

Particle-based Simulation of Convergent-Extension Modifying Finger Instabilities in Chicken Embryo Gastrulation

An UROP funded by the Cambridge Centre for Physical Biology

H. Williams — Summer 2024

We conjecture that there is a causal relationship between convergent-extension motion, forming the primitive streak, and an anterior-posterior asymmetry in multicellular finger-like protrusions observed at the edge (area opaca) of the epiblast cell monolayer during chicken embryo gastrulation. To test this, we first reproduce existing work which computationally simulates the finger instabilities seen in scratch assay “wound healing” experiments. Such simulations model the motion of cell-centroids as particles moving under the influence of effective potentials which represent various key biological processes. We failed to directly answer our primary question regarding the influence of convergent-extension but outline the necessary further work to do so. Source code is made available.

1 Introduction

Particle-based “N-body” computational simulation of collective cell motion in tissues is a well established approach. There are two common flavours: simulating the motion of cell-centroids, as in [4], and simulating the motion of cell-vertices - the junctions formed between adjacent cells [24]. There are also continuum models [18] [10], which treat the cells as an active fluid. Here we simulate the motion of cell-centroids, with effective potentials modelling their finite size and membrane interactions. We chose this as it is the simplest of the three total possible approaches, though more computationally expensive than fluid models where the individual cell motion is in effect blurred.

Typical in-vitro experiments investigating collective cell motion take a “wound healing” scratch assay approach [17] whereby strips of cells are removed from a confluent cell monolayer and the motion of remaining cells into the free space is studied. This is traditionally done via razor or pipette-tip ablation [8]. However, more recent experiments [14] make use of a stencil to avoid causing cell damage (which releases cell contents and distress signals) and investigate the purely mechanical influence of a free edge on collective cell motion. Madin-Darby Canine Kidney (MDCK) cells are a popular cell line for these experiments.

Since in-vitro scratch assays frequently display “finger” formation, a substantial amount of work has already been done to model this mathematically/computationally [4] [21] [17] [10]. This work is based on Tarle et al. (2015) [21] which combines the continuum model for boundary motion described in Mark et al. (2010) [10] with the particle model for bulk motion described in Sepúlveda et al. (2013) [17] and adds a novel force representing the action of an actomyosin “purse-string”.

However, we are not interested in the finger instability per se. Rather we are interested in how the finger instability which is seen at the outer boundary of the area opaca in the gastrulating chicken embryo is coupled with primitive streak formation. Live imaging experi-

ments of excised (but otherwise unperturbed) embryo development show a distinct posterior-anterior asymmetry in the fingering pattern. At the anterior edge, we see longer wavelength, “smooth” fingering and on the posterior edge we see shorter wavelength “spikey” fingering. We conjecture that this polarity is driven by the convergent-extension motion of cells as the primitive streak forms (see figures 1, 2). This is the question we attempt to address.

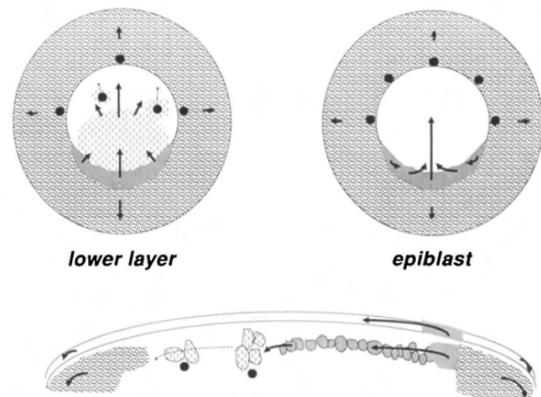


Figure 1: From Stern (1990) [20]. A schematic indicating the two approximate-monolayers of cells present at the time of primitive streak formation and the convergent-extension motion involved: the hypoblast (basal to the yolk and ventral to the eventual chick) and the epiblast (from which all the mature chick cells derive). It is in the epiblast that primitive streak formation occurs and this is henceforth referred to as “the cell monolayer”/“confluent cell monolayer”.

It is worth noting that others suggest the causality may be in reverse [9] and earlier literature suggested that there is little correspondence between the area opaca and primitive streak formation [19]. However, it is further worth noting that in reference [19], area opaca like morphology is seen developing in the cells at the cut edge — preventing us from definitively concluding that

primitive streak formation can succeed in the complete absence of an area opaca.

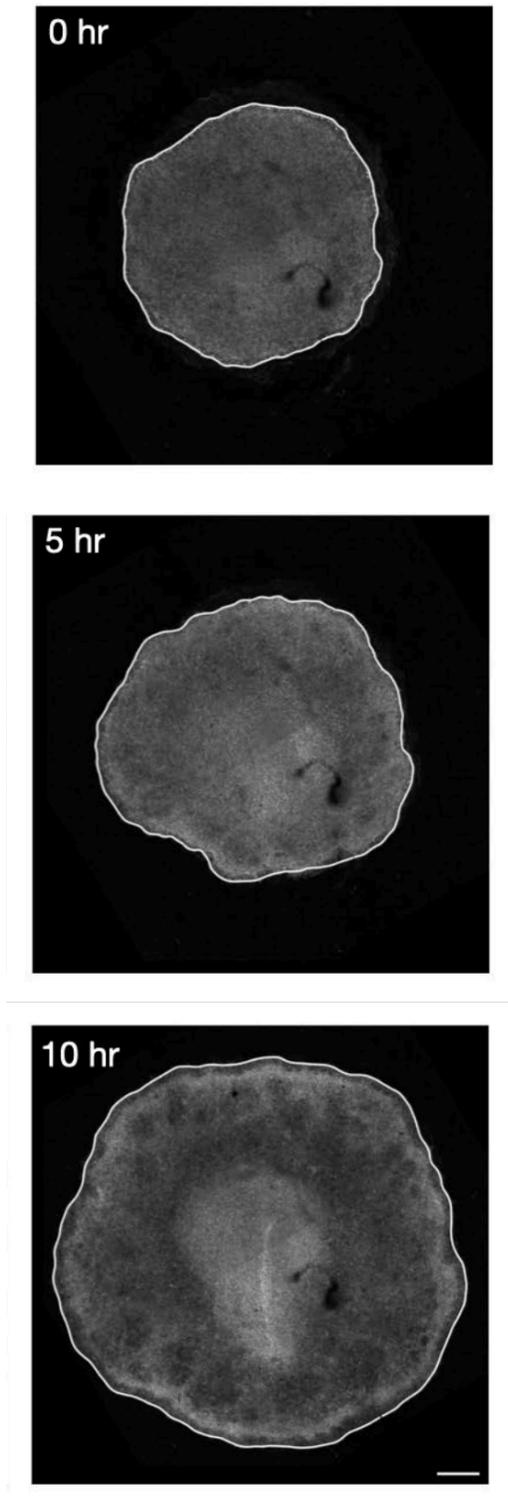


Figure 2: From Michaut (2024) [12] showing primitive streak formation in a quail embryo. This corresponds to Eyal-Giladi and Kochav stages X-XIV in the chicken [7].

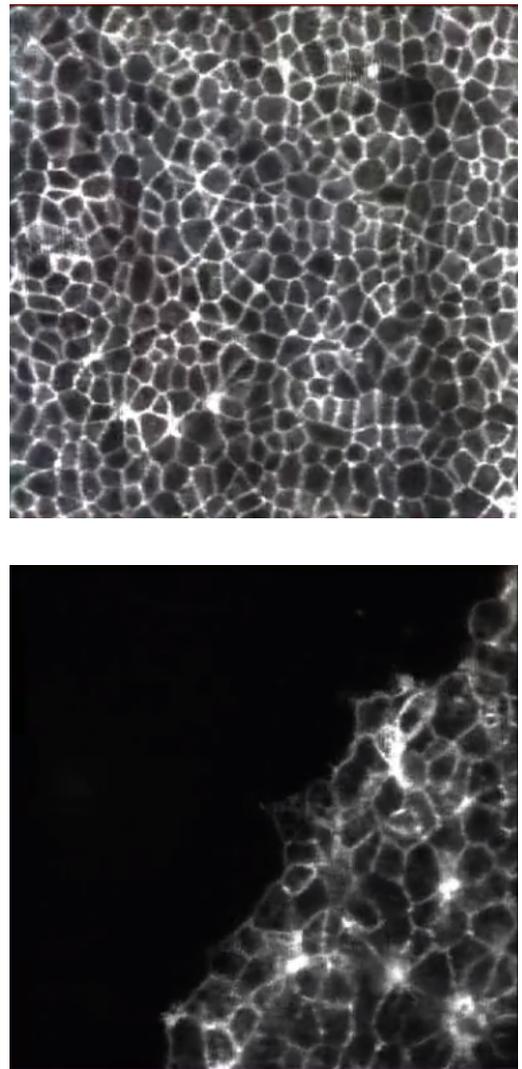


Figure 3: Still video frames from Rozbicki et al. (2015) [15]. The top image shows cells in the bulk (area pellucida) of a gastrulating chicken embryo and the bottom image shows cells at the edge (area opaca). These are to the same scale so that the apparent difference in cell volume is indeed physical.

2 The model

Our mathematical model driving the formation of fingers follows the work of Tarle et. al [21]. This differs from some other models in that the “leader” cells at the tips of fingers are not qualitatively distinct from the other cells. However, the model does distinguish between cells in the bulk and cells along the boundary of the confluent monolayer. The model features four key forces, derived from effective potentials:

1. Cell-cell interaction potential
2. Random noise (active walking, depends on cell density)

3. Vicsek interaction (active velocity alignment)
4. Contour (boundary) forces
 - (a) Passive bending force
 - (b) Outward force from lamellipodia in convex regions
 - (c) “Purse-string” force from acto-myosin in concave regions

For numerical values of the constants used in these potentials, see appendix E

2.1 Cell-Cell Interaction Potential

This is the pairwise, short-range force which acts along the joining line between neighbouring cells. At short distance, it is repulsive in order to represent the “squishyness” of cells due to their finite size. It is worth recalling that eukaryotic cells do not strictly have a fixed volume as they can lose/gain water via many mechanisms, including cytoskeletal changes. Rather, their deformation modulus (squishyness) represents a balance of many competing processes.

At longer range, the potential is attractive, representing cell-cell adhesion. Beyond a certain limit, the potential plateaus (or in practice the code disregards it), representing detachment of the cells.

In the intermediate range, Sarkar et al. [16] have identified the importance of a flat bottom. Without the flat bottom and with low noise (low “temperature”), the particles would settle into a local minimum in the high-dimensional energy landscape — a glassy transition.

$$\text{Interaction Potential}(r) = U_0 \exp\left(-\left(\frac{r}{A_0}\right)^2\right) + U_1(r - A_1)^2\Theta(r - A_1) \quad (1)$$

We follow the potential described in Sepúlveda et al. [17], equation 1 where Θ is the Heaviside step function. Unfortunately in the code we inherited a small mistake introduced by Tarle (which itself is a correction of a more substantial mistake in the potential described in ref [21]). However, this still featured a repulsive core, a relatively flat bottom, and an attractive longer-range force. Since the cell-cell interaction potential has been found to have little impact on the finger instability [17] [10], our simulations are still representative of a more accurate potential.

For future work, we would consider replacing the potential described in equation 1 with that described in equation 2, which actually possesses a hard core (see figure 4). More generally we would recommend actually plotting whatever potential is used!

$$\text{Interaction Potential} = \frac{U_0}{e^2} \exp\left(\left(\frac{A_0}{r}\right)^2\right) + U_1(r - A_1)^2\Theta(r - A_1) \quad (2)$$

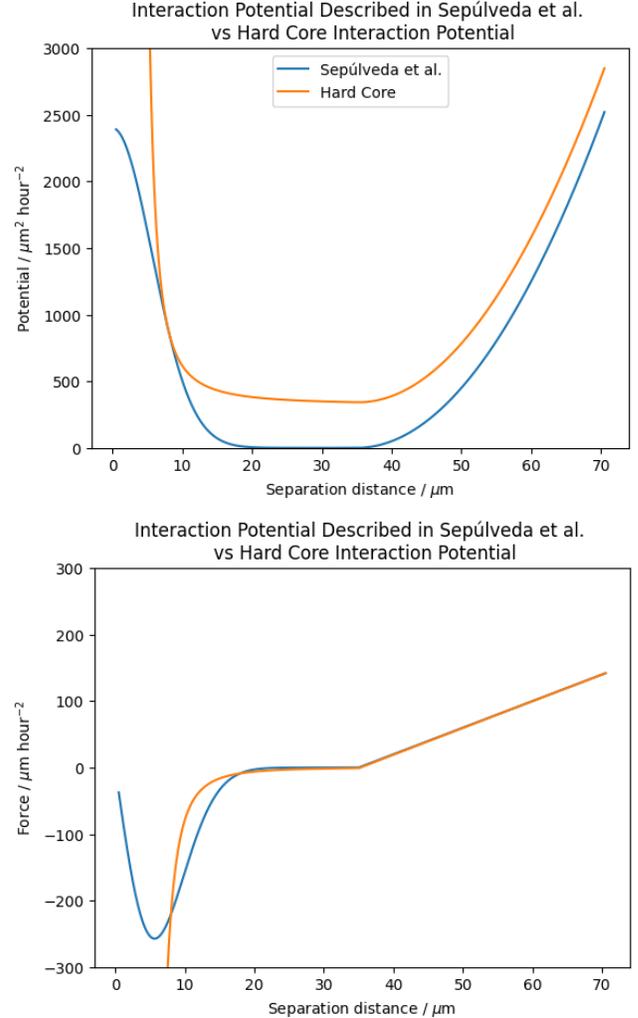


Figure 4: We compare a hard-core interaction potential with the gaussian-core interaction potential described in Sepúlveda et al. (top) and likewise compare the derived forces (bottom). The gaussian results in a short-range repulsive force which peaks before eventually reducing to zero. This is unphysical but acceptable so long as the barrier is sufficiently high to prevent particles crossing it. A compromise would be to raise an error if the particles get closer than the barrier.

2.2 Random Noise

The random noise represents cells’ tendency to explore their environment in a random-walk [17]. The magnitude increases in regions of low cell density, matching in-vitro observations. It resembles Brownian motion

but is in fact an active process. Mathematically, this is modelled as a Orstein-Uhlenbeck system:

$$\sigma(r) = \sigma_0 + (\sigma_1 - \sigma_0) \left(1 - \frac{\rho}{\rho_0}\right) \quad (3)$$

$$\tau \frac{d\vec{\eta}_i}{dt} = -\vec{\eta}_i + \vec{\xi}_i \quad (4)$$

$$\langle \xi_{j,\beta}(t) \xi_{i,\alpha}(t') \rangle = \delta_{i,j} \delta_{\alpha,\beta} \delta(t - t') \quad (5)$$

$$F_{\text{noise}}(\rho) = \sigma(\rho) \vec{\eta}_i \quad (6)$$

Equation 3 adjusts the magnitude σ of the noise depending on cell density ρ . Equation 4 describes how the direction $\vec{\eta}$ of the walk evolves in time t with a delta-correlated stochastic driving force $\vec{\xi}$, in turn described by equation 5. Equations 4 & 5 are well known to physicists as the Langevin equation. Equation 6 shows how we combine magnitude and direction to calculate the force applied to the i 'th particle.

Tarle [21] found that the density-dependent noise is key for leader cells to recruit particles from the bulk into the developing fingers. When noise is reduced or made constant, Tarle observed that fingers tend to be wider and shorter. Presumably by increasing random motion in regions of low density, the cells can more rapidly diffuse into free space opened by growing fingers.

2.3 Vicsek Interaction

This models cells' tendency to reach a quorum regarding their net velocity. In the original paper by Vicsek [22], cells are assumed to actively take on the average velocity of their immediate neighbours. However, this is slightly modified in Tarle/Sepúlveda et al. [21] [17] so that there is an effective force proportional to the *difference* between a cell's current velocity and the average velocity of its neighbours.

$$F_{\text{Vicsek}} = -\alpha \vec{v}_i + \sum_{j \text{ N.N. of } i} \frac{\beta}{N_i} (\vec{v}_j - \vec{v}_i) \quad (7)$$

Equation 7 is the Vicsek-like force, described above. We have also included a straightforward viscosity term, representative of cell-cell adhesive shear forces and motion through the extra-cellular fluid.

Tarle [21] observed that decreasing the magnitude of the Vicsek interaction leads to decreasing finger widths, eventually resulting in single-cell-wide "trains" of cells. Since such trains are commonly found in metastasising cancers, we propose that this indicates a potential drug target: by selectively enhancing the cellular mechanisms underlying the Vicsek interaction, one might reduce the likelihood of metastasis in cancer patients.

2.4 Contour Forces

Whilst the other aspects of the model are biologically relevant, it is the contour forces found at the boundary of the confluent cell monolayer that really drive the

finger instability. This is not one but three different forces, representing three distinct biological processes. In all of these, \vec{H} and H are defined by equations 8 and 9 respectively, where \hat{n} is the outward unit normal along the boundary parameterised by distance s .

$$\vec{H} := \frac{d^2 \vec{r}}{ds^2} \quad (8)$$

$$H := \vec{H} \cdot \hat{n} \quad (9)$$

2.4.1 Passive Bending Force

This reflects the fact that the boundary of the cell monolayer forms a one dimensional membrane. From the Helfrich potential, Mark et al. [10] derive the force:

$$F_{\text{bending}} = -\kappa \frac{d^2 \vec{H}}{ds^2} - \frac{3}{2} \kappa |\vec{H}|^2 \vec{H} \quad (10)$$

The first term in equation 10 reminds us of the gradient-squared term found in diffusion equations. This acts as an averaging term, attempting to smooth any changes in value. However, the value it is trying to smooth is the curvature (itself a second order derivative), as opposed to the displacement of the membrane (zeroth order). This acts to "inflate" sharp tips. Note that we have followed Tarle [21] and removed a term which is linear in \vec{H} since this will be accounted for by the acto-myosin force (equation 12).

The second term in equation 10 is a positive feedback term, which increases finger growth once started, but does not drive the original instability [10].

2.4.2 Lamellipodia along Convex Boundary

This is the most essential part of the model to drive finger development. Theoretically this is somewhat obvious but linear perturbation analysis confirms this [10] and Tarle [21] has also shown via numerical experiments that it is indeed the main force driving the finger instability. The fact that the force increases with increasing curvature is key to finger formation. Indeed experiments whereby the background outward force is increased, so that the curvature-dependent gradient of force is less steep, show that fingering is suppressed and instead the boundary edge moves as a uniform front that can propagate waves of acceleration into the bulk [21] [10] [14].

$$F_{\text{lamellipodia}} = \begin{cases} 0 & H \geq 0 \\ \frac{F_{\text{max}}}{H_{\text{max}}} |H| & 0 > H > -H_{\text{max}} \\ F_{\text{max}} & H \leq -H_{\text{max}} \end{cases} \quad (11)$$

Equation 11 is how we model the force on the cells/particles due to the action of lamellipodia. In-vitro scratch assay experiments show that they are most active when the local curvature is convex [6], hence our

piecewise linear force which is 0 in concave regions and increases linearly up to a maximum possible force in convex regions. The force is directed outward, normal to the cell layer boundary, and thus acts as a positive-feedback mechanism driving finger growth in response to random perturbations.

2.4.3 Acto-Myosin along Concave Boundary

In-vitro experiments [11] [6] show that an inter-cellular, tensile, acto-myosin cable develops along concave segments of the cell-layer boundary. This produces a “purse-string” like effect and a resultant outward force. We model this as a constant tension acting tangentially along the boundary in concave regions (equation 12):

$$F_{\text{cable}} = \begin{cases} 0 & H \leq 0 \\ F_1 & H > 0 \end{cases} \quad (12)$$

3 Detecting Nearest-Neighbours

Both the cell-cell interaction potential and the Vicsek force rely on correctly identifying the immediate (nearest) neighbours of each cell. This is a non-trivial problem.

The algorithm used by Tarle et al. is described in appendix B. For more details regarding Tarle’s algorithm, see the Electronic Supplementary Materials (ESI) [21]. Also, see figure 5 for a visual guide to the cosine-rule blocking criterion.

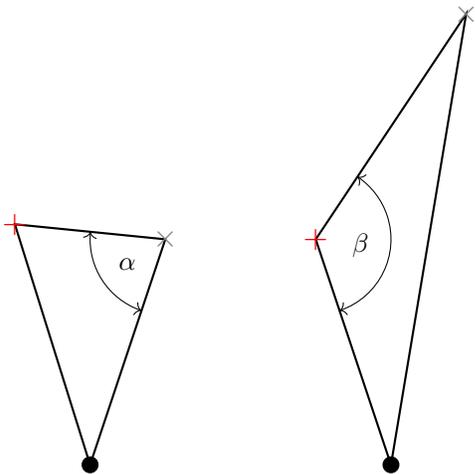


Figure 5: Cosine-rule neighbour-blocking criterion. We check the candidate neighbours of the current vertex (●) to see if one blocks the other. Angle α is sufficiently small that the tentative-neighbour (×) does not block the newest candidate-neighbour (+). In contrast angle β is large enough that we conclude the new candidate-neighbour actually blocks the tentative-neighbour, which should be removed from the neighbour list of the current vertex.

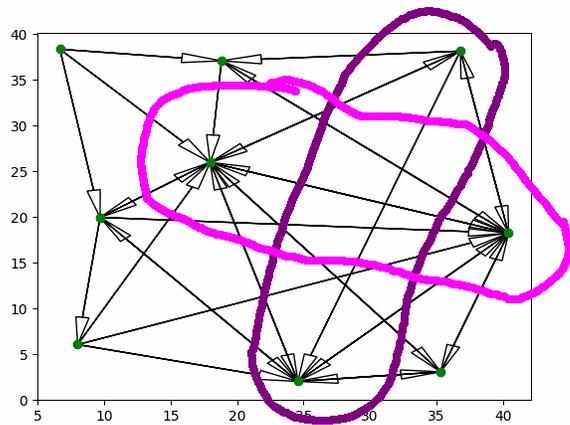


Figure 6: This is not quite the correct neighbour-network but it does highlight a genuine feature of Tarle’s neighbour algorithm: the pink and purple loops highlight two crossed nearest-neighbour connections. This is unphysical/not possible in a densely packed cell monolayer (eg the chicken epiblast).

This algorithm is perfectly acceptable when the cell density is low, so that they remain in a “gaseous” phase. However, the cells in the gastrulating chicken embryo are densely packed (see figure 3) so that Tarle’s nearest-neighbour detecting algorithm is not appropriate for our investigation. The problem is that it allows nearest neighbour interactions to cross each other, which is not possible in a densely packed monolayer of cells. See figure 6 for a sketch of the problem.

After discussing what the particle model is trying to physically represent, it became clear that our nearest-neighbour network should be dual to the outline of the cell membranes (in a similar manner to the duality between the Brillouin zones and the Wigner-Seitz cells in crystalline materials). Thus, what we really want is to form a maximally connected planar graph between the particles.

One way of forming a planar graph from a collection of points is to find the Delaunay triangulation. Indeed, further literature review revealed this is a standard approach not only in tissue simulations [4] but also in many other fields [2]. We implemented the popular Bowyer-Watson algorithm (see appendix D) which is $O(N \ln(N))$. The dual of the Delaunay triangulation is the Voronoi diagram (see figure 7), which is widely used in cell-vertex simulations of tissue motion [1] [3] [4].

More broadly, this problem falls under the title “approximate nearest-neighbour methods.” We are investigating another possible algorithm which is $O(N)$ and apparently novel, to be discussed in a separate text.

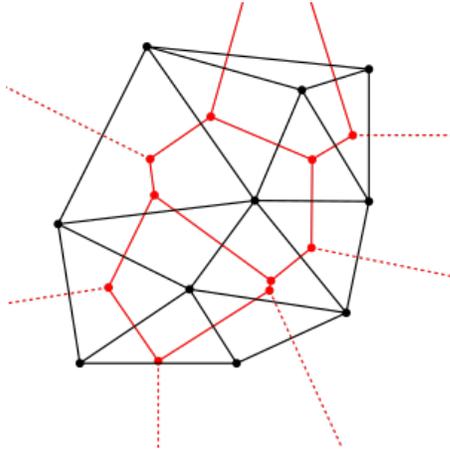


Figure 7: From Bourke (1989) [5]. In black we have the Delaunay triangulation. This is the neighbour network we calculate, where the cell-centroids/particles are the black vertices. In red we have the Voronoi diagram, which is dual to the Delaunay triangulation and appears visually similar to the epiblast cells seen in figure 3.

4 The Coastline Problem

The curvature forces rely on identifying the boundary or “coastline” of the cell monolayer. Famously, this is an ill-defined concept so we must choose a sensible definition and algorithm. For a description of Tarle’s algorithm, see appendix C.

An unforeseen benefit of finding the Delaunay Triangulation is that it defines a coastline for free. Namely, those edges which are the boundary of only one triangle (as opposed to a maximum of two) form the coastline, along with their end particles. There are some artifacts introduced by the supporting structure necessary for the Bowyer-Watson algorithm, and these are discussed in section 7.

5 Results

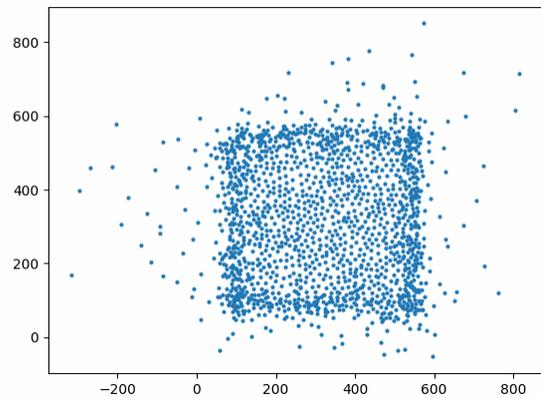
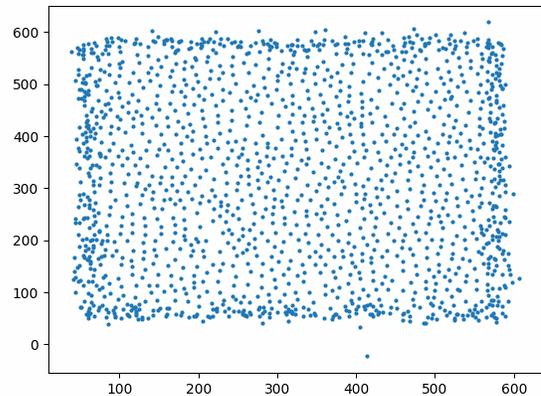


Figure 8: Still frames taken from a simulation of cell motion. The particles (cell-centroids) were initialised in a noisy grid. The top image shows the cells tend to bunch at the edge after some time, possibly due to overcrowding in the bulk or simply diffusion limited by attraction. The lower image shows orbit-like behaviour as cells are slowly ejected. This is because the only part of the model “switched-on” here is the cell-cell interaction potential, with no viscosity etc, and we did not implement detachment.

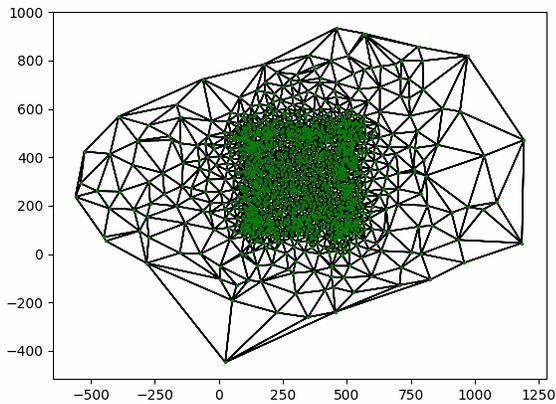


Figure 9: This is another frame taken from the same simulation as above. It shows the Delaunay triangulation which determines neighbour interactions.

In figure 8, we see our preliminary results. Here we only use the cell-cell interaction potential, as described in equation 1. None of the other three aspects of the model are currently well validated in our code. In figure 9 we see the Delaunay triangulation/nearest-neighbour network.

As a result of only effecting the interaction potential, we see the simulation develop orbits. Notably, in this simulation we did not implement a detachment distance, beyond which particles are not considered neighbours, or viscosity. Hence, it behaves like a modified gravity simulation.

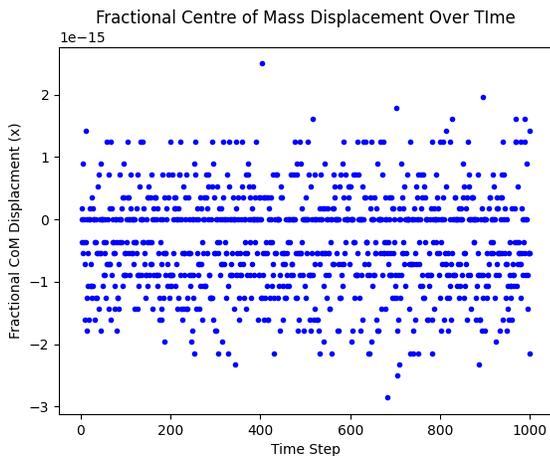


Figure 10: The displacement of the centre of mass over time is taken as a fraction of the initial location of the centre of mass. This is because the initial location is proportional to the size of the confluent layer so gives us a way to make the displacement dimensionless.

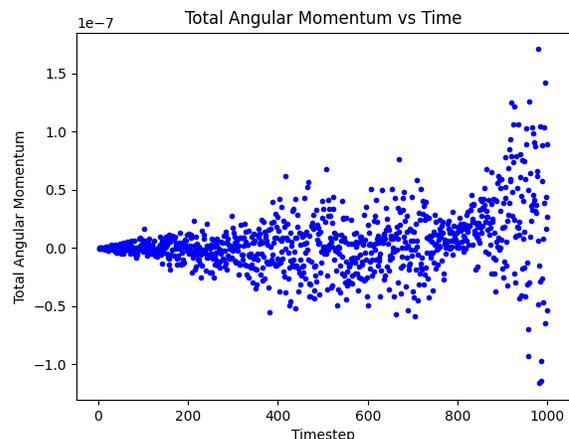


Figure 11: Here we quote raw numerical nett vorticity, in $\mu\text{m}^2 \text{hour}^{-2}$. The best way to make this dimensionless would be to divide it by the sum of the moduli of the particle vorticities at each timestep, so that we can confirm opposing vorticities are cancelling, rather than the number simply being numerically small.

We used a naïve first-order Euler integration scheme. Figures 10 & 11 indicate that that nonetheless, the simulations preserve invariants to a high precision. By using Noether’s theorem, we can check what quantities we expect to be conserved by the model. Two such values are the centre of mass and the total angular momentum. The total angular momentum is an especially important invariant in our investigation since the convergent-extension motion introduces vorticity, but Noether’s theorem tells us the nett vorticity ought to remain null if there are no numerical errors.

6 Discussion

We have successfully implemented code for modelling collective cell motion in densely packed tissue layers. See the appendix A for code availability and appendices B through D for pseudocode describing the key algorithms.

Due to the short duration of the UROP (eight weeks) compared with the time taken to produce the previous work (approximately one year), we did not fully explore the effect of all potentials. Nor did we address our primary question: is there a causal relationship between convergent-extension and the anterior-posterior asymmetry of finger instabilities observed during live imaging of the gastrulating chicken embryo?

The simulations with only the interaction potential “switched on” display orbit-like behaviour. Notably, cell/particle density increases at the edge of the cell monolayer until cells are ejected into orbits. We postulate that this is because initially, cells at the perimeter are under tension from the potential, while cells in the bulk experience very little nett force and thus ex-

hibit fluid-like behaviour. For longer runs of the simulation, we would expect the system to stabilise into a much lower density fluid with substantial vorticity. If we were to “turn-on” detachment and no other parts of the model, we would expect to instead see near total evaporation, with a much smaller number of cells/particles remaining in a low-density fluid-like state. However, neither of these two hypotheses have been tested.

7 Future work

There are a few pieces of further work needed to address our primary question. These are:

- One-by-one “turn-on” and validate the other parts of the model, including detachment, checking that invariants are conserved
- Possibly implement a more sophisticated approach to modify the coastline, removing artifacts from the Bowyer-Watson algorithm
- Carry out a quantitative analysis of the position/velocities of the simulated particles, and compare results with the same analysis of in-vitro scratch assays
- Model the convergent-extension motion which forms the primitive streak during chicken embryo gastrulation and investigate how this modifies finger instabilities at the cell-monolayer boundary
 - Such work might include basic cell-cell signalling dynamics by storing a signal value per cell, and modifying behaviour (including signal level) based on its neighbour’s signal values

It is physically obvious that quantities such as total energy and momentum will not be preserved once we turn-on dissipative effects such as viscosity and active motion. However, this was not derived rigorously from Neother’s theorem. If this were pursued, it would be worth drawing comparisons with the non-relativistic electromagnetic lagrangian, which *does* preserve total energy and momentum (only once the EM field-energy is included) yet also contains a linear velocity term.

8 Conclusions

We have successfully implemented code for modelling collective cell motion in densely packed tissue layers. Whilst our primary question: “is there a causal relationship between convergent-extension and the anterior-posterior asymmetry of finger instabilities observed during live imaging of the gastrulating chicken embryo?” remains unanswered, we have outlined the additional work necessary to investigate this. Very minimal additional code is needed for further work but substantial experimentation and analysis would be necessary.

Funding/Acknowledgements

This work constitutes a summer UROP as part of the Xiong lab in the Gurdon Institute, Cambridge UK. Funding was provided by the Cambridge Centre for Physical Biology. The author would like to thank Dr. Lakshmi Balasubramaniam and Prof. Nir Gov in their capacity as supervisors for the project. Thanks also to the rest of the Xiong group and thanks especially to Dr. Susie McLaren for her patience and valuable input.

References

- [1] Daniel L. Barton et al. “Active Vertex Model for cell-resolution description of epithelial tissue mechanics”. In: *PLOS Computational Biology* 13.6 (June 2017). Ed. by Stanislav Shvartsman, e1005569. ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.1005569. URL: <http://dx.doi.org/10.1371/journal.pcbi.1005569>.
- [2] Mark de Berg et al. *Computational Geometry: Algorithms and Applications*. Springer Berlin Heidelberg, 2008. ISBN: 9783540779742. DOI: 10.1007/978-3-540-77974-2. URL: <http://dx.doi.org/10.1007/978-3-540-77974-2>.
- [3] Dapeng Bi et al. “Motility-Driven Glass and Jamming Transitions in Biological Tissues”. In: *Physical Review X* 6.2 (Apr. 2016). ISSN: 2160-3308. DOI: 10.1103/physrevx.6.021011. URL: <http://dx.doi.org/10.1103/PhysRevX.6.021011>.
- [4] Luis L. Bonilla, Ana Carpio, and Carolina Trenado. “Tracking collective cell motion by topological data analysis”. In: *PLOS Computational Biology* 16.12 (Dec. 2020), pp. 1–43. DOI: 10.1371/journal.pcbi.1008407. URL: <https://doi.org/10.1371/journal.pcbi.1008407>.
- [5] P. Bourke. *Efficient Triangulation Algorithm Suitable for Terrain Modelling*. <https://paulbourke.net/papers/triangulate/>. Accessed: 2024-10-06.
- [6] Agustí Brugués et al. “Forces driving epithelial wound healing”. In: *Nature Physics* 10.9 (Aug. 2014), pp. 683–690. ISSN: 1745-2481. DOI: 10.1038/nphys3040. URL: <http://dx.doi.org/10.1038/nphys3040>.
- [7] Hefzibah Eyal-Giladi and Shimshon Kochav. “From cleavage to primitive streak formation: A complementary normal table and a new look at the first stages of the development of the chick”. In: *Developmental Biology* 49.2 (Apr. 1976), pp. 321–337. ISSN: 0012-1606. DOI: 10.1016/0012-1606(76)90178-0. URL: [http://dx.doi.org/10.1016/0012-1606\(76\)90178-0](http://dx.doi.org/10.1016/0012-1606(76)90178-0).

- [8] Nina Kramer et al. “In vitro cell migration and invasion assays”. In: *Mutation Research/Reviews in Mutation Research* 752.1 (2013), pp. 10–24. ISSN: 1383-5742. DOI: <https://doi.org/10.1016/j.mrrev.2012.08.001>. URL: <https://www.sciencedirect.com/science/article/pii/S1383574212000464>.
- [9] Hyung Chul Lee, Yara Fadaili, and Claudio D. Stern. “Molecular characteristics of the edge cells responsible for expansion of the chick embryo on the vitelline membrane”. In: *Open Biology* 12.9 (Sept. 2022). ISSN: 2046-2441. DOI: [10.1098/rsob.220147](https://doi.org/10.1098/rsob.220147). URL: <http://dx.doi.org/10.1098/rsob.220147>.
- [10] Shirley Mark et al. “Physical Model of the Dynamic Instability in an Expanding Cell Culture”. In: *Biophysical Journal* 98.3 (Feb. 2010), pp. 361–370. ISSN: 0006-3495. DOI: [10.1016/j.bpj.2009.10.022](https://doi.org/10.1016/j.bpj.2009.10.022). URL: <http://dx.doi.org/10.1016/j.bpj.2009.10.022>.
- [11] Paul Martin and Julian Lewis. “Actin cables and epidermal movement in embryonic wound healing”. In: *Nature* 360.6400 (Nov. 1992), pp. 179–183. ISSN: 1476-4687. DOI: [10.1038/360179a0](https://doi.org/10.1038/360179a0). URL: <http://dx.doi.org/10.1038/360179a0>.
- [12] Arthur Michaut et al. “A tension-induced morphological transition shapes the avian extra-embryonic territory”. In: *bioRxiv* (Feb. 2024). DOI: [10.1101/2024.02.08.579502](https://doi.org/10.1101/2024.02.08.579502). URL: <http://dx.doi.org/10.1101/2024.02.08.579502>.
- [13] Rui Pereira et al. “Energy efficiency across programming languages: how do energy, time, and memory relate?” In: *Proceedings of the 10th ACM SIGPLAN International Conference on Software Language Engineering*. SPLASH ’17. ACM, Oct. 2017. DOI: [10.1145/3136014.3136031](https://doi.org/10.1145/3136014.3136031). URL: <http://dx.doi.org/10.1145/3136014.3136031>.
- [14] M. Poujade et al. “Collective migration of an epithelial monolayer in response to a model wound”. In: *Proceedings of the National Academy of Sciences* 104.41 (Oct. 2007), pp. 15988–15993. ISSN: 1091-6490. DOI: [10.1073/pnas.0705062104](https://doi.org/10.1073/pnas.0705062104). URL: <http://dx.doi.org/10.1073/pnas.0705062104>.
- [15] Emil Rozbicki et al. “Myosin-II-mediated cell shape changes and cell intercalation contribute to primitive streak formation”. In: *Nature Cell Biology* 17.4 (Mar. 2015), pp. 397–408. ISSN: 1476-4679. DOI: [10.1038/ncb3138](https://doi.org/10.1038/ncb3138). URL: <http://dx.doi.org/10.1038/ncb3138>.
- [16] Debarati Sarkar, Gerhard Gompper, and Jens Elgeti. “A minimal model for structure, dynamics, and tension of monolayered cell colonies”. In: *Communications Physics* 4.1 (Feb. 2021). ISSN: 2399-3650. DOI: [10.1038/s42005-020-00515-x](https://doi.org/10.1038/s42005-020-00515-x). URL: <http://dx.doi.org/10.1038/s42005-020-00515-x>.
- [17] Néstor Sepúlveda et al. “Collective Cell Motion in an Epithelial Sheet Can Be Quantitatively Described by a Stochastic Interacting Particle Model”. In: *PLOS Computational Biology* 9.3 (Mar. 2013), pp. 1–12. DOI: [10.1371/journal.pcbi.1002944](https://doi.org/10.1371/journal.pcbi.1002944). URL: <https://doi.org/10.1371/journal.pcbi.1002944>.
- [18] Guillermo Serrano Nájera et al. “Control of Modular Tissue Flows Shaping the Embryo in Avian Gastrulation”. In: (July 2024). DOI: [10.1101/2024.07.04.601785](https://doi.org/10.1101/2024.07.04.601785). URL: <http://dx.doi.org/10.1101/2024.07.04.601785>.
- [19] Nelson T. Spratt and Hermann Haas. “Integrative mechanisms in development of the early chick blastoderm. I. Regulative potentiality of separated parts”. In: *Journal of Experimental Zoology* 145.2 (Nov. 1960), pp. 97–137. ISSN: 1097-010X. DOI: [10.1002/jez.1401450202](https://doi.org/10.1002/jez.1401450202). URL: <http://dx.doi.org/10.1002/jez.1401450202>.
- [20] Claudio D. Stern. “The marginal zone and its contribution to the hypoblast and primitive streak of the chick embryo”. In: *Development* 109.3 (July 1990), pp. 667–682. ISSN: 1477-9129. DOI: [10.1242/dev.109.3.667](https://doi.org/10.1242/dev.109.3.667). URL: <http://dx.doi.org/10.1242/dev.109.3.667>.
- [21] Victoria Tarle et al. “Modeling the finger instability in an expanding cell monolayer”. In: *Integr. Biol.* 7 (10 2015), pp. 1218–1227. DOI: [10.1039/C5IB00092K](https://doi.org/10.1039/C5IB00092K). URL: <http://dx.doi.org/10.1039/C5IB00092K>.
- [22] Tamás Vicsek et al. “Novel Type of Phase Transition in a System of Self-Driven Particles”. In: *Physical Review Letters* 75.6 (Aug. 1995), pp. 1226–1229. ISSN: 1079-7114. DOI: [10.1103/physrevlett.75.1226](https://doi.org/10.1103/physrevlett.75.1226). URL: <http://dx.doi.org/10.1103/PhysRevLett.75.1226>.
- [23] P. Vishnevsky. *Why I no longer recommend Julia*. <https://yuri.is/not-julia/>. Accessed: 2024-10-06.
- [24] Jieling Zhao, Farid Manuchehrfar, and Jie Liang. “Cell-substrate mechanics guide collective cell migration through intercellular adhesion: a dynamic finite element cellular model”. In: *Biomechanics and Modeling in Mechanobiology* 19.5 (Feb. 2020), pp. 1781–1796. ISSN: 1617-7940. DOI: [10.1007/s10237-020-01308-5](https://doi.org/10.1007/s10237-020-01308-5). URL: <http://dx.doi.org/10.1007/s10237-020-01308-5>.

A Code Availability

Any readers desiring a copy of the source code should email both *harvey.john.williams2000@gmail.com* and *harvey.williams562@gmail.com*.

The simulation code was written in Go, as it is a relatively fast [13], minimal, compiled programming language with similar syntax to the C family but, unlike the C family, features automatic memory management (“garbage-collection”). Previous work was written in Julia, which has some serious flaws as a programming language [23]. Plotting and analysis code was written in Python and takes an order of magnitude longer to run than the actual simulation, vindicating our choice to use Go.

The biggest “gotcha” using Go is that *every* assignment operation copies by value. A for loop over a list of objects will return a copy of the corresponding object at each step, unless pointers are explicitly used.

B Tarle’s Nearest-Neighbour Algorithm

Algorithm 1: Tarle’s Nearest-Neighbour Algorithm

Input: Vertex List

Side-Effect: Update All Vertex Nearest Neighbours

```
foreach current vertex  $\in$  vertex list do
  List all of current vertex’s neighbours from the previous timestep and all of their previous neighbours.
  These are candidate neighbours;
  Initialise tentative-neighbour list ;           // this will always remain sorted by increasing angle a
  foreach candidate-neighbour  $\in$  candidate-neighbour list do
    Immediately discard if it is beyond some threshold distance from current vertex;
    Find the first tentative-neighbour at a greater angle than the candidate-neighbour;
    Use the cosine rule to calculate the larger internal angle of  $O\hat{A}B$  or  $O\hat{B}A$ , where  $O$  is the current
    vertex,  $A$  is the tentative-neighbour and  $B$  is the candidate neighbour;
    If the internal angle inferred from the cosine rule is too large, this tells us the closer of  $A$  or  $B$  blocks
    the other from being a neighbour to the current vertex;
    Ensure that the correct combination of  $A$  and/or  $B$  is in the appropriate place in the
    tentative-neighbours list (based on whether there is blocking or not);
    If in the previous step, we replaced the tentative-neighbour with the candidate-neighbour, we must also
    compare the candidate-neighbour with the next greater-angle tentative-neighbour;
    Repeat the previous steps in the opposite direction, making comparison with the first smaller-angle
    tentative-neighbour;
  end
  Once all appropriate comparisons, insertions and replacements are made, we now have a sorted
  tentative-neighbour list;
  Replace the current vertex’s neighbour list with the tentative-neighbour list as these are now all confirmed
  valid neighbours;
end
```

^aby angle, we mean the angle from the positive x-axis, centred on the current vertex

Note that Tarle’s Nearest-Neighbour Algorithm could be greatly simplified by pre-sorting the list of candidates in order of angle. With this modification, one would only need to check the candidate-neighbour against the last-added tentative-neighbour, replacing or appending as appropriate, with no need to check against any other previously added candidates (even in the case of replacement) or to check again “in the other direction”. Both the original and the modified version of Tarle’s algorithm are $O(N)$, since the number of candidate neighbours per particle remains fixed as the number of particles (N) increases.

For a visualisation of the cosine-rule blocking criterion, see figure 5 in the main text.

C Tarle’s Coastline Algorithm

Algorithm 2: Tarle’s Coastline Algorithm

Input: Vertex List

Output: Coastline List

Select the vertex with the greatest x-coordinate (or any from a set of equal x-coordinates) and add it to the coastline list;
Select the first of its neighbours (which is sorted by increasing angle from the positive x-axis centred on the current vertex);
Add this neighbour to the coastline list;
Select the first of this neighbour’s neighbours that is clockwise from the joining line in the last section of the coastline ;
Add this neighbour to the coastline list;
Repeat the above two steps until we return to the starting vertex;

Note that in the code she kindly shared with us, Tarle considers edge cases such as “palindromes” in the coastline list caused by following single-cell wide fingers and she also includes some smoothing criteria to approximate the convex hull of the (confluent) vertices. Furthermore there is code to detect islands of cells or individual cells which have detached from the main bulk.

D Bowyer-Watson Algorithm

This is used to find the nearest-neighbour network. See the pseudocode below for a description of the algorithm. An actual implementation should use sensible data-structures (not necessarily plain lists).

Algorithm 3: Bowyer-Watson Algorithm

Input: Vertex List

Output: Triangles List

Side-Effect: Update All Vertex Nearest Neighbours

Initialise the triangles list;
Initialise a “supertriangle” which surrounds all the vertices in the vertex list;
Append the supertriangle vertices to the vertex list;
Append the supertriangle to the triangles list;
foreach current vertex \in vertex list **do**
 Initialise bad-edge list;
 foreach current triangle \in triangle list **do**
 Calculate current triangle’s circumcentre and circumradius;
 if current vertex lies inside current triangle’s circumcircle **then**
 Add the three triangle edges to the bad-edge list;
 Remove the current triangle from the triangles list;
 end
 end
 Remove all doubly-added edges in the bad-edge list;
 Add to the triangle list all triangles formed between the current vertex and the singly-added bad edges, which form an enclosing polygon around the current vertex;
end
Remove any triangles from the triangle list that are formed using any supertriangle vertex;
Remove the supertriangle vertices from the vertex list;

Nota Bene: Any time an edge is created, the bounding vertices should have each other added as nearest neighbours. Likewise any deletions should have the bounding vertices delete the corresponding nearest neighbour entries

E Numerical Values

The numerical values of constants used in our model. These are taken from Tarle et al. and Sepúlveda et al. Since the potentials are effective, we set the mass to be non-dimensional so that force is described in units of acceleration.

Variable	Description	Value
—	Maximum nearest neighbor distance	$70 \mu\text{m}$
U_0	Short-range repulsion magnitude	$2400 \mu\text{m}^2\text{h}^{-2}$
A_0	Range of short-range repulsion	$8 \mu\text{m}$
U_1	Long-range attraction magnitude	2h^{-2}
A_1	Beginning of long-range attraction	$35 \mu\text{m}$

Table 1: Model parameters used in the simulation.