ARTICLE IN PRESS

Available online at www.sciencedirect.com
ScienceDirect

Comput. Methods Appl. Mech. Engrg. xxx (xxxx) xxx

Computer methods in applied mechanics and engineering

www.elsevier.com/locate/cma

Interpreting stochastic agent-based models of cell death

Emma Lejeune[a,b,*], Christian Linder[b,c]

[a] Department of Mechanical Engineering, Boston University, United States of America
[b] Department of Civil and Environmental Engineering, Stanford University, United States of America
[c] Department of Mechanical Engineering, Stanford University, United States of America

Available online xxxx

Abstract

Linking individual cell behavior to changes on the cell population and ultimately tissue scale is a key step towards connecting in vitro laboratory experiments to clinically relevant organ-scale behavior. For example, understanding how an individual tumor cell's response to drug treatments propagates to the macroscale could help guide clinical decision making when tumors are located in high stakes regions. Based on how dying cells interact, macroscale tumor response can be dominated by tumor shrinkage, tumor damage, change in tumor porosity, or some combination of these phenomena. Here we focus on methods of modeling collective cell behavior with the overall goal of understanding how stochastic cell behavior will ultimately manifest on the cell population scale. To do this, we define and implement an agent-based model problem, introduce novel methods for interpreting simulation results, implement a recently developed technique for conducting a variance-based sensitivity analysis, and investigate key model parameters. Operating within an in silico framework allows us to compare our proposed methods with and without reproducible stochastic variations. This is critical because methods that will be functional for interpreting experimental data must also be able to function when aspects of stochastic variation are not reproducible. To demonstrate the utility of our proposed methods, we identify important simulation parameters for our agent-based model, show that two-dimensional and three-dimensional results are not directly equivalent, and highlight the macroscale implications of cell shrinkage. Looking forward, we anticipate that the methods and results presented here will contribute to significant future research in modeling stochastic cell behavior in the multiscale setting.

Keywords: Biomechanics; Multiscale modeling; Agent-based modeling

1. Introduction

The ability to robustly model cell death has medical applications ranging from understanding anomalous organ development [1] to neurodegeneration [2]. A particularly compelling example where an enhanced understanding of cell death on the organ scale would help guide clinical decision making is that of a tumor located in a high stakes region [3–5], illustrated in Fig. 1. In this example, medical interventions such as radiation therapy and chemotherapy largely function by inducing cell death [6,7]. Even when the response of individual cells to treatment is well understood, it is not necessarily straightforward how the cellular-scale process of cell death should be represented in a continuum mechanics based whole-organ simulation. On the macroscale, many different fundamental modeling

* Corresponding author at: Department of Mechanical Engineering, Boston University, United States of America.
E-mail address: elejeune@bu.edu (E. Lejeune).

https://doi.org/10.1016/j.cma.2019.112700
0045-7825/

Please cite this article as: E. Lejeune and C. Linder, Interpreting stochastic agent-based models of cell death, Computer Methods in Applied Mechanics and Engineering (2019) 112700, https://doi.org/10.1016/j.cma.2019.112700.

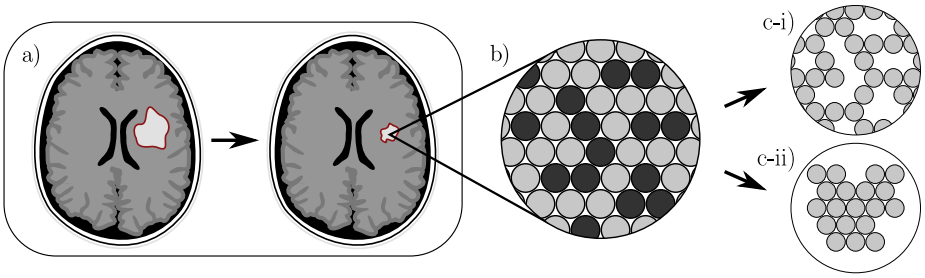

Fig. 1. (a) Understanding tumor response to intervention is critical when tumors are located in high stakes regions; (b) Through agent-based modeling, we are able to directly model the process of cell death; (c-i) Under some circumstance, cell death leads to gaps in the tissue; (c-ii) Under other circumstances, cell death leads to tissue shrinkage.

choices are potentially justifiable. For example, macroscale tissue models may implement cell death as a change in a growth, a damage term, species concentration, or some combination of these mechanisms [8–10]. Fundamentally, macroscale behavior will arise from the way an individual cell behaves and interacts with its neighbors [11]. The most applicable macroscale modeling choice will vary based on the specific type of cell death [12] and the system in question [13–15].

In this work, we take a bottom-up approach to show how small differences in cell death on the cellular scale (i.e. changes in individual behavior) are linked to different interpretations of cell death on larger length scales. Specifically, we show that in some cases cell death leads to gaps between cells, illustrated in Fig. 1c-i, while in other cases it lead to tissue shrinkage, illustrated in Fig. 1c-ii. To accomplish this, we introduce multiple highly extensible computational tools for interpreting agent-based models where each cell acts as an autonomous agent following a set of prescribed rules [16]. While there are numerous examples of agent-based models in the literature [17,18], there are limited examples of agent-based models designed specifically to inform macroscale simulation [19,20]. In particular, there are few tools for interpreting the results of agent-based models in a way that is meaningful for typical examples of tissue scale modeling informed by cellular processes [21,22]. This work is intended to address this gap by introducing transferable and highly extensible procedures for model interpretation. Given recent advances in high-throughput experiments [23], three-dimensional cell culture [24], live cell imaging [25], and cell tracking [26], it is plausible that the computational model used here could ultimately be replaced by in vitro experimentation. In the computational setting, we can run simulations where we control and intentionally reproduce the stochastic components (reproducible stochastic behavior) or where we intentionally do not reproduce the stochastic behavior (irreproducible stochastic behavior). To address this possibility of implementing these methods on experimental data, which would contain irreproducible stochastic behavior, we investigate the performance of our methods on agent-based model data that contains both reproducible and irreproducible stochastic behavior.

The remainder of the paper is organized as follows. In Section 2, we introduce our methods for implementing and analyzing cell death in the agent-based model setting. Specifically, in Section 2.1 we formulate our model, in Section 2.2 we describe novel methods for quantifying model results, in Section 2.3 we describe different approaches for implementing a meta-model to capture the results of a stochastic simulation, and in Section 2.4 we describe the procedure for conducting a sensitivity analysis on a stochastic model. Then, in Section 3, we present the results of applying these methods to interpreting the results from our agent-based model system. Finally, we conclude in Section 4.

2. Methods

In this section, we outline our proposed methods towards the ultimate goal of incorporating agent-based models into a multi-scale computational framework. First, in Section 2.1, we define our problem of interest and define the mechanical model that we use to approach it [27]. Then, in Section 2.2, we introduce strategies for extracting meaningful quantities of interest from the agent-based model. In Section 2.3, we introduce two methods for creating a meta-model of a stochastic simulation and in Section 2.4 we describe the procedure for conducting a sensitivity analysis on a stochastic model. We note that the methods presented in Sections 2.2, 2.3, and 2.4 are all data source agnostic.

Please cite this article as: E. Lejeune and C. Linder, Interpreting stochastic agent-based models of cell death, Computer Methods in Applied Mechanics and Engineering (2019) 112700, https://doi.org/10.1016/j.cma.2019.112700.

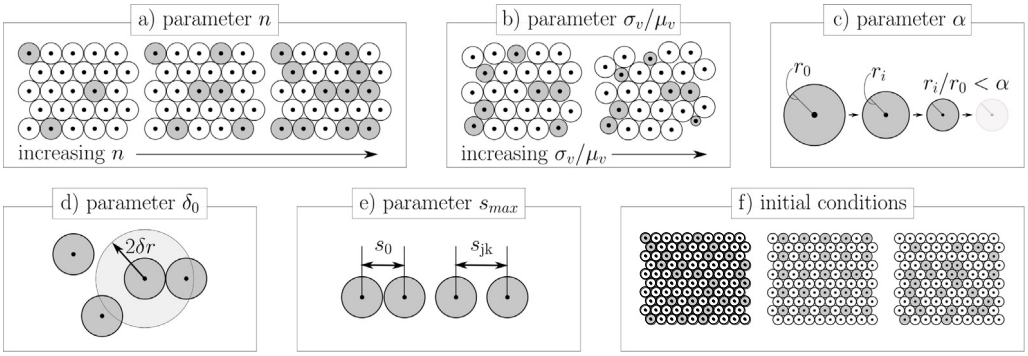

Fig. 2. We investigate the sensitivity of this model to five parameters illustrated here: (a) fraction of dying cells n; (b) initial cell size distribution σ_v/μ_v; (c) cell shrinkage α; (d) cell horizon δ_0 where $\delta = (1+g)\delta_0$; (e) maximum allowable stretch $s_{\max}$. We also run simulations with stochastic initial conditions: (f) different cells in the population are initialized to undergo the cell death procedure.

2.1. Formulating an in silico cell population model

In this paper, we aim to understand how the cell death mechanism implemented in an agent-based model will manifest on the population and tissue scales. To do this, we start by defining an initial problem with an off-lattice agent-based model [28,29] where each cell is represented by a single node [30]. Our initial problem is illustrated in Fig. 2. In the two-dimensional case, we model a $m \times m$ grid of nodes where the cell centers all begin equally spaced on a two-dimensional hexagonal lattice. We run "small" (model 1), "medium" (model 2), and "large" (model 3) simulations to ensure that the methods and conclusions drawn from this work are not restricted to a single model size. In the three-dimensional case, we model a $m \times m \times m$ grid of nodes where the cell centers all begin equally spaced on a three-dimensional face-centered cubic lattice. In both the two-dimensional and three-dimensional cases, the initial distance between nodes is equal to twice their baseline radius r_0. We note that although the cells are initially positioned on a grid, they have full freedom to move in space once the simulation begins.

Given this initial grid of cells, we define a problem where some fraction n of these cells will undergo cell death. We define n as $n = \#\,\text{dying cells} \,/\, \#\,\text{total cells}$. The meaning of parameter n is illustrated in Fig. 2a. For a given instance of random behavior ω, we randomly select a combination of $[nm^2]$ (2D) or $[nm^3]$ (3D) nodes to undergo cell death. Variations in this "initial condition", where a specific set of cells are selected, are illustrated in Fig. 2f. In the numerical setting, we make the selection of cells to remove reproducible by setting an initial seed unique to ω and then generating a randomly ordered list of all of the nodes present at the start of the simulation. Then, the first $[nm^2]$ (2D) or $[nm^3]$ (3D) nodes on the list are chosen to undergo cell death. With this strategy, the random component of node selection is preserved even as parameter n is varied. In this section, we take care to clearly identify the random behavior that we introduce in the model because reproducible stochastic behavior is required for the methods introduced in Section 2.3.

We then model the process of cell death as follows. Cell death is treated simply as a period of radial cell shrinkage followed by node removal when the node has decreased in size below a defined threshold. To model radial shrinkage, we define incremental shrinkage parameter υ where the mean μ_υ and standard deviation σ_υ are model parameters. Then, υ is a random variable computed as

$$\upsilon_j \sim \mathcal{N}(\mu_\upsilon, \sigma_\upsilon) \qquad \text{if } \upsilon_j < \upsilon_{\min} \;\rightarrow\; \upsilon_j := \upsilon_{\min} \tag{1}$$

where $\upsilon_{\min}$ prevents the case of $\upsilon \approx 0$. In the numerical setting, we set a pre-defined random seed and then random values of υ_j are generated and assigned to each node j. This procedure means that for a given instance of random behavior ω, values of υ are reproduced independently of the simulation input parameters. If $\sigma_\upsilon \approx 0$, the nodes will shrink and subsequently be removed simultaneously. With υ_j defined, we then incrementally apply radial shrinkage at each load step by defining radial growth as

$$g_j^{t+1} = g_j^t - \upsilon_j \tag{2}$$

Please cite this article as: E. Lejeune and C. Linder, Interpreting stochastic agent-based models of cell death, Computer Methods in Applied Mechanics and Engineering (2019) 112700, https://doi.org/10.1016/j.cma.2019.112700.

where $g_j^0 = 0$. Preferred node radius is then defined as a function of baseline radius r_0 and growth g_j^t written as

$$r_j = (1 + g_j^t)\, r_0 \,. \tag{3}$$

To model node removal, we define constant parameter α. The condition for node removal is

$$\frac{r_j}{r_0} = \frac{(1 + g_j)\, r_0}{r_0} < \alpha \,. \tag{4}$$

Parameter α is bounded as $0 < \alpha < 1$. However, we typically set a more realistic lower bound with $\alpha_{\min} > 0$. The influence of parameters σ_υ and α is illustrated in Fig. 2b and c respectively.

So far, we have defined initial conditions and the algorithm for cell shrinkage and removal. Now, we briefly discuss aspects of the agent-based model itself. To maintain mechanical equilibrium in our model, we use the peridynamic equation of motion, defined in the discrete setting at static equilibrium as

$$0 = \sum_{\mathrm{k} \in \mathcal{H}'_{\mathrm{j}}} \boldsymbol{f}_{\mathrm{jk}}(\boldsymbol{y}_{\mathrm{j}}, \boldsymbol{y}_{\mathrm{k}})\, \Delta V_{\mathrm{k}} - \sum_{\mathrm{k} \in \mathcal{H}_{\mathrm{j}}} \boldsymbol{f}_{\mathrm{kj}}(\boldsymbol{y}_{\mathrm{j}}, \boldsymbol{y}_{\mathrm{k}})\, \Delta V_{\mathrm{k}} \tag{5}$$

where $\boldsymbol{f}_{\mathrm{jk}}$ is the force density acting at node j due to node k, $\boldsymbol{f}_{\mathrm{kj}}$ is the force density acting at node k due to node j, ΔV_k is node associated volume computed from r_0 and g_k, and $\boldsymbol{y}$ is nodal position [27,31,32]. Horizon $\mathcal{H}_{\mathrm{j}}$ and dual-horizon $\mathcal{H}'_{\mathrm{j}}$ are defined as

$$\mathcal{H}_{\mathrm{j}} = \{\mathrm{k} \mid \quad \|\boldsymbol{y}_{\mathrm{j}} - \boldsymbol{y}_{\mathrm{k}}\| < 2\,\delta_j r_0\} \qquad\qquad \mathcal{H}'_{\mathrm{j}} = \{\mathrm{k} \mid \quad \mathrm{j} \in \mathcal{H}_{\mathrm{k}}\} \tag{6}$$

where δ_j controls the size of the horizon. In defining a peridynamic state-based linear-elastic constitutive law, multiple constants enter this equation: elastic modulus E, Poisson's ratio ν, initial radius size r_0, initial horizon size δ_0, and maximum allowable stretch between nodes $s_{\max}$ [33]. We choose two model input parameters δ_0 and $s_{\max}$ to include in the sensitivity analysis step because their appropriate value and contributions to population behavior are not necessarily intuitive. The meaning of parameter δ_0 is illustrated in Fig. 2d. Specifically, cells interact with other cells that are within distance δ of their center where δ_0 enters the equation as

$$\delta = (1 + g_j)\, \delta_0 \,. \tag{7}$$

The meaning of parameter $s_{\max}$ is illustrated in Fig. 2e. Essentially, the stretch s between cells is defined as

$$s_{\mathrm{jk}} = \frac{\|\boldsymbol{y}_{\mathrm{k}} - \boldsymbol{y}_{\mathrm{j}}\| - \|\boldsymbol{\xi}_{\mathrm{jk}}\|}{\|\boldsymbol{\xi}_{\mathrm{jk}}\|} \tag{8}$$

where $\boldsymbol{\xi}_{\mathrm{jk}}$ is the stretch free separation distance between cell j and cell k defined in Appendix A. When stretch s_{jk} exceeds $s_{\max}$ the bond between cells is considered broken and that specific cell–cell interaction will no longer enter the constitutive law. To run a direct simulation, we specify initial geometry and parameters in an input file and call the peridynamics simulation software described in detail in previous publications [27]. For convenience, more information on defining and implementing this model is available in Appendix A. We also note that a sensitivity analysis including additional model parameters would follow directly from the methods presented here.

2.2. Defining population shrinkage and connectivity Quantities of Interest

In the previous section, we defined a computational model where select nodes within a grid of nodes undergo cell death. In this section, we define two Quantities of Interest (QoI) to capture how the cell death mechanism influences population scale behavior. These QoI are both computed through post-processing routines where the only information needed is the change in nodal position at each step of the simulation. These post-processing methods could be applied to any agent-based model or experimental system where cell tracking data is accessible.

The first QoI is position-based change in population volume J. The procedure for computing J follows from our previously published work [34]. First, as illustrated in Fig. 3a, we consider the p nodes that are present at both the start and the end of the simulation. By connecting every node j to every other node k we have $p(p-1)/2$ vectors $\boldsymbol{\lambda}_0$ at the start of the simulation and $\boldsymbol{\lambda}_t$ at each simulation step. Then, we define matrices

$$\begin{aligned} \boldsymbol{\Lambda_0} &= [\boldsymbol{\lambda}_0^1 \;\; \boldsymbol{\lambda}_0^2 \;\; \boldsymbol{\lambda}_0^3 \;\; \ldots \;\; \boldsymbol{\lambda}_0^{p(p-1)/2}] \\ \boldsymbol{\Lambda_t} &= [\boldsymbol{\lambda}_t^1 \;\; \boldsymbol{\lambda}_t^2 \;\; \boldsymbol{\lambda}_t^3 \;\; \ldots \;\; \boldsymbol{\lambda}_t^{p(p-1)/2}] \,. \end{aligned} \tag{9}$$

Please cite this article as: E. Lejeune and C. Linder, Interpreting stochastic agent-based models of cell death, Computer Methods in Applied Mechanics and Engineering (2019) 112700, https://doi.org/10.1016/j.cma.2019.112700.

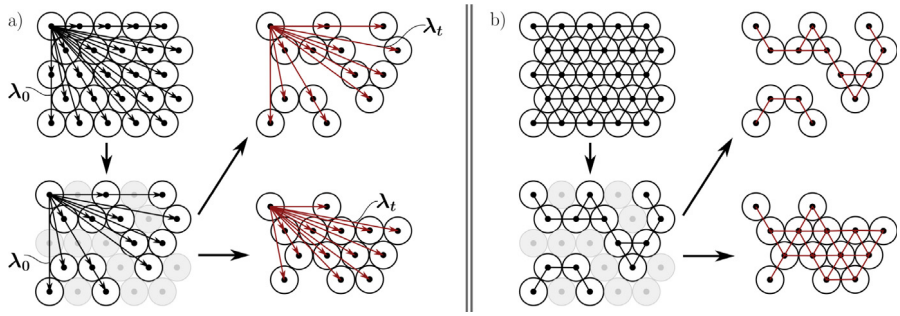

Fig. 3. (a) Schematic of the method used to compute shrinkage J through tracking stretch vectors $\boldsymbol{\lambda}$; (b) Schematic of the method used to compute connectivity C by treating cells as nodes on a graph. The values of shrinkage J and connectivity C will depend on the degree to which the cells reorganize during the simulation. In this schematic, we illustrate cells that do not move (upper) and cells that reorganize substantially (lower).

With matrices $\boldsymbol{\Lambda}_0$ and $\boldsymbol{\Lambda}_t$, we then compute approximate deformation gradient $\boldsymbol{F}$ as

$$\mathbf{F}_t \Lambda_0 = \Lambda_t \qquad \rightarrow \qquad \mathbf{F}_t = \Lambda_t \Lambda_0^T (\Lambda_0 \Lambda_0^T)^{-1} \tag{10}$$

and compute the position-based change in population volume J as

$$J_t = \det \mathbf{F}_t \,. \tag{11}$$

This QoI captures the population shrinkage.

Population shrinkage summarizes one major aspect of simulation results. However, it does not directly capture information about the structural change of the material. On the macroscale, the behavior that J does not capture is the change in material microstructure that depends on nodal connectivity. Therefore, we define a new QoI that is designed to capture additional information about structural change. This second QoI is average population connectivity C. We formulate C as a dimensionless quantity that will monotonically decrease as the cell population becomes on average less connected with more interstitial space. To compute C, we treat our population of cells as a mathematical graph structure G [35]. Each cell becomes a node in graph G. For every cell pair (j, k) where the physical distance between node j and node k is less than or equal to $r_{\mathrm{j}} + r_{\mathrm{k}}$, we add an edge to G between nodes j and k. This is illustrated in Fig. 3b. We are then able to define C as

$$C = \frac{N_G^{1/d}}{m_{SG}} \times \frac{N_{SG}}{N_G} \tag{12}$$

where N_G is the number of nodes in G, N_{SG} is the number of nodes in the largest connected sub-group, $d = 2$ or $d = 3$ is the dimension of the problem, and m_{SG} is the dimensionless mean shortest path in the largest connected sub-group. Once G is constructed, these quantities can be computed easily with existing packages such as the NetworkX Python package [36]. In our results, we report the ratio of connectivity at the end of the simulation to connectivity at the start of the simulation C_f/C_0. In Fig. 4, we show plots of C_f/C_0 with respect to the fraction of dying cells n for simulations with no nodal displacement where $J = 1$. When the nodes are simply removed, C_f/C_0 decreases with respect to n in both the two-dimensional and three-dimensional cases. Clearly, the ratio C_f/C_0 reflects changes that are not captured with QoI J alone.

2.3. Meta-model formulation

One limitation of agent-based models is that they are computationally expensive. This becomes even more prohibitive in multi-scale modeling, where the results of the agent-based simulation could be required at every Gauss point at every time step in a macroscale finite element simulation [37]. One way to avoid this limitation is to construct a meta-model, where input parameters are mapped directly to a specific defined QoI [38,39]. Here we consider the QoIs defined in Section 2.2. To construct a meta-model, we obtain direct simulation data and compute the resulting QoIs for each input parameter set. For a deterministic computational model we would

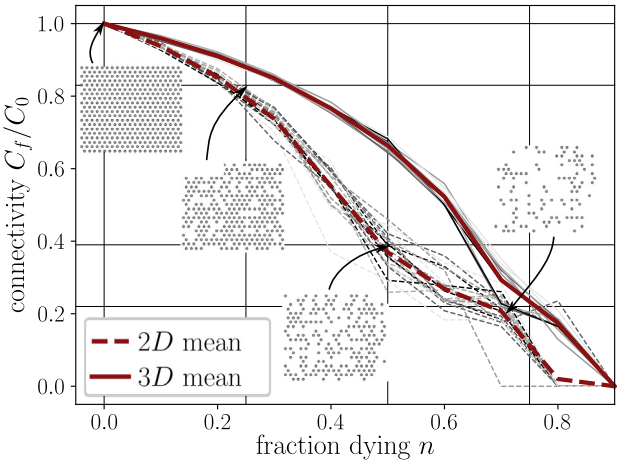

Fig. 4. Plot of connectivity ratio C_f/C_0 for the case where cell centers do not move. This plot shows what happens when nodes are removed and the simulation does not allow for any position updating. Dashed lines represent the two-dimensional case, Solid lines represent the three-dimensional case. The thinner gray lines are individual simulations. Note that C_f/C_0 consistently decreases as the fraction of dying cells n increases.

then train the meta-model on this data following standard procedures [40]. Procedures are also in place for stochastic computational models and sets of experimental data where making stochastic behavior reproducible is not required [41]. However, in this work we are dealing with a stochastic computational model where stochastic behavior is reproducible. Therefore, we are able to investigate two potential frameworks for meta-model construction. First, we define meta-modeling strategy 1 (mm-1) as:

1. run R sets of N simulations
2. create 1 meta-model from $R \times N$ simulations

where R is a random instance (simulations with identical random seeds or multiple random seeds) and N refers to a set of input parameters used to sample the direct simulation. The alternative strategy, (mm-2) is:

1. run R sets of N simulations
2. build R meta-models
3. create 1 model that outputs the average results of R meta-models

where mm-2 takes advantage of the ability to run simulations with reproducible stochastic behavior while mm-1 does not. We briefly note that in the experimental setting, the scenario of reproducible stochastic behavior might also arise in the case where initial conditions are stochastic (i.e. the realization of the microstructure of a tissue sample) but the remainder of the parameters of interest do not cause stochastic behavior and can be measured on the same sample.

Based on preliminary investigation, neural networks [42] provide the best fit to the model data when compared to multivariate adaptive regression splines [43] and Gaussian process regression [44], two popular alternative choices. Though the preliminary investigation is not exhaustive (i.e. neural networks are not unequivocally the best model type for this data), the flexibility of neural networks makes them a broadly attractive choice for capturing potentially complex, highly nonlinear, and not necessarily continuous model behavior [45]. Therefore, the results presented in Sections 3.3 and 3.4 utilize neural networks. We build our neural network using the Keras library [46] running on top of TensorFlow [47]. In Appendix B, we also confirm that neural networks are a valid approach for conducting sensitivity analysis.

2.4. Sensitivity analysis with a stochastic model

With our agent-based model, simulation results will depend on the actions and interactions between different parameters in a manner that is not necessarily straightforward. Therefore, we rely on global sensitivity analysis to understand the relative importance of different model parameters [48–50]. To do this, we define the model as a

Table 1
This table summarizes the parameters selected for simulation. Parameters marked with ⋆ are included in the sensitivity analysis.

Parameter	SA	Value	Source
n	⋆	$0.0 - 0.9$	Range of interest
$\sigma_\upsilon/\mu_\upsilon$	⋆	$0.00001 - 0.5$	Plausible values
α	⋆	$0.3 - 1.0$	Plausible values [54]
δ_0	⋆	$1.01 - 1.15$	Plausible values [27]
$s_{\max}$	⋆	$1.0 - 2.0$	Plausible values [27]
E		1 kPa	Plausible value [55]
ν		0.45	Nearly incompressible material
r_0		5 μm	Approximate cell size [56]

black box f written as

$$Y = f(\boldsymbol{X}, \omega) \tag{13}$$

where Y is a QoI defined in Section 2.2, $\boldsymbol{X}$ contains model input parameters, and ω is an instance of reproducible random behavior [51]. In Section 2.1, we described our strategy for making the random behavior ω in our model reproducible. For a fixed instance of reproducible random behavior ω, first-order Sobol indices S and total Sobol indices T are defined as

$$S_j = \frac{\mathrm{Var}\{\mathbb{E}\{f(\boldsymbol{X}, \omega)|\boldsymbol{X}_j\}\}}{\mathrm{Var}\{f(\boldsymbol{X}, \omega)\}} \qquad T_j = \frac{\mathbb{E}_{\boldsymbol{X}\sim j}\{\mathrm{Var}\{f(\boldsymbol{X}, \omega)|\boldsymbol{X}_{\sim j}\}\}}{\mathrm{Var}\{f(\boldsymbol{X}, \omega)\}}. \tag{14}$$

These quantities, S_j and T_j, are computed numerically through Saltelli sampling [52,53] where we sample P parameter combinations for R random instances [51]. Because this step is computationally limiting due to the high number of simulations required for convergence of the sensitivity analysis, we first construct a meta-model $\hat{f}$ so that N (with $N < P$) rather than P direct model samples are executed. Details of the meta-model implementation were given in Section 2.3. For sensitivity analysis, even run times on the order of minutes can be prohibitive because of the high number of simulations required. For this step, a unique meta-model is constructed for each instance of random behavior. In full, the sensitivity analysis workflow is:

1. run R sets of N direct model simulations
2. build R meta-models
3. run R sets of P meta-model simulations
4. compute the first-order and total Sobol indices from P simulations R times
5. compute the average of the R first-order and total Sobol indices for each input parameter .

Further details of this procedure can be found in recent literature on sensitivity analysis for stochastic models [51]. Table 1 summarizes the parameters included in the sensitivity analysis presented in Section 3.3.

2.4.1. Note on an alternative sensitivity analysis procedure

The ability to run simulations where stochastic behavior is reproducible is crucial to the sensitivity analysis procedure detailed above. In our in silico model, parameters $\boldsymbol{X}$ and stochastic behavior ω can be controlled independently. However, this will not be the case for most experimental systems. With this limitation, the procedure for running a sensitivity analysis is limited to:

1. run $R \times N$ direct model simulations
2. build 1 meta-model with $R \times N$ data points
3. run 1 set of P meta-model simulations
4. compute first-order and total Sobol indices from the P simulations

In an effort to better understand how this strategy [51] may limit model results for systems similar to the one we are investigating, we also run sensitivity analysis with this method for comparison purposes.

3. Results and discussion

The methods presented in Section 2 detail how to implement and interpret our agent-based model. We begin in Section 3.1 with a summary of the simulations that were performed for this study. Then, in Section 3.2 by showing

Table 2
This table summarizes the results presented in Section 3.

① exploratory direct simulation →	② meta-model construction →	③ sensitivity analysis →	④ investigate key parameters with meta-models →	⑤ investigate key parameters with direct simulations
Fig. 5	Fig. 6 Fig. C.1 Fig. D.1	Fig. 7 Fig. 8 Fig. B.1 Fig. C.2	Fig. 9 Fig. C.3	Fig. 10 Fig. 11 Fig. 12

individual model results, where Quantities of Interest are plotted for multiple parameter combinations. Then, in Section 3.3, we show the results of the sensitivity analysis, and in Section 3.4 we show the results of implementing the two meta-modeling procedures described in Section 2.3. Finally, in Section 3.5, we present the key mechanistic insights gained by our model.

3.1. Summary of simulations performed

For clarity, a summary of the workflow that points towards the relevant figures for each step of the analysis is presented in Table 2. We run both two-dimensional (2D) and three-dimensional (3D) simulations. In the 2D case, the cells begin the simulation in a grid with a size of either 4×4 (referred to as model 1, small), 8×8 (model 2, medium), or 20×20 (model 3, large). In the 3D case, the cells begin the simulation in a grid with a size of either $4 \times 4 \times 4$ (model 1, small), $6 \times 6 \times 6$ (model 2, medium), or $8 \times 8 \times 8$ (model 3, large). For each investigation, we directly run each model considered N times and use the data generated by N model runs to construct our meta-models. In Section 2.2, we first determine what N must be such that meta-model error is sufficiently low. We determine that $N = 200$ is sufficient. Because our simulations are stochastic, we run each simulation R times with different initial seeds. We determine that $R = 10$ is sufficient to capture mean behavior, though we also show mean and standard deviation of later results with $R = 50$ because we ran additional simulations to ensure that our conclusions were robust. When comparing different meta-model procedures, we use the same directly sampled data, which corresponds to $N \times R$ direct model runs. When it comes to sampling the meta-model for sensitivity analysis, we run P computationally cheap meta-model simulations up to $P = 10{,}000$ to show convergence. Likewise, we call the meta-model 10,000 times at regularly spaced intervals to construct contour plots for easy visualization of the parameter space. Notably, the ability to construct a meta-model is the key ingredient that makes our analysis computationally feasible.

3.2. Individual model results

In Fig. 5, we plot the change in population shrinkage J and average connectivity C, the two QoIs defined in Section 2.2, over the course of the simulation for multiple model runs. In the upper row, we plot the results of 5 simulations on different grid sizes, 4×4, 8×8, and 20×20, with identical parameter combinations but different instances of random behavior. This row of plots illustrates that for this agent-based model, increasing the grid size leads to a decrease in the influence of the stochastic behavior. This is an important check because we cannot assume that this will occur in all agent-based models. In the lower row, we show a plot of J and a plot of C over the course of the simulation for 15 different parameter combinations with fixed random behavior on a 20×20 grid. Fixed random behavior means that in all simulations initial cell size is perturbed identically and the cells that are selected to undergo cell death are in an identical ranked order in every simulation. In the remainder of this section, we focus on the state of the cell population at the final point in each simulation.

3.3. Critical model parameters identified with sensitivity analysis

In Section 2.4, we described the procedure for conducting a sensitivity analysis on a stochastic agent-based model. We begin by training neural networks on data sets with different parameter combinations but identical random behavior. In Fig. 6, we show what this learning curve looks like for "small" simulations with a 4×4 grid

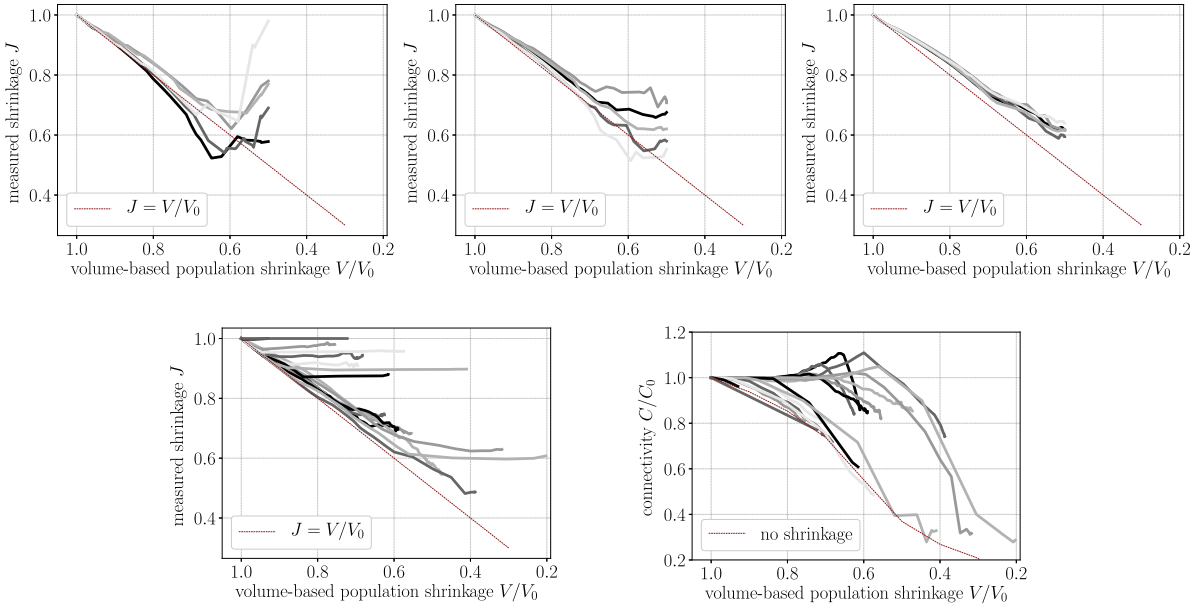

Fig. 5. Upper row: Change in shrinkage J with respect to loss in volume over the course of the simulation for 1 parameter combination with 5 different instances of random behavior. Left: 4×4 starting grid of nodes; Center: 8×8 starting grid; Right: 20×20 starting grid. Lower row: different parameter combinations with fixed random behavior; Left: Change in shrinkage J with respect to loss in volume over the course of the simulation for 15 different parameter combinations with fixed random behavior. The red dashed line shows J directly proportional to volume loss; Right: Change in connectivity C/C_0 with respect to loss in volume over the course of the simulation for 15 different parameter combinations with fixed random behavior. The red dashed line shows C/C_0 when there is no nodal displacement.

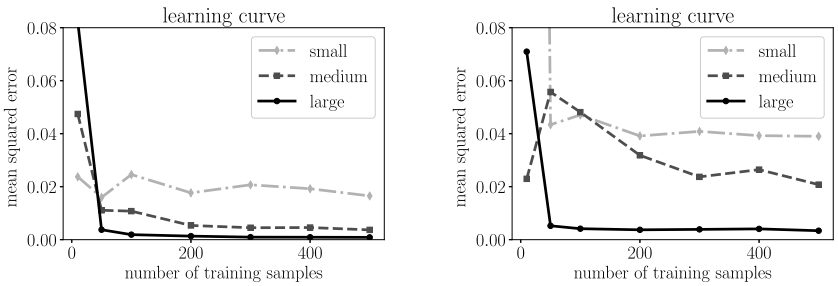

Fig. 6. Left: 5-fold cross-validation error with respect to the number of training samples for predicting J from input parameters with fixed random behavior in two-dimensions; Right: 5-fold cross-validation error with respect to the number of training samples for predicting C from input parameters with fixed random behavior in two-dimensions. The "small" simulations are on a 4×4 grid, the "medium" simulations are on an 8×8 grid, the "large" simulations are on a 20×20 grid. Learning curves for the three-dimensional case are presented in Fig. C.1.

(model 1), "medium" simulations with an 8×8 grid (model 2), and "large" simulations with a 20×20 grid (model 3). Three-dimensional meta-model performance is similar. Based on the learning curves, 200 training samples are pragmatic choice for constructing the meta-models required for sensitivity analysis. In Fig. 7, we show the results of the sensitivity analysis on the "large" model size for the two-dimensional model and the three-dimensional model. In our analysis, we determine that $R = 10$ random behavior instances is sufficient, and demonstrate convergence of the total Sobol indices with respect to P meta-model samples. From the sensitivity analysis, where we show the 95% bootstrapped confidence interval around the mean total Sobol indices, it is clear that parameter n, fraction of cells dying, and parameter α, cell shrinkage, are important while the other parameters investigated have a much smaller influence. This qualitative trend is held true for QoI J and QoI C, all of the model sizes considered, and for both the two-dimensional and three-dimensional cases.

Please cite this article as: E. Lejeune and C. Linder, Interpreting stochastic agent-based models of cell death, Computer Methods in Applied Mechanics and Engineering (2019) 112700, https://doi.org/10.1016/j.cma.2019.112700.

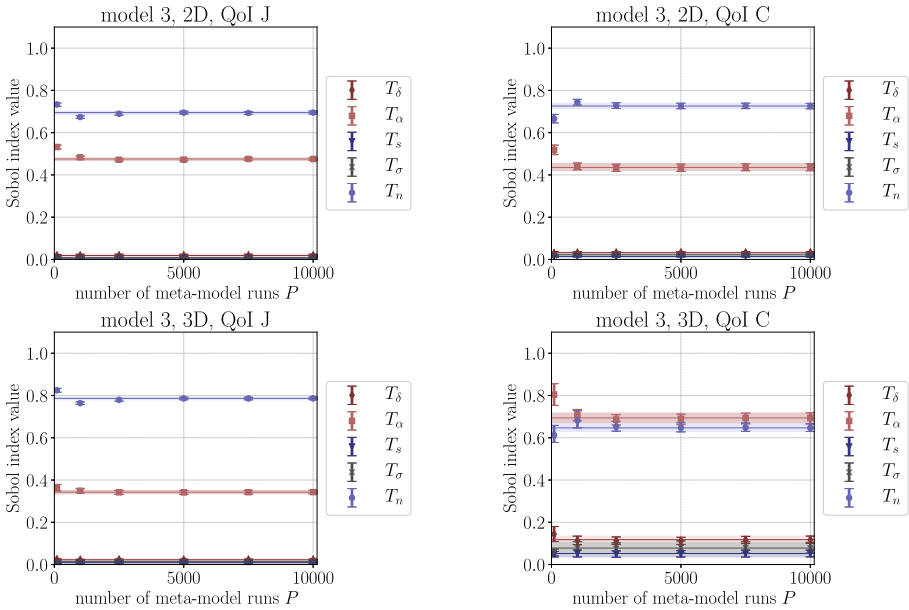

Fig. 7. Upper left: sensitivity analysis results from model 3, 2D (20 × 20 grid) for QoI J; Upper right: sensitivity analysis results from model 3, 2D (20 × 20 grid) for QoI C. Lower left: sensitivity analysis results from model 3, 3D (8 × 8 × 8 grid) for QoI J; Lower right: sensitivity analysis results from model 3, 3D (8 × 8 × 8 grid) for QoI C. The error bars show the 95% bootstrap confidence interval around the mean of each value of T. Horizontal lines show the confidence interval bounds for $P = 10{,}000$. These plots show that the total Sobol index values T converge with respect to the number of meta-model runs P, and that α and n are consistently the most important parameters. Additional sensitivity analysis results are presented in Fig. C.2.

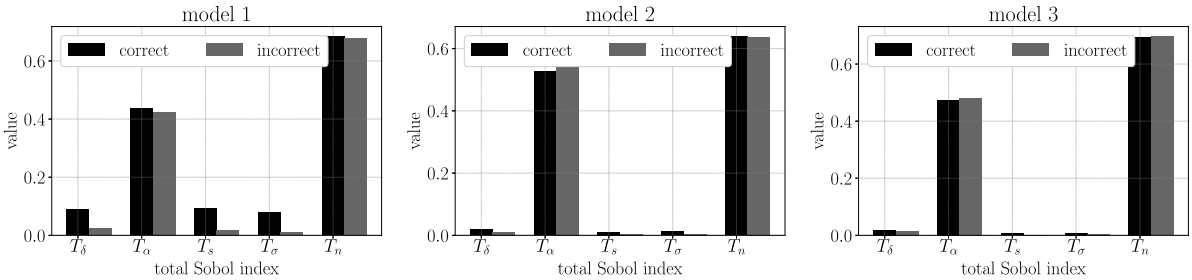

Fig. 8. Comparison of the "correct" sensitivity analysis procedure (Section 2.4) to the "incorrect" sensitivity analysis procedure (Section 2.4.1) for the three model sizes in the two-dimensional setting. To build these histograms, we sampled the direct model $N = 200$ times and the meta-model $P = 10{,}000$ times.

In Fig. 8, we compare the correct method for conducting a sensitivity analysis on a stochastic simulation to the incorrect method described in Section 2.4.1. For the smallest model size (4 × 4 grid, model 1), there is noticeable difference in the sensitivity analysis results. Specifically, the influence of parameters δ, $s_{\max}$ and σ_υ is underreported when the incorrect method is used on the smallest grid size. However, for the larger model sizes, this difference is not substantial. As follows from intuition, employing the correct method is more important when there is less "self-averaging" within the model. The similarity between the two methods indicates that meaningful results can potentially still be obtained in analogous experimental systems even when stochastic behavior is not reproducible.

3.4. Comparison of meta-modeling procedures

Beyond sensitivity analysis, meta-modeling has substantial promise as a tool for scale-bridging in a multi-scale computational framework. For this application, we are interested in the performance of a given meta-modeling

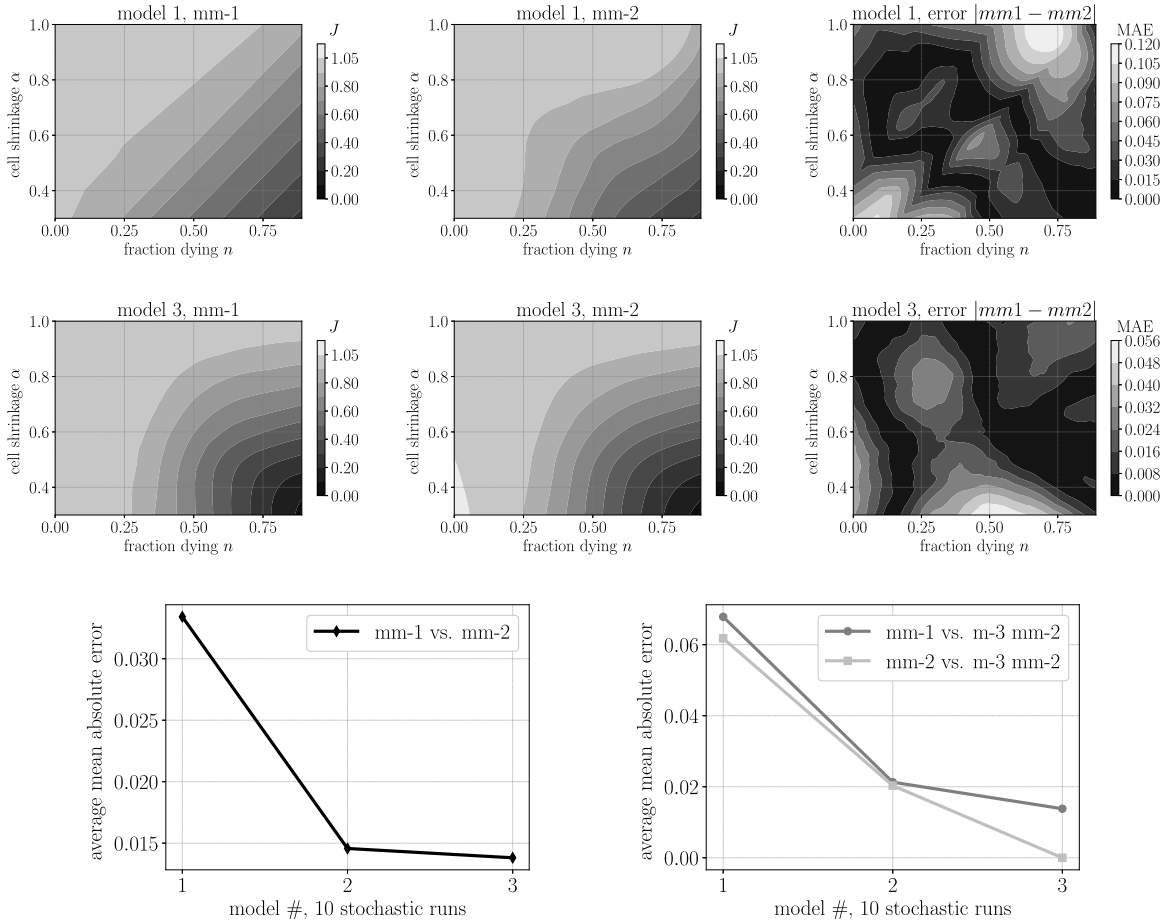

Fig. 9. Upper left: contour plot of mm-1 trained to model 1 data (mean absolute test error: 0.10); Upper center: contour plot of mm-2 trained to model 1 data (mean absolute test error: 0.073); Upper right: contour plot comparing mm-1 and mm-2 for model 1; Middle left: contour plot of mm-1 trained to model 3 data (mean absolute test error: 0.024); Middle center: contour plot of mm-2 trained to model 3 data (mean absolute test error: 0.020); Middle right: contour plot comparing mm-1 and mm-2 for model 3; Lower left: difference between mm-1 and mm-2 for the different models; Lower right: Comparison between all models and mm-2 trained on model 3. The results depicted in this figure are for the two-dimensional case. Additional contour plots are presented in Fig. C.3.

procedure on unseen model data that does not have identical random behavior to the data in the test set. In Fig. 9, we show a comparison between the two meta-modeling procedures described in Section 2.3. The contour plots show predicted QoI J with respect to the two most important parameters α and n. For our data, we found that mm-2, where multiple meta-models are constructed and then averaged, outperformed mm-1, where a single meta-model is fit to the entire data set. The difference in behavior between mm-1 and mm-2 is highly apparent for model 1 (the 4×4 grid), as seen in the upper row of Fig. 9, and is substantially less apparent for model 3 (the 20×20 grid), as seen in the middle row of Fig. 9. We show that for this model increasing the number of cells in the initial grid decreases the difference between mm-1 and mm-2. We observe qualitatively similar behavior for both the three-dimensional results and the results for QoI C. In both cases mm-2 performs better, but mm-1 can perform comparably for the larger models. In this work, we use Latin Hypercube sampling to cover the input parameter space [57]. In future work, more in depth analysis of the tradeoff between quantifying stochastic variation and exploring the parameter space may be warranted [58]. A brief discussion on sampling across different random instances is presented in Appendix D.

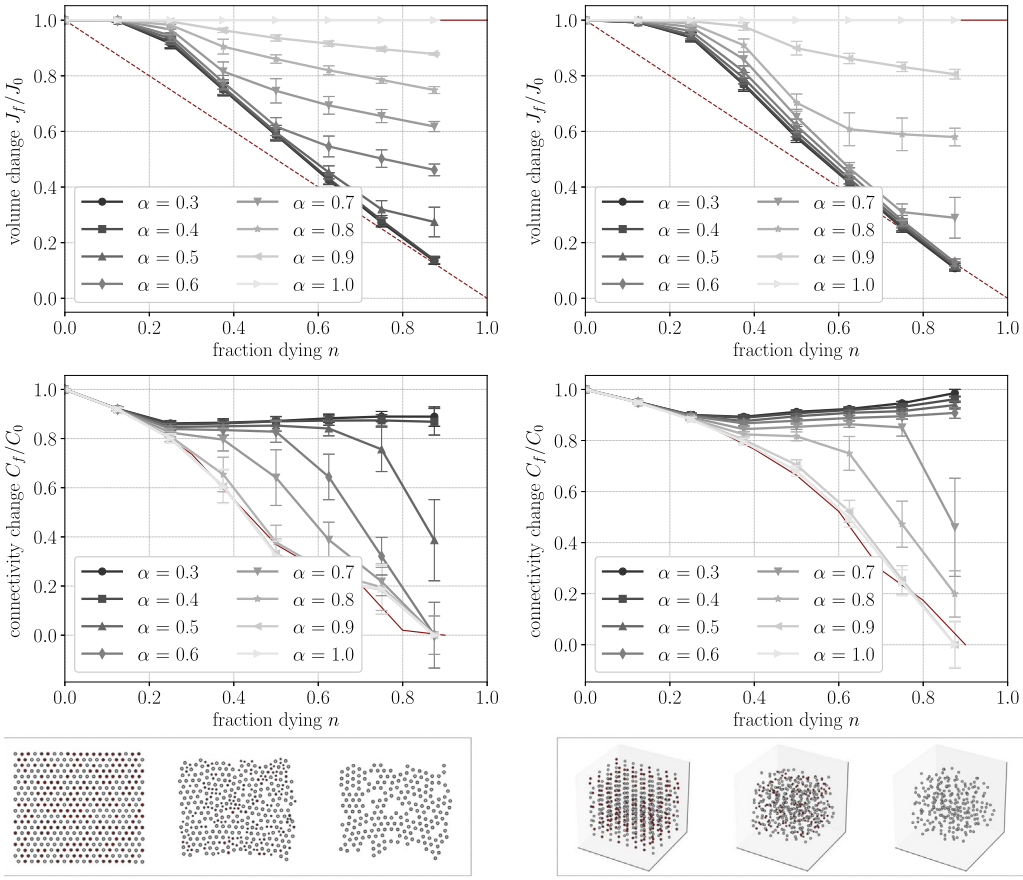

Fig. 10. Parameter sweep of the two most important parameters identified in the sensitivity analysis n and α. Upper left: volume change J_f/J_0 in two-dimensions; Upper right: volume change J_f/J_0 in three-dimensions; Lower left: connectivity change C_f/C_0 in two-dimensions; Lower right: connectivity change C_f/C_0 in three-dimensions. The illustrations on the lower row are representative simulation results where each cell is illustrated as a single node. Each point shows the mean and standard deviation of 50 simulations. For QoI J, the red dashed line represents $J = V/V_0$. For QoI C, the red dashed line represents C_f/C_0 for the case of no position updating as illustrated in Fig. 4.

3.5. Key model results relevant to understanding cell death

Thus far, we have demonstrated the efficacy of our methods for interpreting agent-based models. Here we show the key insights gained from the specific model described in Section 2.1. To begin, full parameter sweeps of the most important parameters α and n for both J and C in two-dimensional and three-dimensional space are shown in Fig. 10. The two key insights from these simulations are as follows.

3.5.1. The two-dimensional and three-dimensional cases are quantitatively different

One notable result from our analysis is that the two-dimensional and three-dimensional cases are quantitatively different. This result is seen broadly in Fig. 10, and more explicitly in Fig. 11. In Fig. 11, shrinkage J_f/J_0 is plotted with respect to cell shrinkage α for different values of fraction dying n in two-dimensions and three-dimensions. Qualitatively, simulation results are similar. In both cases, smaller α and larger n both result in more population shrinkage. However, the simulation results differ quantitatively. This difference is likely because of the inherent difference in connectivity between discrete cells in two-dimensional and three-dimensional space. Unlike with a continuum model, our discrete model allows this difference to naturally emerge. This is an important result because it shows that for a problem like trying to understand tumor shrinkage, where the cellular micro-environment is very much three-dimensional, two-dimensional simulations – and likely two-dimensional experiments – are not

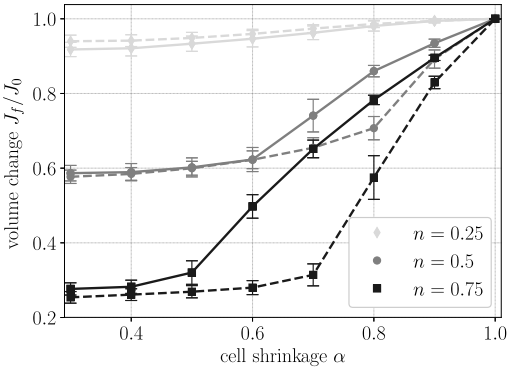

Fig. 11. Plot of measured volume change J_f/J_0 with respect to cell shrinkage α for different values of n with fixed $s_{\max}$, $\sigma_\upsilon/\mu_\upsilon$, and δ_0 at the mean of the parameter space. The solid line represents the two-dimensional case, the dashed line represents the three-dimensional case. Each point shows the mean and standard deviation of 50 simulations. Notably, when $n = 0.25$, two-dimensional and three-dimensional results are not substantially different. When $n = 0.5$, two-dimensional and three-dimensional results are substantially different for $\alpha = 0.7, 0.8$. When $n = 0.75$, two-dimensional and three-dimensional results are substantially different for $\alpha = 0.6, 0.7, 0.8, 0.9$.

sufficient. Looking forward, this result motivates constructing three-dimensional agent-based models of microscale cell population shrinkage, even if three-dimensional models are more computationally expensive.

3.5.2. There is a trade-off between QoI J and QoI C with respect to shrinkage parameter α

As the cell shrinkage α increases for a fixed fraction of dying cells n, QoI J increases while QoI C decreases. This is shown broadly in Fig. 10 and more explicitly in Fig. 12. In Fig. 12, the left axis shows population volume change J_f/J_0 and the right axis shows connectivity change C_f/C_0 with respect to α for a fixed value of n for both the two-dimensional and three-dimensional cases. The trade-off between J and C is interesting for two reasons. First, it emphasizes that modeling cell death in a cell population is not trivial, and care should be taken to investigate and select the mechanisms and parameters designed to capture the cell death process in the agent-based model setting. Second, it implies that different implementations of cell death on the cellular scale will lead to quite different manifestations of cell death on the macroscale. In some cases, cell death may be best captured by applying volume change, related to changes in J, while in the other cases cell death may be better captured by a damage parameter or change in density or porosity, related to changes in C. On the macroscale, this may have substantial implications for the way in which fluid and subsequently dissolved species can percolate through the tissue. And, of course, in the intermediate scenario, some combination of macroscale representations may be necessary.

4. Conclusion

In this paper, we explore the process of cell death with the ultimate goal of being able to incorporate microscale data describing cell population response to cell death into a multi-scale computational framework. Here we focused on agent-based modeling of cell death and the methods and procedures required to construct, interpret, and ultimately successfully utilize agent-based models. We began in Section 2 by introducing our methods. In Section 2.1, we defined our model wherein a given fraction of cells in a grid undergoes cell death. Then, in Section 2.2, we introduced the procedure for post-processing model results by computing shrinkage J and average connectivity C QoIs. In Section 2.4 we described the process of conducting a sensitivity analysis for a stochastic agent-based model, and in Section 2.3 we introduced two potential meta-modeling frameworks for simulations with reproducible stochastic behavior.

In Section 3, we present exploratory model results. Specifically, in Section 3.3, we showed through sensitivity analysis that population fraction undergoing cell death n and cell shrinkage α are the most important model parameters. In Section 3.4, we compare multiple meta-modeling strategies for simulations with reproducible stochastic behavior and illustrated that the performance of multiple frameworks should be investigated when possible. In Section 3.5.1 we showed that the two-dimensional and three-dimensional versions of the model lead to different quantitative behavior. Finally, in Section 3.5.2, we showed that there is a trade-off between shrinkage J and connectivity C for variations in cell shrinkage α.

Please cite this article as: E. Lejeune and C. Linder, Interpreting stochastic agent-based models of cell death, Computer Methods in Applied Mechanics and Engineering (2019) 112700, https://doi.org/10.1016/j.cma.2019.112700.

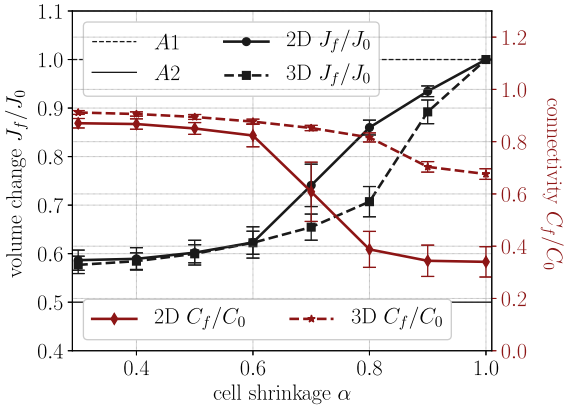

Fig. 12. Plot of measured volume change J_f/J_0 (left axis) and connectivity change C_f/C_0 (right axis) with respect to cell shrinkage α for both two-dimensional and three-dimensional cases. All other parameters are fixed with $n = 0.5$, and $s_{\max}$, $\sigma_\upsilon/\mu_\upsilon$, and δ_0 at the mean of the parameter space. Each point shows the mean and standard deviation of 50 simulations. Here, $A1$ shows population volume change in the case of no shrinkage while $A2$ shows population volume change in proportion to the fraction of dying cells. The difference between the two-dimensional and three-dimensional simulations is substantial for $\alpha = 0.7$, 0.8, 0.9.

The methods and results presented here are a starting point towards future work in approaching cell-population scale data with multi-scale modeling as the ultimate goal. Notably, the QoI defined in Section 2.2, and both the sensitivity analysis and meta modeling frameworks described in Sections 2.4 and 2.3 are all data source agnostic. In future work, data can be derived from more robust and tailored computational models or in vitro experiments. The exploratory results with our simple agent-based model indicate that from a mechanics perspective, the macro-scale strategy for modeling cell death will vary substantially based on the underlying cellular scale behavior of the tissue in question.

We anticipate that future work in this direction will include three major components. First, future work will involve developing a robust platform for comparing experimental observations at multiple scales. Second, future cellular-scale models of cell death will necessarily be much more complex and should take into account the multiphysics nature of the cell death process and additional aspects of cellular scale structure. For example, future models should contain explicit representations of the cell membrane, cell–cell adhesions, and interstitial fluid. Third, for given types of cell death observed on the cellular and population scales, physically meaningful macro-scale models must be formulated and coupled to cellular-scale behavior. In particular, the direct mechanistic influence of microscale cell death on macroscale tissue porosity is a promising future area of investigation. With these components, rigorous multi-scale modeling of phenomena such as tumor response to intervention will be possible.

Acknowledgments

This work was supported by the National Science Foundation, United States of America Graduate Research Fellowship Grant No. DGE-114747 to EL and National Science Foundation, United States of America CAREER Grant No. 1553638 to CL. We thank Benjamin Carter for assistance with proofreading this manuscript.

Appendix A. Implementing the agent-based cell model

In Section 2.1, we briefly introduced our cell population model. Here, we provide additional detail. The mechanical aspects of our cell population model are based on peridynamics, a theoretical and computational framework that is typically implemented numerically as a mesh-free method [32,59]. With this framework, we treat each cell as an individual node and use the peridynamic equation of motion to maintain mechanical equilibrium [27]. For further information, we direct the reader to the peridynamics literature [60–63].

In Eq. (5) we define the discrete form of the peridynamic equation of motion at static equilibrium, and in Eq. (6), we defined the horizon and dual-horizon. After each simulation step of algorithmically defined cell behavior, the entire system is relaxed back to mechanical equilibrium via an adaptive dynamic relaxation procedure [64]. Now,

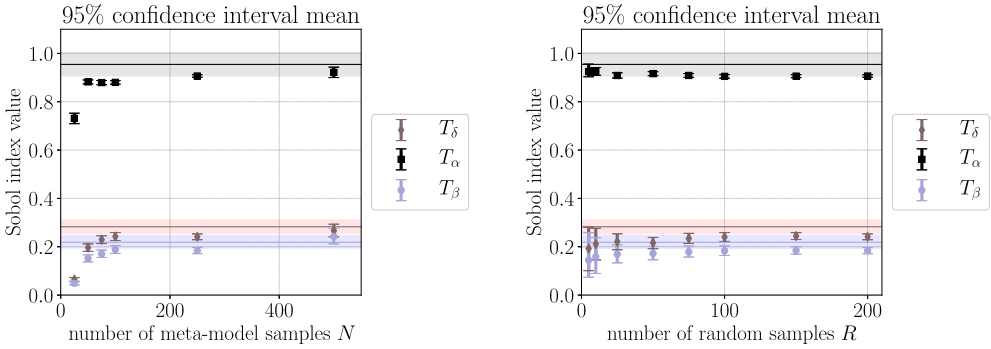

Fig. B.1. The total Sobol indices computed with the meta-model (symbol markers) converge to the total Sobol indices computed by direct model sampling (horizontal lines, 95% confidence interval shown) as both the number of parameter combinations used to train the meta-model N increases (left plot) and as the number of random model samples R increases. Further details of this simulation are available in [54].

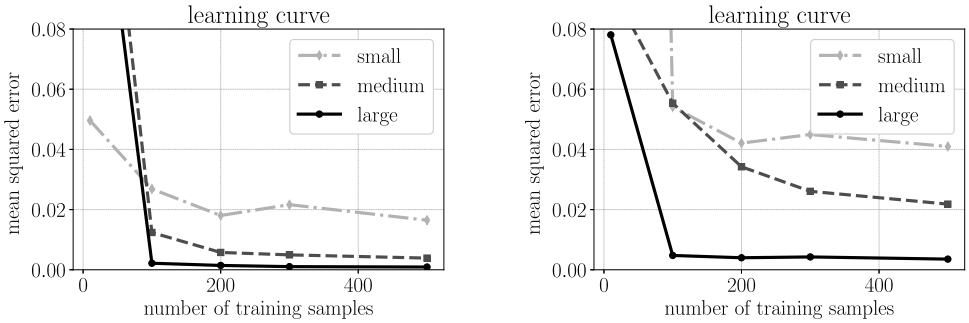

Fig. C.1. Left: 5-fold cross-validation error with respect to the number of training samples for predicting J from input parameters with fixed random behavior in three-dimensions; Right: 5-fold cross-validation error with respect to the number of training samples for predicting C from input parameters with fixed random behavior in three-dimensions. The "small" simulations are on a $4 \times 4 \times 4$ grid, the "medium" simulations are on an $6 \times 6 \times 6$ grid, the "large" simulations are on a $8 \times 8 \times 8$ grid.

we provide the equations necessary to define force density $\boldsymbol{f}$. First, we define the stretch free separation distance between nodes as

$$\|\boldsymbol{\xi}_{\mathrm{jk}}\| = (1 + g_{\mathrm{j}})r_{\mathrm{j}} + (1 + g_{\mathrm{k}})r_{\mathrm{k}} \tag{A.1}$$

where r is the initial radius associated with each node. Given $\|\boldsymbol{\xi}_{\mathrm{jk}}\|$, we define the stretch between node j and node k as

$$s_{\mathrm{jk}} = \frac{\|\boldsymbol{y}_{\mathrm{k}} - \boldsymbol{y}_{\mathrm{j}}\| - \|\boldsymbol{\xi}_{\mathrm{jk}}\|}{\|\boldsymbol{\xi}_{\mathrm{jk}}\|} \tag{A.2}$$

which is used to determine bond damage γ_{jk} as

$$\gamma_{\mathrm{jk}} = \begin{cases} 1 & \text{if } s < s_{\max} \\ 0 & \text{otherwise} \end{cases} \tag{A.3}$$

with $s_{\max}$ defined as the maximum allowable stretch between bonds. Parameter $s_{\max}$ is investigated in the sensitivity analysis. In this work, influence function ω is defined simply as

$$\omega_{\mathrm{jk}} = \gamma_{\mathrm{jk}}\,. \tag{A.4}$$

Next, we define horizon weighted volume m as

$$m_{\mathrm{j}} = \sum_{\mathrm{k} \in \mathcal{H}_{\mathrm{j}}} \omega_{\mathrm{jk}}\, \|\boldsymbol{\xi}_{\mathrm{jk}}\|^2\, \Delta V_{\mathrm{k}} \tag{A.5}$$

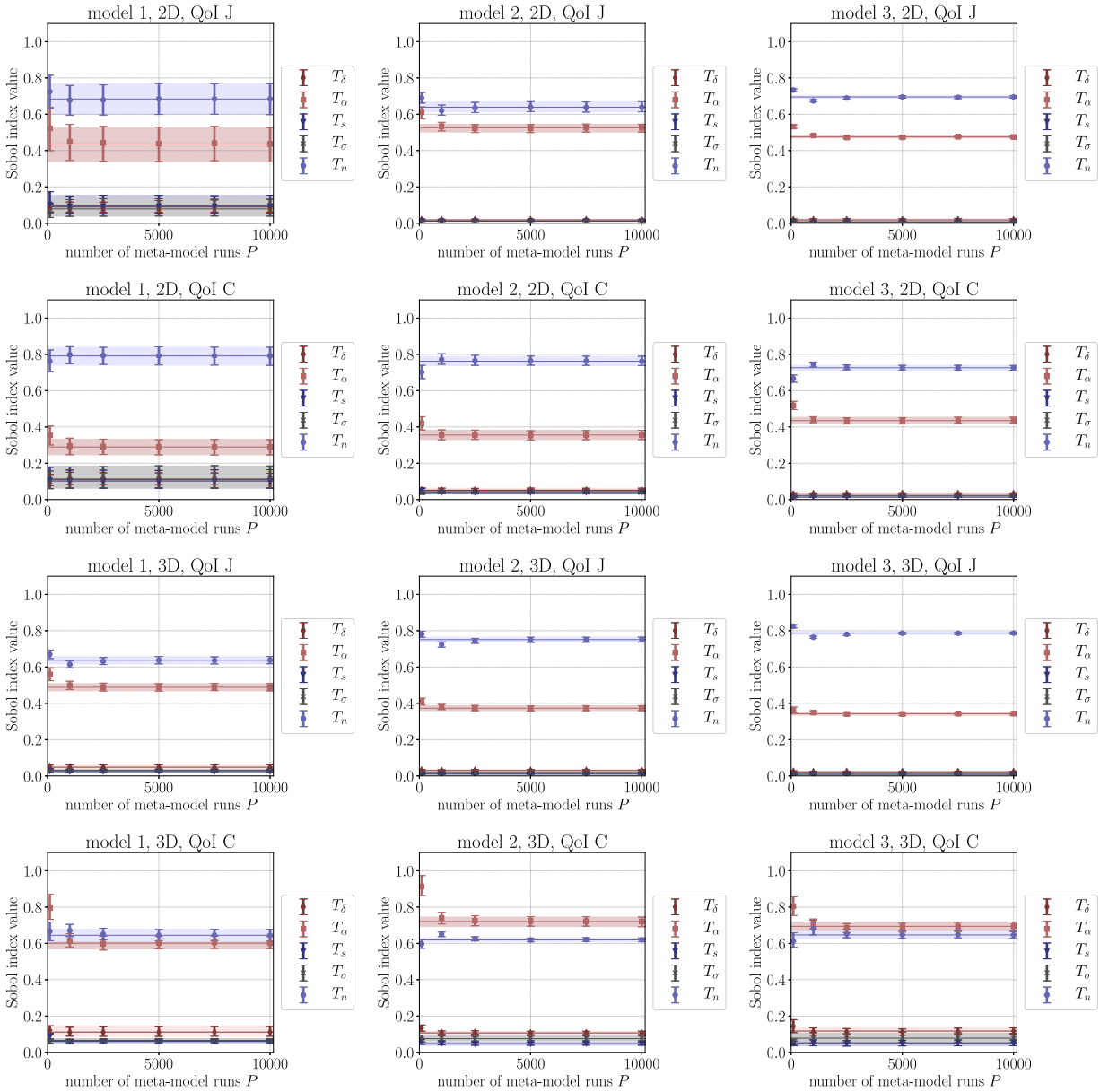

Fig. C.2. Sensitivity analysis results for QoI J and QoI C for all sizes of two-dimensional and three-dimensional simulation. The error bars show the 95% bootstrap confidence interval around the mean of each value of T. These plots show that the total Sobol index values T converge with respect to the number of meta-model runs P, and that α and n are consistently the most important parameters.

bond elongation e as

$$e_{\mathrm{jk}} = \|\boldsymbol{y}_{\mathrm{k}} - \boldsymbol{y}_{\mathrm{j}}\| - \|\boldsymbol{\xi}_{\mathrm{jk}}\| \tag{A.6}$$

dilation θ as

$$\theta_{\mathrm{j}} = \frac{n}{m_{\mathrm{j}}} \sum_{\mathrm{k}\in\mathcal{H}_{\mathrm{j}}} \omega_{\mathrm{jk}} \, \|\boldsymbol{\xi}_{\mathrm{jk}}\| \, e_{\mathrm{jk}} \, \Delta V_{\mathrm{k}} \tag{A.7}$$

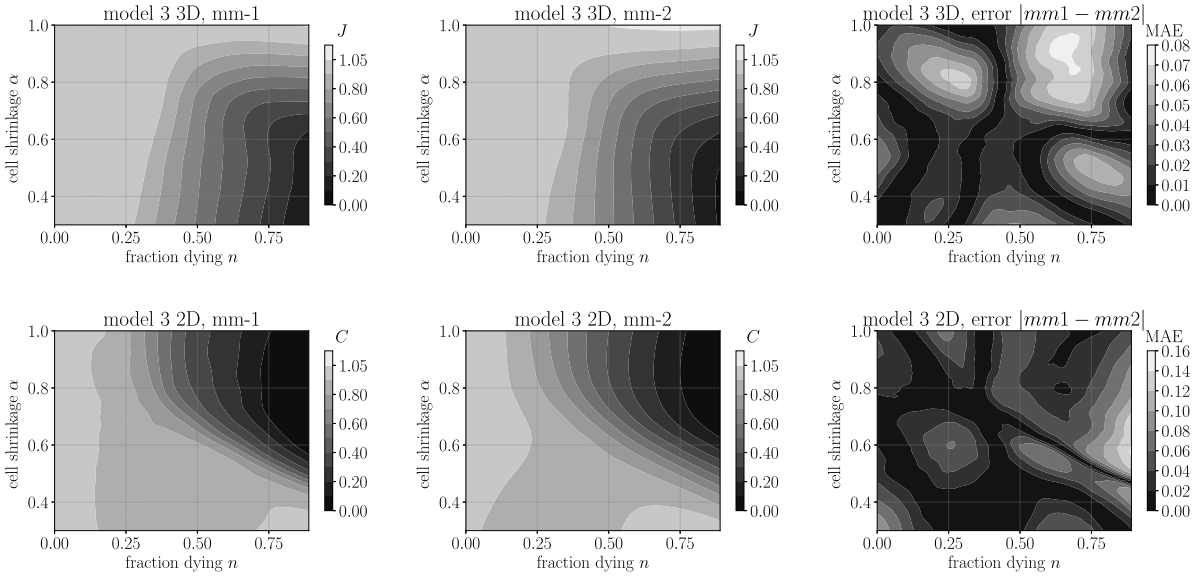

Fig. C.3. Upper left: contour plot of mm-1 trained to model 3 3D data for QoI J; Upper center: contour plot of mm-2 trained to model 3 3D data for QoI J; Upper right: contour plot comparing mm-1 and mm-2 for model 3 3D for QoI J; Lower left: contour plot of mm-1 trained to model 3 2D for QoI C; Lower center: contour plot of mm-2 trained to model 3 2D for QoI C; Lower right: contour plot comparing mm-1 and mm-2 for model 3 2D for QoI C.

and deviatoric bond elongation e^d as

$$e^d_{\mathrm{jk}} = e_{\mathrm{jk}} - \frac{\theta_{\mathrm{j}}\,\|\boldsymbol{\xi}_{\mathrm{jk}}\|}{n} \tag{A.8}$$

where $n = 3$ for a three-dimensional problem. With these terms defined, the equation for the magnitude of force density that arises at node k due to node j is

$$t_{\mathrm{kj}} = \frac{n\,\kappa\,\theta_{\mathrm{j}}}{m_{\mathrm{j}}}\,\omega_{\mathrm{jk}}\,\|\boldsymbol{\xi}_{\mathrm{jk}}\| \;+\; \frac{n\,(n+2)\,\mu}{m_{\mathrm{j}}}\,\omega_{\mathrm{jk}}\,e^d{}_{\mathrm{jk}} \tag{A.9}$$

where κ and μ are Lamé parameters. Building on these definitions, force density is defined as

$$\begin{aligned} \boldsymbol{f}_{\mathrm{jk}}(\boldsymbol{y}_{\mathrm{j}}, \boldsymbol{y}_{\mathrm{k}}) &= t_{\mathrm{jk}} \cdot \frac{\boldsymbol{y}_{\mathrm{k}} - \boldsymbol{y}_{\mathrm{j}}}{\|\boldsymbol{y}_{\mathrm{k}} - \boldsymbol{y}_{\mathrm{j}}\|} \\ \boldsymbol{f}_{\mathrm{kj}}(\boldsymbol{y}_{\mathrm{j}}, \boldsymbol{y}_{\mathrm{k}}) &= t_{\mathrm{kj}} \cdot \frac{-(\boldsymbol{y}_{\mathrm{k}} - \boldsymbol{y}_{\mathrm{j}})}{\|\boldsymbol{y}_{\mathrm{k}} - \boldsymbol{y}_{\mathrm{j}}\|}\,. \end{aligned} \tag{A.10}$$

In Table 1 we list the parameters required to implement our simulations.

Appendix B. Efficacy of the neural network meta-model during sensitivity analysis

In our previous work [54], we conducted a sensitivity analysis on a similar system without the intermediate step of constructing a meta-model. This was computationally feasible because the prior system was two-dimensional, contained fewer cells than the simulations presented here, and only three parameters were investigated. With the simulation data from our previous publication, we are able to further verify the efficacy of our neural network based meta-model by comparing the results of the sensitivity analysis with and without the meta-modeling step. In Fig. B.1, we show that as the number of parameter samples used to construct the meta-model N and random samples R increase, the meta-model based predictions of Sobol indices converge to the Sobol indices computed with direct model sampling. This result confirms that the meta-modeling strategy is effective.

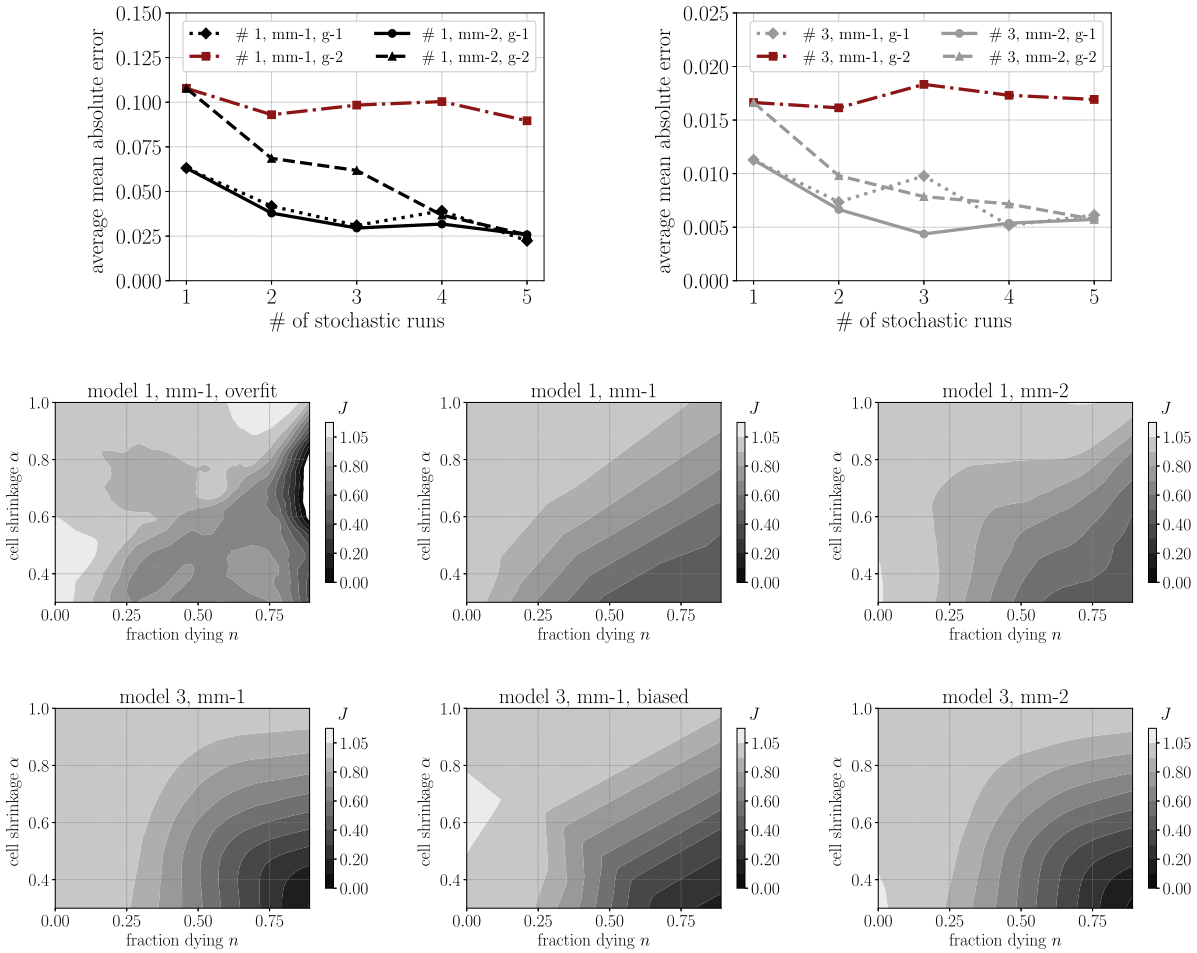

Fig. D.1. The upper row shows a comparison between g-1 and g-2 sampled data for mm-1 and mm-2. Upper left: comparing different meta-models of model 1 data to mm-2 with all 10 stochastic runs included; Upper right: comparing different meta-models of model 3 data to mm-2 with all 10 stochastic runs included; The middle and lower rows illustrate the effects of different meta-modeling strategies for g-2 sampled data. Middle left: contour plot of mm-1 nn-1 trained to model 1 data (mean absolute test error: 0.14); Middle center: contour plot of mm-1 nn-2 trained to model 1 data (mean absolute test error: 0.10); Middle right: contour plot of mm-2 nn-1 trained to model 1 data (mean absolute test error: 0.097); Lower left: contour plot of mm-1 nn-1 trained to model 1 data (mean absolute test error: 0.024); Lower center: contour plot of mm-1 nn-1 trained to model 1 data (mean absolute test error: 0.048); Lower right: contour plot of mm-2 nn-2 trained to model 1 data (mean absolute test error: 0.023). Here, neural network 1 (nn-1) is trained with the same parameters as in Fig. 6 while neural network 2 (nn-2) is trained with different parameters tuned to avoid overfitting for mm-1 model 1.

Appendix C. Additional simulation results

In this Appendix we include supplementary simulation results. First, in Fig. C.1, we show the performance of our neural network based meta-model on three-dimensional simulation data. Notably, this figure is similar to Fig. 6 in the main body of the text. Then, in Fig. C.2, we show the results of the sensitivity analysis for two-dimensional data and three-dimensional data for all model sizes with respect to QoI J and QoI C. As with the results reported in the main body of the text, n and α are the most important parameters for all simulation types. Finally, in Fig. C.3, we show additional contour plots of QoI J and QoI C with respect to the most important parameters n and α. Fig. C.3 is consistent with the data presented in Fig. 9.

Please cite this article as: E. Lejeune and C. Linder, Interpreting stochastic agent-based models of cell death, Computer Methods in Applied Mechanics and Engineering (2019) 112700, https://doi.org/10.1016/j.cma.2019.112700.

Table D.3
This table summarizes the meta-models investigated for data sampled from model 1 and model 3. Results are shown in Fig. D.1.

Model 1	mm-1, nn-1	mm-1, nn-2	mm-2, nn-1
g-1 sampling	Best performance	n/a	Best performance
g-2 sampling	Overfit	Good performance	Best performance
Model 3	**mm-1, nn-1**	**mm-1, nn-2**	**mm-2, nn-1**
g-1 sampling	Best performance	n/a	Best performance
g-2 sampling	Good performance	Biased	Best performance

Appendix D. Additional details of meta-model selection

As stated in the main body of the text, we use Latin Hypercube Sampling to sample the input parameters space. For our model with repeatable stochastic behavior, we can generate new data by running the model multiple times with the same input parameters (herein referred to as group 1 or g-1) or instead generate a new set of input parameters with the Latin Hypercube Sampling algorithm for each run of repeatable stochastic behavior (herein referred to group 2 or g-2). We found that for our model, generating a new set of input parameters for each stochastic run (g-2) makes the meta-model prone to overfitting when meta-modeling strategy 1 (mm-1) is implemented. In this section, we briefly illustrate the difference between mm-1 and mm-2 for sampling algorithms g-1 and g-2.

We have summarized our analysis and the performance of each model in Table D.3. The results of our comparison are also shown in Fig. D.1. Critically, we show that the results obtained when mm-1 is implemented for g-2 sampled data differ from the mm-1 g-1, mm-2 g-1, and mm-2 g-2 groups. In Fig. D.1 we also show examples of a meta-model that is overfitting on model 1 data but performing acceptably on model 3 data, and a meta-model that is biased in the context of model 3 data but helps avoid overfitting with model 1 data. The meta-models implemented here are multi-layer perceptron regressors. The first neural network (nn-1) contains 2 hidden layers with 500 neurons each. The second neural network (nn-2) contains 2 hidden layers with 5 neurons each. As illustrated in Fig. D.1, nn-2 is less flexible though less prone to overfitting. This is a typical example of the bias/variance tradeoff [40]. Though future work is needed in this area, these initial results indicate that when we are restricted to mm-1, g-1 sampling might be preferable to g-2 sampling because it is more likely to avoid model overfitting.

References

[1] Y. Yamaguchi, N. Shinotsuka, K. Nonomura, K. Takemoto, K. Kuida, H. Yosida, M. Miura, Live imaging of apoptosis in a novel transgenic mouse highlights its role in neural tube closure, J. Cell Biol. 195 (6) (2011) 1047–1060.
[2] J. Weickenmeier, M. Jucker, A. Goriely, E. Kuhl, A physics-based model explains the prion-like features of neurodegeneration in alzheimer's disease, parkinson's disease, and amyotrophic lateral sclerosis, J. Mech. Phys. Solids 124 (2019) 264–281.
[3] N. Bellomo, N.K. Li, P.K. Maini, On the foundations of cancer modelling: selected topics, speculations, and perspectives, Math. Models Methods Appl. Sci. 18 (4) (2008) 593–646.
[4] O. Clatz, M. Sermesant, P.-Y. Bondiau, H. Delingette, S.K. Warfield, G. Malandain, N. Ayache, Realistic simulation of the 3-d growth of brain tumors in mr images coupling diffusion with biomechanical deformation, IEEE Trans. Med. Imaging 24 (10) (2005) 1334–1346.
[5] T.S. Deisboeck, Z. Wang, P. Macklin, V. Cristini, Multiscale cancer modeling, Ann. Rev. Biomed. Eng. 13 (2011) 127–155.
[6] R. Baskar, K. Lee, R. Yeo, K. Yeoh, Cancer and radiation therapy: current advances and future directions, Int. J. Med. Sci. 9 (3) (2012) 193.
[7] E. Cohen-Jonathan, E. Bernhard, W. McKenna, How does radiation kill cells? Curr. Opin. Chem. Biol. 3 (1) (1999) 77–83.
[8] T.C. Harris, R. de Rooij, E. Kuhl, The shrinking brain: Cerebral atrophy following traumatic brain injury, Ann. Biomed. Eng. (2018) 1–19.
[9] R. Jain, J. Martin, T. Stylianopoulos, The role of mechanical forces in tumor growth and therapy, Annu. Rev. Biomed. Eng. 16 (321–346) (2014).
[10] L. Taber, Biomechanics of growth, remodeling, and morphogenesis., Appl. Mech. Rev. 48 (8) (1995) 487–545.
[11] E. Lejeune, C. Linder, Quantifying the relationship between cell division angle and morphogenesis through computational modeling, J. Theoret. Biol. 418 (2017) 1–7.
[12] G. Majno, I. Joris, Apoptosis, oncosis, and necrosis. an overview of cell death, Amer. J. Pathol. 146 (1) (1995) 3.
[13] D. Ambrosi, S. Pezzuto, D. Riccobelli, T. Stylianopoulos, P. Ciarletta, Solid tumors are poroelastic solids with a chemo mechanical feedback on growth, J. Elasticity (2016) 1–18.
[14] M. Suzanne, H. Steller, Shaping organisms with apoptosis, Cell Death Differ. 20 (5) (2013) 669.
[15] T. Stylianopoulos, The solid mechanics of cancer and strategies for improved therapy, J. Biomech. Eng. 139 (2) (2017) 021004.

Please cite this article as: E. Lejeune and C. Linder, Interpreting stochastic agent-based models of cell death, Computer Methods in Applied Mechanics and Engineering (2019) 112700, https://doi.org/10.1016/j.cma.2019.112700.

[16] E. Lejeune, C. Linder, Modeling mechanical inhomogeneities in small populations of proliferating monolayers and spheroids, Biomech. Model. Mechanobiol. 17 (3) (2018) 727–743.
[17] G. An, Q. Mi, J. Dutta-Moscato, Y. Vodovotz, Agent-based models in translational systems biology, Wiley Interdiscip. Rev.: Syst. Biol. Med. 1 (2) (2009) 159–171.
[18] Z. Wang, J. Butner, R. Kerketta, V. Cristini, T.S. Deisboeck, Simulating cancer growth with multiscale agent-based modeling, Semin. Cancer Biol. 30 (2015) 70–78.
[19] M. Rahman, Y. Feng, T. Yankeelov, J. Oden, A fully coupled space–time multiscale modeling framework for predicting tumor growth, Comput. Methods Appl. Mech. Engrg. 320 (2017) 261–286.
[20] H. Rocha, R. Almeida, E. Lima, A. Resende, J. Oden, T. Yankeelov, A hybrid three-scale model of tumor growth, Math. Models Methods Appl. Sci. 28 (1) (2018) 61–93.
[21] M.K. Rausch, J.D. Humphrey, A computational model of the biochemomechanics of an evolving occlusive thrombus, J. Elasticity 129 (1–2) (2017) 125–144.
[22] A.B. Tepole, E. Kuhl, Computational modeling of chemo-bio-mechanical coupling: a systems-biology approach toward wound healing, Comput. Methods Biomech. Biomed. Eng. 19 (1) (2016) 13–30.
[23] Y.-C. Tung, A.Y. Hsiao, S.G. Allen, Y.-s. Torisawa, M. Ho, S. Takayama, High-throughput 3d spheroid culture and drug testing using a 384 hanging drop array, Analyst 136 (3) (2011) 473–478.
[24] S. Breslin, L. O'Driscoll, Three-dimensional cell culture: the missing link in drug discovery, Drug Discov. Today 18 (5) (2013) 240–249.
[25] D.A. Van Valen, T. Kudo, K.M. Lane, D.N. Macklin, N.T. Quach, M.M. DeFelice, I. Maayan, Y. Tanouchi, E.A. Ashley, M.W. Covert, Deep learning automates the quantitative analysis of individual cells in live-cell imaging experiments, PLoS Comput. Biol. 12 (11) (2016) e1005177.
[26] E. Meijering, O. Dzyubachyk, I. Smal, Methods for cell and particle tracking, in: Methods in Enzymology, Vol. 504, Elsevier, 2012, pp. 183–200.
[27] E. Lejeune, C. Linder, Modeling tumor growth with peridynamics, Biomech. Model. Mechanobiol. (2017) 1–17.
[28] D. Drasdo, A. Buttenschön, P. Van Liedekerke, Agent-based lattice models of multicellular systems: numerical methods, implementation, and applications, in: Numerical Methods and Advanced Simulation in Biomechanics and Biological Processes, Elsevier, 2018, pp. 223–238.
[29] P. Van Liedekerke, A. Buttenschön, D. Drasdo, Off-lattice agent-based models for cell and tumor growth: numerical methods, implementation, and applications, in: Numerical Methods and Advanced Simulation in Biomechanics and Biological Processes, Elsevier, 2018, pp. 245–267.
[30] H. Byrne, D. Drasdo, Individual-based and continuum models of growing cell populations: a comparison, J. Math. Biol. 58 (4–5) (2008) 657–687.
[31] H. Ren, X. Zhuang, Y. Cai, T. Rabczuk, Dual-horizon peridynamics, Internat. J. Numer. Methods Engrg. 108 (12) (2016) 1451–1476.
[32] S.A. Silling, Reformulation of elasticity theory for discontinuities and long-range forces, J. Mech. Phys. Solids 48 (2000) 175–209.
[33] S.A. Silling, M. Epton, O. Weckner, J. Xu, E. Askari, Peridynamic states and constitutive modeling, J. Elasticity 88 (2) (2007) 151–184.
[34] E. Lejeune, B. Dortdivanlioglu, E. Kuhl, C. Linder, Understanding the mechanical link between oriented cell division and cerebellar morphogenesis, Soft Matter 15 (10) (2019) 2204–2215.
[35] M. Newman, Networks: an introduction, Oxford university press, 2010.
[36] A. Hagberg, P. Swart, D. S Chult, Exploring Network Structure, Dynamics, and Function using NetworkX, Technical Report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008.
[37] V.K. Lai, M.F. Hadi, R.T. Tranquillo, V.H. Barocas, A multiscale approach to modeling the passive mechanical contribution of cells in tissues, J. Biomech. Eng. 135 (7) (2013) 071007.
[38] T. Lee, A.K. Gosain, I. Bilionis, A.B. Tepole, Predicting the effect of aging and defect size on the stress profiles of skin from advancement, rotation and transposition flap surgeries, J. Mech. Phys. Solids 125 (2019) 572–590.
[39] G.H. Teichert, K. Garikipati, Machine learning materials physics: Surrogate optimization and multi-fidelity algorithms predict precipitate morphology in an alternative to phase field dynamics, Comput. Methods Appl. Mech. Engrg. 344 (2019) 666–693.
[40] G. James, D. Witten, T. Hastie, R. Tibshirani, An Introduction to Statistical Learning, Vol. 112, Springer, 2013.
[41] B. Ankenman, B.L. Nelson, J. Staum, Stochastic kriging for simulation metamodeling, Oper. Res. 58 (2) (2010) 371–382.
[42] J.A. Anderson, An Introduction to Neural Networks, MIT press, 1995.
[43] J.H. Friedman, Multivariate adaptive regression splines, Ann. Statist. 19 (1) (1991) 1–67.
[44] C.K. Williams, C.E. Rasmussen, Gaussian Processes for Machine Learning, MIT Press Cambridge, MA, 2006.
[45] G.H. Teichert, K. Garikipati, Machine learning materials physics: Surrogate optimization and multi-fidelity algorithms predict precipitate morphology in an alternative to phase field dynamics, Comput. Methods Appl. Mech. Engrg. 344 (2019) 666–693.
[46] F. Chollet, et al., Keras, 2015, https://keras.io.
[47] M. Abadi, et al., Tensorflow: Large-scale machine learning on heterogeneous systems, 2015, http://tensorflow.org/.
[48] A. Saltelli, Global sensitivity analysis: an introduction, in: Proc. 4th International Conference on Sensitivity Analysis of Model Output, 2004, pp. 27–43.
[49] I. Sobol, Sensitivity estimates for nonlinear mathematical models, Math. Modelling Comput. Exp. 1 (4) (1993) 407–414.
[50] I. Sobol, Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates, Math. Comput. Simulation 55 (1) (2001) 271–280.
[51] J. Hart, A. Alexanderian, P. Gremaud, Efficient computation of sobol' indices for stochastic models, SIAM J. Sci. Comput. 39 (4) (2017) 1514–1539.

Please cite this article as: E. Lejeune and C. Linder, Interpreting stochastic agent-based models of cell death, Computer Methods in Applied Mechanics and Engineering (2019) 112700, https://doi.org/10.1016/j.cma.2019.112700.

[52] J. Herman, W. Usher, Salib: An open-source python library for sensitivity analysis, J. Open Source Softw. 2 (9) (2017).
[53] A. Saltelli, Making best use of model evaluations to compute sensitivity indices, Comput. Phys. Comm. 145 (2) (2002) 280–297.
[54] E. Lejeune, C. Linder, Understanding the relationship between cell death and tissue shrinkage via a stochastic agent-based model, J. Biomech. 73 (2018) 9–17.
[55] W. Xu, R. Mezencev, B. Kim, L. Wang, J. McDonald, T. Sulchek, Cell stiffness is a biomarker of the metastatic potential of ovarian cancer cells, PLoS One 7 (10) (2012) e46609.
[56] D. Drasdo, S. Hohme, A single cell based model of tumor growth in vitro monolayers and spheroids, Phys. Biol. 2 (3) (2005) 133–147.
[57] A.B. Owen, Orthogonal arrays for computer experiments, integration and visualization, Statist. Sinica (1992) 439–452.
[58] S. Sankaran, C. Audet, A.L. Marsden, A method for stochastic constrained optimization using derivative-free surrogate pattern search and collocation, J. Comput. Phys. 229 (12) (2010) 4664–4682.
[59] S. Silling, E. Askari, A meshfree method based on the peridynamic model of solid mechanics, Comput. Struct. 83 (17) (2005) 1526–1535.
[60] D. Littlewood, Roadmap for peridynamic software implementation, SAND Report, Sandia National Laboratories, Albuquerque, NM and Livermore, CA, 2015.
[61] E. Madenci, E. Oterkus, Peridynamic Theory and its Applications, Springer, 2014.
[62] S. Oterkus, Peridynamics for the Solution of Multiphysics Problems, (Ph.D. thesis), The University of Arizona, 2015.
[63] S.A. Silling, R.B. Lehoucq, Peridynamic theory of solid mechanics, Adv. Appl. Mech. 44 (2010) 73–168.
[64] B. Kilic, E. Madenci, An adaptive dynamic relaxation method for quasi-static simulations using the peridynamic theory, Theor. Appl. Fract. Mech. (ISSN: 0167-8442) 53 (3) (2010) 194–204.

Please cite this article as: E. Lejeune and C. Linder, Interpreting stochastic agent-based models of cell death, Computer Methods in Applied Mechanics and Engineering (2019) 112700, https://doi.org/10.1016/j.cma.2019.112700.