (2019) 21–34.
[49] S. B. Pope, Algorithms for Ellipsoids, Cornell University, Ithaca, NY, 2008.