



*Citation for published version:*

Smith, C, Mailler, C & Yates, K 2019, 'Unbiased on lattice domain growth', *Physical Review E*, vol. 100, no. 6, 063307. <https://doi.org/10.1103/PhysRevE.100.063307>

*DOI:*

[10.1103/PhysRevE.100.063307](https://doi.org/10.1103/PhysRevE.100.063307)

*Publication date:*

2019

*Document Version*

Peer reviewed version

[Link to publication](#)

(C) American Physical Society, 2019.

## University of Bath

### General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

### Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

# Unbiased on-lattice domain growth

Cameron A. Smith<sup>1,\*</sup>, Cécile Mailler<sup>2</sup>, Christian A. Yates<sup>1</sup>

<sup>1</sup>Centre for Mathematical Biology, Department of Mathematical Sciences, University of Bath, Claverton Down, Bath, BA2 7AY, United Kingdom

<sup>2</sup>Probability Laboratory, Department of Mathematical Sciences, University of Bath, Claverton Down, Bath, BA2 7AY, United Kingdom

\*E-mail: c.smith3@bath.ac.uk

Key words: domain growth, mesoscopic modelling, reaction-diffusion master equation, morphogen gradient, uniform growth, on-lattice modelling

## Abstract

Domain growth is a key process in many areas of biology, including embryonic development, the growth of tissue, and limb regeneration. As a result, mechanisms for incorporating it into traditional models for cell movement, interaction, and proliferation are of great importance. A previously well-used method in order to incorporate domain growth into on-lattice reaction-diffusion models causes a build up of particles on the boundaries of the domain, which is particularly evident when diffusion is low in comparison to the rate of domain growth. Here, we present a new method which addresses this unphysical build up of particles at the boundaries, and demonstrate that it is accurate for scenarios in which the previous method fails. Further, we discuss for which parameter regimes it is feasible to continue using the original method due to diffusion dominating the domain growth mechanism.

## 1 Introduction

Domain growth is an inherent feature of many biological systems, from neural crest cell migration [1, 2] to the growth and shrinkage of tissue [3, 4], and it has been investigated in the context of pattern formation in reaction-diffusion systems [5, 6]. It is therefore important that we have reliable mathematical tools in order to model such systems.

One method of modelling general reaction-diffusion systems is to compartmentalise the spatial domain into a lattice of small regions, in which particles are considered to reside. Typically, particles are permitted to jump between neighbouring compartments (although non-local jumping can also be incorporated [7]), and particles may react with others in their current compartment (similarly, in some methods, particles may be allowed to interact with particles in neighbouring compartments [8]). These events, under the assumption that updates happen in continuous time, are considered Markovian and, as a result, have associated exponentially distributed waiting times. The most commonly used method to simulate such systems is the Gillespie algorithm [9], however there are many others that are also used within the literature (see, for example, the next reaction method [10], the next subvolume method [11, 12], and the sorting direct method [13]).

The word ‘particle’ in this context can refer to a multitude of biological or physical entities; a particle may be a cell when modelling biological systems such as neural crest cell migration in embryonic development, a chemical species when considering pattern formation or different classes of individual in the case of the spread of an epidemic. The discretisation of the domain does not necessarily correspond to any physical attribute of the space being modelled, but is usually used as a mathematical tool through which we are able to model stochasticity in particle positions and domain growth. As an example, consider the formation of a morphogen gradient on a growing domain [14]. In this example, the particles are the morphogen molecules, and the domain may be discretised to arbitrary accuracy, with no physical meaning necessarily being attached to the compartments. This case study will be investigated in more detail in Section 3.1.

One possible option for incorporating domain growth into these on-lattice position-jump processes has been previously suggested by Baker et al. [15]. Growth is achieved by choosing compartments to

divide uniformly at random at a given rate. Once chosen, a compartment instantaneously doubles in length and splits down the middle to produce a new compartment (pushing all compartments to the right of the one chosen by one compartment's width). Particles that resided in the original compartment are then redistributed into the newly created compartments by placing each particle into one or the other with equal probability. This is illustrated schematically in Figure 1, and a method for its implementation is detailed in Algorithm 1. A commonly implemented simulation allows particles to diffuse by jumping between neighbouring boxes while, at the same time, attempting to grow the domain uniformly [15, 3, 16]. When simulating this scenario using the method of Baker et al. [15] and averaging over multiple repeats, it becomes apparent that there is an issue with particles building up at the ends of the domain, an effect that can be expected when diffusion is low in comparison to the domain growth rate. When diffusion is larger, there is enough time for particles to relax to equilibrium (particles have enough time to occupy newly created sites) before the next domain growth event.

**Algorithm 1: The original method [15]**

**(1a)** Initialise time  $t = 0$  and set the final time  $T > 0$ . Initialise the number of compartments  $k$  and the particle numbers in each compartment  $m_i$ ,  $i \in \{1, \dots, k\}$ . Specify the size of compartment  $h$ , the jump rate  $d = D/h^2$  and the growth rate  $\rho$ .

**(1b)** Calculate the propensity functions  $\alpha_i^L = dm_i$ ,  $\alpha_i^R = dm_i$  and  $\alpha_i^G = \rho$  for each compartment  $i \in \{1, \dots, k\}$  for left jumps, right jumps and growth events respectively, with  $\alpha_1^L = 0 = \alpha_k^R$ . Calculate the sum of the propensity functions:

$$\alpha_0 = \sum_{i=1}^k (\alpha_i^L + \alpha_i^R + \alpha_i^G).$$

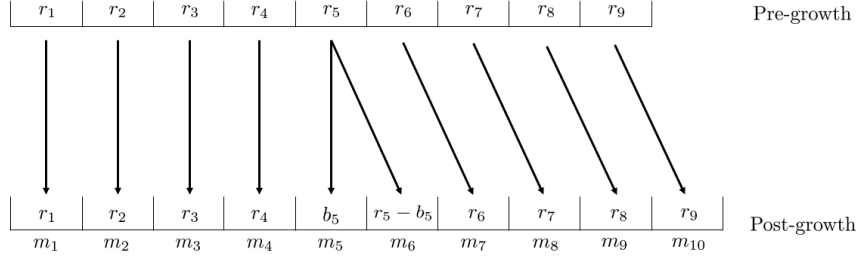
**(1c)** Calculate the time until the next event by firstly drawing a uniform random variable between zero and one,  $u_1 \sim \text{Unif}(0, 1)$ , and setting  $\tau = 1/\alpha_0 \log(1/u_1)$ . Update the time  $t \leftarrow t + \tau$ .

**(1d)** Determine which event, amongst all compartments, is next to occur by choosing one at random with probability proportional to the propensity function.

1. If the event corresponds to a left (resp. right) jump event from compartment  $i$ , remove a particle from compartment  $i$  and add one to compartment  $i - 1$  (resp.  $i + 1$ ).
2. If the event corresponds to a growth event in compartment  $i$ : Record the pregrowth particle numbers  $\mathbf{r} \leftarrow \mathbf{m}$ , create an extra compartment at the right end of the postgrowth domain (increasing  $k$  by 1 by setting  $k \leftarrow k + 1$ ), draw a binomial random variable with  $r_i$  trials and probability of success  $1/2$ ,  $b \sim \text{Bin}(r_i, 1/2)$ , and redistribute particles as follows (for  $j \in \{1, \dots, k\}$ ):

$$m_j = \begin{cases} r_j, & \text{if } j \in \{1, \dots, i - 1\}, \\ b, & \text{if } j = i, \\ r_i - b, & \text{if } j = i + 1, \\ r_{j-1}, & \text{if } j \in \{i + 2, \dots, k\}. \end{cases}$$

**(1e)** If  $t < T$  then return to step **(1b)**, else end.



**Figure 1.** A schematic showing a possible splitting event when there are nine pregrowth compartments (compartments identified by the subscripts on the particle numbers). Pregrowth particle numbers are denoted  $r_i$ . In this case, compartment 5 is chosen to divide, and its contents are split binomially with probability of success  $1/2$  ( $b_5 \sim \text{Bin}(r_5, 1/2)$ ) between the compartment in the original position (compartment 5 in this case) and the new one that is created to its right. The compartments originally numbered 6, 7, 8 and 9 are moved one position to the right and become compartments 7, 8, 9 and 10, respectively. Postgrowth particle numbers are denoted by the  $m_i$ 's.

In order to rectify this problem we have designed a novel mechanism which enables the implementation of unbiased domain growth — one that prevents the accumulation of particles at the boundaries. This method enacts a more continuous approach to domain growth compared to Baker et al. [15], in which compartments are first stretched and then renormalised with concurrent redistribution of their particles in to neighbouring compartments. In general, we will consider growth dynamics in isolation, without considering any other mechanisms such as diffusion and reactions. This is in order to demonstrate the possible issues without confounding or hiding growth induced phenomena with other dynamics. All models will be presented in one dimension, but extensions to higher dimensions will be discussed in Section 4.

This paper will be set out in the following way. In Section 2 we investigate and explain the problems with the commonly employed domain growth mechanism in more detail. In Section 3, we present a novel, but distinct and importantly unbiased domain growth method that we use to address this issue. Finally, we present our conclusions in Section 4.

## 2 Problems with an existing domain growth mechanism

Within this section, we describe and demonstrate problems with an existing domain growth mechanism at the mesoscale [15], which we will refer to as the “original method”. We then postulate why the phenomenon has not been noticed before and confirm the existence of the problem using three different techniques.

The original method causes an accumulation of particles at the ends of the domain. This build up is caused by the inherent bias in the way in which growth is implemented: compartments are always shifted to the right when one is chosen to split. When a single compartment is chosen to split, most postgrowth compartments will either retain their pregrowth contents (if a pregrowth compartment to its right splits) or will take on the contents of the pregrowth compartment to its left (if the split is to the left of that pregrowth compartment). The notable exceptions to these rules are when postgrowth compartments retain or gain (respectively) only half of the particles (on average) from its or a neighbour’s (respectively) pregrowth compartment. These events only affect two compartments per splitting event, but are nevertheless important.

The only way a postgrowth compartment retains half of its particles (on average) is if it is the compartment chosen to split. Similarly, the only way a postgrowth compartment gains half of the particles (on average) of a neighbouring compartment is if the pregrowth neighbour immediately to the left splits. All the postgrowth compartments, therefore, gain or retain half a compartment's worth of particles (on average) in two different ways. The only exceptions are the first and last postgrowth compartments. The first postgrowth compartment can only retain half of its particles (on average) if it is chosen to split. It can never gain half of the particles from another compartment as there are no pregrowth compartments to its left that can split. Similarly, the last postgrowth compartment can never retain half of its particles since it did not exist on the pregrowth domain. Instead it can only gain half of the particles from a neighbouring compartment when the pregrowth compartment in the final position splits.

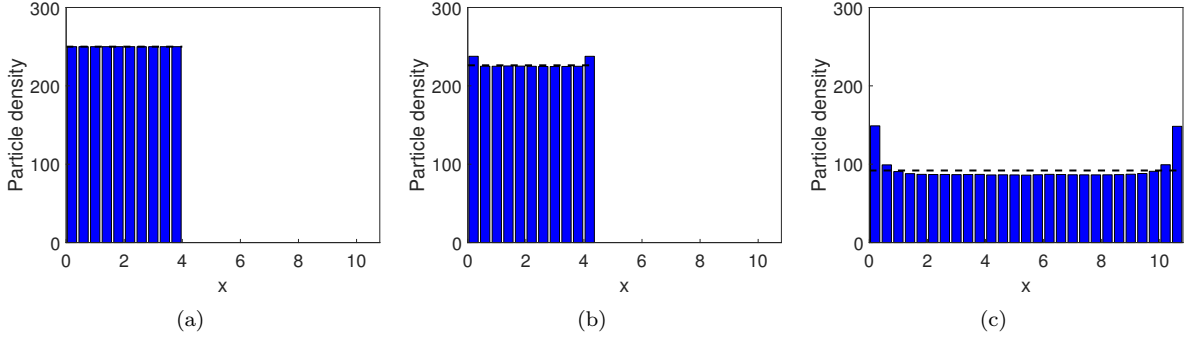
If particles are initially distributed uniformly, then an unbiased domain growth method will maintain this uniform distribution (on average, and provided no particles enter or leave the domain). For the method of Baker et al. [15] (see Figure 1 and Algorithm 1), splitting events will redistribute the particles into the postgrowth compartments which correspond to the pregrowth compartment chosen to split and its neighbour to the right. For example, in Figure 1, the particle redistribution event affects postgrowth compartments 5 and 6 when pregrowth compartment 5 is chosen to split. If these splitting events affected all compartments equally then the domain growth method would be unbiased and the particle profile would remain uniform as the domain grew. However, since the two end compartments suffer (on average) only half of the particle-reducing splitting events that the non-end compartments suffer, this leads to particles accumulating at the ends of the domains.

The build up of particles at either end of the domain has previously been hidden by the smoothing effect of fast diffusion [15]. If diffusion is large in comparison to the rate of domain growth, particles are able to diffuse away from the high concentration regions at the end of the domain, leading to a near-uniform particle profile (given a uniform initial condition and zero-flux boundary conditions).

We will illustrate the problems with the compartment-based domain growth method of Baker et al. [15] in three different ways. For the first we undertake a stochastic simulation of the original method using the Gillespie algorithm [9] and average particle densities over many repeats. For the second demonstration, we calculate the numerical solution of the mean-field equations that stem from the corresponding domain growth master equation of the stochastic process in the absence of diffusion. Thirdly, we employ an analytical mathematical argument based on local redistribution of particles to assess the particle distribution in the limit of large numbers of boxes. All code written in order to simulate these, and all other examples within this manuscript, can be found in the electronic supplementary material [17].

For the stochastic simulation, we initialise a total of 1000 particles uniformly in the domain  $[0, 4]$  which grows exponentially in time with rate  $\rho$  (we will use exponential growth as our primary test simulation, however other types of growth exhibit similar problems). We set the compartment width to be 0.4 so that initially we have 10 compartments. The growth rate,  $\rho$ , is set to be 0.01 (note that all units here are arbitrary in both space and time). We choose compartment splitting times to be deterministic, meaning that we pre-calculate the times at which a compartment should split (on average) and always enact a splitting event at those times, while the Gillespie algorithm handles the stochastic events between these pre-determined times. Alternatively, growth events could be incorporated stochastically as part of the Gillespie algorithm, but for the purposes of demonstrating the problem, and for ease of visualisation, we use deterministic growth in this exposition. The compartment to divide at each predetermined splitting time is chosen uniformly at random from amongst the current compartments. Diffusion is set to be 0, so that the only effect on compartment occupancy is domain growth. Using this domain growth configuration and averaging over 50,000 repeats, we see the clear build up of particles at both ends of the domain (see Figure 2).

To further corroborate these results and explain where this behaviour originates from, we consider the master equation for the splitting algorithm outlined above, which was first derived in [15]. Let



**Figure 2.** Three snapshots of the stochastic simulation which demonstrate the issue with the original domain growth method [15]. The blue bars denote particle densities in each compartment, and the black-dashed lines denote the expected behaviour of an unbiased growth method. Particle numbers are averaged over 50,000 repeats. All other parameter values are as in the text. (a) The initial configuration shows 1,000 particles spread uniformly across the initial domain  $[0, 4]$ . (b) At time 10, particle build up at the domain ends is already evident. (c) Once the system has evolved to time 100, particles have accumulated significantly close to the domain ends.

$p(\mathbf{m}, k, t) = \mathbb{P}(\mathbf{N}(t) = \mathbf{m}, K(t) = k)$  be the probability that the state variable (the number of particles in each compartment) at time  $t$  is  $\mathbf{m} = (m_1, \dots, m_k)^T$  and the number of compartments at time  $t$  is  $k$ . This quantity evolves according to:

$$\frac{dp}{dt}(\mathbf{m}, k, t) = \rho \sum_{j=1}^{k-1} \pi(m_j, m_{j+1} | m_j + m_{j+1}) p(G_j \mathbf{m}, k-1, t) - \rho k p(\mathbf{m}, k, t). \quad (1)$$

Here  $\rho$  is the splitting rate, which is the rate at which each compartment is chosen to divide,  $\pi(x, y | z)$  is a distribution describing the probability that, given there are  $z$  particles in a compartment before splitting, there are  $x$  and  $y$  particles in the two post-split compartments (where  $x + y = z$ ), and  $G_j : \mathbb{R}^k \rightarrow \mathbb{R}^{k-1}$  is an operator that combines the contents of compartments  $j$  and  $j + 1$  (the opposite process to splitting), so that

$$G_j : (m_1, \dots, m_j, m_{j+1}, \dots, m_k)^T \mapsto (m_1, \dots, m_j + m_{j+1}, \dots, m_k)^T.$$

Note that the growth considered for the master equation is stochastic in both position and timing.

We can calculate the mean-field equations for the evolution of the mean number of particles in each compartment, under the splitting probability corresponding to a symmetric binomial distribution

$$\pi(x, y | z) = \binom{z}{x} \left(\frac{1}{2}\right)^z,$$

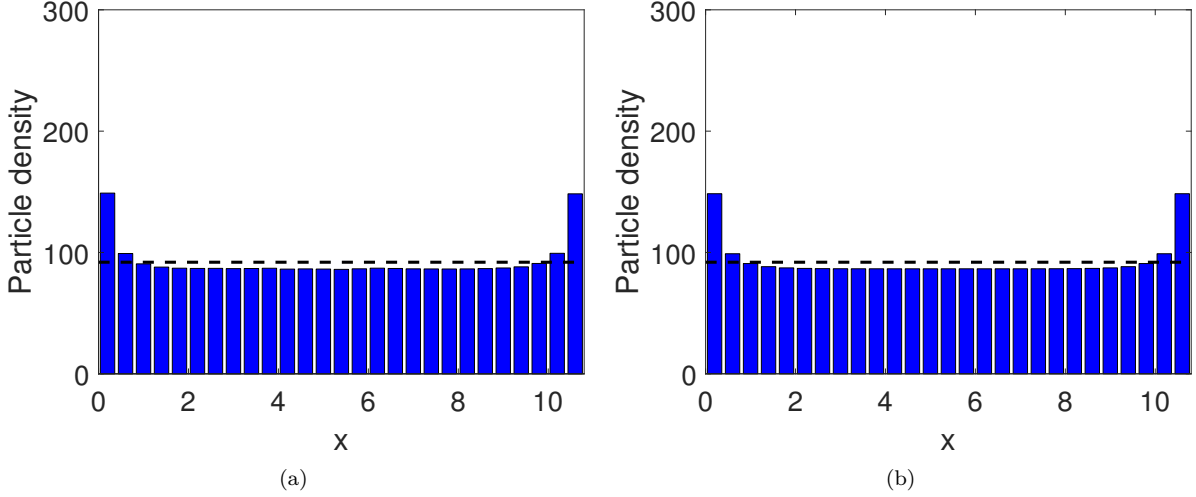
by multiplying both sides of equation (1) by  $m_i$  for each index  $i \in \{1, \dots, k\}$ , and summing over the entire postgrowth state variable. Define the quantity

$$M_i^k(t) = \sum_{\mathbf{m} \in \mathbb{N}_0^k} m_i p(\mathbf{m}, k, t),$$

which is the mean number of particles in compartment  $i$  at time  $t$ , given that there are  $k$  compartments in the system overall. These corresponding mean-field equations are given by:

$$\frac{dM_i^k}{dt}(t) = \rho \left[ \left(i - \frac{3}{2}\right) M_{i-1}^{k-1}(t) + \left(k - \frac{1}{2} - i\right) M_i^{k-1}(t) - k M_i^k(t) \right], \quad (2)$$

which holds for  $k \in \mathbb{N}$ ,  $j \in \{1, \dots, k\}$  and  $t > 0$  [15], noting that  $M_k^{k-1} \equiv 0$ . The full derivation is omitted here, however we direct the interested reader to Baker et al. [15], or Appendix D.1 for a similar calculation. We plot the solutions to equation (2) in order to illustrate the average behaviour of the system under the biased domain growth algorithm, in Figure 3. We show that in this case, we exhibit the same build up of particles at the boundaries that is evident in the stochastic simulation, shown in Figure 2.



**Figure 3.** A comparison between (a) the particle density found by using Algorithm 1, averaged over 50,000 repeats (reproduced from Figure 2(c)), and (b) the corresponding particle density found by solving mean-field equations (2). The mean-field equations are initialised in the same way as the stochastic algorithm (see Figure 2(a) and related caption).

Finally, we gain a more quantitative insight into the bias engendered by the domain growth method of Baker et al. [15] by using an analytical approach. In particular, we derive coefficients which describe the densities of particles in each compartment. We let  $u_i^k$  denote the normalised density of particles in compartment  $i$  when there are  $k$  compartments in total (a random variable),  $M_i^k = \mathbb{E}[u_i^k]$ , and  $I^k$  is the index of the compartment which splits when there are  $k$  compartments in total (chosen uniformly at random for each event). Finally, we define  $q$  to be a sample from some symmetric distribution with mean  $1/2$ . This will denote the random proportion of density that is placed in the left-hand postgrowth compartment when there are  $k$  compartments in total, and is independent of the particle density and the compartment chosen to split. We begin with all of the density in the first (and only) compartment, so that  $u_1^1 = 1$ . We will set up recursion relationships between the  $u_i^k$ 's for different  $i$  and  $k$  in order to approximate  $u_i^k$  for large  $k$ . Specifically, we express  $u_i^{k+1}$  in terms of  $u_j^k$  for  $j \in \{1, \dots, k\}$ :

$$u_i^{k+1} = \underbrace{u_i^k \mathbb{1}_{[I^k > i]}}_{\text{Compartment to right of } i \text{ splits}} + \underbrace{qu_i^k \mathbb{1}_{[I^k = i]}}_{\text{Compartment } i \text{ splits}} + \underbrace{(1-q)u_{i-1}^k \mathbb{1}_{[I^k = i-1]}}_{\text{Compartment } i-1 \text{ splits}} + \underbrace{u_{i-1}^k \mathbb{1}_{[I^k < i-1]}}_{\text{Compartment to left of } i-1 \text{ splits}}, \quad i \in \{2, \dots, k\}, \quad (3)$$

with similar expressions for  $i = 1$  and  $k + 1$ . Here  $\mathbb{1}_{[\text{condition}]}$  is the indicator function, which is unity when the subscripted condition is satisfied and zero otherwise.

Considering the first compartment, relationship (3) stipulates that

$$u_1^{k+1} = u_1^k \mathbb{1}_{[I^k > 1]} + qu_1^k \mathbb{1}_{[I^k = 1]}.$$

Taking expectations of this expression, noting that the choice of compartment, the density in each compartment and the proportion of the density placed in the left pregrowth compartment are independent (so that  $u_i^k$ ,  $I^k$  and  $q$  are independent for every  $i \in \{1, \dots, k\}$ ) we find that:

$$\begin{aligned} M_1^{k+1} &= M_1^k \mathbb{P}(I^k > 1) + \frac{1}{2} M_1^k \mathbb{P}(I^k = 1), \\ &= M_1^k \left( \frac{k-1}{k} + \frac{1/2}{k} \right), \\ &= M_1^k \left( 1 - \frac{1/2}{k} \right). \end{aligned}$$

Applying this recursive relation  $k$  times:

$$M_1^{k+1} = M_1^1 \prod_{n=1}^k \left( 1 - \frac{1/2}{n} \right) = \prod_{n=1}^k \left( 1 - \frac{1/2}{n} \right), \quad (4)$$

since  $M_1^1 = \mathbb{E}[u_1^1] = 1$ . In order to simplify this expression, we can use the following relationship, which is derived in Appendix A:

$$\prod_{n=a}^b \left( 1 + \frac{c}{n} \right) \approx \frac{\Gamma(a)}{\Gamma(a+c)} e^{bc}, \quad (5)$$

where  $a, b \in \mathbb{N}$  such that  $a < b$  and  $b$  is large, and  $c \in \mathbb{R}$ . Applying approximation (5) to equation (4) for large  $k$  gives:

$$M_1^{k+1} = \prod_{n=1}^k \left( 1 - \frac{1/2}{n} \right) \approx \frac{\Gamma(1)}{\Gamma(1/2)e^{-1/2}} k^{-1/2} = \frac{1}{\sqrt{\pi}e^{-1/2}} k^{-1/2}. \quad (6)$$

Now consider the second compartment. As with the first, we write the recursion relation (3):

$$u_2^{k+1} = u_2^k \mathbb{1}_{[I^k > 2]} + \frac{1}{2} u_2^k \mathbb{1}_{[I^k = 2]} + \frac{1}{2} u_1^k \mathbb{1}_{[I^k = 1]}.$$

Once again, we take expectations and simplify by applying relation (3) recursively (as in equation (4)):

$$\begin{aligned} M_2^{k+1} &= M_2^k \mathbb{P}(I^k > 2) + \frac{1}{2} M_2^k \mathbb{P}(I^k = 2) + \frac{1}{2} M_1^k \mathbb{P}(I^k = 1), \\ &\vdots \\ &= \underbrace{M_2^2 \prod_{n=2}^k \left( 1 - \frac{3/2}{n} \right)}_{\text{Term 1}} + \underbrace{\sum_{n=2}^{k-1} \left( \prod_{j=n+1}^k \left( 1 - \frac{3/2}{j} \right) \right) M_1^n \left( \frac{1/2}{n} \right)}_{\text{Term 2}} + \underbrace{M_1^k \left( \frac{1/2}{k} \right)}_{\text{Term 3}}. \end{aligned} \quad (7)$$

We consider each of the three terms in equation (7) sequentially. Using equation (5), we can approximate term 1, for large  $k$ , as

$$\prod_{n=2}^k \left( 1 - \frac{3/2}{n} \right) \approx \frac{\Gamma(2)}{\Gamma(1/2)e^{-3/2}} k^{-3/2}. \quad (8)$$

It can be shown (see Appendix B) that for large  $k$ , term 2 can be approximated by

$$\sum_{n=2}^{k-1} \left( \prod_{j=n+1}^k \left( 1 - \frac{3/2}{j} \right) \right) M_1^n \left( \frac{1/2}{n} \right) \approx \frac{1}{2} \frac{1}{\sqrt{\pi}e^{-1/2}} k^{-1/2} - \frac{1}{\sqrt{\pi}e^{-1/2}} k^{-3/2}. \quad (9)$$



To simplify term 3 we make use of result (6):

$$M_1^k \left( \frac{1/2}{k} \right) \approx \frac{1}{2} \frac{1}{\sqrt{\pi}e^{-1/2}} k^{-3/2} \quad (10)$$

Substituting the resultant expressions (8), (9) and (10) into (7) gives, for large  $k$ :

$$M_2^{k+1} \approx M_2^2 \frac{\Gamma(2)}{\Gamma(1/2)e^{-3/2}} k^{-3/2} + \frac{1}{2} \frac{1}{\sqrt{\pi}e^{-1/2}} k^{-1/2} - \frac{1}{\sqrt{\pi}e^{-1/2}} k^{-3/2} + \frac{1}{2} \frac{1}{\sqrt{\pi}e^{-1/2}} k^{-3/2},$$

For large  $k$ , the  $O(k^{-1/2})$  terms dominate leaving us with the following approximation:

$$M_2^{k+1} \approx \frac{1}{2} \frac{1}{\sqrt{\pi}e^{-1/2}} k^{-1/2}. \quad (11)$$

Following the same procedure, we can find the approximate expressions for the asymptotic particle density in each of the compartments. In particular, it can be shown that

$$M_i^k \approx c_i k^{-1/2}, \quad (12)$$

where

$$c_1 = \frac{1}{\sqrt{\pi}e^{-1/2}},$$

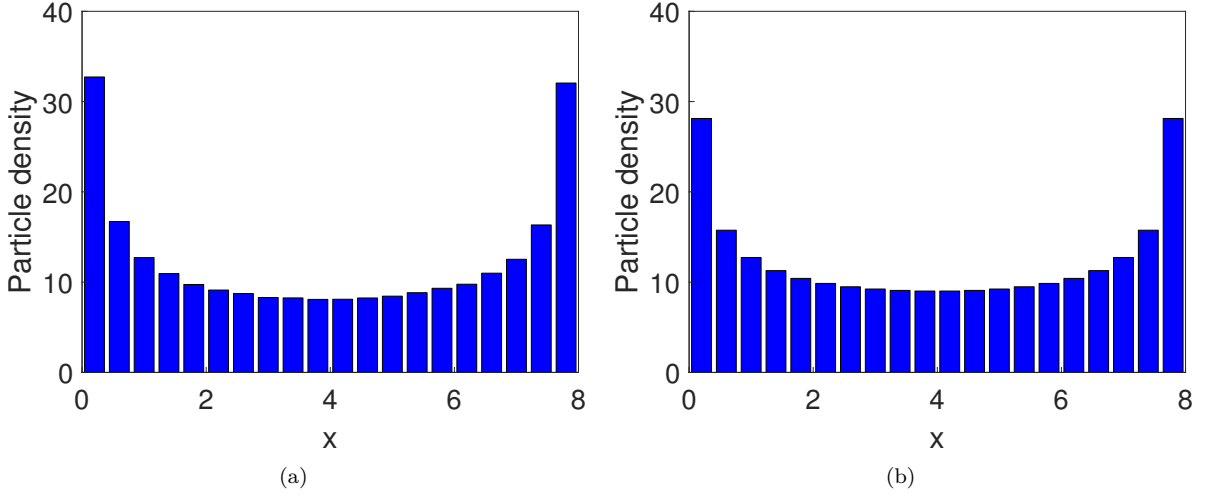
$$c_i = c_1 \prod_{j=2}^i \frac{2j-3}{2j-2}, \quad i \in \{2, 3, \dots\}.$$

To assess the accuracy of this approximation, we undertake a stochastic simulation initialised with a single compartment containing 100 particles. At time  $t = 100$ , under our time-deterministic splitting mechanism each simulation finishes with 20 compartments. We then compare the particle numbers in each compartment, averaged over 10,000 repeats, to equation (12). Since the simulation domain has only finitely many compartments but our mathematical analysis considers an infinite number of compartments, we average  $M_i^{20}$  and  $M_{21-i}^{20}$  when we plot compartment  $i$  under the assumption that densities are initialised, and subsequently remain, symmetric. Finally, we scale each of the plots so that they have the same number of particles. Although a quantitative agreement is not expected, since our results hold strictly only on an infinite domain, the results in Figure 4 demonstrate that our mathematical analysis matches the simulation results qualitatively.

The original domain growth method proposed by Baker et al. [15] has been used in many compartment-based studies of domain growth [18, 19, 20, 3]. Despite not having previously been evident, we have been able to demonstrate in three distinct ways, that this domain growth method is biased. The consequence of this bias is that particles tends to accumulate at the extreme ends of the domain. In the next section, we introduce the stretching method, which prevents this build up of particles and gives genuinely uniform, unbiased domain growth.

### 3 Stretching method

We now introduce the stretching method. This differs from the original method because it is a global method as opposed to a local one. That is, instead of choosing a single compartment to instantaneously grow to twice its length and split, we stretch all compartments by a small amount and redistribute particles amongst all compartments (for a brief discussion of local growth mechanisms, please see Appendix C). We will firstly explain the method, before demonstrating its effectiveness. We do this by showing that it



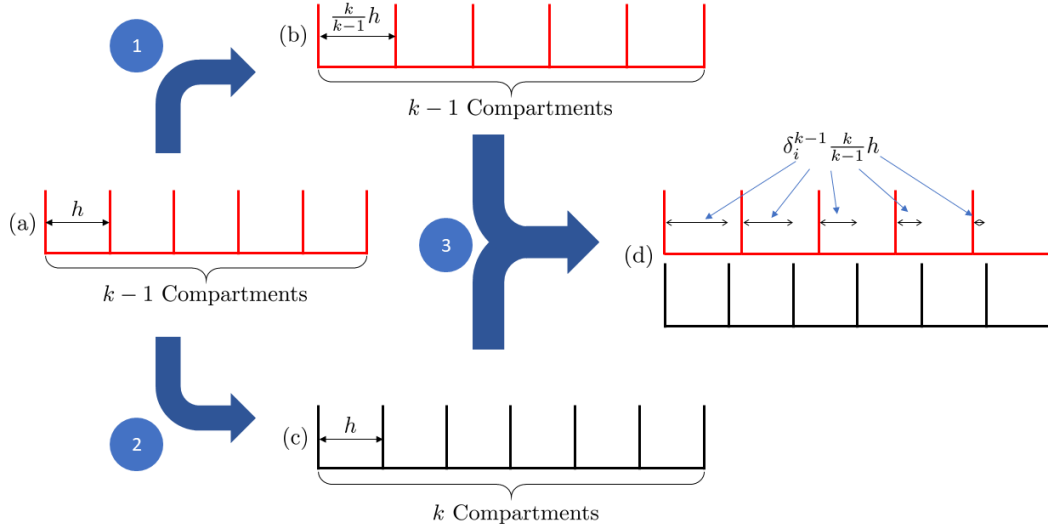
**Figure 4.** A comparison between (a) a stochastic simulation averaged over 10,000 repeats and initialised with a single compartment and (b) the result of the mathematical argument, where the displayed compartment  $i$  is given by the average of  $M_i^{20}$  and  $M_{21-i}^{20}$ , calculated using equation (12). Both plots have the same total area.

can correctly maintain a uniform particle profile on a uniformly growing domain. We then look at a case study, the formation of a morphogen gradient on an exponentially growing domain, in order to directly compare the original and stretching methods using an example with its roots in biology (see Section 3.1). Finally, in Section 3.2, we investigate the parameter regimes in which the spatial inhomogeneities in the original method are negligible.

We begin by describing the method. Assume the number of compartments before a growth event is  $k - 1$  for some  $k > 2$ , and define each compartment to be of width  $h$  (see Figure 5(a)). We will define the state variable before growth to be  $\mathbf{r} \in \mathbb{N}_0^{k-1}$  and after growth to be  $\mathbf{m} \in \mathbb{N}_0^k$  in order to be consistent with the original method. The method proceeds as follows:

1. When a growth event is chosen to occur, we stretch the domain to be of size  $kh$  rather than  $(k-1)h$  (see Figure 5(b)). We do this uniformly across the entirety of the domain, so that each compartment on the stretched domain is now of width  $kh/(k-1)$ .
2. In the second step we add a compartment to the right-hand end of the pre-stretched domain (see Figure 5(c)). It is on this postgrowth domain that we define the state variable  $\mathbf{m}$ . Note that we now have two domains, each with a different number of compartments, but both of the same length.
3. For the third step, we compare the two meshes. Note that for every stretched compartment, exactly two of the postgrowth compartments intersect it (see Figure 5(d)). Assuming particles are uniformly distributed across each stretched compartment, we can calculate the proportion of particles,  $\delta_i^{k-1}$ , that should be placed in the left overlapping compartment. If we denote the right-hand end of compartment  $i$  in the renormalised domain as  $x_i = ih$  and use  $\tilde{x}_i = [ki/(k-1)]h$  the quantity for the stretched domain, then:

$$\begin{aligned} \delta_i^{k-1} \frac{k}{k-1} h &= x_i - \tilde{x}_{i-1} \\ &= ih - \frac{k}{k-1} ih + \frac{k}{k-1} h \end{aligned}$$



**Figure 5.** The process of domain growth for the stretching method. We start with  $k - 1$  compartments (a), and in step 1, stretch each of these compartments to be  $k/(k - 1)$  times their original length (b). This has the effect of increasing the domain length by  $h$ , a compartment's width. In step 2, we add a new compartment to the original  $k - 1$  compartments of size  $h$  (c), which again yields a domain that is a compartment's width bigger. In step 3, we redistribute the particles in the stretched  $k - 1$  compartments (c) into the  $k$  compartments of size  $h$  (b) by calculating the  $\delta_i^{k-1}$  values, which tell us how much overlap there is between the two meshes (d). These  $\delta_i^{k-1}$  values are treated as the probability for each particle in a stretched compartment to move to the renormalised compartment which overlaps its left-hand boundary. If a particle does not move to the renormalised compartment which overlaps with its left hand boundary then it moves to the renormalised compartment which overlaps with its right hand boundary.

$$\begin{aligned} &= \frac{k-i}{k-1}h. \\ \Rightarrow \delta_i^{k-1} &= \frac{k-i}{k}. \end{aligned} \quad (13)$$

4. Finally we calculate the new state  $\mathbf{m}$  by drawing  $k-1$  binomial random variables  $b_i \sim \text{Bin}(r_i, \delta_i^{k-1})$  and calculating  $\mathbf{m}$  as:

$$\mathbf{m} = \sum_{i=1}^{k-1} b_i \mathbf{e}_i + (r_i - b_i) \mathbf{e}_{i+1},$$

where the  $\mathbf{e}_i$  are the standard  $k$ -dimensional basis vectors.

**Algorithm 2: The stretching method**

- (2a)** Initialise time  $t = 0$  and set the final time  $T > 0$ . Initialise the number of compartments,  $k$ , and the particle numbers in each compartment  $m_i$ ,  $i \in \{1, \dots, k\}$ . Specify the size of compartment  $h$ , the jump rate  $d = D/h^2$  and the growth rate  $\rho$ .
- (2b)** Calculate the propensity functions  $\alpha_i^L = dm_i$ ,  $\alpha_i^R = dm_i$  for left and right jumps from each compartment for  $i = 1, \dots, k$  with  $\alpha_1^L = 0 = \alpha_k^R$ , and set the propensity function for a growth event to be  $\alpha^G = \rho k$ . Calculate the sum of the propensity functions:

$$\alpha_0 = \sum_{i=1}^k (\alpha_i^L + \alpha_i^R) + \alpha^G.$$

- (2c)** Calculate the time until the next event by firstly drawing  $u_1 \sim \text{Unif}(0, 1)$  and setting  $\tau = 1/\alpha_0 \log(1/u_1)$ . Set the time to be the next event that occurs  $t \leftarrow t + \tau$ .
- (2d)** Determine which event is next to occur by choosing one at random with probability proportional to the propensity function.
1. If the event corresponds to a left (resp. right) jump event from box  $i$ , remove a particle from compartment  $i$  and add one to compartment  $i - 1$  (resp.  $i + 1$ ).
  2. If the event corresponds to a growth event: firstly, define the pregrowth state to be  $\mathbf{r} \leftarrow \mathbf{m}$ , calculate the overlap proportions  $\delta_i^k = (k + 1 - i)/(k + 1)$ , and use these in order to draw  $k$  binomial random variables  $b_i \sim \text{Bin}(m_i, \delta_i^k)$ . Create an extra compartment at the right end of the postgrowth domain (increasing  $k$  by 1 by setting  $k \leftarrow k + 1$ ). Then set, for  $j \in \{1, \dots, k\}$ :

$$m_j = \begin{cases} b_1, & \text{if } j = 1, \\ b_j + (r_{j-1} - b_{j-1}), & \text{if } j \in \{2, \dots, k - 1\}, \\ r_{k-1} - b_{k-1}, & \text{if } j = k. \end{cases}$$

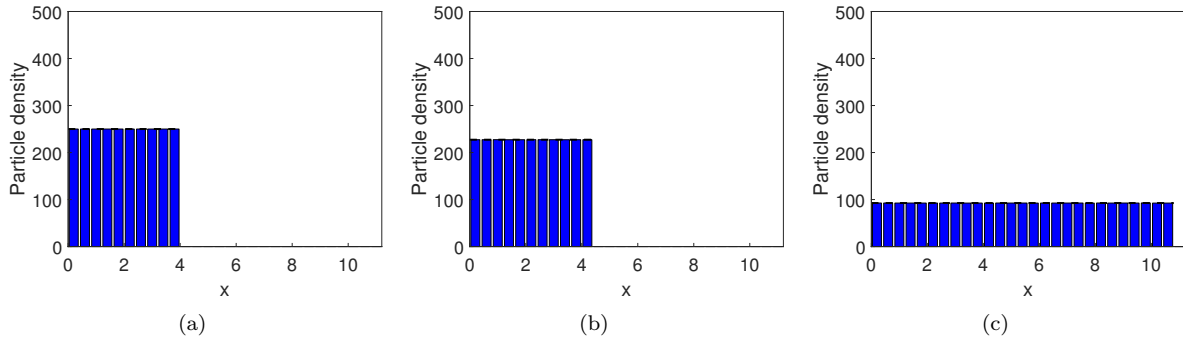
- (2e)** If  $t < T$  then return to step **(2b)**, else end.

We assess the stretching method by initialising a uniform profile and test to see whether uniformity is maintained under the stretching domain growth method. In Figure 6, we can see that the stretching method performs very well in comparison to the original method.

### 3.1 Case study: Morphogen gradient

For our case study, we apply the original and stretching methods to the formation of a morphogen gradient on an exponentially growing domain [21, 22]. We begin with an initial domain of length  $L_0$ , which grows with rate  $\rho$ , and on which particles with density  $u(x, t)$  move and interact. Particles diffuse with diffusion coefficient  $D$ , and they decay uniformly at a rate  $\mu$ . There is also an influx of particles at the left-hand boundary, at rate  $D\lambda$ . We will only allow particles to jump between adjacent compartments and to interact within their own compartments. In order to compare the results of the stochastic simulation, we compute the solution to the associated mean-field PDE, which is given by:

$$\begin{aligned} \text{PDE : } \frac{\partial u}{\partial t} &= D \frac{\partial^2 u}{\partial x^2} - \frac{\partial(\rho x u)}{\partial x} - \mu u, & x \in (0, L_0 \exp(\rho t)), \quad t > 0, \\ \text{BC : } \frac{\partial u}{\partial x} \Big|_{x=0} &= -\lambda, \quad \frac{\partial u}{\partial x} \Big|_{x=L_0 \exp(\rho t)} = 0, & t > 0, \end{aligned} \tag{14}$$



**Figure 6.** Three snapshots of the particle density on a domain growing according to the stretching method (a) initially, at time 0, (b) at time 10 and (c) at the final time 100. All descriptions and parameter values are the same as in Figure 2.

$$\text{IC} : u(x, 0) = 0,$$

$$x \in [0, L_0].$$

The PDE comprises three terms on the right-hand side — the first is the diffusive term, the second is dilution due to domain growth and the third is degradation of particles over the spatial domain [5]. The first boundary condition is the in-flux of particles at the left of the domain, while the second boundary condition is a reflective boundary. For our on-lattice simulations, diffusion is implemented by allowing particles to jump between neighbouring compartments with rate  $d = D/h^2$ . Particle degradation is achieved by allowing each particle in a compartment to be removed with rate  $\mu$ , while production is included through a production reaction in the compartment adjacent to the left-hand boundary, as is described by Taylor et al. [7].

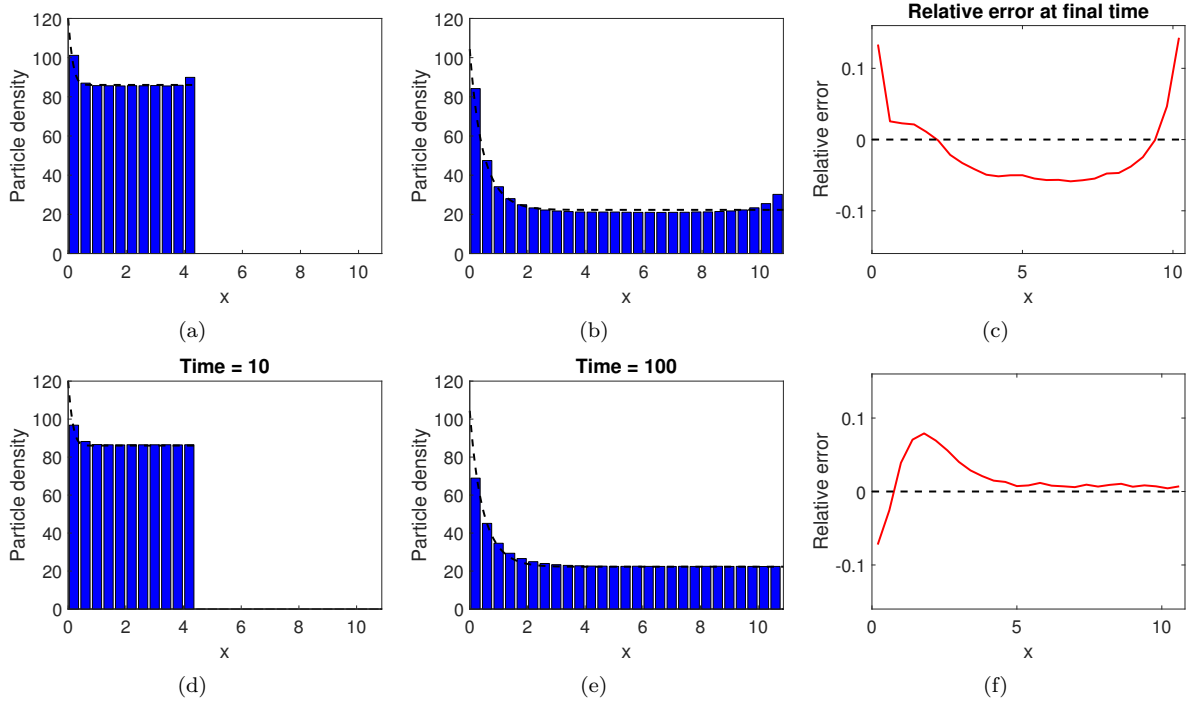
We simulate this system with exponential growth rate  $\rho = 0.01$  on a domain of initial length  $L_0 = 4$ . We set the diffusion coefficient to be  $D = 0.0025$ , the influx rate is specified by setting  $\lambda = 200$  and the degradation rate  $\mu = 0.005$ . We simulate until a final time,  $t = 100$  and average over 50,000 repeats. The results are displayed in Figure 7.

As with the case of maintaining a uniform gradient, particle densities for the stretching method agree well with the associated mean-field PDE [15] (see Figures 7(d)-7(f)), however we still observe the same collection of particles at the boundaries for the original method (see Figures 7(a)-7(c)). This is particularly evident at the right-hand side of the domain, indicating that we are able to correctly simulate a reaction-diffusion system which incorporates first-order reactions using the stretching method. We anticipate that the extension to second- and higher-order reactions will yield similar results since the domain growth mechanisms is decoupled from the reaction mechanism.

### 3.2 Comparison of methods

In this section, we will investigate the two methods, and the parameter regimes in which the errors from the original method are acceptable due to the interplay between diffusion, the growth rate and initial domain length. We explore this in two ways. The first is through a heuristic argument.

The mean squared displacement describes how the variance in position of a Brownian particle changes in time. If multiple particles are initialised at the origin and diffuse for a time,  $t$ , then  $\langle x^2 \rangle = 2Dt$ , where the angled brackets denote an ensemble average of the squared distances from the origin. If a particle is to diffuse over the entire domain before the domain grows, then the squared distance from the origin would be  $x^2 \sim L_0^2$ . Likewise, the typical time frame for growth is given by setting  $t \sim 1/\rho$ . Substituting these into the expression for the mean squared displacement yields  $D \sim L_0^2\rho$ . Therefore, we say we are



**Figure 7.** Two simulations of a morphogen gradient formation using (a)-(c) the original method and (d)-(f) the stretching method. In the first two columns, the blue bars represent the ensemble average of the stochastic algorithm over 50,000 repeats (a), (d) at time 10 and (b), (e) at time 100, while the black dashed line is the numerical solution to the PDE (14). In the final column we plot the relative errors between the on-lattice method and the PDE. All parameters are as in the text.

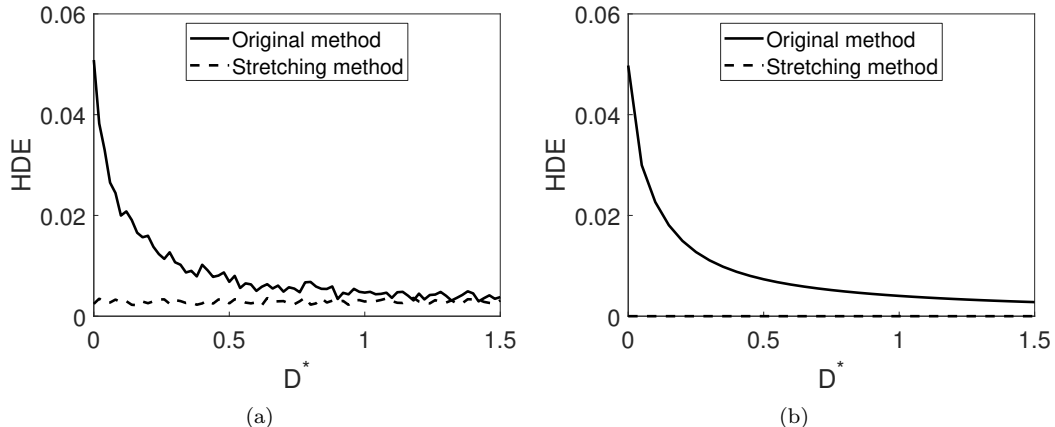
in a ‘high diffusion’ parameter regime when  $D > L_0^2 \rho$ . When in the low diffusion regime ( $D < L_0^2 \rho$ ), particles are unable to spread and equilibrate before the domain grows, which is when we see the build up of particles in the original method.

We verify this heuristic result with a second argument. In order to do this, we have simulated a non-dimensionalised stochastic system and compared it to the solution of the mean-field continuum diffusion equation in order to determine a threshold value for diffusion. The results can be seen in Figure 8, where we plot the histogram distance error (HDE), which is given by the expression

$$\text{HDE}(D^*) = \frac{1}{2} \sum_{n=1}^N |p_n^s(D^*) - p_n^m(D^*)|. \quad (15)$$

Here,  $p_n^s(D^*)$  is the value of the normalised solution of the stochastic simulation (original or stretching method) at the final time with non-dimensional diffusion  $D^*$ ,  $p_n^m(D^*)$  is the solution for the associated non-dimensional mean-field PDE, and  $n$  indexes a common mesh on which we compare the two solutions. The non-dimensional diffusion parameter is equal to  $D^* = D/(L_0^2 \rho)$  where, as before,  $L_0$  is the initial domain length and  $\rho$  is the exponential growth rate. From Figure 8(a), a value  $D^*$  greater than 1 yields similar HDE values for both the original and stretching methods. This indicates that, if  $D > L_0^2 \rho$  then the original method should have a similar performance to the stretching method. However, if the inequality is reversed, so that  $D < L_0^2 \rho$ , then the stretching method should be used. We also note that the same

pattern is apparent when comparing the solutions of the mean equations (2) and (24), which can be seen in Figure 8(b). The HDE for the stretching method is exactly zero because it maintains uniformity precisely in the mean field.



**Figure 8.** (a) HDEs for stochastic simulations of a non-dimensional diffusion mechanism on a growing domain, and their corresponding histogram distance errors (HDEs) for varying diffusion values. The solid line corresponds to the original method [15], while the dashed line denotes the stretching method. Each simulation is averaged over 5000 repeats. (b) A similar plot, but using the solutions of the mean equations (2) and (24). Note that the dotted line in this case is on top of the horizontal axis since there is no stochastic error in the mean equations and no bias in the stretching domain growth method.

## 4 Discussion

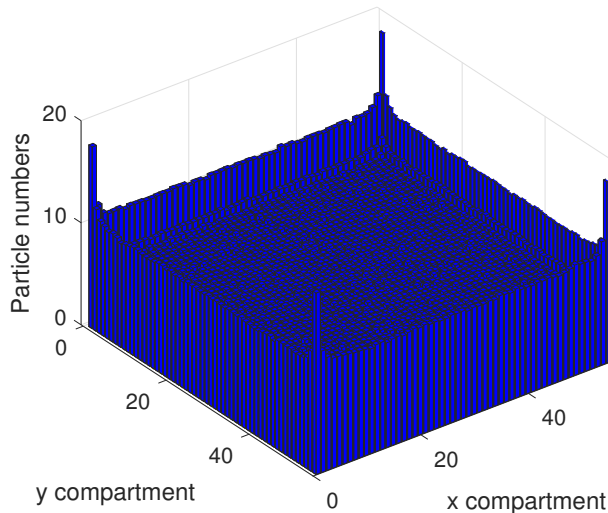
Domain growth mechanisms for on-lattice models are of importance for the accurate representation of many biological processes. We have demonstrated beyond doubt that the original method, suggested by Baker et al. [15], causes a build up of particles at the boundaries of the spatial domain (see Figure 2(c)). Consequently, we have developed a method for implementing domain growth when modelling reaction-diffusion systems at the mesoscale in order to correct this build-up. This technique involves stretching all compartments by a small amount (leading to the creation of a new compartment) and the appropriate re-distribution of the particles. We have demonstrated that this method agrees with the corresponding mean-field equations derived in the continuum limit, while maintaining a uniform profile, and have shown that it correctly models morphogen gradient formation on a growing domain.

The stretching method will be particularly useful when developing spatially extended hybrid methods on growing domains. These methods split the spatial domain into subdomains, in which different modelling paradigms are used, separated by an interface or overlap region [23]. We envisage that the stretching method can be used in the compartment-based subdomain of a hybrid model for reaction-diffusion on a growing domain, without causing a build-up of particles at the interface.

There are still several open questions regarding on-lattice domain growth whose answers go beyond the scope of this paper. The first of these concerns modelling domain growth in higher dimensions. To induce on-lattice domain growth in higher dimensions we can employ the following method by Binder et al. [24]. Consider, for the purposes of this example, a two-dimensional domain (although higher dimensional growth is straightforward to implemented by analogy). In order to maintain a rectangular domain, a “growth event” must increase the total number of rows or columns by one. For example, when a growth event is chosen to occur in the vertical direction, we must increase in the number of rows. To do this,

we temporarily treat each column as its own one-dimensional domain, and implement a single vertical growth event in each column, independent of the others. Doing this for every column results in the whole domain increasing in height by a single row.

We have simulated such a domain growth process using the original method of Baker et al. [15] to implement the independent row or column elongations when carrying out a horizontal or vertical (respectively) growth event. Specifically, for clarity, we carry out horizontal and vertical growth events simultaneously, which maintains the aspect ratio of the initially square domain we begin with. Diffusion of particles is turned off in order to clearly demonstrate the bias induced by this two-dimensional version of domain growth as illustrated in Figure 9. The same effect that we have observed in one dimension (namely a preponderance of particles towards the boundaries of the domain) is also present in higher dimensions. Extending the stretching method will provide a straightforward fix to this problem in higher dimensions. Overlap fractions will be calculated as ratios of (hyper-)volumes as opposed to ratios of lengths, and particles in a pregrowth compartment will be distributed between multiple overlapping postgrowth compartments using multinomial distributions (the natural generalisation of the binomial distributions used in the one-dimensional case).



**Figure 9.** A simulation of the domain growth method of Baker et al. [15] extended to two dimensions. The method of domain growth is described in the text. The domain contains 1,000 particles, and particle redistribution is due to domain growth events alone as diffusion is set to zero. The growth rate is  $\rho = 0.01$ , and the particle densities are averaged over 10,000 independent repeats.

Whilst domain growth is important, equally, domain shrinkage is also of significant biological interest. Domain shrinkage through directed apoptosis (programmed cell death) is an important component of many wound healing processes [25, 26] for example. Without further investigation it is not immediately clear whether domain shrinkage implemented by the removal of a randomly chosen compartment (in analogy to the domain *growth* method of Baker et al. [15] and as implemented by Yates [3]) would induce bias in cell densities. In contrast we are confident that implementing domain shrinkage into the stretching method by considering pre- and post-shrink overlap regions compartments will not induce bias. Nevertheless these hypotheses remain to be tested.

Finally, it is of interest to adapt the stretching method to account for non-uniform growth, which has been shown to be important in biological scenarios [27]. All of the examples we have presented have implemented uniform growth — in which all regions of space grow at the same rate. The stretching method could be adapted to incorporate non-uniform growth by splitting the domain into groups of



compartments each of which have a different growth rate. The stretching method can then be used on each of the groups individually.

Many authors have used the method presented by Baker et al. [15] to incorporate growth into on-lattice simulations of reaction-diffusion processes. For example, Woolley et al. [18] investigate the role that domain growth plays on modelling stochastic reaction-diffusion systems. Thompson et al. [19] explore cell migration and adhesion during biological development, while tissue growth and shrinkage are studied by Yates [3]. However, as demonstrated in this paper, particularly in the case of low diffusivity, the inherent bias in the domain growth method suggests that the conclusions drawn from these studies may require re-evaluation. We suggest that the ‘stretching’ domain growth method we propose in this paper is an appropriate alternative with which to re-evaluate these results and which should be employed in future studies of reaction-diffusion processes on growing domains.

## Acknowledgements

CAS is supported by a scholarship from the EPSRC Centre for Doctoral Training in Statistical Applied Mathematics at Bath (SAMBa), under the project EP/L015684/1. CM is grateful to EPSRC for support through the fellowship EP/R022186/1.

# Appendices

## A Deriving equation (5)

Within this appendix, we derive the expression in equation (5):

$$\prod_{n=a}^b \left(1 + \frac{c}{n}\right) \approx \frac{\Gamma(a)}{\Gamma(a+c)} b^c, \quad (5)$$

which holds for  $a, b \in \mathbb{N}$  such that  $a < b$  and  $b$  is large, and for  $c \in \mathbb{R}$ . We start by representing the product on the left-hand side of equation (5) using gamma functions:

$$\begin{aligned} \prod_{n=a}^b \left(1 + \frac{c}{n}\right) &= \prod_{n=a}^b \left(\frac{n+c}{n}\right) \\ &= \frac{\Gamma(a)\Gamma(b+c+1)}{\Gamma(a+c)\Gamma(b+1)}. \end{aligned} \quad (A1)$$

We apply Stirling’s approximation, which says that for large  $z$ ,  $\Gamma(z) \approx \sqrt{2\pi(z-1)}(z-1)^{z-1}e^{-(z-1)}$ . Applying this to equation (A1), we obtain:

$$\begin{aligned} \prod_{n=a}^b \left(1 + \frac{c}{n}\right) &\approx \frac{\Gamma(a)}{\Gamma(a+c)} \frac{\sqrt{2\pi(b+c)}(b+c)^{b+c}}{\sqrt{2\pi b} b^b} \frac{e^b}{e^{b+c}} \\ &= \frac{\Gamma(a)}{\Gamma(a+c)} \sqrt{\frac{b+c}{b}} \left(\frac{b+c}{b}\right)^b (b+c)^c e^{-c}. \end{aligned}$$

We assume that  $b$  is large, and thus,  $b+c \approx b$ . Applying this:

$$\prod_{n=a}^b \left(1 + \frac{c}{n}\right) \approx \frac{\Gamma(a)}{\Gamma(a+c)} \sqrt{\frac{b}{b}} \left(\frac{b}{b}\right)^b b^c e^{-c}$$

$$= \frac{\Gamma(a)}{\Gamma(a+c)} e^c b^c.$$

## B Deriving equation (9)

Within this appendix, we derive the expression in equation (9):

$$\sum_{n=2}^{k-1} \left( \prod_{j=n+1}^k \left( 1 - \frac{3/2}{j} \right) \right) M_1^n \left( \frac{1/2}{n} \right) \approx \frac{1}{2} \frac{1}{\sqrt{\pi} e^{-1/2}} k^{-1/2} - \frac{1}{\sqrt{\pi} e^{-1/2}} k^{-3/2}, \quad (9)$$

where the approximation is for large values of  $k$ .

$$\begin{aligned} \sum_{n=2}^{k-1} \left( \prod_{j=n+1}^k \left( 1 - \frac{3/2}{j} \right) \right) M_1^n \left( \frac{1/2}{n} \right) &= \sum_{n=2}^{k-1} \left( \prod_{j=1}^k \left( 1 - \frac{3/2}{j} \right) \right) \left( \prod_{j=1}^n \left( 1 - \frac{3/2}{j} \right) \right)^{-1} M_1^n \left( \frac{1/2}{n} \right) \\ &= \left( \prod_{j=1}^k \left( 1 - \frac{3/2}{j} \right) \right) \sum_{n=2}^{k-1} \left( \prod_{j=1}^n \left( 1 - \frac{3/2}{j} \right) \right)^{-1} M_1^n \left( \frac{1/2}{n} \right) \\ &\approx k^{-3/2} \sum_{n=2}^{k-1} \left( n^{-3/2} \right)^{-1} M_1^n \left( \frac{1/2}{n} \right), \end{aligned}$$

where, in the final approximation, we have used equation (5) in order to simplify the two products. Simplifying yields:

$$\sum_{n=2}^{k-1} \left( \prod_{j=n+1}^k \left( 1 - \frac{3/2}{j} \right) \right) M_1^n \left( \frac{1/2}{n} \right) \approx k^{-3/2} \sum_{n=2}^{k-1} n^{3/2} M_1^n \left( \frac{1/2}{n} \right).$$

We now utilise equation (6) (approximation is better for large  $k$ ), in order to obtain:

$$\begin{aligned} \sum_{n=2}^{k-1} \left( \prod_{j=n+1}^k \left( 1 - \frac{3/2}{j} \right) \right) M_1^n \left( \frac{1/2}{n} \right) &\approx k^{-3/2} \sum_{n=2}^{k-1} n^{3/2} \frac{1}{\sqrt{\pi} e^{-1/2}} n^{-1/2} \frac{1}{2} n^{-1} \\ &= k^{-3/2} \frac{1}{\sqrt{\pi} e^{-1/2}} \sum_{n=2}^{k-1} \frac{1}{2} \\ &= k^{-3/2} \frac{1}{\sqrt{\pi} e^{-1/2}} \frac{1}{2} (k-2) \\ &= \frac{1}{2} \frac{1}{\sqrt{\pi} e^{-1/2}} k^{-1/2} - \frac{1}{\sqrt{\pi} e^{-1/2}} k^{-3/2}. \end{aligned}$$

## C Discussion on local methods

Throughout Section 2, we have demonstrated that at low diffusion levels, the method presented by Baker et al. [15] fails to correctly model the growth of a uniform profile. This is a local method, which means that on every occasion that the domain is due to grow, we choose a compartment at random at which

the growth event occurs. Over many repeats of the same process, different compartments will be chosen and so, when averaging over these repeats, each compartment is chosen an equal number of times (when considering only a single growth event).

There are three elements that define a local method:

1. The probability of choosing each compartment,
2. The direction of growth,
3. The redistribution of particles.

In the case of Baker et al. [15], we (1) choose each compartment uniformly at random, (2) always grow to the right and (3) redistribute the particles using a symmetric binomial distribution. We now discuss which of these we can change in order to create a local method that correctly maintains a uniform profile on a growing domain.

The simplest way of creating a different local method is to change one of the three elements. However, it can be shown that changing only a single element does not yield the expected uniform growth. As a result, the next simplest is to change two of the elements, whilst fixing one. One such way would be to (1) set the probability of choosing each compartment to be general, (2) set the direction of growth to be left or right with a probability of a half each, and (3) redistribute the particles using a binomial distribution with a generic probability of success. These properties yield an algorithm which has  $k$  degrees of freedom (where there are  $k - 1$  pregrowth compartments), which can be used in order to solve a series of equations to ensure that a uniform profile is maintained.

In order to calculate the values for the probability distribution, and the probability for success in the binomial distribution when redistributing particles post-split, we can write a series of equations that relate the pre- and postgrowth states, on average. Using these, we are able to use numerical optimisation techniques in order to find the optimum values for the unknown parameters. However, while this method fixes the main issue with the original method, it introduces some new ones. Firstly, the centre of growth is no longer fixed at one end of the domain. This may cause problems for certain methodological applications (e.g. employing the method for spatial hybrid methods [28, 29, 30]) or for some biological applications in which tissues genuinely grow from a fixed origin at one end of the domain grow from one end of the domain (e.g. hyphal tip growth [31]). Secondly, we have to solve an overdetermined system for every possible number of compartments that might occur. This can be computationally expensive, especially for large compartment numbers. In practice, the unknown parameters can be computed and stored *a priori*, although, if the timing of domain growth events is stochastic, it may not be clear in advance exactly how many compartments the domain will comprise.

We also note that we have changed the direction of growth from being always to the right in the original method, to being left or right with equal probability. This is because there is no probability distribution for the compartments together with redistribution probability that maintains a uniform profile when the direction of growth is always the same.

## D Justifying the stretching method

In Section 3, we introduced the stretching method as an alternative to the original method presented by Baker et al. [15]. In order to demonstrate that the original method fails, we have simulated the corresponding mean equations, and also analytically calculated the density of particles in each compartment with a large number of compartments. In this section, we will conduct a similar analysis of the stretching method.

## D.1 The mean equations

In this section, we will calculate the mean equations for the stretching method by firstly considering the master equation for the process. We will then simulate the solutions to the mean equations, demonstrating that the solution remains uniform.

We begin by defining the sets  $\mathcal{M}_k^N = \{\mathbf{m} \in \mathbb{N}^k : \sum_{i=1}^k m_i = N\}$ . This is the set of all state vectors when there are  $k$  compartments and  $N$  particles in the system in total. We would like an expression for the probability that there are  $k$  compartments in total, and the state variable is  $\mathbf{m} \in \mathcal{M}_k^N$  with  $N$  a fixed integer. We call this probability  $p(\mathbf{m}, k, t)$  as we did in Section 2. Then, defining  $\pi(\mathbf{m}|\mathbf{r})$  to be the transition probability from state  $\mathbf{r}$  to state  $\mathbf{m}$ :

$$\frac{dp}{dt}(\mathbf{m}, k, t) = \rho(k-1) \sum_{\mathbf{r} \in \mathcal{M}_{k-1}^N} [p(\mathbf{r}, k-1, t)\pi(\mathbf{m}|\mathbf{r})] - \rho k p(\mathbf{m}, k, t). \quad (16)$$

Here, the summand in the first term of the right hand side represents the rate of moving from a state  $\mathbf{r} \in \mathcal{M}_{k-1}^N$  to the state  $\mathbf{m} \in \mathcal{M}_k^N$ . The second term is the rate at which the process leaves the state  $\mathbf{m}$ .

We next derive the mean equations. Similarly to the original method, define:

$$M_i^k(t) = \sum_{\mathbf{m} \in \mathcal{M}_k^N} m_i p(\mathbf{m}, k, t). \quad (17)$$

Multiplying the CME (16) by  $m_i$ , summing over all possible  $\mathbf{m} \in \mathcal{M}_k^N$ , and applying equation (17), we obtain:

$$\frac{dM_i^k}{dt}(t) = \rho(k-1) \sum_{\mathbf{m} \in \mathcal{M}_k^N} \sum_{\mathbf{r} \in \mathcal{M}_{k-1}^N} [m_i p(\mathbf{r}, k-1, t)\pi(\mathbf{m}|\mathbf{r})] - \rho k M_i^k(t). \quad (18)$$

From now on, we will drop the range of the sums, and simply write  $\mathbf{m}$  or  $\mathbf{r}$  and implicitly assume we are summing over the correct sets in order to simplify the notation. Recall that for each  $\mathbf{r}$ , there is an associated vector of binomial random variables  $\mathbf{b}$  that re-distributes the particles from the pregrowth state to the postgrowth state. Further, for any  $j \in \{1, \dots, k-1\}$ , the probability of drawing  $b_j$  is given by the probability mass function:

$$\mathbb{P}(b_j|\mathbf{r}) = \binom{r_j}{b_j} (\delta_j^{k-1})^{b_j} (1 - \delta_j^{k-1})^{r_j - b_j}, \quad (19)$$

where  $\delta_j^{k-1}$  is the ‘overlap’ proportion defined in equation (13). We can then use the relationships set out in Algorithm 2, namely:

$$m_j = \begin{cases} b_1, & \text{if } j = 1, \\ b_j + (r_{j-1} - b_{j-1}), & \text{if } j \in \{2, \dots, k-1\}, \\ r_{k-1} - b_{k-1}, & \text{if } j = k. \end{cases} \quad (20)$$

Assuming that  $i \in \{2, \dots, k-1\}$  (a similar argument can be applied to the case  $i \in \{1, k\}$ ) and substituting the relationships in (20) into the sum in equation (18), we obtain:

$$\begin{aligned} \sum_{\mathbf{m}} \sum_{\mathbf{r}} m_i p(\mathbf{r}, k-1, t)\pi(\mathbf{m}|\mathbf{r}) &= \sum_{\mathbf{m}} \sum_{\mathbf{r}} (r_{i-1} - b_{i-1} + b_i) p(\mathbf{r}, k-1, t)\pi(\mathbf{m}|\mathbf{r}) \\ &= \sum_{\mathbf{r}} \sum_{\mathbf{m}} r_{i-1} p(\mathbf{r}, k-1, t)\pi(\mathbf{m}|\mathbf{r}) - \sum_{\mathbf{r}} \sum_{\mathbf{m}} (b_{i-1} - b_i) p(\mathbf{r}, k-1, t)\pi(\mathbf{m}|\mathbf{r}), \end{aligned} \quad (21)$$

where in the final step, we have split the sum and also switched the order of summation. We will now make use of the binomial random variables (the  $b_j$ ’s). We note that in order to transition from the

pregrowth state  $\mathbf{r} \in \mathcal{M}_{k-1}^N$  to the postgrowth state  $\mathbf{m} \in \mathcal{M}_k^N$ , we need to find a vector  $\mathbf{b}$ . However, only a small subset of these  $\mathbf{b}$  vectors have a non-zero probability of occurring. Therefore, the sum over  $\mathbf{m}$  can be re-written as a sum over the possible  $\mathbf{b}$  values that have a non-zero probability. Using this, and substituting for  $\pi(\mathbf{m}|\mathbf{r})$  the specific probability  $\mathbb{P}(\mathbf{b}|\mathbf{r})$  of the binomial redistribution which takes us from state  $\mathbf{r}$  to state  $\mathbf{m}$ , equation (21) becomes:

$$\begin{aligned} \sum_{\mathbf{r}} \sum_{\mathbf{m}} m_i p(\mathbf{r}, k-1, t) \pi(\mathbf{m}|\mathbf{r}) &= \sum_{\mathbf{r}} \sum_{\mathbf{m}} r_{i-1} p(\mathbf{r}, k-1, t) \pi(\mathbf{m}|\mathbf{r}) - \sum_{\mathbf{r}} \sum_{\mathbf{m}} (b_{i-1} - b_i) p(\mathbf{r}, k-1, t) \pi(\mathbf{m}|\mathbf{r}) \\ &= \sum_{\mathbf{r}} \sum_{\mathbf{b}} r_{i-1} p(\mathbf{r}, k-1, t) \mathbb{P}(\mathbf{b}|\mathbf{r}) - \sum_{\mathbf{r}} \sum_{\mathbf{b}} (b_{i-1} - b_i) p(\mathbf{r}, k-1, t) \mathbb{P}(\mathbf{b}|\mathbf{r}). \end{aligned} \quad (22)$$

Here  $\mathbb{P}(\mathbf{b}|\mathbf{r}) = \prod_{j=1}^{k-1} \mathbb{P}(b_j|\mathbf{r})$  (where  $\mathbb{P}(b_j|\mathbf{r})$  are given in equation (19)), and the sum over  $\mathbf{b}$  is a sum over the set  $\{\mathbf{b} : b_j \in \{0, \dots, r_j\} \forall j \in \{1, \dots, k-1\}\}$ . We now re-arrange the two sums in expression (22):

$$\begin{aligned} \sum_{\mathbf{r}} \sum_{\mathbf{b}} r_{i-1} p(\mathbf{r}, k-1, t) \mathbb{P}(\mathbf{b}|\mathbf{r}) - \sum_{\mathbf{r}} \sum_{\mathbf{b}} (b_{i-1} - b_i) p(\mathbf{r}, k-1, t) \mathbb{P}(\mathbf{b}|\mathbf{r}) \\ = \sum_{\mathbf{r}} r_{i-1} p(\mathbf{r}, k-1, t) \left[ \sum_{\mathbf{b}} \mathbb{P}(\mathbf{b}|\mathbf{r}) \right] - \sum_{\mathbf{r}} p(\mathbf{r}, k-1, t) \left[ \sum_{\mathbf{b}} (b_{i-1} - b_i) \mathbb{P}(\mathbf{b}|\mathbf{r}) \right]. \end{aligned}$$

The sum in the square brackets in the first term is equal to 1 as it is the sum of a probability distribution and the sum in the square brackets in the second term is a difference of two expectations:

$$\begin{aligned} \sum_{\mathbf{r}} r_{i-1} p(\mathbf{r}, k-1, t) \left[ \sum_{\mathbf{b}} \mathbb{P}(\mathbf{b}|\mathbf{r}) \right] - \sum_{\mathbf{r}} p(\mathbf{r}, k-1, t) \left[ \sum_{\mathbf{b}} (b_{i-1} - b_i) \mathbb{P}(\mathbf{b}|\mathbf{r}) \right] \\ = \sum_{\mathbf{r}} r_{i-1} p(\mathbf{r}, k-1, t) - \sum_{\mathbf{r}} p(\mathbf{r}, k-1, t) [\mathbb{E}[b_{i-1}|\mathbf{r}] - \mathbb{E}[b_i|\mathbf{r}]]. \end{aligned}$$

Finally, using the standard expectation for the binomial distribution, and bringing terms together, we find that:

$$\begin{aligned} \sum_{\mathbf{m}} \sum_{\mathbf{r}} m_i p(\mathbf{r}, k-1, t) \pi(\mathbf{m}|\mathbf{r}) &= (1 - \delta_{i-1}^{k-1}) \sum_{\mathbf{r}} r_{i-1} p(\mathbf{r}, k-1, t) + \delta_{i-1}^{k-1} \sum_{\mathbf{r}} r_i p(\mathbf{r}, k-1, t) \\ &= (1 - \delta_{i-1}^{k-1}) M_{i-1}^{k-1} + \delta_{i-1}^{k-1} M_i^{k-1}, \end{aligned} \quad (23)$$

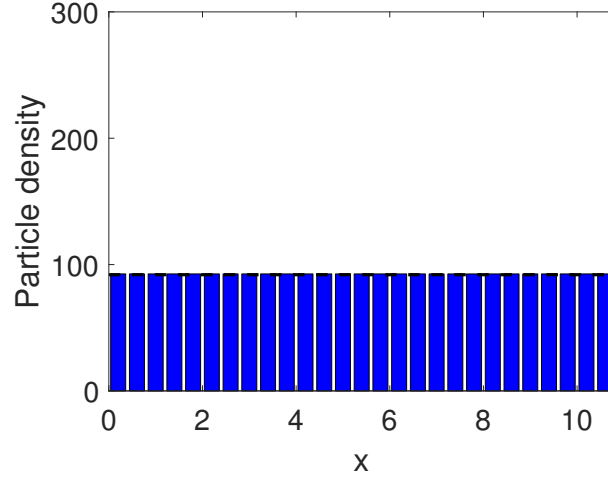
where the final equality uses the definition (17). Substituting this expression into equation (18) yields the mean-field density evolution equations for the stretching method:

$$\frac{dM_i^k}{dt} = \rho(k-1) [(1 - \delta_{i-1}^{k-1}) M_{i-1}^{k-1} + \delta_{i-1}^{k-1} M_i^{k-1}] - \rho k M_i^k. \quad (24)$$

## D.2 Analytical density

Here we will use a similar approach to Section 3 in order to calculate the average density of particles when the domain grows according to the stretching method. We let  $u_i^k$  be the random density of particles in compartment  $i$  when there are  $k$  compartments in total,  $M_i^k = \mathbb{E}[u_i^k]$ , and  $q(u_i^k, \delta_i^k)$  a realisation from a general probability distribution with mean value  $M_i^k \delta_i^k$ . The  $q(u_i^k, \delta_i^k)$  values denote the density that is redistributed into postgrowth compartment  $i$  from pregrowth compartment  $i$ , analogous to the  $\mathbb{P}(b_i|r_i)$  in equation (19) for the discretised process. Then the  $u_i^k$ 's are related by the following recursive relation:

$$u_i^k = q(u_i^{k-1}, \delta_i^{k-1}) + (u_{i-1}^{k-1} - q(u_{i-1}^{k-1}, \delta_{i-1}^{k-1})) \quad (25)$$



**Figure 10.** The solution to the mean-field equations (24). All parameters are the same as Figure 3(b).

with an initial condition  $u_1^1 = 1$ . Here, the first term on the right-hand side is the fraction of the density that is provided to compartment  $i$  on the postgrowth domain by the pregrowth compartment to the right, while the second term is the density provided from the left pregrowth compartment. We now take expectations to yield the recursive equations:

$$M_i^k = \delta_i^{k-1} M_i^{k-1} + (1 - \delta_{i-1}^{k-1}) M_{i-1}^{k-1}. \quad (26)$$

We show that, under this recursion relation,  $M_i^k = 1/k$  by induction on the number of compartments,  $k$ . Clearly, using the initial condition, we have the base case  $M_1^1 = 1$ . Now assume that  $M_i^{k-1} = 1/(k-1)$  for all  $i \in \{1, \dots, k-1\}$ . Then, using (26):

$$\begin{aligned} M_i^k &= \delta_i^{k-1} M_i^{k-1} + (1 - \delta_{i-1}^{k-1}) M_{i-1}^{k-1} \\ &= \frac{\delta_i^{k-1}}{k-1} + \frac{1 - \delta_{i-1}^{k-1}}{k-1} \\ &= \frac{k-i + (k - (k-i-1))}{k(k-1)} \\ &= \frac{1}{k}. \end{aligned}$$

Here, the second line uses the inductive hypothesis and the third line employs the definition of  $\delta_i^{k-1}$ . This completes the inductive step, and hence the proof that  $M_i^k = 1/k$ .

## References

1. R. McLennan, L. Dyson, K.W. Prather, J.A. Morrison, R.E. Baker, P.K. Maini, and Kulesa P.M. Multiscale mechanisms of cell migration during development: theory and experiment. *Development*, 139(16):2935–2944, 2012.
2. P.M. Kulesa, C.M. Bailey, J.C. Kasemeier-Kulesa, and R. McLennan. Cranial neural crest migration: new rules for an old road. *Dev. Biol.*, 344(2):543–554, 2010.
3. C.A. Yates. Discrete and continuous models for tissue growth and shrinkage. *J. Theor. Biol.*, 350: 37–48, 2014.

4. L. Wolpert, C. Tickle, and A.M. Arias. *Principles of development*. Oxford University Press, 5 edition, 2015.
5. E.J. Crampin, E.A. Gaffney, and P.K. Maini. Reaction and diffusion on growing domains: scenarios for robust pattern formation. *Bull. Math. Biol.*, 61(6):1093–1120, 1999.
6. E.J. Crampin. *Reaction-diffusion patterns on growing domains*. PhD thesis, University of Oxford, 2000.
7. P.R. Taylor, R.E. Baker, and C.A. Yates. Deriving appropriate boundary conditions, and accelerating position-jump simulations, of diffusion using non-local jumping. *Phys. Biol.*, 12(1):016006, 2015.
8. S.A. Isaacson. A convergent reaction-diffusion master equation. *J. Chem. Phys.*, 139(5):054101, 2013.
9. D.T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, 81(25):2340–2361, 1977.
10. M.A. Gibson and J. Bruck. Efficient exact stochastic simulation of chemical systems with many species and many channels. *J. Phys. Chem. A.*, 104(9):1876–1889, 2000.
11. J. Elf and M. Ehrenberg. Spontaneous separation of bi-stable biochemical systems into spatial domains of opposite phases. *Syst. Biol.*, 1(2):230–236, 2004.
12. J. Hattne, D. Fange, and J. Elf. Stochastic reaction-diffusion simulation with MesoRD. *Bioinformatics*, 21(12):2923–2924, 2005.
13. J.M. McCollum, G.D. Peterson, C.D. Cox, M.L. Simpson, and N.F. Samatova. The sorting direct method for stochastic simulation of biochemical systems with varying reaction execution behavior. *Comput. Biol. Chem.*, 30(1):39–49, 2006.
14. A.M. Smith, R.E. Baker, D. Kay, and P.K. Maini. Incorporating chemical signalling factors into cell-based models of growing epithelial tissues. *J. Math. Biol.*, 65(3):441–463, 2012.
15. R.E. Baker, C.A. Yates, and R. Erban. From microscopic to macroscopic descriptions of cell migration on growing domains. *Bull. Math. Biol.*, 72(3):719–762, 2010.
16. C.A. Yates and R.E. Baker. Importance of the voronoi domain partition for position-jump reaction-diffusion processes on non-uniform rectilinear lattices. *Phys. Rev. E*, 88(5):054701, 2013.
17. See Supplemental Material at [URL will be inserted by publisher] for the accompanying MATLAB code.
18. T.E. Woolley, R.E. Baker, E.A. Gaffney, and P.K. Maini. Stochastic reaction and diffusion on growing domains: understanding the breakdown of robust pattern formation. *Phys. Rev. E*, 84(4):046216, 2011.
19. R.N. Thompson, C.A. Yates, and R.E. Baker. Modelling cell migration and adhesion during development. *Bull. Math. Biol.*, 74(12):2793–2809, 2012.
20. C.A. Yates, R.E. Baker, R. Erban, and P.K. Maini. Going from microscopic to macroscopic on non-uniform growing domains. *Phys. Rev. E*, 86(2):021921, 2012.
21. L. Wolpert. Positional information and the spatial pattern of cellular differentiation. *J. Theor. Biol.*, 25(1):1–47, 1969.

22. K.J. Painter, P.K. Maini, and H.G. Othmer. Stripe formation in juvenile pomacanthus explained by a generalized turing mechanism with chemotaxis. *Proc. Natl. Acad. Sci. USA*, 96(10):5549–5554, 1999.
23. C.A. Smith and C.A. Yates. Spatially extended hybrid methods: a review. *J. Roy. Soc. Interface*, 15(139), 2018.
24. B.J. Binder, K.A. Landman, M.J. Simpson, M. Mariani, and D.F. Newgreen. Modeling proliferative tissue growth: a general approach and an avian case study. *Phys. Rev. E*, 78(3):031912, 2008.
25. D.G. Greenhalgh. The role of apoptosis in wound healing. *Int. J. Biochem. Cell Biol.*, 30(9): 1019–1030, 1998.
26. F. Grinnell, M. Zhu, M.A. Carlson, and J.M. Abrams. Release of mechanical tension triggers apoptosis of human fibroblasts in a model of regressing granulation tissue. *Exp. Cell. Res.*, 248(2): 608–619, 1999.
27. M.J. Simpson, K.A. Landman, and D.F. Newgreen. Chemotactic and diffusive migration on a nonuniformly growing domain: numerical algorithm development and applications. *J. Comput. Appl. Math.*, 192(2):282–300, 2006.
28. C.A. Yates and M.B. Flegg. The pseudo-compartment method for coupling partial differential equation and compartment-based models of diffusion. *J. Roy. Soc. Interface*, 12(106):20150141, 2015.
29. M.B. Flegg, S. Hellander, and R. Erban. Convergence of methods for coupling of microscopic and mesoscopic reaction-diffusion simulations. *J. Comput. Phys.*, 289(C):1–17, May 2015. ISSN 0021-9991.
30. C.A. Smith and C.A. Yates. The auxiliary region method: A hybrid method for coupling a PDE to Brownian-based dynamics for reaction-diffusion systems. *R. Soc. Open Sci.*, 5(8), 2018.
31. A. Goriely and M. Tabor. Mathematical modeling of hyphal tip growth. *Fungal Biol. Rev.*, 22(2): 77–83, 2008.