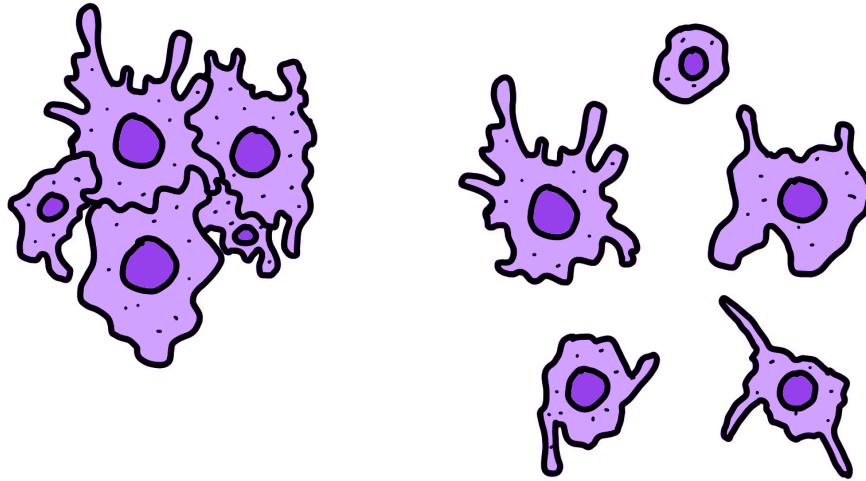




CHALMERS
UNIVERSITY OF TECHNOLOGY



Modelling and inference of interacting cell dynamics using single time-point images

Master's thesis in Engineering Mathematics and Computational Science

PATRIK FRANZÉN DENNIS

DEPARTMENT OF MATHEMATICAL SCIENCES

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2026
www.chalmers.se

MASTER'S THESIS 2026

Modelling and inference of interacting cell dynamics using single time-point images

PATRIK FRANZÉN DENNIS



CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Mathematical Sciences
Division of Applied Mathematics and Statistics
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2026

Modelling and inference of interacting cell dynamics using single time-point images
PATRIK FRANZÉN DENNIS

© PATRIK FRANZÉN DENNIS, 2026.

Supervisors: Adam Malik, Applied Mathematics and Statistics & Philip Gerlee,
Applied Mathematics and Statistics
Examiner: Philip Gerlee, Applied Mathematics and Statistics

Master's Thesis 2026
Department of Mathematical Sciences
Division of Applied Mathematics and Statistics
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Hand-drawn illustration of a cell population.

Typeset in L^AT_EX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2026

Modelling and inference of interacting cell dynamics using single time-point images
Patrik Franzén Dennis
Department of Mathematical Sciences
Chalmers University of Technology

Abstract

Understanding cell dynamics, such as migration, proliferation, and interactions are central in cancer biology, particularly for modeling invasive tumors like glioblastoma. High-throughput *in vitro* assays frequently rely on single time-point (endpoint) images, which lack direct temporal data, presenting a significant methodological challenge for parameter estimation. This thesis develops a comprehensive computational pipeline to infer interacting cell dynamics exclusively from endpoint phase-contrast microscopy images using Approximate Bayesian Computation (ABC).

The workflow integrates a robust U-Net segmentation model with a ResNet34 backbone and boundary-aware probability shaping, alongside a velocity-predictive tracking algorithm to mitigate survival bias and extract unbiased biological features. We evaluate an agent-based model utilizing overdamped Langevin dynamics proposed and used in against an advanced Fractional Brownian Motion (fBm) framework designed to capture directional persistence. Our results demonstrate that while the fBm model accurately captures the super-diffusive memory inherent to glioblastoma motility, this increased model capacity introduces parameter degeneracy at a single temporal endpoint. To navigate the resulting non-linear parameter manifolds, Sequential Monte Carlo (SMC-ABC) proved strictly superior to standard Rejection and Regression-Adjusted ABC variants.

Furthermore, we expose an important "Sim-to-Real Gap" regarding summary statistics: while continuous Topological Data Analysis (TDA) via Persistence Images excels on idealized *in vitro* data, it suffers from "topological fragility" when exposed to real-world biological noise and segmentation artifacts. Consequently, a hybrid summary statistic combining the Pair Correlation Function (PCF) with discrete Betti curves emerged as the most robust metric for experimental inference. Ultimately, this work establishes that extracting complex dynamic memory from static snapshots is mathematically viable, provided that physical model capacity is carefully balanced with robust spatial statistics and adaptive, non-linear Bayesian algorithms.

Keywords: Approximate Bayesian Computation, Fractional Brownian Motion, Topological Data Analysis, Cell Dynamics, Image Segmentation, Glioblastoma

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my supervisors, Adam Malik and Philip Gerlee. Thank you for your invaluable guidance and, most importantly, for granting me the absolute freedom to truly "scratch every itch" I had during this project. Your trust allowed me to chase down complex research questions, experiment with diverse methodologies, and independently explore how to solve them. This intellectual autonomy made the research process incredibly fulfilling.

On a personal note, this thesis would not have crossed the finish line without the unwavering support of my partner. Thank you for your endless patience, and especially for acting as my designated system administrator, checking my terminal far too many times while I was away from the desk to ensure my simulations hadn't crashed or thrown any errors.

Finally, a special thank you to my dog, Panko, whose quiet companionship during countless late nights at the keyboard provided the perfect grounding presence when the code refused to cooperate.

Patrik Franzén Dennis, Gothenburg, June 2026

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

ABC	Approximate Bayesian Computation
Cdk	Cyclin-dependent kinase
CNN	Convolutional Neural Network
DFT	Discrete Fourier Transform
DSC	Dice Similarity Coefficient
ECM	Extracellular Matrix
fBm	Fractional Brownian Motion
FCN	Fully Convolutional Network
FFT	Fast Fourier Transform
fGn	Fractional Gaussian Noise
HGCC	Human Glioblastoma Cell Culture
IFFT	Inverse Fast Fourier Transform
IoU	Intersection over Union
KDE	Kernel Density Estimate
MAE	Mean Absolute Error
MSD	Mean Squared Displacement
PCF	Pair Correlation Function
SDE	Stochastic Differential Equation
SMC	Sequential Monte Carlo
TDA	Topological Data Analysis
VR	Vietoris-Rips

Nomenclature

Below is the nomenclature of indices, sets, parameters, and variables that have been used throughout this thesis.

Indices

d	Spatial dimension
i, j	Index for individual cells or particles
k	Index for lag in autocovariance / SMC generation
p	Homological dimension
t	Index for time step
u	Index for spatial coordinates/pixels in probability maps

Sets

$\mathbb{R}^d$	d -dimensional Euclidean space
X	Set of spatial cell coordinates (point cloud)
$\mathcal{S}$	Simplicial complex
$\mathcal{A}(X)$	Alpha complex of the point cloud X
V	Vertex set of a simplicial complex

Parameters

D	Diffusion coefficient
H	Hurst parameter
N	Total number of simulated cells
N_p	Number of particles in the SMC ABC algorithm
T	Total simulation time

r	Cell radius
α	Anomalous diffusion exponent / Proliferation rate
β	Crowding sensitivity
Δt	Time step size
ϵ_k	Tolerance threshold for ABC at generation k
ρ	Growth slope

Variables

$B^H(t)$	Fractional Brownian Motion process
$F(X)$	Forces acting on cells
$g(r)$	Pair correlation function
$P(u)$	Boundary-aware probability map
$p_{adj}(u)$	Adjusted boundary-aware probability map
$s(x)$	Summary statistics of the data
$v_i(t)$	Velocity of cell i at time t
$W_k^{(i)}$	Weight of particle i at SMC generation k
$X(t)$	Position of a cell at time t
β_0	0-dimensional Betti number
β_1	1-dimensional Betti number
β_p	p -dimensional Betti number
$\gamma(k)$	Autocovariance
ΔW^H	Correlated Gaussian increments
θ	Vector of model parameters
$\rho(x, y)$	Distance metric between summary statistics

Contents

List of Acronyms	ix
Nomenclature	xi
List of Figures	xvii
List of Tables	xxi
1 Introduction	1
1.1 Historical Context and Motivation	1
1.2 Purpose and Objective	2
1.3 Research Questions	2
1.4 Contributions and pipeline overview	3
2 Theory	5
2.1 Biological background: cell growth, motility and interactions	5
2.1.1 Cell migration as a mechanochemical cycle	5
2.1.2 Environmental guidance cues and mechanotransduction	6
2.1.3 Modes of migration and phenotypic plasticity	7
2.1.4 Proliferation and density regulation	7
2.1.5 Cell-cell and cell-matrix interactions	8
2.1.6 Connection to the modelling and inference pipeline	9
2.2 Models for segmentation	9
2.2.1 Probabilistic segmentation framework	10
2.2.1.1 Boundary-aware probability mapping	10
2.2.2 Convolutional neural networks and encoder-decoder architectures	11
2.2.2.1 U-Net and residual encoders	12
2.2.2.2 U-Net++: nested skip connections	13
2.2.3 Transformer-based segmentation and SegFormer	13
2.2.4 Fogbank	14
2.2.5 Training objective and loss functions	14
2.2.6 Evaluation metrics	15
2.3 The Role of Trajectory Statistics	16
2.4 Approximate Bayesian Computation	17

2.4.1	Distance functions and summary statistics	17
2.4.2	Rejection ABC	18
2.4.3	Regression-adjusted ABC	19
2.4.3.1	Kernel weighting	21
2.4.4	Sequential Monte Carlo ABC	23
2.5	Modelling Cell Population Dynamics	26
2.6	Velocity Driven models	27
2.6.1	Fractional stochastic calculus and fractional Brownian motion	27
2.6.1.1	Fractional calculus preliminaries	28
2.6.1.2	How the same kernels reappear in fractional Gaussian processes.	28
2.6.1.3	Fractional Brownian motion	29
2.6.1.3.1	Properties.	29
2.6.1.3.2	Non-stationarity.	29
2.6.1.3.3	FBm has stationary increments.	29
2.6.1.3.4	Self-similarity.	29
2.6.1.3.5	Persistence vs. anti-persistence.	29
2.6.1.4	Fractional Gaussian noise and increment covariance	30
2.6.1.5	Long-range dependence	31
2.6.1.6	Spectral density of fractional Gaussian noise	32
2.6.1.7	Davies–Harte simulation of fGn and fBm	32
2.6.1.7.1	Circulant embedding of size $M = 2N$	33
2.6.1.7.2	Remark (alternative embedding size $M = 2(N - 1)$).	33
2.6.1.7.3	FFT diagonalisation and the sampling formula	33
2.6.1.7.4	Sampling in the Fourier domain.	34
2.6.1.7.5	From fGn to fBm by cumulative summation	34
2.7	Topological Data Analysis	34
2.7.1	Foundations of Point-Set Topology and Metric Spaces	35
2.7.2	Simplicial Complexes: Bridging the Discrete and Continuous	35
2.7.2.1	Čech and Vietoris-Rips Complexes	36
2.7.3	Algebraic Topology and Simplicial Homology	37
2.7.3.1	Chain Complexes and the Boundary Operator	37
2.7.3.2	Cycles, Boundaries, and the Homology Quotient	38
2.7.4	Persistent Homology and Stability	39
2.7.5	Geometric Exactness: Delaunay Triangulations and the Alpha Complex	40
2.7.6	Continuous Topological Summaries: Persistence Images	41
3	Methods	45
3.1	Data and experimental design	45
3.1.1	Cell lines, time intervals, and dataset splitting	46
3.2	Segmentation and tracking	46
3.2.1	Preprocessing and annotation protocol	47
3.2.2	Model configurations and training procedure	48

3.2.2.1	Training objective and optimisation protocol	48
3.2.3	Inference-time probability mapping and instance extraction . .	49
3.2.4	Evaluation protocol and metrics	50
3.3	Cell Tracking and Trajectory Reconstruction	50
3.3.1	Temporal Resolution and Precise Timestamping	50
3.3.2	Evaluated Tracking Strategies	51
3.3.3	Gap Closing and Filtering	51
3.3.4	Data-Driven Validation of Velocity Persistence	51
3.4	Feature extraction	52
3.4.1	Radius feature from instance masks	52
3.4.2	Diffusion feature via mean squared displacement	53
3.4.3	Directional persistence via the Hurst parameter	54
3.4.3.0.1	Drift correction.	54
3.4.4	Proliferation rate from cell-count time series	54
3.5	Temporal consistency of derived quantities	54
3.5.1	Radius stability under increasing density	55
3.5.2	Consistency of mean squared displacement	55
3.6	Simulation models and numerical algorithms	55
3.6.1	Numerical integration via the Euler-Maruyama scheme	55
3.6.2	Algorithmic implementation of edge forces	56
3.6.3	Generation of fractional Gaussian noise via the Davies-Harte method	56
3.6.3.0.1	Implementation note (NumPy FFT scaling).	56
3.7	Summary statistics for Approximate Bayesian Computation	57
3.7.1	Baseline endpoint summaries and distance	57
3.7.2	Advanced topological extraction and hybrid summaries	58
3.7.2.1	Hybrid summary framework	59
3.8	Data-driven prior extraction	60
3.9	Approximate Bayesian Computation implementation	60
3.10	Simulation models and numerical algorithms	61
3.10.1	State space and domain	61
3.10.2	Baseline agent-based model	61
3.10.2.1	Overdamped stochastic dynamics	61
3.10.2.2	Cell-cell exclusion force	62
3.10.2.3	Step-aware edge forces for bounded stochastic motion	62
3.10.3	Stochastic proliferation with contact inhibition	65
3.10.4	Numerical scheme: coupled diffusion and events	66
3.10.5	Implementation for computational efficiency	66
3.10.6	Neighbour search via KD-trees	67
3.10.7	Vectorised force accumulation and memory layout	67
3.10.8	Parallel simulation of independent replicates	67
3.10.9	Summary	67
3.10.10	Modelling Discrete-time velocity and Davies-Harte simulation	67
4	Results	71
4.1	Segmentation results	71

4.1.1	Quantitative comparison	71
4.1.2	Density-Dependent Performance and Boundary Preservation	71
4.1.3	Qualitative comparison	73
4.1.4	Temporal radius stability and MSD consistency	73
4.1.5	Evaluation of Tracking Strategies	74
4.2	Feature extraction & Characterization of the Prior Search Space	75
4.3	Inference results	75
4.3.1	Performance on Synthetic Data	75
4.3.2	Parameter Inference on Experimental Data	76
4.3.3	Inference Incorporating Fractional Brownian Motion	79
4.3.4	Model Capacity and Parameter Identifiability	80
4.3.5	Topological Feature Extraction and Synthetic Performance	83
5	Discussion	87
5.1	Segmentation Mechanics: Inductive Bias and Boundary Preservation	87
5.2	Segmentation and Tracking: Overcoming Survival Bias	88
5.3	Model Selection and the Cost of Capacity (fBm)	88
5.4	ABC Algorithmic Hierarchy	89
5.5	The Sim-to-Real Gap and Topological Fragility	90
5.5.1	The Information Limit of Endpoint Assays: Integrals vs. Suprema	90
5.6	The Mechanism of the Hybrid Summary Statistic	91
5.7	Limitations	92
5.8	Future Work	92
5.8.1	Bridging the Sim-to-Real Inference Gap	92
5.8.2	Preliminary Exploration of Gradient-Based Contour Extraction (The Franzeg Algorithm)	93
6	Conclusion	95
6.1	Segmentation and Feature Extraction	95
6.2	Parameter Estimation and Mechanistic Modelling	95
6.3	The Efficacy of Summary Statistics	96
6.4	Final Remarks	96
	Bibliography	97

List of Figures

1.1	High-level pipeline overview.	4
2.1	Illustration of cell migration. Cells polarize and extend protrusions to drive motion, engaging in a continuous cycle of leading-edge protrusion and rear retraction.	6
2.2	The eukaryotic cell cycle. The cell grows during Interphase (G_1 , S, and G_2) before undergoing active division during the Mitotic (M) phase.	8
2.3	Illustration of cell proliferation and the impact of local density. As cultures transition from a freely dividing sparse state (b, right) to a densely packed state (b, left), contact inhibition suppresses the cell cycle machinery.	8
2.4	Illustration of short-range steric repulsion between cells. When physical boundaries overlap or interact, repulsive forces (red arrows) enforce volume exclusion and spatial organization.	9
2.5	Comparison of low and high-density cell populations in projection. In the sparse culture (a), individual cells and their boundaries are isolated and clearly distinguishable. In the dense culture (b), cells frequently touch and physically overlap, forming complex, interconnected clusters complicating accurate boundary segmentation.	11
2.6		16
2.7	Schematic of the ABC Rejection algorithm. Parameters are sampled from the prior and accepted only if the simulated summary statistics are sufficiently close to the observed statistics.	19
2.8	Schematic of ABC Regression adjustment. The model learns the local relationship between statistics and parameters to correct the posterior approximation.	21
2.9	Schematic of SMC-ABC. The algorithm evolves a population of particles through a sequence of decreasing tolerances. In each generation, particles are perturbed, filtered, and re-weighted to approximate the posterior distribution more closely.	24
2.10	Schematic illustration of the qualitative effect of the Hurst parameter on increment correlations: persistent paths tend to keep direction, while anti-persistent paths tend to revert.	30

2.11	Geometric representations of k -simplices for $k \in \{0, 1, 2, 3\}$. A simplicial complex is constructed by gluing these fundamental geometric building blocks together along their faces.	36
2.12	Construction of a Vietoris-Rips complex $VR_\epsilon(X)$. Disks of radius $\epsilon/2$ are drawn around each point. When two disks intersect (pairwise distance $\leq \epsilon$), an edge is formed. Because all three vertices of the central triangle are pairwise connected, the 2-simplex (shaded blue) is automatically added.	37
2.13	A VR complex filtration over increasing spatial scales ϵ . At ϵ_1 , the complex consists of discrete points ($\beta_0 = 4, \beta_1 = 0$). At ϵ_2 , an empty triangle forms ($\beta_0 = 2, \beta_1 = 1$). At ϵ_3 , the triangle is filled by a 2-simplex, and the isolated point connects, yielding a single connected component and eliminating the hole ($\beta_0 = 1, \beta_1 = 0$).	39
2.14	The geometric flaw of the Vietoris-Rips complex in 2D space. When four cells (black nodes) are pairwise connected within the threshold ϵ , the VR flag complex automatically hallucinates a solid 3-simplex (a 3D tetrahedron, shaded red). This forces 2D biological data into a false 3D geometry.	40
2.15	Resolving the geometric flaw using spatial partitioning. (a) The 2D plane is partitioned into Voronoi territories, where red dashed borders indicate equidistant boundaries between cells. (b) The dual Delaunay Triangulation connects cells only if their territories share a border. Because 2D borders cannot cross, the resulting planar web mathematically forbids 3D self-intersections.	41
3.1	Visualisation of the well-plate coordinate system.	45
3.2	Schematic of the data curation and splitting strategy. From the raw HGCC dataset [56], 127 frames were manually selected to maximize morphological variance (e.g., distinguishing filopodia and lamellipodia) and temporal coverage. Validation and Test sets are blinded to ensure the model generalizes to unseen cell lines and timepoints. . . .	47
3.3	Example of inference-time probability fields produced by the segmentation model: original input, cell-interior probability, boundary probability, and the adjusted map constructed by Eq. (2.1). Bright regions correspond to high class confidence. Boundary-aware subtraction suppresses connectivity across cell-cell contacts, facilitating instance separation after thresholding.	49
3.4	Comparison of instance segmentation prediction overlays (left) and the resulting transformed geometric radius models (right).	53
3.5	Conceptual illustration of the Pair Correlation Function ($g(r)$). Left: A simulated centroid pattern showing a reference cell and its steric exclusion zone ($2R$). Right: The corresponding continuous $g(r)$ curve, capturing the hard-core exclusion limit at small radii, followed by a local clustering peak before decaying to spatial randomness.	58
3.6	The one-step exit probability in (3.13) is a Gaussian left-tail probability induced by the Euler-Maruyama update.	63

3.7	Rectangular domain with interior boundary bands (light blue). Inside a band, a constant inward drift is applied along the wall normal; outside the band, no edge force is applied.	64
3.8	Schematic of the Davies–Harte method: embed the Toeplitz covariance in a circulant matrix, use FFT to obtain eigenvalues, sample in the Fourier domain, and invert by FFT to recover correlated increments.	68
4.1	Distribution of Dice scores across evaluated segmentation configurations. Each row corresponds to one trained model or baseline method (as encoded by experiment identifiers). The selected model is the one that provides the best trade-off between median overlap and robustness to outliers in dense regimes.	72
4.2	Example of result from mismatch evaluation software, showing False Negatives (green), False Positives (magenta) and True Positives (white).	74
4.3	Sanity check on A10: radius evolution over time. A stable mean radius without systematic drift supports the assumption that the segmentation model does not progressively merge cells at high density. Details of the radius estimator are provided in the later cell-dynamics section.	75
4.4	Sanity check on A10: mean squared displacement (MSD) curve and fitted diffusion trend over the fitting regime. Agreement supports that segmentation and tracking yield stable displacement statistics. The diffusion estimator is described in the later cell-dynamics methodology section.	76
4.5	Data-Driven Definition of Simulation Priors. (a) Time evolution of the mean cell radius, showing stability over 4 days. (b) Distribution of growth rates ($\log_{10} \alpha_{birth}$), indicating a slow-cycling population. (c) Distribution of Diffusion Coefficients ($\log_{10} D$), spanning an order of magnitude. (d) Distribution of the Hurst Parameter (H). The dashed black line at $H = 0.5$ represents random Brownian motion; the observed shift to higher values ($H \in [0.47, 0.83]$) indicates super-diffusive persistence.	77
4.6	Scatter plot of estimated parameter values against true values in the collection $\theta = (\log_{10} \alpha, \log_{10} D, r)$ generated by the synthetic data set. Visual tightening around the identity line demonstrates the correction of tolerance-induced bias.	78
4.7	Scatter plot of estimated parameter values against true values in the collection $\theta = (\log_{10} \alpha, \log_{10} D, r)$ for the experimental datasets using the Langevin model.	79
4.8	Scatter plot of estimated parameter values against true values in the collection $\theta = (\log_{10} \alpha, \log_{10} D, r, H)$ for the Fractional Brownian Motion model.	82
4.9	Multimodal feature extraction for a single simulation, showing local spatial mechanics (PCF), discrete topological features (Betti curves), and continuous topological robustness (Persistence Images).	83

4.10	Alpha complex and persistence diagram for synthetic data. The strict physical rules of the simulator generate clean, mathematically perfect voids (H_1) that are easily captured by TDA.	85
4.11	Alpha complex and persistence diagram extracted from real in-vitro data. Segmentation errors and diffuse boundaries shatter the pristine topology, generating massive amounts of low-persistence "topological dust" (features clustered near the diagonal line).	85
4.12	Model performance (Mean Absolute Error) stratified by cellular diffusion. Lower values indicate better parameter recovery. High diffusion generally decreases the accuracy of proliferation and radius estimates.	86
4.13	Predicted vs. True parameters for in-vitro experimental data. The horizontal banding indicates metric collapse in the continuous TDA models, while discrete topology (Betti) and spatial mechanics (PCF) demonstrate greater robustness to real-world morphological and proliferative noise, respectively.	86

List of Tables

4.1	Segmentation performance on the test set (sorted by mean Dice). . .	73
4.2	Global Comparison of Tracking Strategies ($N = 96$ Replicates)	74
4.3	Data-Driven Bounds for Simulation Parameters & Characterization .	76
4.4	Pearson Correlation Coefficients (R) for parameter inference across models and ABC methods on in-vitro data.	80

1

Introduction

Cell populations show rich dynamical behaviour driven by the combined effects of migration, cell-cell interactions, and proliferation. Quantifying these mechanisms is central in understanding and modelling cell growth. More specifically, invasive cancers such as glioblastoma can spread through surrounding tissue by persistent single-cell and collective migration, while tumour growth additionally depends on proliferation and density-dependent inhibition together with morphological traits such as cell sizes. Reliable estimates of migration, proliferation rates and cell morphology therefore provide a compact but informative description of how a cell line expands and reorganises over time. Thus furthering our understanding of cancer modelling.

In vitro two-dimensional (2D) cell culture assays are widely used to study cell dynamics, where one can observe how cancer cells migrate, proliferate, and interact in controlled experimental settings. Time-lapse imaging provides direct access to dynamic behaviours through trajectories and division events, but it is resource-intensive and limits experimental throughput. In contrast, end-point imaging records the population at a single final time and is therefore well-suited for high-throughput experimental designs. The drawback however is that it removes direct temporal information. This creates a central methodological challenge: *how can one infer parameters governing dynamic cell behaviours from a single image of the final population configuration?* A framework proposed in [44] addresses this by combining an agent-based model of interacting cells with likelihood-free Bayesian inference, where Approximate Bayesian Computation (ABC) matches summary statistics computed from simulations to those extracted from an observed end-point assay. Given the benefits seen using a novel yet simple framework in [44], one may wonder how far endpoint imagery can take us, if one enhances the framework.

1.1 Historical Context and Motivation

To extract information from static endpoint images, recent methodologies have utilized likelihood-free inference, specifically Approximate Bayesian Computation (ABC) [54, 9, 44]. The framework established by [44] demonstrates parameter estimation by simulating an agent-based model and comparing the final simulated

configuration to the experimental data using carefully chosen summary statistics.

In this framework, parameter estimation is driven by two primary summary statistics that quantify the discrepancy between the simulated data s and the experimental observation e . The first statistic calculates the normalized error in the final cell count, N

$$E_n = \frac{N_e - N_s}{1 + N_e}. \quad (1.1)$$

The second statistic captures the spatial arrangement and clustering of the Pair Correlation Function (PCF), evaluated across K discrete spatial bins,

$$E_{pcf} = \frac{1}{K} \sum_{i=1}^K \left(\frac{\text{PCF}_e(i) - \text{PCF}_s(i)}{1 + \text{PCF}_e(i)} \right)^2. \quad (1.2)$$

The correlation between estimated and real parameter values that followed from the chosen summary statistics and ABC rejection were 0.87, 0.50 and 0.52 for proliferation, diffusion and radius respectively.

1.2 Purpose and Objective

The purpose of this thesis is to advance the modelling and inference of interacting cell dynamics from single end-point images. Concretely, the objective is to improve parameter estimation for end-point 2D cell culture assays by enhancing and systematically comparing key stages of the modelling-inference framework or pipeline, in particular the framework presented in [44].

The enhancements to the baseline framework are the following:

1. Cell segmentation
2. Modelling for migration, interaction and proliferation
3. The choice of summary statistics (and extracted features) used to compare simulated and observed end-point configurations
4. ABC inference strategy used to approximate posterior distributions over model parameters

1.3 Research Questions

This previously stated objective can be summarised and organised around the following layers together with their subsequent questions.

1. **Segmentation and feature extraction:** how reliably can cell configurations (positions, sizes and trajectories when time-lapse is available) be extracted

from microscopy images, and what segmentation choices are required for robust downstream inference?

2. **Tracking:** using a data-driven approach, are velocity driven models with persistence motivated?
3. **Model:** to what extent does a baseline positional interacting-particle model reproduce the spatial structure of end-point assays, and do velocity-driven models that incorporate directional persistence produce more realistic population configurations?
4. **Summary statistics:** which summary statistics are most informative about key parameters (e.g. proliferation, motility, interaction scale) when only an end-point configuration is observed, and how does the choice of summary statistics affect identifiability and uncertainty?
5. **Inference:** how do different ABC variants trade off computational cost and posterior accuracy in this setting, and when are sequential or regression-adjusted methods preferable to a rejection baseline?

1.4 Contributions and pipeline overview

This thesis develops a complete, end-to-end computational pipeline that transforms raw phase-contrast microscopy images into inferred mechanistic parameters. The workflow progresses sequentially: it utilizes a boundary-aware deep learning architecture to extract accurate cellular segmentations from dense cultures, computes macroscopic spatial features using Topological Data Analysis (TDA), and relies on Sequential Monte Carlo Approximate Bayesian Computation (SMC-ABC) to calibrate the underlying agent-based models.

The primary contribution, and the analytical punchline of this work, is demonstrating that sophisticated likelihood-free inference on biological data is only reliable when the intermediate mathematical representations are inherently resistant to noise. A fundamental methodological challenge is that errors introduced early in the pipeline (such as under-segmentation in dense cultures) aggressively propagate into downstream statistics and severely bias the inferred parameters. By integrating exact Alpha Complexes and continuous Persistence Images into the inference pipeline, this thesis provides a topological framework that mathematically filters out the stochastic artifacts of overlapping cells. This robustness allows the SMC-ABC algorithm to confidently and accurately recover the true physical contact-inhibition and motility thresholds from the experimental data.

To further ensure the integrity of this inference, the framework incorporates rigorous data-quality sanity checks. These checks confirm that extracted kinematic and geometric features, such as approximately stable cell radii in non-growing cell lines, exhibit biologically plausible behaviours before executing the computationally

1. Introduction

expensive ABC simulations. The complete integration of these methodologies is summarized in Figure 1.1.

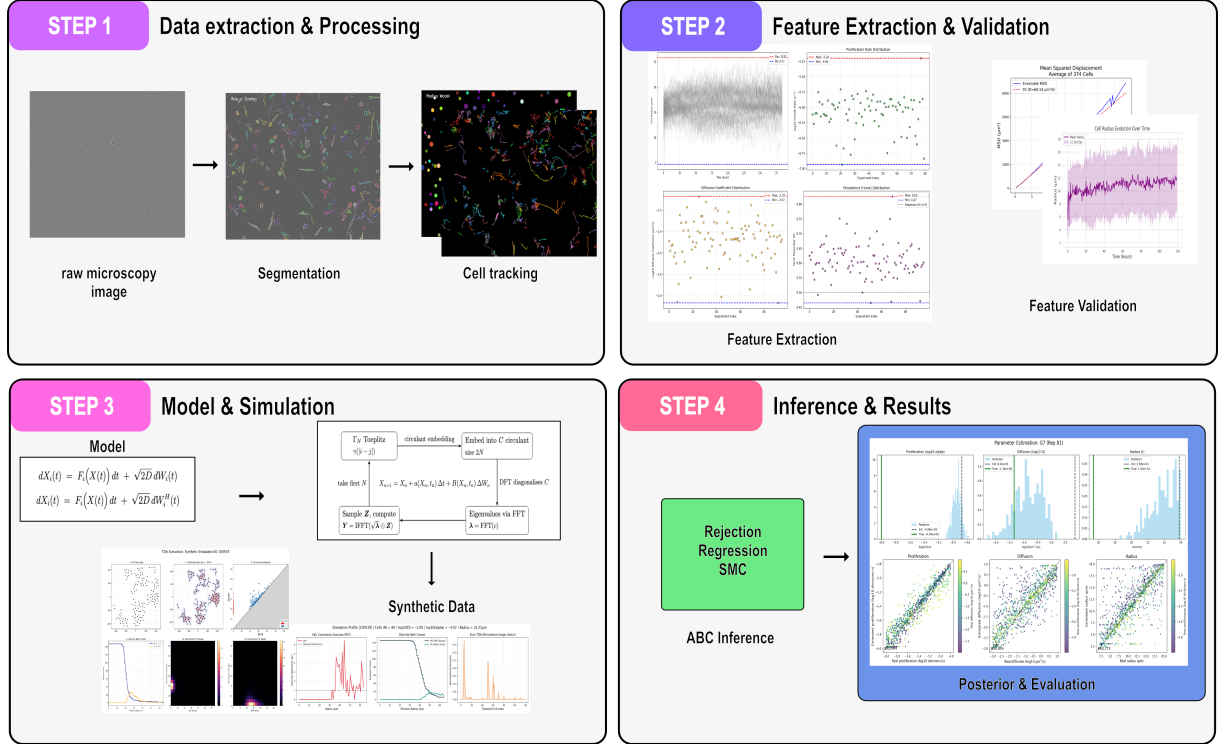


Figure 1.1: High-level pipeline overview.

2

Theory

This chapter aims to give the theoretical background for this thesis to an extent which is necessary to understand the following chapters and in particular all used methods. The reader is assumed to have an understanding of stochastic calculus and probability theory. Apart from the two mentioned, an undergraduate-level knowledge of mathematics suffices. Nice-to-have prerequisites is knowledge of neural networks, topology and fractional stochastic calculus.

2.1 Biological background: cell growth, motility and interactions

Collective cell dynamics observed in *in vitro* assays emerge from coupled processes of cell growth and division, migration, and mechanical or biochemical interactions with neighbouring cells and the extracellular environment. These processes are regulated by the cell cycle machinery and by signal transduction pathways that coordinate cytoskeletal remodelling, adhesion, and force generation. [2]

2.1.1 Cell migration as a mechanochemical cycle

Cells have been found to migrate by repeating the mechanochemical cycle shown in Figure 2.1 consisting of polarity establishment, protrusion at the leading edge, adhesion and traction generation, and rear retraction. Front–rear polarity localises signalling molecules and cytoskeletal regulators so that protrusive activity is biased to one side of the cell [2, 42].

At the leading edge, actin polymerisation generates membrane protrusions such as lamellipodia and filopodia [2, 42]. Filopodia are identified by thin extensions, while lamellipodia have broad extensions at the leading edge (as illustrated by the broad ruffled fronts in Figure 2.1). These protrusions couple to the extracellular matrix (ECM) through protein complexes, also known as focal adhesions [2, 24]. Contractile forces generated by myosin motors acting on actin filaments then transmit traction to the substrate, pulling the cell body forward while the rear detaches and retracts [2, 42]. A key physical feature of this cycle is that actin networks can exhibit retrograde flow; effective forward motion requires mechanical coupling between the

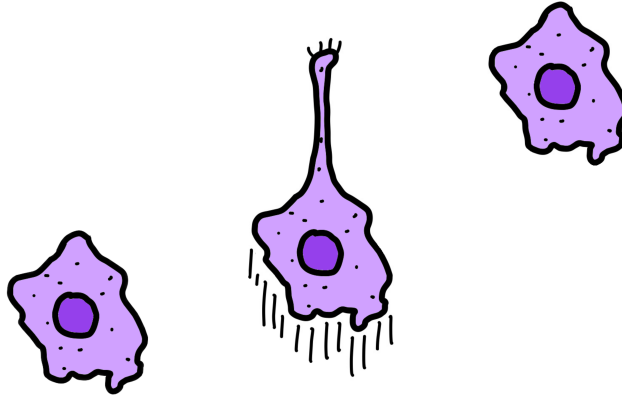


Figure 2.1: Illustration of cell migration. Cells polarize and extend protrusions to drive motion, engaging in a continuous cycle of leading-edge protrusion and rear retraction.

cytoskeleton and the substrate through engaged adhesions [2, 12].

This perspective is often summarised as a "motor-clutch" mechanism. The motor consisting of actin polymerisation and myosin contraction providing active force generation, while the clutch is due to adhesion complexes providing load-bearing coupling [12, 2]. Adhesions are dynamic and mechanosensitive, assembling, maturing, and disassembling in a force-dependent manner, which feeds back into migration speed and persistence [24, 52].

2.1.2 Environmental guidance cues and mechanotransduction

Cell migration is not purely random. Cells bias their polarity and traction in response to chemical and mechanical cues, a process broadly referred to as mechanotransduction and cue-guided motion [2, 52]. Common guidance modes include:

- Chemotaxis: biased migration along gradients of soluble signalling molecules [42].
- Haptotaxis: biased migration along gradients of substrate-bound ligands or adhesivity [42].
- Durotaxis: biased migration towards regions of higher substrate stiffness, often attributed to stiffness-dependent adhesion reinforcement and traction asymmetries [34].
- Contact guidance: alignment and channeling of motion by ECM fibre anisotropy or topographic features [22].

In confined environments, additional cues such as hydraulic resistance and pressure gradients can bias motion, and some cell types also respond to external electric fields. Such electric cues fall under galvanotaxis [52]. The relative importance of these cues depends strongly on cell type, ECM composition, stiffness, and geometric confinement [22, 52].

2.1.3 Modes of migration and phenotypic plasticity

Cells can realise migration through different physical modes. In mesenchymal migration, motion is typically adhesion-dominated: protrusions form, adhesions transmit traction, and the cell advances via coordinated contraction and rear retraction [22, 42]. In contrast, amoeboid or bleb-based migration can dominate when adhesion is weak or confinement is strong [22]. Here, local membrane-cortex detachment and intracellular pressure form blebs, and motion can occur through frictional coupling with surrounding boundaries or through hybrid mechanisms that combine rapid bleb protrusions with transient adhesions [22].

While *in-vivo* environments provide diverse mechanical inputs such as three-dimensional extracellular matrices and structural confinement that can trigger these varied migration modes, cells grown in standard 2D adherent cultures lack this complex environmental context. In the flat *in-vitro* assays analyzed in this work, three-dimensional confinement is negligible and motility is principally governed by substrate adhesion and planar cell-to-cell contacts. Consequently, mesenchymal migration is typically the dominant physical mode in these experimental setups.

Importantly, cells may transition between modes depending on mechanical conditions or external perturbations, implying that a single effective description (e.g. an overdamped stochastic process with an effective diffusion coefficient) may represent an aggregate of multiple underlying motility mechanisms [52]. This motivates treating model parameters later in the thesis as effective or phenomenological summaries of complex mechanochemical behaviour.

2.1.4 Proliferation and density regulation

Population expansion depends on proliferation in addition to migration. Proliferation is governed by the cell-division cycle (Figure 2.2), which alternates M-phase (mitosis and cytokinesis) with interphase (G_1 , S, and G_2). Progress through the cycle is controlled by a kinase-based checkpoint system, where cyclin-dependent kinases (Cdks) activated by cyclins drive transitions, and checkpoints delay progression until key conditions are satisfied (e.g. DNA integrity, replication completion, and correct spindle attachment) [2].

In adherent monolayers, proliferation is often regulated by density-dependent mechanisms such as contact inhibition: as local density increases, cells may slow or halt division [52, 2]. From a modelling perspective, this motivates proliferation mechanisms whose rates decrease with crowding, which is consistent with the observation

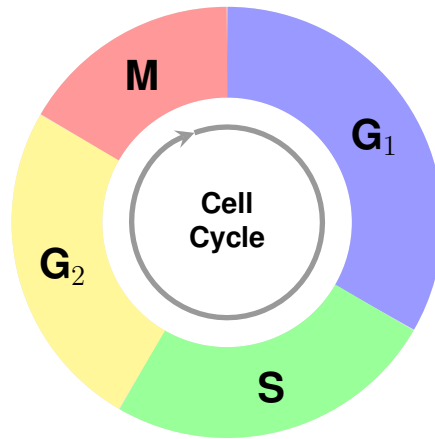


Figure 2.2: The eukaryotic cell cycle. The cell grows during Interphase (G_1 , S , and G_2) before undergoing active division during the Mitotic (M) phase.

that dense cultures can stabilise in a near-confluent state even in growth-permissive conditions (Figure 2.3).

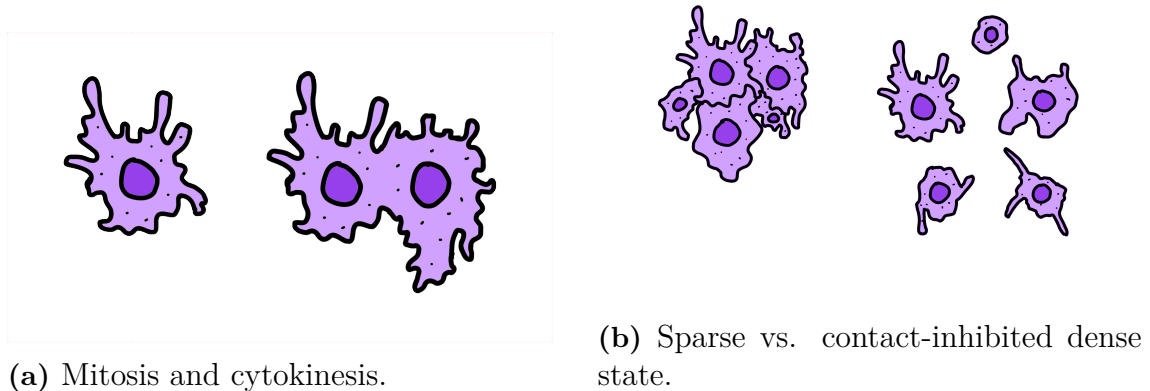


Figure 2.3: Illustration of cell proliferation and the impact of local density. As cultures transition from a freely dividing sparse state (b, right) to a densely packed state (b, left), contact inhibition suppresses the cell cycle machinery.

2.1.5 Cell–cell and cell–matrix interactions

Migration and proliferation occur in the context of physical and biochemical interactions. Cells adhere to the ECM primarily through integrins, which provide bidirectional signalling and mechanical linkage to the cytoskeleton [24, 2]. Cell–cell adhesion is frequently mediated by cadherins in adherens junctions and desmosomes, which provide cohesion and transmit forces across tissues [2]. The ECM itself is a composite material containing fibrous proteins (e.g. collagen), adhesive glycoproteins (e.g. fibronectin, laminin), and hydrated proteoglycan gels, and can be remodelled by cells through protease secretion and force-mediated reorganisation [2, 22].

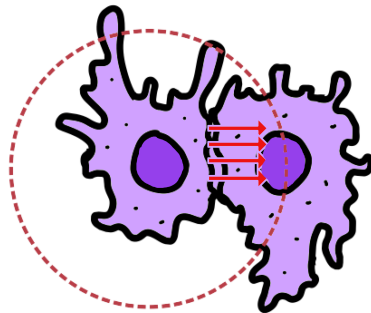


Figure 2.4: Illustration of short-range steric repulsion between cells. When physical boundaries overlap or interact, repulsive forces (red arrows) enforce volume exclusion and spatial organization.

In simplified in-vitro assays on planar substrates, many of these processes are coarse-grained into effective interaction rules: short-range steric repulsion prevents unphysical overlap (Figure 2.4), while longer-range attraction/adhesion can encourage clustering or collective organisation. This coarse-graining motivates the interaction potentials introduced later in the thesis, which aim to capture the net effect of multiple molecular interaction pathways.

2.1.6 Connection to the modelling and inference pipeline

The biological mechanisms above motivate the modelling choices made later in the thesis. At the single-cell level, polarity-driven migration with many small stochastic perturbations can often be represented by an effective overdamped stochastic process [52]. At the population level, cell-cell interactions and density-dependent proliferation shape collective organisation. Furthermore, the "motor-clutch" mechanism motivates directional persistence in cells [12].

2.2 Models for segmentation

The segmentation stage takes raw phase-contrast microscopy images and produces pixel-wise labels indicating whether each pixel belongs to background, cell interior, or cell boundary. In the implementation used here, segmentation is formulated as a supervised, pixel-wise classification problem. Additionally the identification of every individual object in an image at the pixel level, assigning a unique mask to each instance is named instance segmentation.

To begin with we summarise our deep learning model, which in this thesis, learns the mapping

$$X \in \mathbb{R}^{H \times W \times 1} \mapsto \hat{Y} \in [0, 1]^{H \times W \times 3},$$

where X is a single-channel input image, H and W are the height and width of the image respectively and $\hat{Y}$ is a three-channel output encoding, for each pixel, the predicted probabilities of background, cell interior, and boundary.

2.2.1 Probabilistic segmentation framework

Modern learning-based segmentation models do not predict a binary mask directly, but instead output a spatial probability field that expresses confidence that each pixel belongs to a particular semantic class.

This stage is set by first u denoting a pixel location in the image domain $\Omega \subset \mathbb{Z}^2$. The network produces, for each $u \in \Omega$, a vector of class probabilities

$$P(u) = (p_{\text{bg}}(u), p_{\text{cell}}(u), p_{\text{bnd}}(u)), \quad p_{\text{background}}(u) + p_{\text{cell}}(u) + p_{\text{boundary}}(u) = 1.$$

Internally, the network outputs real-valued logits $z_c(u)$ for each class $c \in \{\text{background, cell, boundary}\}$, which are converted into probabilities by a softmax activation [26],

$$p_c(u) = \frac{\exp(z_c(u))}{\sum_{c'} \exp(z_{c'}(u))}.$$

The fields $p_{\text{background}}$, p_{cell} and p_{boundary} can be visualised as probability heatmaps, where bright regions indicate high confidence for the corresponding class. An example of the three predicted probability fields together with the adjusted map used for instance extraction is shown in Figure 3.3 in the main segmentation chapter.

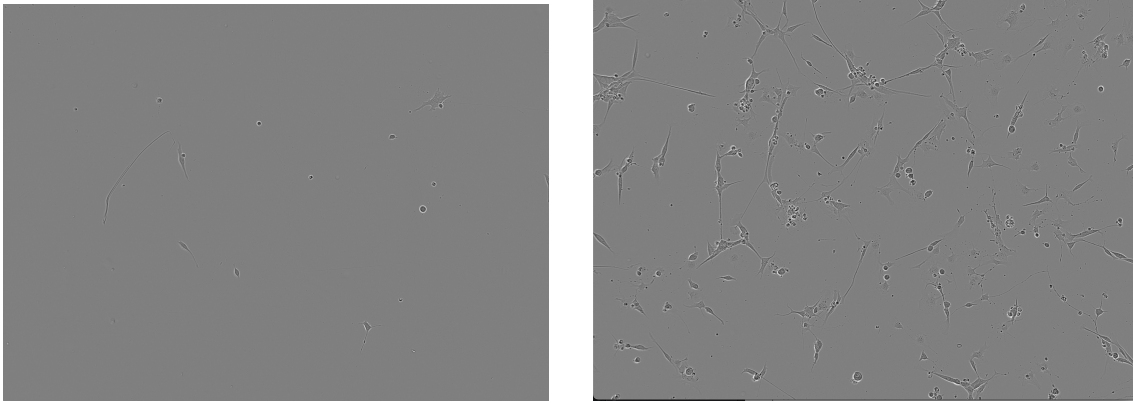
2.2.1.1 Boundary-aware probability mapping

A key difficulty in dense biological cultures is the tendency for neighbouring cells to touch or overlap in projection (see Figure 2.5). If one were to threshold the cell probability $p_{\text{cell}}(u)$ directly, contacting cells would often merge into a single connected component, leading to substantial under-segmentation. To mitigate this, the boundary class probability is explicitly incorporated as a repulsive term that suppresses high-confidence cell predictions in regions that the network believes correspond to cell-to-cell boundaries.

Concretely, we construct a modified probability field

$$p_{\text{adjusted}}(u) = p_{\text{cell}}(u) - \lambda p_{\text{boundary}}(u), \tag{2.1}$$

where $\lambda > 0$ is a scalar hyperparameter that controls the strength of boundary suppression. In practice, λ is determined empirically via a one-dimensional grid search over a validation subset, selecting the value that optimally resolves dense cell clusters without spuriously fragmenting isolated cells. Intuitively, where the



(a) Low density (VID558_A12_....)

(b) High density (VID558_A1_....)

Figure 2.5: Comparison of low and high-density cell populations in projection. In the sparse culture (a), individual cells and their boundaries are isolated and clearly distinguishable. In the dense culture (b), cells frequently touch and physically overlap, forming complex, interconnected clusters complicating accurate boundary segmentation.

model is confident that a pixel is boundary, $p_{\text{boundary}}(u)$ is large and $p_{\text{adjusted}}(u)$ is reduced, effectively carving a "trench" between neighbouring cells. In regions well inside cells, $p_{\text{boundary}}(u)$ is small and $p_{\text{adjusted}}(u)$ remains close to $p_{\text{cell}}(u)$, so interior regions remain intact.

A binary cell mask is then obtained by thresholding

$$\hat{Y}(u) = \mathbb{I}\{p_{\text{adjusted}}(u) \geq \tau\},$$

for a suitable threshold $\tau \in (0, 1)$. Finally, morphological cleaning and connected-component analysis are applied to $\hat{Y}$ to produce instance labels that are used for downstream tracking, radius estimation, and statistical analysis of cell dynamics.

This probability manipulation provides two important benefits. First, it enables separation of touching clusters without the need for manually engineered watershed or contour heuristics. Second, it yields a probabilistically interpretable mechanism: boundary pixels provide negative evidence for the cell-interior class, rather than segmentation relying purely on hard geometric post-processing. The trenching technique is illustrated in figure 3.3.

2.2.2 Convolutional neural networks and encoder-decoder architectures

Convolutional neural networks (CNNs) form the backbone of most modern biomedical segmentation methods [30, 26]. A CNN processes an input image by applying a sequence of learnable convolutional filters [30]. Each convolutional layer replaces the original pixel intensities with feature maps that encode higher-level patterns such as edges, textures, or cell-like shapes.

Formally, a two-dimensional convolution with filter k maps an input feature map f to

$$(f * k)(i, j) = \sum_{u, v} f(i - u, j - v) k(u, v),$$

where the filter coefficients $k(u, v)$ are learned from data [26]. Stacking many such layers, interleaved with nonlinear activations (e.g. rectified linear units), batch normalisation, and pooling, allows the network to represent complex, hierarchical features.

A common design for segmentation is the encoder-decoder (or "U-shaped") architecture [33, 43]. Fully convolutional networks (FCNs) were among the first to popularize dense, per-pixel prediction using downsampling and upsampling within a single convolutional model [33]. The encoder gradually reduces spatial resolution by alternating convolutional blocks with downsampling (e.g. max pooling or strided convolutions). This process increases the receptive field, so deeper layers can aggregate information over larger portions of the image and capture global context. The decoder mirrors this structure: it progressively upsamples the coarse feature maps back to the original resolution, combining coarse semantic information with fine spatial detail to produce dense, pixel-level predictions.

2.2.2.1 U-Net and residual encoders

The U-Net architecture is a widely used encoder-decoder model tailored for biomedical image segmentation [43]. Its defining feature is the use of skip connections between corresponding encoder and decoder levels.

Let E_ℓ denote the feature map at encoder level ℓ and D_ℓ the decoder feature map at the corresponding spatial resolution. The decoder update has the form

$$D_\ell = \phi(\text{Up}(D_{\ell+1}) \parallel E_\ell),$$

where $\text{Up}(\cdot)$ denotes an upsampling operation (e.g. transposed convolution), $\parallel$ denotes channel-wise concatenation, and ϕ is a small stack of convolutions and nonlinearities.

Intuitively, the encoder learns increasingly abstract features but loses fine-grained spatial information through pooling. The skip connections pass along high-resolution features from the encoder so that the decoder can recover precise boundaries and shapes. This is particularly important in cell segmentation, where neighbouring cells may be separated by only a thin boundary of pixels.

In this work, a U-Net variant with a ResNet34 encoder is used as the primary segmentation model. Residual networks (ResNets) are deep CNNs built from residual blocks [28]. Instead of learning a direct mapping $x \mapsto F(x)$, a residual block learns a perturbation $R(x)$ and adds it to the input

$$y = x + R(x).$$

Here $R(x)$ is typically implemented as a small stack of convolutions, normalisation layers, and nonlinearities. The skip connection $x \mapsto x + R(x)$ helps to stabilise optimisation in deep networks by making it easier to learn modifications around the identity mapping. Replacing the plain U-Net encoder with a ResNet34 encoder allows the model to leverage rich features pre-trained on large natural-image datasets and tends to improve convergence and robustness.

2.2.2.2 U-Net++: nested skip connections

U-Net++ extends U-Net by redesigning the skip pathways between encoder and decoder [57]. Instead of connecting encoder and decoder features directly at each resolution, U-Net++ introduces intermediate convolutional blocks along the skip paths. These blocks progressively transform encoder features before they are fused into the decoder, effectively bridging the semantic gap between low-level encoder features and high-level decoder features.

From a structural perspective, the decoder of U-Net++ becomes a densely connected grid of convolutional blocks, where each node receives inputs both from the encoder at the corresponding level and from multiple decoder nodes at neighbouring levels. This nested architecture can provide sharper boundaries and improved delineation of small structures, at the cost of more parameters and increased computation.

2.2.3 Transformer-based segmentation and SegFormer

While CNNs build representations using local convolutions with limited receptive fields, transformer-based models employ self-attention to capture long-range dependencies. The self-attention mechanism was introduced in the transformer architecture [53]. Originally developed for natural language processing, transformers have been adapted to images by treating an image as a sequence of patches.

Given an input sequence of feature vectors $\{x_1, \dots, x_n\}$, self-attention transforms them into queries Q , keys K , and values V via learned linear projections. The attention mechanism computes

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d}}\right)V,$$

where d is the feature dimension. Each output position is thus a weighted average of all input positions, with weights determined by learned similarity between queries and keys. In image transformers, this allows the model to relate distant pixels directly, which is beneficial for capturing global context and long-range cell interactions.

SegFormer is a transformer-based segmentation architecture that combines a hierarchical transformer encoder with a lightweight decoder [55]. The encoder (a "Mix Transformer") processes the image at multiple resolutions, similar in spirit to

the multi-scale feature maps of CNNs, but uses self-attention within each scale to capture non-local relationships. The decoder aggregates these multi-scale features using simple feed-forward layers and upsamples them to the original resolution to produce dense predictions.

In the present setting, SegFormer serves as a conceptually different alternative to CNN-based models. It can better exploit global context (for example, distinguishing overlapping cells from background artefacts) but typically requires more data and careful regularisation to reach its full potential.

2.2.4 Fogbank

As a non-deep learning reference, a classical segmentation method is included in the comparison. This algorithm relies on hand-crafted image processing steps such as intensity thresholding, filtering, and morphological operations [11]. Such methods have the advantage of being interpretable and lightweight, and do not require training data. However, in heterogeneous phase-contrast images they often struggle with varying backgrounds, low contrast, and complex cell shapes. In this thesis, Fogbank serves as a baseline against which the deep models are evaluated.

2.2.5 Training objective and loss functions

All deep models considered here are trained to predict, for each input pixel, a three-dimensional vector of class probabilities corresponding to background, cell interior, and boundary. Let $\hat{p}_c(i, j)$ denote the predicted probability that pixel (i, j) belongs to class $c \in \{0, 1, 2\}$, and let $y_c(i, j) \in \{0, 1\}$ denote the corresponding one-hot encoded ground-truth label. Training minimises a combination of a pixel-wise cross-entropy term and a Dice loss term [26, 36].

The cross-entropy component penalises misclassification at each pixel,

$$\mathcal{L}_{\text{CE}} = - \sum_{i,j} \sum_c w_c y_c(i, j) \log \hat{p}_c(i, j),$$

where $w_c > 0$ are class weights. The weights are chosen to counteract class imbalance and emphasise boundary pixels, which are much less frequent than background or cell-interior pixels but crucial for separating touching cells.

The Dice loss component targets the overlap between predicted and true segmentation masks. For a given class c , the (soft) Dice coefficient is

$$\text{Dice}_c = \frac{2 \sum_{i,j} y_c(i, j) \hat{p}_c(i, j)}{\sum_{i,j} y_c(i, j) + \sum_{i,j} \hat{p}_c(i, j) + \varepsilon},$$

with a small constant $\varepsilon > 0$ to avoid division by zero. The Dice loss for class c is $1 - \text{Dice}_c$; in practice these are averaged (or summed with class-specific weights) across the three classes.

The total loss used for training is

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{Dice}},$$

with $\mathcal{L}_{\text{Dice}}$ denoting the suitably weighted average of $1 - \text{Dice}_c$. This combination leverages the complementary strengths of cross-entropy (good calibration of probabilities and stable optimisation) and Dice (directly optimising spatial overlap, particularly useful under class imbalance).

2.2.6 Evaluation metrics

To quantify segmentation performance, predicted masks $\hat{Y}$ are compared with annotated ground truth masks Y , we used overlap-based metrics. The two metrics employed are the Dice Similarity Coefficient (DSC) and the Jaccard index, also known as Intersection over Union (IoU).

First, let $Y \subset \Omega$ denote the ground-truth set of foreground pixels and $\hat{Y} \subset \Omega$ the predicted set. The Dice score is defined as

$$\text{DSC}(Y, \hat{Y}) = \frac{2|Y \cap \hat{Y}|}{|Y| + |\hat{Y}|}, \quad 0 \leq \text{DSC} \leq 1,$$

originally introduced as a similarity measure in ecology and now widely used in medical image analysis [17]. A Dice value of 1 corresponds to perfect agreement, whereas values near 0 indicate almost complete mismatch. Geometrically, Dice measures how much the predicted region overlaps the annotated one relative to their combined area. Figure 2.6 illustrates two schematic cases: strong overlap leading to a high Dice score, and weak overlap producing a low Dice score.

Dice is particularly suitable in the present context for three reasons. First, it is symmetric in prediction and truth, avoiding bias towards either. Second, it is sensitive to boundary errors, which is crucial because biological interpretation depends on accurate separation of neighbouring cells. Third, Dice is relatively robust to moderate class imbalance, so the metric remains meaningful even when background heavily outweighs foreground, as is typical in microscopy images.

Originally introduced by Jaccard [29], and now widely used as a segmentation metric [48], the Jaccard index is defined as

$$\text{IoU}(P, G) = \frac{|P \cap G|}{|P \cup G|} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}}.$$

IoU penalises both over-segmentation (false positives) and under-segmentation (false negatives) by normalising the intersection by the union of the predicted and ground truth sets. In practice, Dice and IoU are closely related and often tell a similar story. The main difference is that IoU is stricter, where for the same prediction, IoU will always be less than or equal to Dice. Using both metrics allows us to assess overlap in a way that is comparable to other segmentation strategies

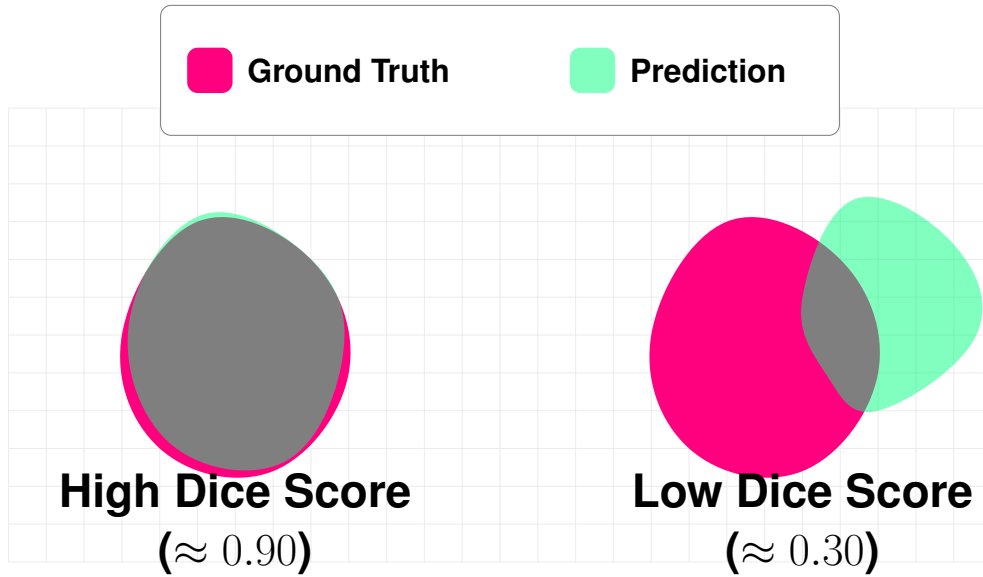


Figure 2.6

High Dice scores indicate accurate reconstruction of cell shapes and precise boundary delineation, which in turn ensures reliable centroid extraction, radius estimation, and diffusion analysis in later stages of the pipeline. The boundary-aware probability subtraction framework is therefore not simply a technical detail of the segmentation network, but a structural design choice that directly influences the validity of the Approximate Bayesian Computation procedures described in the subsequent section.

In summary, the segmentation methodology combines probabilistic modelling, boundary-aware probability shaping, encoder-decoder and transformer-based architectures, and overlap-based evaluation. Together, these elements ensure that the produced segmentation masks are not only visually plausible but also statistically reliable inputs for subsequent biological inference.

2.3 The Role of Trajectory Statistics

Historically, the extraction of specific cellular properties from trajectory data has relied on a few fundamental, well-established mathematical relationships. For cellular motility, the standard for estimating the diffusion coefficient D is the Mean Squared Displacement (MSD). Rooted in [21], the MSD for a purely diffusive particle scales linearly with time τ ,

$$\text{MSD}(\tau) = 4D\tau \quad (2.2)$$

Similarly, baseline proliferation is evaluated by assuming exponential growth before reaching carrying capacity, where the population $N(t)$ at time t depends on the initial population N_0 and the proliferation rate ρ

$$N(t) = N_0 e^{\rho t}. \quad (2.3)$$

These established physical relationships provide the critical ground truth when validating whether a simulated model accurately reflects the biological reality of the cell

culture. They are also used for extraction of the values for diffusion, proliferation and radius.

2.4 Approximate Bayesian Computation

Approximate Bayesian Computation (ABC) is a family of likelihood-free Bayesian methods for parameter inference [4, 47]. The aim is to learn a posterior distribution for unknown parameters θ from observed data y .

The foundational setting is a parametric model

$$\theta \sim \pi(\theta), \quad y \sim p(\cdot | \theta),$$

where θ denotes unknown parameters, $\pi(\theta)$ is a prior distribution, and $p(y | \theta)$ is an intractable distribution. ABC replaces explicit evaluation of the likelihood $p(y | \theta)$ by using forward simulations of ones model.

The key assumption is that for any proposed θ one can simulate synthetic data

$$x \sim p(\cdot | \theta),$$

and reduce both x and the observed data y to low-dimensional summary statistics

$$s(x), s(y) \in \mathbb{R}^K.$$

A non-negative error or distance $E(\cdot, \cdot)$ is used to quantify how close a simulated data set is to the observation. For a tolerance $\epsilon > 0$, the ABC posterior takes the form

$$\pi_\epsilon(\theta | y) \propto \pi(\theta) \int \mathbb{I}\{E(s(x), s(y)) \leq \epsilon\} p(x | \theta) dx,$$

where $\mathbb{I}\{\cdot\}$ is the indicator function. If $s(\cdot)$ is sufficient for θ and $\epsilon \rightarrow 0$, then $\pi_\epsilon(\theta | y)$ approaches the exact posterior $\pi(\theta | y)$.

In practical applications, one works with non-zero tolerances and with summary statistics that are not strictly sufficient, so ABC produces an approximation to the exact posterior.

2.4.1 Distance functions and summary statistics

The design of summary statistics and the choice of distance are central to ABC methods [4, 32]. If the summary map $s(\cdot)$ is sufficient, then no information about θ is lost, but in complex models sufficiency is rarely attainable. Instead, one selects summaries that are low-dimensional, interpretable, and empirically informative about θ .

Given summaries $s(x)$ and $s(y)$, a common choice of distance is a weighted Euclidean norm

$$E(s(x), s(y)) = \|W(s(x) - s(y))\|_2,$$

where W is a diagonal or full weighting matrix. Weights are typically chosen to normalize components to comparable scales, to emphasize particularly informative summaries. When the dimension K is large, distances between independent draws tend to concentrate; this is one manifestation of the curse of dimensionality and limits how small ϵ can be chosen for a given number of simulations.

2.4.2 Rejection ABC

The simplest ABC algorithm is rejection sampling. At its core, rejection ABC approximates the posterior by repeatedly proposing parameters from the prior, simulating data under each candidate, and retaining only those parameter values whose simulated summaries are sufficiently close to the observed summaries [41, 5].

In its standard form, the algorithm proceeds as follows and can be followed in greater detail in Algorithm 1 and Figure 4.7b.

Algorithm 1 Rejection ABC

Require: Prior $\pi(\theta)$; simulation $x \sim p(\cdot \mid \theta)$; summary $s(\cdot)$; distance $E(\cdot, \cdot)$; observed data y ; total simulations N ; tolerance ϵ **or** desired number of accepted draws k .

- 1: Compute observed summaries $s_y \leftarrow s(y)$.
 - 2: **for** $i = 1, 2, \dots, N$ **do**
 - 3: Draw parameter $\theta^{(i)} \sim \pi(\theta)$.
 - 4: Simulate synthetic data $x^{(i)} \sim p(\cdot \mid \theta^{(i)})$.
 - 5: Compute summaries $s^{(i)} \leftarrow s(x^{(i)})$.
 - 6: Compute distance $d_i \leftarrow E(s^{(i)}, s_y)$.
 - 7: **end for**
 - 8: **if** a fixed tolerance ϵ is used **then**
 - 9: Define acceptance set $\mathcal{A} \leftarrow \{i : d_i \leq \epsilon\}$.
 - 10: **else**
 - 11: Let $\mathcal{A}$ be the indices of the k smallest distances $\{d_i\}$ (“best- k ” rule).
 - 12: **end if**
 - 13: Form the approximate posterior sample $\{\theta^{(i)} : i \in \mathcal{A}\}$.
 - 14: Optionally, estimate posterior summaries (mean, credible intervals, modes) from the accepted sample.
-

1. Draw $\theta^{(i)} \sim \pi(\theta)$ from the prior.
2. Simulate $x^{(i)} \sim p(\cdot \mid \theta^{(i)})$ and compute summaries $s^{(i)} = s(x^{(i)})$.
3. Compute the distance $d_i = E(s^{(i)}, s(y))$.
4. Accept $\theta^{(i)}$ if $d_i \leq \epsilon$; otherwise reject.

The set of accepted draws $\{\theta^{(i)} : d_i \leq \epsilon\}$ forms an approximate sample from $\pi_\epsilon(\theta \mid y)$. In practice one typically chooses ϵ implicitly by fixing a target number of accepted

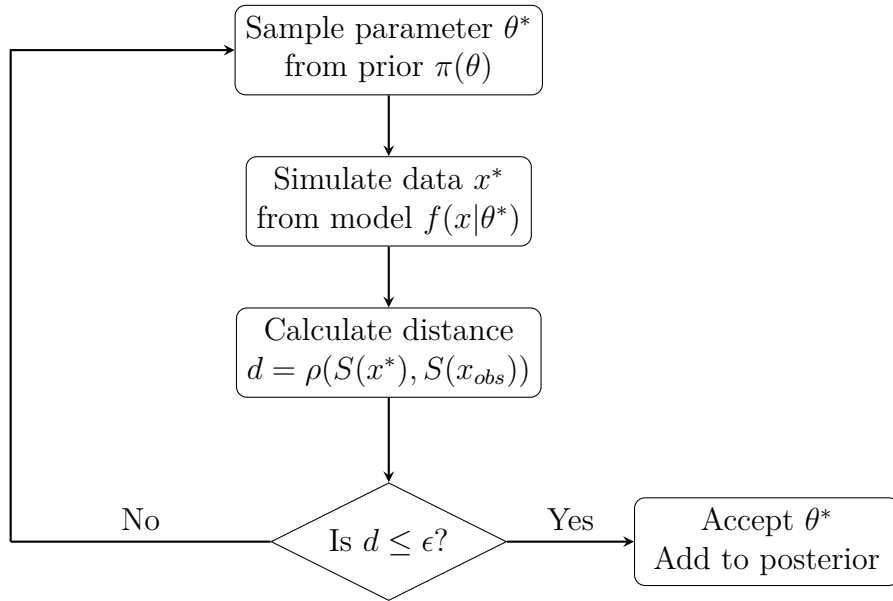


Figure 2.7: Schematic of the ABC Rejection algorithm. Parameters are sampled from the prior and accepted only if the simulated summary statistics are sufficiently close to the observed statistics.

draws and retaining the k smallest distances; this is often referred to as the “best- k ” rule.

When many simulations are available, for example from a precomputed collection of pairs (θ_i, s_i) , rejection ABC can be implemented by computing distances from s_i to $s(y)$ and retaining the k smallest. The accepted parameters are then treated as a discrete approximation of the ABC posterior. Point estimates such as posterior modes can be obtained by applying a histogram or kernel density estimate to the accepted parameters.

2.4.3 Regression-adjusted ABC

Regression-adjusted ABC modifies the rejection scheme in order to reduce bias when the tolerance is not extremely small. The idea is to regard the accepted (θ_i, s_i) pairs as a local cloud around the observed summaries and to model the local dependence of parameters on summaries with a regression model. One then uses the fitted local linear relationship to adjust each accepted parameter value to the value it would have if its summary equalled $s(y)$ [5].

To build intuition, consider a simple setting with a single parameter θ and a single summary statistic s (for instance, the final cell count or one bin of the pair correlation function). If infinitely many simulations were available and we plotted all pairs (θ, s) , there would be some underlying functional relationship mapping the summaries back to the parameters. We can denote this unknown regression function as $g(\cdot)$, such that

$$\theta \approx g(s).$$

Regression adjustment assumes that, in a neighbourhood of the observed summary $s(y)$, this relationship is approximately linear

$$\theta \approx \beta_0 + \beta_1 s.$$

In basic rejection ABC we keep simulations for which s_i is "close enough" to $s(y)$. However, the accepted s_i are not exactly equal to $s(y)$: some are slightly smaller, some are slightly larger. If θ tends to increase with s , then simulations with $s_i > s(y)$ will on average have θ_i that are too large relative to $\theta | s(y)$, and simulations with $s_i < s(y)$ will have θ_i that are too small. The cloud of accepted points (θ_i, s_i) is therefore biased relative to the distribution of θ conditional on $s = s(y)$.

Regression adjustment seeks to correct this by fitting the local trend and then "sliding" each accepted draw back to $s(y)$ along that trend. In the one-dimensional case, a local regression fit gives

$$\theta_i \approx \hat{\beta}_0 + \hat{\beta}_1 s_i,$$

while the fitted value at the observed summary is

$$\hat{\theta}(s(y)) = \hat{\beta}_0 + \hat{\beta}_1 s(y).$$

The difference between these two fitted values is

$$\hat{\theta}(s_i) - \hat{\theta}(s(y)) = \hat{\beta}_1 (s_i - s(y)),$$

so an adjusted draw is obtained by subtracting this predicted shift

$$\theta_{i,\text{adj}} = \theta_i - \hat{\beta}_1 (s_i - s(y)).$$

If $s_i = s(y)$ there is no adjustment; if s_i is above $s(y)$ and θ increases with s , the adjustment moves θ_i down; if s_i is below $s(y)$, it moves θ_i up. Collecting all $\theta_{i,\text{adj}}$ yields a sample that is closer to the conditional distribution $\theta | s(y)$ than the raw accepted sample.

We now turn to the general setting. Let $\mathcal{A}$ denote a neighbourhood of k accepted simulations with small distances d_i . For each $i \in \mathcal{A}$, we define the augmented summary vector

$$z_i = \begin{pmatrix} 1 \\ s_i \end{pmatrix} \in \mathbb{R}^{K+1},$$

and a kernel weight $w_i = K(d_i/h)$, where $K(\cdot)$ is a kernel and h plays the role of a bandwidth. Let Z denote the $k \times (K+1)$ matrix with rows $z_i^\top$, let W be the $k \times k$ diagonal matrix with entries w_i , and consider one parameter component $\theta^{(j)}$.

Weighted least squares yields

$$\hat{\beta}^{(j)} = (Z^\top W Z)^{-1} Z^\top W \theta^{(j)},$$

where $\theta^{(j)}$ is the k -vector of the j th component across accepted draws. Expressing $\hat{\beta}^{(j)} = (\hat{\beta}_0^{(j)}, \hat{B}^{(j)})$ with slope vector $\hat{B}^{(j)} \in \mathbb{R}^K$, the adjusted draws are

$$\theta_{i,\text{adj}}^{(j)} = \theta_i^{(j)} - (s_i - s(y))^\top \hat{B}^{(j)}, \quad i \in \mathcal{A}.$$

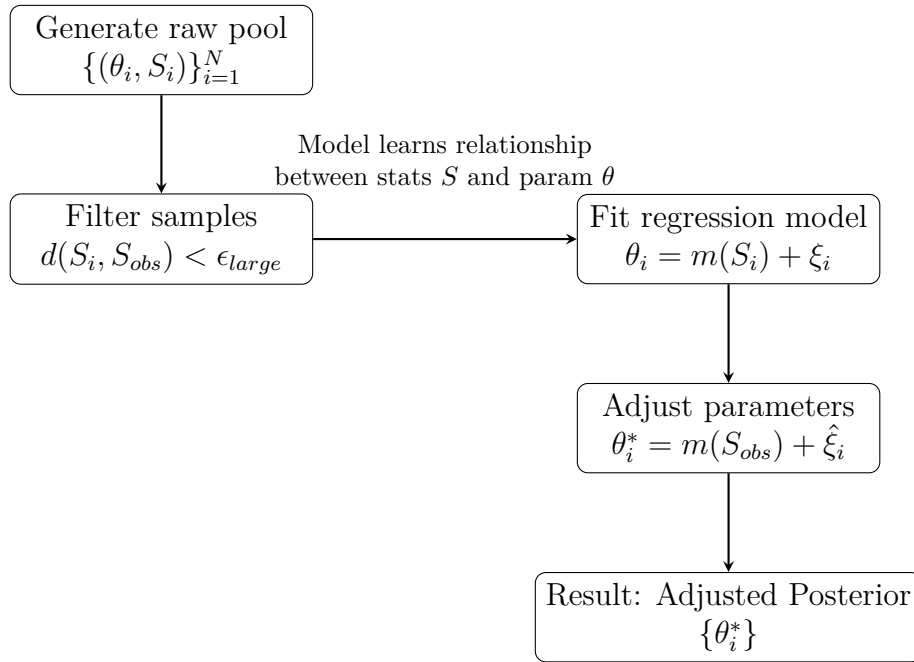


Figure 2.8: Schematic of ABC Regression adjustment. The model learns the local relationship between statistics and parameters to correct the posterior approximation.

Stacking over all components of θ yields adjusted parameter vectors $\theta_{i,\text{adj}}$. These adjusted draws are then analysed in place of the raw accepted draws. For example, one may compute posterior means, medians, modes, and credible intervals based on $\{\theta_{i,\text{adj}}\}_{i \in \mathcal{A}}$.

Intuitively, the adjustment can be described as in a local neighbourhood around $s(y)$, the regression intends to describe how θ changes with s , where subtracting $(s_i - s(y))^\top \hat{B}^{(j)}$ removes the linear trend and maps the draw from the location s_i to $s(y)$. When the true conditional relationship between θ and s is approximately linear in a suitable parametrisation, the regression-adjusted sample is typically less biased and more tightly concentrated near the true parameter value than the unadjusted accepted sample.

To summarise the regression-adjusted ABC procedure, the algorithm is provided in Algorithm 2 and an overview is shown in figure 4.7b.

2.4.3.1 Kernel weighting

In practice, the kernel function $K(\cdot)$ controls how strongly each accepted simulation contributes to the local regression, as a function of its distance from the observed summaries. This is accomplished with turning distances d_i into weights $w_i = K(d_i/h)$ via the kernel function K and a bandwidth $h > 0$.

The bandwidth controls how local the regression fit is, where a small h means that

Algorithm 2 Regression-adjusted ABC

Require: Prior $\pi(\theta)$; simulation $x \sim p(\cdot \mid \theta)$; summary map $s(\cdot)$; distance $E(\cdot, \cdot)$; observed data y ; kernel $K(\cdot)$ and bandwidth h ; desired neighborhood size k .

- 1: Compute observed summaries $s_y \leftarrow s(y)$.
- 2: Run rejection ABC as in Algorithm 1 to obtain N pairs $(\theta^{(i)}, s^{(i)})$ with distances d_i .
- 3: Let $\mathcal{A}$ be the indices of the k smallest distances; these form the local neighborhood around s_y .
- 4: **for** $i \in \mathcal{A}$ **do**
- 5: Define augmented summary vector $z_i \leftarrow (1, s^{(i)})^\top$.
- 6: Compute kernel weight $w_i \leftarrow K(d_i/h)$.
- 7: **end for**
- 8: Form design matrix Z with rows $z_i^\top$ for $i \in \mathcal{A}$.
- 9: Form diagonal weight matrix $W = \text{diag}(w_i : i \in \mathcal{A})$.
- 10: **for** each parameter component j of θ **do**
- 11: Let $\theta^{(j)}$ be the vector of the j th component across $\{\theta^{(i)} : i \in \mathcal{A}\}$.
- 12: Compute weighted least-squares estimate

$$\hat{\beta}^{(j)} \leftarrow (Z^\top W Z)^{-1} Z^\top W \theta^{(j)}.$$

- 13: Split $\hat{\beta}^{(j)} = (\hat{\beta}_0^{(j)}, \hat{B}^{(j)})$ with slope vector $\hat{B}^{(j)}$ corresponding to the summary components.
- 14: **for** each $i \in \mathcal{A}$ **do**
- 15: Adjust the j th parameter component:

$$\theta_{i,\text{adj}}^{(j)} \leftarrow \theta_i^{(j)} - (s^{(i)} - s_y)^\top \hat{B}^{(j)}.$$

- 16: **end for**
 - 17: **end for**
 - 18: Collect adjusted parameter vectors $\theta_{i,\text{adj}}$ for $i \in \mathcal{A}$.
 - 19: Use $\{\theta_{i,\text{adj}}\}_{i \in \mathcal{A}}$ to compute posterior summaries (means, modes, credible intervals, *etc.*).
-

only simulations close to $s(y)$ receive substantial weight, while large h leads to a smoother, more global fit that uses a broader neighbourhood.

Furthermore, we use compactly supported kernels based on the rescaled distance

$$u_i = \frac{d_i}{h},$$

so that simulations with $u_i > 1$ receive zero weight and do not influence the regression fit.

Two common choices of kernel in ABC are the Epanechnikov and triangular kernels.

The first is given by

$$K_{\text{Epa}}(u) = \max(0, 1 - u^2),$$

which assigns its largest weight at $u = 0$ and then decreases quadratically to zero at $|u| = 1$. This can be rephrased as giving the highest weight to simulations that are close to the observed summaries and decreases quadratically to zero at the boundary of the neighbourhood.

The Epanechnikov kernel is optimal in a mean-square error sense for local linear regression under standard regularity conditions, in the sense that, among symmetric kernels with compact support, it minimises the asymptotic variance of the estimator for a given bandwidth. In the implementation used here, the normalising constant is omitted and the weights take the simplified form

$$w_i \propto \max\{0, 1 - u_i^2\},$$

since only relative weights matter in the weighted least squares problem.

As previously mentioned, one can instead use the triangular kernel

$$K_{\text{tri}}(u) = \max(0, 1 - |u|),$$

which instead decreases linearly to zero at $|u| = 1$. In both cases, $K(u) = 0$ for $|u| > 1$, so simulations with $d_i > h$ receive zero weight and do not influence the regression fit.

From a practical point of view, the bandwidth h is usually the dominant tuning parameter. Changing h alters how many simulations effectively contribute to the regression, and therefore has a large impact on bias and variance. By contrast, switching between Epanechnikov and triangular kernels tends to have a much smaller effect on the resulting estimates, as both kernels are smooth, give highest weight to simulations closest to $s(y)$, and down-weight more distant simulations to zero. In this sense, the precise kernel shape is less important than choosing a sensible bandwidth, provided that a standard compactly supported kernel (such as Epanechnikov or triangular) is used. For this work the Epanechnikov kernel is used throughout.

2.4.4 Sequential Monte Carlo ABC

Sequential Monte Carlo ABC (SMC-ABC) addresses the inefficiency of naive rejection ABC, especially when the prior is diffuse and the posterior is concentrated. Rather than drawing parameters independently from the prior and rejecting the majority of them, SMC-ABC maintains a population of parameter candidates, often called particles, and gradually moves this population from the prior towards regions that are compatible with the data [51, 16, 47]. An overview of this method is shown in figure 2.9.

In this context, a particle is simply a parameter vector $\theta^{(p)}$ together with an importance weight $w^{(p)}$. At iteration (or “population”) t the algorithm stores P such

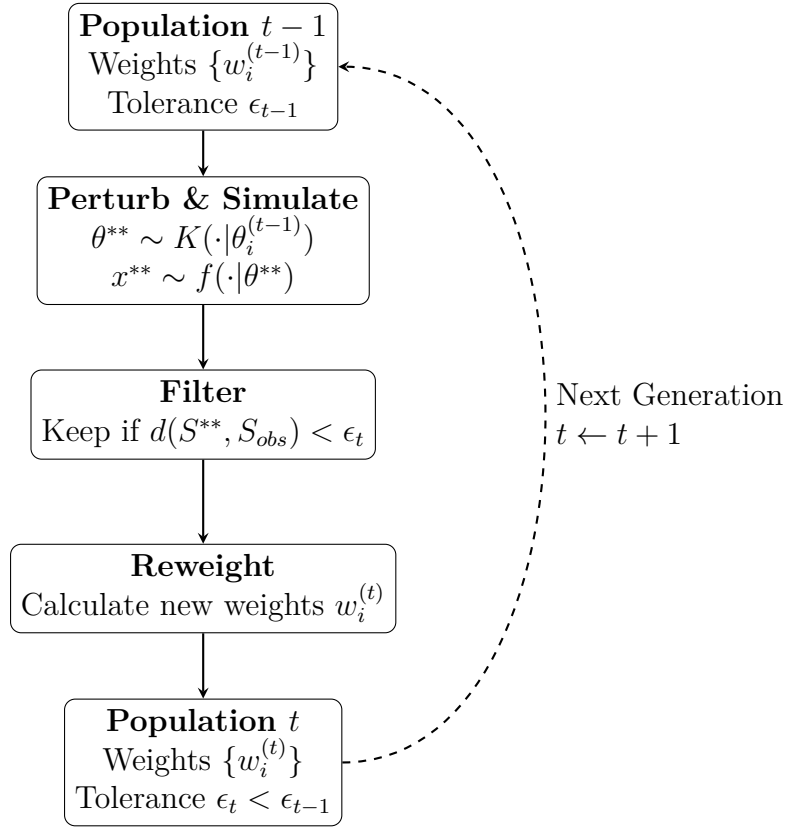


Figure 2.9: Schematic of SMC-ABC. The algorithm evolves a population of particles through a sequence of decreasing tolerances. In each generation, particles are perturbed, filtered, and re-weighted to approximate the posterior distribution more closely.

particles

$$\{(\theta_t^{(p)}, w_t^{(p)})\}_{p=1}^P,$$

which form a discrete approximation of an ABC posterior distribution at tolerance ϵ_t . The transition from population $t - 1$ to t is driven by three basic operations:

- *Resampling*: good particles with large weights are copied, while particles with small weights are discarded.
- *Perturbation and Simulation*: each resampled particle is randomly "jittered" to explore nearby parameter values. This perturbation is defined by a proposal kernel $K_t(\theta | \tilde{\theta})$, for example a multivariate Gaussian centred at $\tilde{\theta}$. Using this proposed parameter, a synthetic dataset x^{**} is generated from the simulator (the forward generative model), denoted as $f(\cdot | \theta^{**})$.
- *Re-weighting*: the importance weights $w_t^{(p)}$ are updated to correct for the fact that particles were proposed via the kernel K_t rather than directly from the target distribution at level ϵ_t .

SMC–ABC also replaces the single tolerance ϵ of rejection ABC by a decreasing sequence

$$\epsilon_1 > \epsilon_2 > \dots > \epsilon_T \geq 0.$$

At the first population ($t = 1$), a set of particles $\{\theta_1^{(p)}\}_{p=1}^P$ is drawn from the prior. For each particle, synthetic data are simulated, summary statistics and distances are computed, and only those with $E(s(x_1^{(p)}), s(y)) \leq \epsilon_1$ are retained. The corresponding weights are typically set equal and normalised.

At population $t > 1$ the algorithm proceeds sequentially:

1. Resample particles from population $t - 1$ according to their weights $\{w_{t-1}^{(j)}\}_{j=1}^P$.
2. Perturb the resampled particles via a candidate kernel $K_t(\theta \mid \tilde{\theta})$ to obtain candidates $\theta_t^{(p)}$.
3. For each candidate, simulate data, compute summary statistics, and accept the candidate if $E(s(x_t^{(p)}), s(y)) \leq \epsilon_t$. Otherwise, draw a new candidate from step 2 until acceptance.
4. Update the importance weights to correct for the bias introduced by sampling from the perturbation kernel rather than the true prior. The unnormalized weight $W_t^{(p)}$ for particle p is calculated as

$$W_t^{(p)} = \frac{\pi(\theta_t^{(p)})}{\sum_{j=1}^P w_{t-1}^{(j)} K_t(\theta_t^{(p)} \mid \theta_{t-1}^{(j)})},$$

where $\pi(\cdot)$ denotes the prior distribution and P is the total particle count. The weights are subsequently normalized such that $\sum_{p=1}^P w_t^{(p)} = 1$.

The above is detailed in Algorithm 3.

The sequence of tolerances may be chosen in advance or adapted during runtime, for instance by setting ϵ_t to a quantile of the empirical distances at population $t - 1$.

By gradually tightening the tolerance and re-using information across populations, SMC–ABC concentrates simulation effort in regions of high posterior density and often attains a much smaller final tolerance than is feasible with a single-stage rejection sampler.

The final weighted population at the smallest tolerance serves as an approximate sample from a low-tolerance ABC posterior. As in the previous methods, point estimates and credible intervals can be obtained by summarising this population.

A brief note on computational complexity

To evaluate and compare the computational footprint of these methods, we define the following variables:

- N : number of simulated data sets available for a rejection or regression-adjusted run (e.g., the size of a precomputed simulation bank).
- k : number of accepted simulations (or the size of the local neighborhood).
- K : dimension of the summary statistic vector.
- P : number of particles per population in SMC–ABC.
- T : number of SMC populations.
- S : computational cost of one full forward simulation.

For rejection ABC based on N simulated summaries, computing the pairwise distances between the simulated summary statistics and the observed summary statistics scales as $O(NK)$, and selecting the k smallest distances can be done in $O(N)$ expected time using partial selection algorithms (or $O(N \log N)$ with a total sort). Memory requirements are $O(N)$ for the stored simulated summaries and $O(k)$ for the accepted subset.

Regression-adjusted ABC incurs all of the costs of rejection ABC, plus the cost of fitting local regressions in a neighbourhood of size k . A straightforward implementation of weighted least squares on K summaries has complexity on the order of $O(kK^2)$ per parameter, or up to $O(K^3)$ with naive linear algebra routines and solvers. The dominating simulation cost is incurred during the construction of synthetic data rather than during inference.

SMC–ABC relies on runtime simulation. If each of the P particles in each of the T populations requires, on average $c_t \geq 1$ simulator calls at population t before satisfying the tolerance, then the total number of simulator calls is approximately

$$\sum_{t=1}^T c_t P,$$

and the dominant cost is thus of order $O(TPS)$. Distance evaluations within each population scale computationally as $O(TPK)$ and $O(TP)$, respectively, which are typically negligible compared with the simulation cost in complex models.

2.5 Modelling Cell Population Dynamics

Understanding the trajectories of individual cells within a population requires mathematical frameworks capable of handling both deterministic interactions and inherent biological randomness. At the single-cell level, polarity-driven migration subject to numerous small stochastic perturbations is naturally framed using stochastic calculus.

Between proliferation events, the motion of a generic cell can be modelled using an

overdamped Langevin stochastic differential equation (SDE)

$$dX(t) = F(X(t)) dt + \sqrt{2D} dW(t), \quad (2.4)$$

where $D > 0$ is the macroscopic diffusion coefficient and $W(t)$ represents a standard Brownian motion [39, 23]. The drift term $F(X)$ is a generalized vector field that aggregates the deterministic interactions and physical forces governing the cell's spatial organization.

To simulate continuous-time SDEs on a discrete computational grid, the Euler–Maruyama numerical scheme is standard practice. Given a discrete time step Δt , the continuous position X is updated as

$$X(t + \Delta t) = X(t) + F(X(t)) \Delta t + \sqrt{2D \Delta t} \xi, \quad \xi \sim \mathcal{N}(0, I). \quad (2.5)$$

This scheme ensures that the variance of the simulated Brownian increments correctly scales with the time step, forming the theoretical mathematical basis for calculating continuous stochastic motion across all spatial models in this work.

2.6 Velocity Driven models

While positional Langevin models are mathematically convenient, they can be biologically restrictive because the driving noise $\mathbf{W}_i(t)$ has independent increments, implying no directional persistence. This means that the instantaneous displacement over a short interval is independent of the previous one. Thus a 180 degree pivot is equally probable to remaining on the same course, which is unrealistic. Empirically, many motile cells exhibit correlated velocity increments (persistence) on intermediate time scales, as discussed for microorganisms and cell migration in, e.g., [49, 31]. A natural remedy is to model velocity as the primary stochastic variable and recover positions by time integration. This leads to velocity-driven systems of the form

$$\begin{aligned} d\mathbf{X}_i(t) &= \mathbf{V}_i(t) dt, \\ d\mathbf{V}_i(t) &= -\left[\mu \mathbf{V}_i(t) + \frac{1}{m} \mathbf{F}_i(\mathbf{X}_i(t))\right] dt + \sigma d\mathbf{G}_i(t), \end{aligned} \quad (2.6)$$

where the stochastic forcing $\mathbf{G}_i(t)$ is chosen to encode temporal memory.

2.6.1 Fractional stochastic calculus and fractional Brownian motion

The aim for this section is to lay the foundation to then motivate why we can use The Davies Harte method to simulate (2.6) by replacing $\mathbf{G}_i(t)$ with fractional Brownian motion $\mathbf{W}^H(t)$.

2.6.1.1 Fractional calculus preliminaries

Fractional stochastic calculus is tightly connected to fractional calculus, in which differentiation and integration are extended to non-integer orders [40]. Fractional stochastic calculus can be viewed as extending classical stochastic calculus by replacing the standard time derivative (d/dt) with a fractional derivative (e.g., d^α/dt^α) and incorporating fractional noise.

This is done by letting f be locally integrable on $[0, T]$. The (left-sided) Riemann–Liouville fractional integral of order $\alpha > 0$ is defined by

$$(I_{0+}^\alpha f)(t) = \frac{1}{\Gamma(\alpha)} \int_0^t (t-s)^{\alpha-1} f(s) ds, \quad t \in [0, T], \quad (2.7)$$

where $\Gamma(\cdot)$ is the Gamma function. For $\alpha = 1$ this reduces to the ordinary integral. If $n-1 < \alpha < n$ with $n \in \mathbb{N}$, the (left-sided) Riemann–Liouville fractional derivative is

$$(D_{0+}^\alpha f)(t) = \frac{d^n}{dt^n} (I_{0+}^{n-\alpha} f)(t). \quad (2.8)$$

A related alternative, often preferred in initial-value problems, is the Caputo derivative,

$$({}^C D_{0+}^\alpha f)(t) = (I_{0+}^{n-\alpha} f^{(n)})(t), \quad n-1 < \alpha < n, \quad (2.9)$$

which differentiates first and fractional-integrates afterwards. These operators capture long-memory kernels of the form $(t-s)^{\alpha-1}$, and the same kernel structure reappears in representations of fractional Gaussian processes.

2.6.1.2 How the same kernels reappear in fractional Gaussian processes.

A central fact is that fractional Brownian motion (fBm) admits an integral representation with respect to standard Brownian motion where the integrand is a fractional power-law kernel. One classical representation is the Mandelbrot–Van Ness formula. [35, 7]

Theorem 1 (Mandelbrot–Van Ness representation). *Let $H \in (0, 1)$ and let W be a two-sided standard Brownian motion. Then fBm W^H can be represented in law as*

$$W^H(t) = \frac{\sigma}{\Gamma(H + \frac{1}{2})} \left(\int_{-\infty}^0 \left((t-s)^{H-\frac{1}{2}} - (-s)^{H-\frac{1}{2}} \right) dW(s) + \int_0^t (t-s)^{H-\frac{1}{2}} dW(s) \right), \quad t \geq 0. \quad (2.10)$$

The integrand in (2.10) is precisely of the power-law form $(t-s)^{H-\frac{1}{2}}$, matching the fractional integral kernel $(t-s)^{\alpha-1}$ in (2.7) with $\alpha = H + \frac{1}{2}$. In this sense, fBm can be viewed as a fractional integral of white noise/Brownian increments, which is the origin of its long-memory behaviour. A detailed construction and the associated Hilbert-space calculus can be found in [7].

2.6.1.3 Fractional Brownian motion

Fractional Brownian motion $\{W^H(t)\}_{t \geq 0}$ with Hurst parameter $H \in (0, 1)$ is a centred Gaussian process with $W^H(0) = 0$ and covariance function [45]

$$\mathbb{E}[W^H(t) W^H(s)] = \frac{\sigma^2}{2} (t^{2H} + s^{2H} - |t - s|^{2H}), \quad s, t \geq 0. \quad (2.11)$$

The special case $H = \frac{1}{2}$ recovers standard Brownian motion (up to σ).

2.6.1.3.1 Properties. For fBm we have multiple key structural consequences and properties of (2.11). The first being non-stationarity.

2.6.1.3.2 Non-stationarity. A process $X(t)$ is (weakly) stationary if $\mathbb{E}[X(t)]$ is constant and $\text{Cov}(X(t), X(s))$ depends only on $t - s$. Although $\mathbb{E}[W^H(t)] = 0$, its variance is

$$\text{Var}(W^H(t)) = \mathbb{E}[(W^H(t))^2] = \sigma^2 t^{2H},$$

which depends on t . Hence W^H is not stationary. This matters because many tools for spectral analysis and fast simulation (Toeplitz/circulant structure) require stationarity. To resolve this, we instead consider fBms increments, which turn out to be, to our pleasure, stationary.

2.6.1.3.3 FBM has stationary increments. For $\tau \geq 0$,

$$W^H(t + \tau) - W^H(t) \stackrel{d}{=} W^H(\tau) - W^H(0),$$

and, using (2.11),

$$\text{Var}(W^H(t + \tau) - W^H(t)) = \sigma^2 \tau^{2H},$$

which depends only on τ . As mentioned, this motivates working with the increment process, also named fractional Gaussian motion (fGm), which is stationary and therefore admits a spectral density and efficient FFT-based simulation.

2.6.1.3.4 Self-similarity. For any $c > 0$,

$$\{W^H(ct)\}_{t \geq 0} \stackrel{d}{=} \{c^H W^H(t)\}_{t \geq 0}. \quad (2.12)$$

In particular, the mean-squared displacement (MSD) scales as $\text{MSD}(t) \propto t^{2H}$, which connects H directly to log-log MSD slopes and is central in trajectory analysis (see 3.4.3 for further details).

2.6.1.3.5 Persistence vs. anti-persistence. Although increments are stationary, they are generally correlated. When $H > 1/2$, increments are positively correlated (persistent motion); when $H < 1/2$, increments are negatively correlated (anti-persistent). This memory property is the reason fBm is a natural Gaussian driver for velocity models with directional persistence [7, 35].

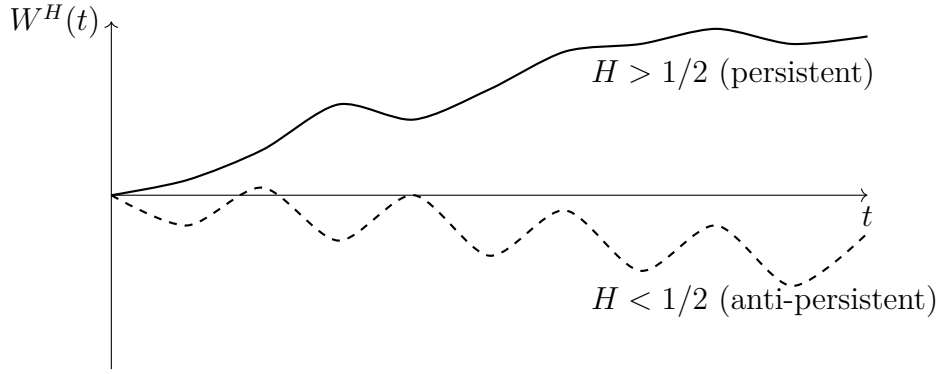


Figure 2.10: Schematic illustration of the qualitative effect of the Hurst parameter on increment correlations: persistent paths tend to keep direction, while anti-persistent paths tend to revert.

2.6.1.4 Fractional Gaussian noise and increment covariance

As motivated, for numerical work, it is often convenient to work with the increments of fBm. Thus, let $\Delta > 0$ and define the sampled fBm $B_n := W^H(n\Delta)$. We name the increment sequence

$$G_n := B_{n+1} - B_n, \quad n \in \mathbb{Z}, \quad (2.13)$$

as fractional Gaussian noise (fGn) which is a mean-zero stationary Gaussian sequence. Here we have claimed stationarity, which is shown below together with the autocovariance of fGn.

Proposition 2.6.1 (Autocovariance of fGn). *For $k \in \mathbb{Z}$,*

$$\gamma(k) := \text{Cov}(G_n, G_{n+k}) = \frac{\sigma^2}{2} \Delta^{2H} \left(|k+1|^{2H} - 2|k|^{2H} + |k-1|^{2H} \right). \quad (2.14)$$

Proof. Using bilinearity of covariance and stationarity of increments, for any n ,

$$\gamma(k) = \text{Cov}(B_{n+1} - B_n, B_{n+k+1} - B_{n+k}).$$

By expanding, we obtain

$$\gamma(k) = \text{Cov}(B_{n+1}, B_{n+k+1}) - \text{Cov}(B_{n+1}, B_{n+k}) - \text{Cov}(B_n, B_{n+k+1}) + \text{Cov}(B_n, B_{n+k}).$$

Now apply the fBm covariance (2.11) to each term with $B_m = W^H(m\Delta)$. Write $C(a, b) := \text{Cov}(W^H(a\Delta), W^H(b\Delta))$; then

$$C(a, b) = \frac{\sigma^2}{2} \Delta^{2H} \left(a^{2H} + b^{2H} - |a - b|^{2H} \right).$$

Hence

$$\gamma(k) = C(n+1, n+k+1) - C(n+1, n+k) - C(n, n+k+1) + C(n, n+k).$$

Substitute the expression for $C(\cdot, \cdot)$ and observe that the a^{2H} and b^{2H} terms cancel pairwise

$$\begin{aligned}\gamma(k) &= \frac{\sigma^2}{2} \Delta^{2H} \left[-(|(n+1) - (n+k+1)|^{2H}) + (|(n+1) - (n+k)|^{2H}) \right. \\ &\quad \left. + (|n - (n+k+1)|^{2H}) - (|n - (n+k)|^{2H}) \right] \\ &= \frac{\sigma^2}{2} \Delta^{2H} \left[-|k|^{2H} + |k-1|^{2H} + |k+1|^{2H} - |k|^{2H} \right] \\ &= \frac{\sigma^2}{2} \Delta^{2H} \left(|k+1|^{2H} - 2|k|^{2H} + |k-1|^{2H} \right),\end{aligned}$$

which is (2.14). The result is independent of n , hence $\{G_n\}$ is stationary. $\square$

2.6.1.5 Long-range dependence

A defining characteristic of fGn is the long-range dependence and asymptotic decay of the autocovariance function $\gamma(k)$ [6]. These two properties are the backbone and synonymous with long-range memory.

Proposition 2.6.2 (Asymptotic behaviour of the fGn autocovariance). *As $|k| \rightarrow \infty$,*

$$\gamma(k) \sim \sigma^2 \Delta^{2H} H(2H-1) |k|^{2H-2}. \quad (2.15)$$

Proof. Assume $k \rightarrow +\infty$ and by symmetry $\gamma(-k) = \gamma(k)$. Let $p = 2H$ and consider the function $f(x) = x^p$ for $x > 0$. A second-order Taylor expansion around k gives, for ± 1 increments,

$$f(k \pm 1) = f(k) \pm f'(k) + \frac{1}{2} f''(k) + o(f''(k)), \quad k \rightarrow \infty,$$

since the remainder is $O(f^{(3)}(k))$ and $f^{(3)}(k) = O(k^{p-3})$.

Using the derivatives

$$f'(x) = px^{p-1}, \quad f''(x) = p(p-1)x^{p-2},$$

we obtain

$$\begin{aligned}f(k+1) + f(k-1) - 2f(k) &= \left(f(k) + f'(k) + \frac{1}{2} f''(k) + o(f''(k)) \right) \\ &\quad + \left(f(k) - f'(k) + \frac{1}{2} f''(k) + o(f''(k)) \right) - 2f(k) \\ &= f''(k) + o(f''(k)) = p(p-1)k^{p-2} + o(k^{p-2}).\end{aligned}$$

Insert this into (2.14) and dropping absolute values since $k \rightarrow +\infty$, we obtain

$$\gamma(k) = \frac{\sigma^2}{2} \Delta^{2H} \left((k+1)^p - 2k^p + (k-1)^p \right) \sim \frac{\sigma^2}{2} \Delta^{2H} p(p-1) k^{p-2}.$$

Finally, substitute $p = 2H$

$$\frac{1}{2} p(p-1) = \frac{1}{2} (2H)(2H-1) = H(2H-1),$$

which yields (2.15). $\square$

The exponent $2H - 2$ lies in $(-1, 0)$ when $H \in (1/2, 1)$, implying slow (non-summable) decay of $\sum_k |\gamma(k)|$ and thus long-range dependence and memory.

2.6.1.6 Spectral density of fractional Gaussian noise

Since fGn is a stationary sequence, it admits a discrete-time spectral density $f(\lambda)$ on $[-\pi, \pi]$

$$\gamma(k) = \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{ik\lambda} f(\lambda) d\lambda.$$

An explicit representation of this density, which is mathematically crucial for efficient simulation via the circulant embedding method is given by the periodic sum

$$f(\lambda) = 2 \sin(\pi H) \Gamma(2H + 1) (1 - \cos \lambda) \sum_{m=-\infty}^{\infty} \frac{1}{|\lambda + 2\pi m|^{2H+1}}, \quad \lambda \in [-\pi, \pi]. \quad (2.16)$$

For the purposes of this work, the most important feature of $f(\lambda)$ is its asymptotic behaviour near the origin. As $\lambda \rightarrow 0$, the $m = 0$ term dominates, and since $1 - \cos \lambda \approx \lambda^2/2$, the density scales as

$$f(\lambda) \approx |\lambda|^{1-2H}$$

When $H > 1/2$, this spectrum diverges at zero frequency. This strong low-frequency content is the frequency-domain manifestation of long-range dependence, explicitly mathematically confirming the directional persistence sought for our velocity-driven models.

2.6.1.7 Davies–Harte simulation of fGn and fBm

The numerical experiments and parameter inference pipelines in this thesis rely strictly on discrete-time observations. Therefore, the core computational requirement is not the evaluation of continuous pathwise integrals, but rather the ability to efficiently and exactly generate the correlated Gaussian increments G_n governed by the covariance structure in (2.14). This is achieved using the Davies-Harte method [15].

Let $\mathbf{G} = (G_0, \dots, G_{N-1})^\top$ denote fGn on an equidistant grid $t_n = n\Delta$, obtained as increments of fBm,

$$G_n := B^H(t_{n+1}) - B^H(t_n), \quad n = 0, \dots, N-1.$$

Since fGn is a centred stationary Gaussian sequence, it is fully characterised by its autocovariance $\gamma(k) = \text{Cov}(G_n, G_{n+k})$, and the covariance matrix of $\mathbf{G}$ is the Toeplitz matrix

$$\Gamma_N = \left(\gamma(|i - j|) \right)_{i,j=0}^{N-1}.$$

A direct Cholesky factorisation of Γ_N costs $O(N^3)$ and becomes prohibitive for large N .

The Davies–Harte method accelerates sampling by embedding Γ_N into a larger circulant covariance matrix, which is diagonalised by discrete Fourier modes and can therefore be sampled in $O(N \log N)$ time using FFTs [15, 19, 18].

2.6.1.7.1 Circulant embedding of size $M = 2N$ Let $M := 2N$. Define the length- M embedding vector $c = (c_0, \dots, c_{M-1})$ by

$$c_0 := \gamma(0), \quad c_k := \gamma(k) \quad (1 \leq k \leq N-1), \quad c_N := 0, \quad c_{M-k} := \gamma(k) \quad (1 \leq k \leq N-1). \quad (2.17)$$

Let $C \in \mathbb{R}^{M \times M}$ be the circulant matrix with first row $(c_0, \dots, c_{M-1})$. By construction, the top-left $N \times N$ block of C equals Γ_N . Hence, if one can sample a Gaussian vector $\mathbf{Y} \in \mathbb{R}^M$ with covariance C , then the first N components of $\mathbf{Y}$ form an exact sample from $\mathcal{N}(0, \Gamma_N)$.

2.6.1.7.2 Remark (alternative embedding size $M = 2(N-1)$). Some implementations take $M := 2(N-1)$ and embed using

$$c_0 = \gamma(0), \quad c_k = \gamma(k) \quad (1 \leq k \leq N-1), \quad c_{M-k} = \gamma(k) \quad (1 \leq k \leq N-2),$$

i.e. without the middle entry c_N because N is not an index in that convention. This produces a circulant matrix of size $2N-2$. Either convention is valid; what matters is that the resulting circulant matrix contains Γ_N as a principal block and has nonnegative FFT eigenvalues.

2.6.1.7.3 FFT diagonalisation and the sampling formula As usual, matrix diagonalisation is desired for further computations. Circulant matrices are diagonalised by the discrete Fourier transform (DFT).

Let $F \in \mathbb{C}^{M \times M}$ denote the unitary DFT matrix,

$$F_{jk} = \frac{1}{\sqrt{M}} e^{-2\pi i(j-1)(k-1)/M}, \quad j, k = 1, \dots, M, \quad \text{so that} \quad F^* F = I.$$

Let $\boldsymbol{\lambda} = (\lambda_0, \dots, \lambda_{M-1})^\top$ be the DFT of the first row c ,

$$\lambda_j = \sum_{k=0}^{M-1} c_k e^{-2\pi i j k / M}, \quad j = 0, \dots, M-1. \quad (2.18)$$

Lemma 1 (Circulant diagonalisation). *With the above definitions,*

$$C = F^* \text{diag}(\boldsymbol{\lambda}) F. \quad (2.19)$$

Proof. The columns of F^* are eigenvectors of any circulant matrix and the eigenvalues are given by the DFT of the first row [19, 18]. $\square$

2.6.1.7.4 Sampling in the Fourier domain. Assume $\lambda_j \geq 0$ for all j , which is checked numerically. Let $\mathbf{Z} \in \mathbb{C}^M$ be a standard complex Gaussian vector with $\mathbb{E}[\mathbf{Z}] = 0$ and $\mathbb{E}[\mathbf{Z}\mathbf{Z}^*] = I$, constrained by conjugate symmetry so that the inverse DFT is real

$$Z_{M-j} = \overline{Z_j}, \quad j = 1, \dots, \frac{M}{2} - 1, \quad \text{and} \quad Z_0, Z_{M/2} \in \mathbb{R}.$$

Define

$$\mathbf{Y} := F^* \text{diag}(\sqrt{\boldsymbol{\lambda}}) \mathbf{Z}. \quad (2.20)$$

Proposition 2.6.3 (Correct covariance and real-valuedness). *If $\lambda_j \geq 0$ for all j and $\mathbf{Z}$ satisfies the above symmetry, then $\mathbf{Y}$ in (2.20) is real-valued, centred Gaussian, and*

$$\text{Cov}(\mathbf{Y}) = C.$$

Consequently, $(Y_0, \dots, Y_{N-1})^\top \sim \mathcal{N}(0, \Gamma_N)$.

Proof. Gaussianity follows because $\mathbf{Y}$ is a linear transform of a Gaussian vector. Conjugate symmetry implies that $F^*(\sqrt{\boldsymbol{\lambda}} \odot \mathbf{Z})$ is real (a standard DFT fact). Finally,

$$\text{Cov}(\mathbf{Y}) = \mathbb{E}[\mathbf{Y}\mathbf{Y}^*] = F^* \text{diag}(\sqrt{\boldsymbol{\lambda}}) \mathbb{E}[\mathbf{Z}\mathbf{Z}^*] \text{diag}(\sqrt{\boldsymbol{\lambda}}) F = F^* \text{diag}(\boldsymbol{\lambda}) F = C,$$

where we used $\mathbb{E}[\mathbf{Z}\mathbf{Z}^*] = I$ and Lemma 1. The final claim follows from the block-embedding property of C . $\square$

2.6.1.7.5 From fGn to fBm by cumulative summation Once the increments $(G_0, \dots, G_{N-1})$ have been generated (e.g. by setting $G_n := Y_n$ for $n = 0, \dots, N-1$), an fBm path on the grid is obtained by

$$B_0 = 0, \quad B_{n+1} = B_n + G_n, \quad n = 0, \dots, N-1. \quad (2.21)$$

A full schematic view of the Davies-Harte simulation can be seen in Figure 3.8.

2.7 Topological Data Analysis

While classical spatial statistics, such as the Pair Correlation Function (PCF), are highly effective at capturing localized, pairwise interactions, they compress the data and fail to register macroscopic, higher-order structural information. In biological systems, cellular interactions give rise to complex geometric formations such as densely packed cohesive clusters and expansive, ring-like voids driven by steric exclusion and motility. To capture these spatial formations and shapes quantitatively we use Topological Data Analysis (TDA). TDA is a set of methodologies rooted in algebraic topology, designed to infer qualitative features of data irrespective of coordinate transformations, continuous deformations, or stochastic noise [10, 20]. Consequently, the features extracted via TDA will be utilized later in this thesis as robust summary statistics for parameter inference, providing a powerful, higher-order alternative to traditional pairwise measures like the PCF.

2.7.1 Foundations of Point-Set Topology and Metric Spaces

Topology is the study of properties that are preserved through continuous deformations, such as stretching, twisting, and bending, but not tearing or gluing. Unlike standard Euclidean geometry, which is heavily reliant on rigid metrics containing exact distances and angles, topology focuses on the generalized concepts of nearness and connectivity [37].

The power of topology lies in the definition of topological equivalence. Two spaces (X, τ_X) and (Y, τ_Y) are considered topologically equivalent if there exists a function $f : X \rightarrow Y$ that is a bijection, continuous, and possesses a continuous inverse f^{-1} , known as a homeomorphism. Properties that remain invariant under such transformations, such as the number of independent connected components or the existence of structural voids—are termed topological invariants [37].

Within the context of this thesis, the dataset consists of a finite set of two-dimensional coordinates representing cell centroids, $X = \{x_1, x_2, \dots, x_N\} \subset \mathbb{R}^2$. Using standard subspace topology from $\mathbb{R}^2$, this discrete finite set possesses a trivial and uninformative topology: it is simply a collection of disconnected points. To extract meaningful geometric information, we must systematically build a continuous topological space on top of these discrete points by leveraging their underlying metric structure.

2.7.2 Simplicial Complexes: Bridging the Discrete and Continuous

To bridge the gap between discrete point clouds and continuous topological spaces, TDA utilizes combinatorial structures known as simplicial complexes [25]. These complexes allow us to approximate continuous, complex shapes using simple, discrete geometric building blocks called simplices.

A geometric k -simplex σ is defined as the convex hull of $k + 1$ affinely independent points $V = \{v_0, v_1, \dots, v_k\}$ in $\mathbb{R}^d$, where $d \geq k$. Algebraically, this is expressed as

$$\sigma = \left\{ \sum_{i=0}^k \lambda_i v_i \mid \sum_{i=0}^k \lambda_i = 1, \lambda_i \geq 0 \text{ for all } i \right\}. \quad (2.22)$$

Intuitively, simplices represent the simplest and most fundamental possible shapes and building blocks in any given dimension. This meaning that a 0-simplex is a single point (vertex), a 1-simplex is a line segment (edge), a 2-simplex is a solid triangle, and a 3-simplex is a solid tetrahedron. A face of a k -simplex σ is the convex hull of any non-empty subset of its vertices.

A geometric simplicial complex K is formed by a finite collection of geometric simplices that satisfies two strict geometric rules. The first being that every face of a simplex in K is also in K , and the non-empty intersection of any two simplices $\sigma_1, \sigma_2 \in K$ is exactly a face of both σ_1 and σ_2 .

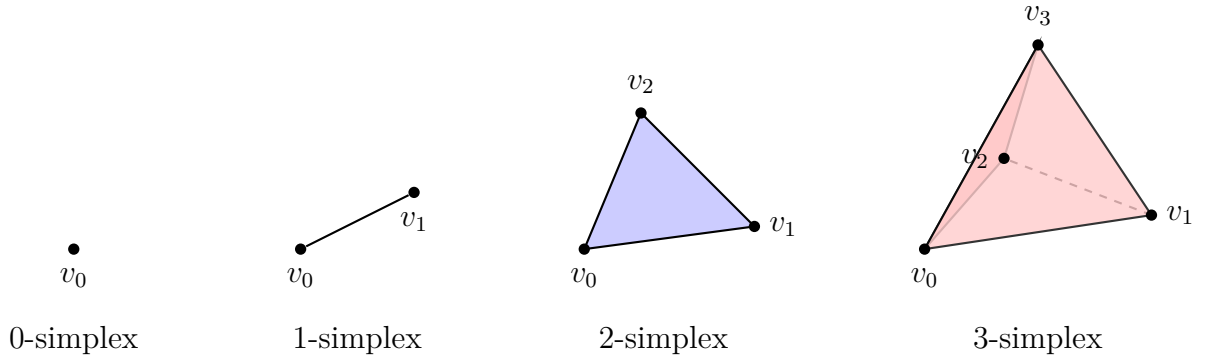


Figure 2.11: Geometric representations of k -simplices for $k \in \{0, 1, 2, 3\}$. A simplicial complex is constructed by gluing these fundamental geometric building blocks together along their faces.

While geometric simplices are intuitive to visualize, tracking continuous floating-point coordinates for intersecting shapes in high-dimensional space is computationally prohibitive. Therefore, algorithms rely on the abstract simplicial complex. Given a vertex set V , an abstract simplicial complex is a purely combinatorial family of subsets $K \subseteq \mathcal{P}(V)$, where $\mathcal{P}(V)$ denotes the power set (the set of all possible subsets) of V . This family is defined strictly by the rule that if a set $\sigma \in K$, all of its subsets $\tau \subseteq \sigma$ must also belong to K . This strips away the coordinate geometry, reducing the topology to a purely discrete, graph-like list of vertex connections.

2.7.2.1 Čech and Vietoris-Rips Complexes

To build a biologically meaningful abstract simplicial complex from the cell centroid point cloud X , we must establish a rule for when vertices should be connected. The most direct theoretical approach is to treat each cell as a physical body with a defined interaction radius.

Let $\mathcal{U} = \{B_{\epsilon/2}(x_i)\}_{i=1}^N$ be a collection of closed Euclidean balls, each with radius $\epsilon/2$, centered at every data point $x_i \in X$. The Čech complex, denoted $\check{C}_\epsilon(X)$, dictates that a higher-dimensional k -simplex $[x_{i_0}, \dots, x_{i_k}]$ is formed if and only if the exact common intersection of their corresponding spheres is strictly non-empty

$$\bigcap_{j=0}^k B_{\epsilon/2}(x_{i_j}) \neq \emptyset. \quad (2.23)$$

The Čech complex is considered the theoretical gold standard because it is governed by the Nerve Theorem [27, 20]. Because convex Euclidean balls form a "good cover", the Nerve Theorem mathematically guarantees that the combinatorial Čech complex is strictly homotopy equivalent to the true continuous union of the physical balls, $\bigcup_{i=1}^N B_{\epsilon/2}(x_i)$.

However, verifying the simultaneous intersection of $k + 1$ spheres becomes computationally intractable as data scales. To circumvent this bottleneck, applied TDA

predominantly relies on the Vietoris-Rips (VR) complex, $\text{VR}_\epsilon(X)$. The VR complex relaxes the intersection constraint, requiring only that all points within the simplex are pairwise within a maximum distance of ϵ ,

$$\text{VR}_\epsilon(X) = \left\{ \sigma \subseteq X \mid \max_{u,v \in \sigma} d(u,v) \leq \epsilon \right\}. \quad (2.24)$$

Because it is a flag complex, the VR complex is entirely determined by its 1-dimensional edges. Whenever a group of vertices is fully pairwise connected, the corresponding solid higher-dimensional simplex is automatically generated. While it is an approximation and lacks the strict homotopy guarantee of the Čech complex, the VR complex computationally bounds it through $\check{C}_\epsilon(X) \subseteq \text{VR}_\epsilon(X) \subseteq \check{C}_{\sqrt{2}\epsilon}(X)$, ensuring its validity as a topological proxy [25].

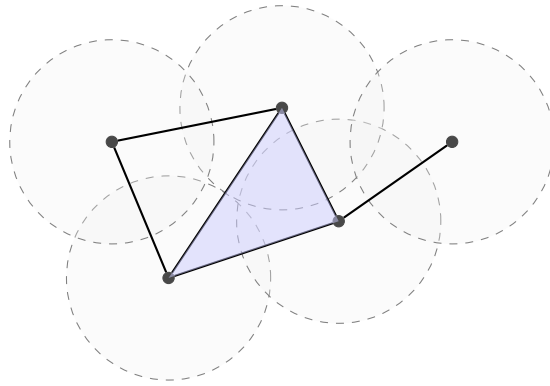


Figure 2.12: Construction of a Vietoris-Rips complex $\text{VR}_\epsilon(X)$. Disks of radius $\epsilon/2$ are drawn around each point. When two disks intersect (pairwise distance $\leq \epsilon$), an edge is formed. Because all three vertices of the central triangle are pairwise connected, the 2-simplex (shaded blue) is automatically added.

2.7.3 Algebraic Topology and Simplicial Homology

Once the point cloud is structured as an abstract simplicial complex K , we require a systematic mathematical framework to identify and count its topological invariants such as holes and clusters. This is achieved through Simplicial Homology, which transforms topological classification problems into computationally tractable linear algebra [27].

2.7.3.1 Chain Complexes and the Boundary Operator

To formalize the algebraic structure of the complex, we define the k -th chain group, denoted $C_k(K)$, as the free vector space over a field $\mathbb{F}$ whose basis elements are the k -simplices present in K . In the context of TDA, computations are almost exclusively performed over the finite field $\mathbb{Z}_2$ for computational efficiency, as it eliminates the need to track the orientation of the simplices. A k -chain $c \in C_k(K)$ is thus a formal linear combination of k -simplices

$$c = \sum_i \alpha_i \sigma_i, \quad \alpha_i \in \mathbb{Z}_2. \quad (2.25)$$

The structural relationship between k -simplices and $(k-1)$ -simplices is mapped by the boundary operator, a linear transformation $\partial_k : C_k \rightarrow C_{k-1}$. The boundary of a k -simplex $\sigma = [v_0, \dots, v_k]$ is evaluated as the sum of its $(k-1)$ -dimensional faces

$$\partial_k([v_0, \dots, v_k]) = \sum_{i=0}^k [v_0, \dots, \hat{v}_i, \dots, v_k], \quad (2.26)$$

where $\hat{v}_i$ denotes the omission of the i -th vertex. Operating over $\mathbb{Z}_2$ implies that $1+1=0$. Thus, if two adjacent k -simplices share a $(k-1)$ -dimensional face (e.g., an edge shared by two connected triangles), that shared internal face appears twice in the algebraic sum and perfectly cancels out, leaving strictly the exterior boundary of the composite shape.

A foundational theorem of algebraic topology states that the boundary of a boundary is empty. Formally, the composition of two consecutive boundary operators is the zero map

$$\partial_{k-1} \circ \partial_k = 0. \quad (2.27)$$

This sequence of vector spaces and boundary operators forms what is known as a chain complex [27]

$$\dots \xrightarrow{\partial_{k+2}} C_{k+1}(K) \xrightarrow{\partial_{k+1}} C_k(K) \xrightarrow{\partial_k} C_{k-1}(K) \xrightarrow{\partial_{k-1}} \dots \xrightarrow{\partial_1} C_0(K) \xrightarrow{\partial_0} 0. \quad (2.28)$$

2.7.3.2 Cycles, Boundaries, and the Homology Quotient

The fundamental identity $\partial^2 = 0$ allows us to define two critical linear subspaces within $C_k(K)$:

1. **The Cycle Group** $Z_k = \ker(\partial_k) = \{c \in C_k \mid \partial_k(c) = 0\}$. These are chains that possess no boundary. For example, a closed loop of edges forms a 1-cycle.
2. **The Boundary Group** $B_k = \text{im}(\partial_{k+1}) = \{\partial_{k+1}(d) \mid d \in C_{k+1}\}$. These are chains that act as the exact boundary of a higher-dimensional geometric structure. For example, the outer edges surrounding a filled solid 2-simplex form a 1-boundary.

Because $\text{im}(\partial_{k+1}) \subseteq \ker(\partial_k)$, it is guaranteed that $B_k \subseteq Z_k$ (i.e., every boundary is inherently a cycle). However, not every cycle is a boundary. A cycle that does not bound a higher-dimensional simplex represents a genuine topological "hole" in the complex. To isolate these true topological features, we construct the quotient vector space.

The k -th Homology Group $H_k(K)$ is defined algebraically as

$$H_k(K) = Z_k / B_k = \ker(\partial_k) / \text{im}(\partial_{k+1}). \quad (2.29)$$

In this quotient space, two cycles $z_1, z_2 \in Z_k$ are considered homologous (belonging to the same equivalence class) if their difference is a boundary, meaning $z_1 - z_2 \in B_k$.

The dimension of this vector space over $\mathbb{Z}_2$ is defined as the k -th Betti number, $\beta_k = \dim(H_k(K))$. The Betti numbers provide a rigorous mathematical count of the independent topological features:

- β_0 : The number of connected components.
- β_1 : The number of 1-dimensional holes (loops).
- β_2 : The number of 2-dimensional voids (enclosed cavities).

2.7.4 Persistent Homology and Stability

Analyzing a static simplicial complex evaluated at a single proximity threshold ϵ is inherently limited, as it requires an arbitrary parameter choice. In physical cellular systems, structures exist simultaneously across a multitude of spatial scales. Persistent Homology circumvents this limitation by analyzing the topological evolution across all possible scales continuously [10].

By sweeping the proximity parameter ϵ continuously from 0 to ∞ , we generate a strictly nested sequence of simplicial complexes, known formally as a filtration

$$\emptyset = K_0 \subseteq K_{\epsilon_1} \subseteq K_{\epsilon_2} \subseteq \dots \subseteq K_{\epsilon_m} = K_{max}. \quad (2.30)$$

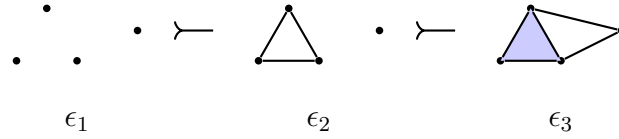


Figure 2.13: A VR complex filtration over increasing spatial scales ϵ . At ϵ_1 , the complex consists of discrete points ($\beta_0 = 4, \beta_1 = 0$). At ϵ_2 , an empty triangle forms ($\beta_0 = 2, \beta_1 = 1$). At ϵ_3 , the triangle is filled by a 2-simplex, and the isolated point connects, yielding a single connected component and eliminating the hole ($\beta_0 = 1, \beta_1 = 0$).

Applying the homology functor to this nested sequence produces a persistence module. As the scale ϵ increases, topological features are born (when a new cycle forms) and subsequently die (when the cycle is filled in by a higher-dimensional solid simplex).

The lifespan of every independent topological feature is recorded as an interval $[b, d]$, representing its birth and death scales. The aggregate collection of these intervals is visualized as a Persistence Diagram. The primary theoretical advantage of extracting these diagrams is mathematically guaranteed by the Stability Theorem [13]. The theorem dictates that the Bottleneck distance (d_B) between two persistence diagrams is strictly upper-bounded by the Hausdorff distance (d_{GH}) between their originating point clouds

$$d_B(D(X), D(Y)) \leq 2d_{GH}(X, Y). \quad (2.31)$$

This establishes that persistent homology is intrinsically resistant to stochastic noise; minor perturbations in cellular coordinates generate only proportional, bounded shifts in the persistence diagram, validating its utility as a reliable summary statistic.

2.7.5 Geometric Exactness: Delaunay Triangulations and the Alpha Complex

While the Vietoris-Rips complex is computationally ubiquitous, it suffers from severe geometric limitations when deployed in dense, low-dimensional physical spaces, such as in vitro 2D cell cultures. Because the VR complex is a flag complex, it automatically inserts higher-dimensional solids whenever the outer edges form a clique. Consequently, if four cellular centroids are densely packed in a 2D plane, the VR filtration will mathematically insert a 3-simplex (a solid tetrahedron). Because a tetrahedron cannot physically exist in 2D without self-intersection, these artificial simplices distort the pipeline’s correspondence to biological reality (Figure 2.14).

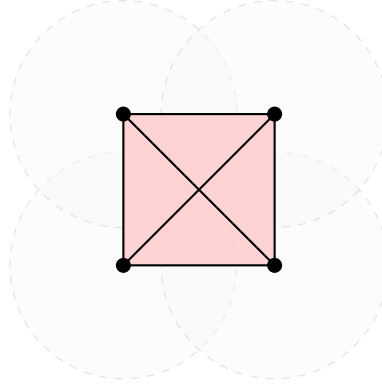


Figure 2.14: The geometric flaw of the Vietoris-Rips complex in 2D space. When four cells (black nodes) are pairwise connected within the threshold ϵ , the VR flag complex automatically hallucinates a solid 3-simplex (a 3D tetrahedron, shaded red). This forces 2D biological data into a false 3D geometry.

To achieve geometric exactness, we utilize the Alpha Complex, an algorithmic construction rooted in spatial packing theory. The foundation of the Alpha Complex is the Voronoi tessellation of the point cloud X . Let $V(x_i)$ denote the Voronoi cell associated with a point $x_i \in X$, defined as the region of space closer to x_i than to any other point in X

$$V(x_i) = \{y \in \mathbb{R}^2 \mid d(y, x_i) \leq d(y, x_j) \text{ for all } x_j \in X\}. \quad (2.32)$$

The dual graph of the Voronoi tessellation is the Delaunay Triangulation, $\text{Del}(X)$, a simplicial complex containing an edge between x_i and x_j if and only if their Voronoi cells share a boundary (Figure 2.15).

To introduce a spatial scale ϵ , we restrict the Voronoi cells to the intersection with closed Euclidean balls of radius $\epsilon/2$, defined as $R_\epsilon(x_i) = V(x_i) \cap B_{\epsilon/2}(x_i)$. The Alpha Complex, denoted $\mathcal{A}_\epsilon(X)$, is defined as the exact nerve of this restricted

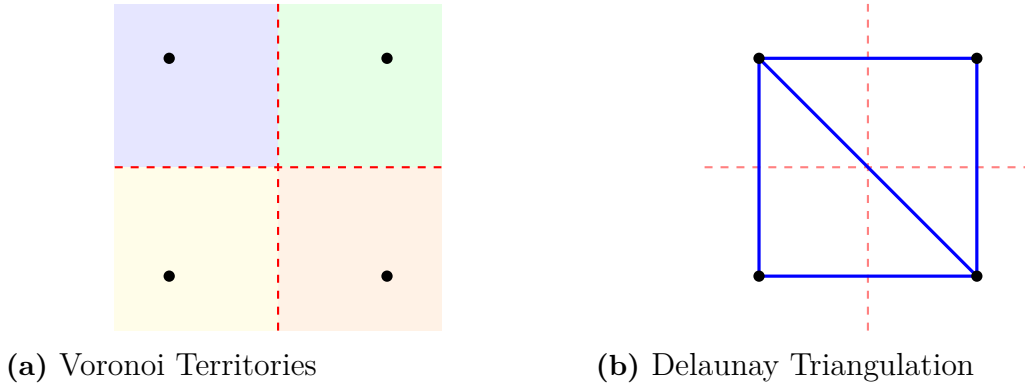


Figure 2.15: Resolving the geometric flaw using spatial partitioning. (a) The 2D plane is partitioned into Voronoi territories, where red dashed borders indicate equidistant boundaries between cells. (b) The dual Delaunay Triangulation connects cells only if their territories share a border. Because 2D borders cannot cross, the resulting planar web mathematically forbids 3D self-intersections.

Voronoi cover. A simplex $\sigma = [x_{i_0}, x_{i_1}, \dots, x_{i_k}]$ is included in $\mathcal{A}_\epsilon(X)$ if and only if the intersection of their restricted regions is non-empty

$$\bigcap_{j=0}^k R_\epsilon(x_{i_j}) \neq \emptyset. \quad (2.33)$$

Because $\mathcal{A}_\epsilon(X)$ is a subcomplex of the Delaunay triangulation, it inherits the dimensional restrictions of the embedding space; it mathematically forbids the formation of 3-simplices in $\mathbb{R}^2$. Furthermore, by the Nerve Theorem, the Alpha Complex is strictly homotopy equivalent to the exact geometric union of the growing cell radii $\bigcup_{i=1}^N B_{\epsilon/2}(x_i)$ at every scale ϵ . Consequently, tracking the persistent homology of the Alpha filtration provides a flawless, noise-reduced topological summary of the physical cell packing geometry [20].

2.7.6 Continuous Topological Summaries: Persistence Images

While persistence diagrams capture the structural evolution of the Alpha filtration, their formulation as multisets of discrete intervals $[b, d)$ renders them inherently unstable for continuous gradient-based or distance-based optimization frameworks. Standard vectorizations, such as discrete Betti curves, inherently rely on step-functions. A Betti curve evaluates the exact integer count of topological features alive at a specific radius r . If a stochastic fluctuation causes a structural biological void to form at $r = 5.1$ instead of $r = 5.0$, the discrete boundary of the step-function triggers an artificially massive spike in the metric discrepancy.

To resolve the extreme sensitivity of discrete topological vectors to stochastic noise, we map the persistence diagrams into a strictly continuous, differentiable functional space known as Persistence Images [1].

The construction of a Persistence Image begins with a linear transformation of the persistence diagram D . Each birth-death coordinate $(b, d) \in D$ is mapped to a birth-persistence coordinate (b, p) , where $p = d - b$ represents the absolute lifespan of the topological feature. This transformation rotates the diagram, reorienting the interval data such that topological noise (short-lived features) is compressed towards the horizontal axis $p = 0$, while robust, macroscopic tissue structures are elevated.

To translate these discrete coordinates into a continuous topology, a two-dimensional Gaussian probability distribution is centered over every point (b_i, p_i) in the transformed space. The cumulative persistence surface, $\rho(x, y)$, is defined as the weighted sum of these continuous distributions

$$\rho(x, y) = \sum_{i=1}^{|D|} w(p_i) \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(x - b_i)^2 + (y - p_i)^2}{2\sigma^2}\right), \quad (2.34)$$

where σ^2 controls the Gaussian variance (blur radius). The crucial mathematical innovation of the Persistence Image is the continuous weighting function $w(p_i)$. By defining a weight function that increases monotonically with persistence, such as $w(p) = p^2$, the formulation mathematically annihilates stochastic noise. The hundreds of transient, minuscule loops generated by random cellular motility are assigned weights near zero, fading seamlessly into the background. Conversely, stable voids formed by steric exclusion are heavily weighted, manifesting as bright, continuous peaks in the probability surface.

Finally, the continuous surface $\rho(x, y)$ is discretized by integrating over a uniform rectangular grid, yielding a finite-dimensional matrix (the Persistence Image). This matrix can be flattened into a robust, noise-resistant vector. Unlike jagged Betti curves, if a biological void shifts slightly due to spatial stochasticity, the corresponding Gaussian peak smoothly shifts across the adjacent pixels, producing a proportional and well-behaved penalty in the mathematical distance metric.

Algorithm 3 Sequential Monte Carlo ABC (SMC-ABC)

Require: Prior $\pi(\theta)$; simulation $x \sim p(\cdot \mid \theta)$; summary map $s(\cdot)$; distance $E(\cdot, \cdot)$; observed data y ; number of particles P ; sequence of tolerances $\epsilon_1 > \epsilon_2 > \dots > \epsilon_T$; candidate kernels $\{K_t(\theta \mid \tilde{\theta})\}_{t=2}^T$.

- 1: Compute observed summaries $s_y \leftarrow s(y)$.
- 2: **Initialization** ($t = 1$).
- 3: **for** $p = 1, 2, \dots, P$ **do**
- 4: **repeat**
- 5: Draw $\theta_1^{(p)} \sim \pi(\theta)$.
- 6: Simulate $x_1^{(p)} \sim p(\cdot \mid \theta_1^{(p)})$ and compute $s_1^{(p)} \leftarrow s(x_1^{(p)})$.
- 7: Compute distance $d_1^{(p)} \leftarrow E(s_1^{(p)}, s_y)$.
- 8: **until** $d_1^{(p)} \leq \epsilon_1$
- 9: Set weight $w_1^{(p)} \leftarrow 1$.
- 10: **end for**
- 11: Normalize weights: $w_1^{(p)} \leftarrow w_1^{(p)} / \sum_{q=1}^P w_1^{(q)}$.
- 12: **Iterative populations** ($t = 2, \dots, T$).
- 13: **for** $t = 2, 3, \dots, T$ **do**
- 14: **for** $p = 1, 2, \dots, P$ **do**
- 15: **repeat**
- 16: Sample ancestor index a from $\{1, \dots, P\}$ with probabilities $\{w_{t-1}^{(q)}\}$.
- 17: Draw candidate $\theta^* \sim K_t(\theta \mid \theta_{t-1}^{(a)})$.
- 18: Simulate $x^* \sim p(\cdot \mid \theta^*)$ and compute $s^* \leftarrow s(x^*)$.
- 19: Compute distance $d^* \leftarrow E(s^*, s_y)$.
- 20: **until** $d^* \leq \epsilon_t$
- 21: Set $\theta_t^{(p)} \leftarrow \theta^*$ and $s_t^{(p)} \leftarrow s^*$.
- 22: Update unnormalised weight

$$\tilde{w}_t^{(p)} \leftarrow \frac{\pi(\theta_t^{(p)})}{\sum_{q=1}^P w_{t-1}^{(q)} K_t(\theta_t^{(p)} \mid \theta_{t-1}^{(q)})}.$$

- 23: **end for**
- 24: Normalize weights: $w_t^{(p)} \leftarrow \tilde{w}_t^{(p)} / \sum_{q=1}^P \tilde{w}_t^{(q)}$.
- 25: **end for**
- 26: **Output:** weighted particle system $\{(\theta_T^{(p)}, w_T^{(p)})\}_{p=1}^P$ approximating a low-tolerance ABC posterior at ϵ_T .
- 27: Posterior summaries and point estimates are obtained by weighting functions of $\theta_T^{(p)}$ with $w_T^{(p)}$.

3

Methods

3.1 Data and experimental design

The experimental data for this work consists of in vitro time-lapse imagery derived from 2D cell culture assays. The cell lines were obtained from the Human Glioblastoma Cell Culture (HGCC) database [56], which provides a multifaceted repository of patient-derived cell lines.

The imaging data was captured using phase-contrast microscopy, chosen for its ability to visualize cellular morphology without the need for exogenous staining. The raw dataset comprises high-resolution image sequences (1388×1023 pixels) recorded systematically using a robotic imaging device. The data is organized into well-plate coordinates ranging from A1 to H12, as illustrated in Figure 3.1, with each coordinate representing a distinct experimental replicate or cell line condition.

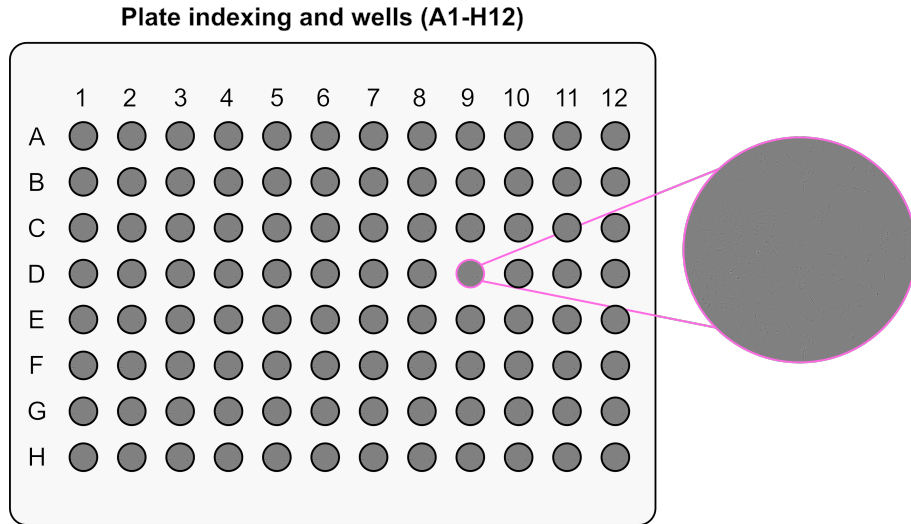


Figure 3.1: Visualisation of the well-plate coordinate system.

Each time-series consists of exactly 370 frames, covering a total observation period from inoculation ($t = 0$) up to a terminal timestamp of 4 days, 23 hours, and 7 minutes. This temporal resolution of 20 minutes was chosen to capture both rapid motility events, such as "run-and-tumble" reorientations, and slower population-level

dynamics, such as proliferation and confluence saturation.

3.1.1 Cell lines, time intervals, and dataset splitting

To ensure the robust generalization of the segmentation and inference pipeline, the dataset was curated to maximize phenotypic diversity. Glioblastoma cells exhibit significant morphological plasticity, transitioning between mesenchymal (elongated, spindle-like) and amoeboid (rounded, deformable) states [42]. As detailed in the theoretical framework (Section 2.1), resolving these states relies on identifying specific protrusive structures filopodia and lamellipodia [2].

Training, Validation, and Test Split

Ground truth generation for segmentation requires pixel-level annotation, which is highly resource-intensive. Therefore, a representative subset of 127 images, containing approximately 35,000 cells across the A1–H12 cell lines, was manually annotated using Napari [38]. Each annotated frame was exported as a labelled instance `.tif` file.

A stratified sampling strategy was employed to capture the full distribution of experimental variability along two primary axes:

1. **Morphological Distribution:** Images were chosen to include both sparse cultures (dominated by single-cell migration) and dense, confluent monolayers (dominated by cell-cell interactions and contact inhibition).
2. **Temporal Variance:** Frames were extracted from varied timepoints across the five-day duration to ensure robustness against temporal changes in contrast and debris accumulation.

As visualized in Figure 3.2, these 127 annotated images were partitioned into Training, Validation, and Test sets. Crucially, the Validation and Test sets were blinded, meaning they contain cell lines and time intervals distinctly separate from those seen during training. This strict separation is enforced to prevent temporal leakage and avoid overfitting to specific well-plate conditions.

3.2 Segmentation and tracking

The first crucial component of the inference pipeline is the construction of reliable cell trajectories and morphology estimates from the phase-contrast images. Because all downstream parameter inference depends heavily on the correctness of detected objects, segmentation is treated as a foundational methodological problem. Errors that merge neighbouring cells in dense cultures propagate nonlinearly: merged instances bias centroid locations, inflate apparent cell areas, produce cascading errors in radius estimation, and distort displacement statistics by inducing spurious track identity changes.

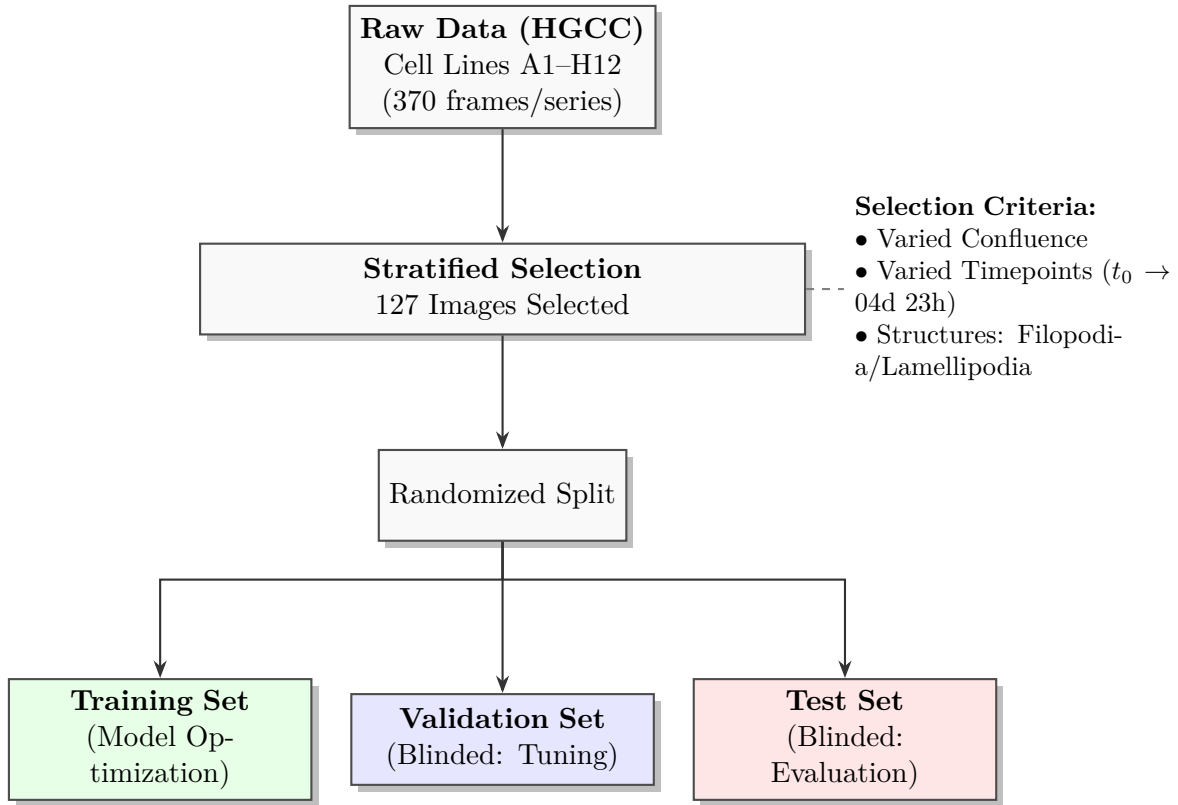


Figure 3.2: Schematic of the data curation and splitting strategy. From the raw HGCC dataset [56], 127 frames were manually selected to maximize morphological variance (e.g., distinguishing filopodia and lamellipodia) and temporal coverage. Validation and Test sets are blinded to ensure the model generalizes to unseen cell lines and timepoints.

To mitigate this, we benchmark multiple segmentation strategies on the annotated dataset and adopt a boundary-aware probabilistic post-processing step to enforce separation in crowded scenes. The theoretical foundations for the chosen architectures—including probabilistic outputs, encoder–decoder CNNs, residual encoders, and transformer-based segmentation are detailed in Section 2.2.

This section describes the concrete configuration and implementation used in this thesis, including hyperparameter selection, the boundary-aware probability mapping used at inference time, and the sanity checks deployed to validate the temporal consistency of derived biological quantities.

3.2.1 Preprocessing and annotation protocol

To train the boundary-aware segmentation models, the instance label masks generated during the dataset curation (Section 1.1) required transformation. The raw ground truth masks were constructed as instance images, where the background is labelled 0 and each unique cell is assigned a distinct integer label.

For model training, these instance labels were transformed into a three-class seman-

tic representation: background, cell interior, and cell boundary. The explicit boundary class is critical; in dense cultures, separation errors are dominated by cell-cell contacts. Training the network to predict an explicit boundary label provides a dedicated channel for learning separation cues that would otherwise be ambiguously absorbed into the cell-interior probability.

3.2.2 Model configurations and training procedure

We evaluate three methodological families of segmentation strategies:

1. **CNN encoder-decoder models:** Standard U-Net, U-Net++, and a U-Net variant equipped with a ResNet34 backbone.
2. **Transformer-based segmentation:** SegFormer (utilising the MiT-B0 backbone).
3. **Classical baseline:** Fogbank method, a non-learning pipeline based on hand-crafted filtering and morphological operations.

All learning-based models are trained to predict the same three-channel probability field $\hat{Y} \in [0, 1]^{H \times W \times 3}$, representing the background, cell interior, and cell boundary. Maintaining a consistent output format ensures that differences in performance can be attributed primarily to the representational capacity and inductive bias of the architectures rather than incompatible objective functions.

To rigorously evaluate the impact of spatial context and memory constraints on segmentation performance, the respective deep learning architectures were systematically trained and tested across various hyperparameter configurations. Specifically, the neural networks were evaluated using combinations of fixed image crop sizes (1024×1024 , 768×768 , and 512×512 pixels) alongside varying batch sizes (1, 2, and 4). The classical Fogbank method was excluded from this parameter search, as it serves as a non-learning baseline.

3.2.2.1 Training objective and optimisation protocol

All deep models were trained as supervised pixel-wise classifiers. Let $z_c(u)$ denote the logit produced by the network for class c at pixel u , and let $p_c(u)$ denote the corresponding softmax probability. The training target is a one-hot encoded label field $y_c(u) \in \{0, 1\}$. Optimisation minimises a combination of weighted cross-entropy and Dice loss, as detailed in Section 2.2. The central practical motivation for this composite objective is twofold: cross-entropy stabilises learning and encourages calibrated probabilities, while the Dice loss directly optimises overlap under severe class imbalance.

To manage variable image sizes and increase the diversity of local contexts observed during training, the networks were trained on the aforementioned fixed-size crops sampled from the original microscopy frames. Treating the crop size as a controlled

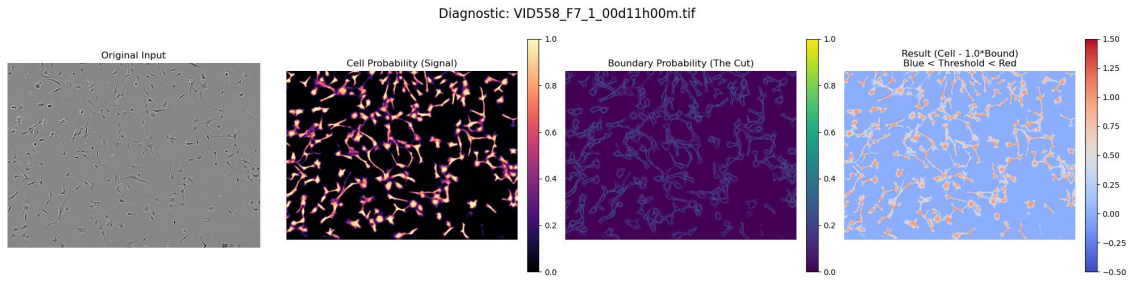


Figure 3.3: Example of inference-time probability fields produced by the segmentation model: original input, cell-interior probability, boundary probability, and the adjusted map constructed by Eq. (2.1). Bright regions correspond to high class confidence. Boundary-aware subtraction suppresses connectivity across cell–cell contacts, facilitating instance separation after thresholding.

parameter allows for a deliberate trade-off between global spatial context (larger crops) and batch-level diversity or memory efficiency (smaller crops). Furthermore, standard geometric and intensity augmentations were applied to reduce overfitting and improve invariance to minor changes in acquisition conditions.

Validation was performed periodically during training. Model selection was conducted by choosing the configuration that maximised overlap-based segmentation performance while maintaining qualitative robustness in highly confluent regimes. The outcome of this process is summarised in the quantitative comparison (Figure 4.1), resulting in the U-Net with a ResNet34 encoder being selected as the default segmentation engine for the downstream inference pipeline.

3.2.3 Inference-time probability mapping and instance extraction

The trained networks output class probabilities rather than binary masks. However, direct thresholding of the cell-interior probability $p_{\text{cell}}(u)$ tends to merge touching cells into single connected components at high densities. To address this, we employ the boundary-aware probability mapping using the adjusted cell score introduced in Section 2.2.

For all deep models evaluated in this work, the mapping hyperparameters were set to $\lambda = 1.0$ and $\tau = 0.5$. These values were selected via cross-validation, specifically optimising to minimise the instance-level false negative rate (i.e., the number of completely missed cells). In dense cultures, pixel-level metrics can be misleading; a model may achieve a high global Dice Similarity Coefficient by correctly classifying the vast majority of cell-interior pixels, yet fail to delineate individual boundaries, effectively merging multiple cells together. By tuning λ and τ to minimise missed cell instances rather than strictly maximising pixel overlap, the pipeline actively penalises under-segmentation and ensures robust object separation.

3.2.4 Evaluation protocol and metrics

Segmentation quality is quantified by comparing predicted masks against the expert-annotated ground truth. The principal metric is the Dice Similarity Coefficient (DSC), complemented by the Jaccard index (IoU) as a stricter overlap criterion; both metrics are formally defined in Section 2.2.

Beyond scalar metrics, qualitative inspection is performed to confirm that the highest-scoring models do not achieve their accuracy by systematically over-smoothing boundaries or failing entirely in dense regimes. This verification step is essential, as aggregate overlap metrics can often mask biologically relevant failure modes, such as occasional catastrophic cell merges. This is visually validated by generating overlay imagery that explicitly highlights spatial discrepancies through the assignment of False Negative, False Positive, and True Positive pixels (see Section 3.2.4).

3.3 Cell Tracking and Trajectory Reconstruction

Following instance segmentation, individual cell trajectories were reconstructed by linking cell centroids across consecutive temporal frames. Let $\mathbf{x}_i(t)$ denote the centroid of the i -th segmented instance at time t . The tracking problem consists of identifying the corresponding centroid $\mathbf{x}_j(t + \Delta t)$ in the subsequent frame that represents the same biological cell.

We performed particle linking using the `trackpy` library [3], which implements the Crocker-Grier algorithm [14]. This framework resolves identity assignments by minimizing a global cost function based on the squared displacements between candidate pairs.

3.3.1 Temporal Resolution and Precise Timestamping

A critical challenge in this dataset is the coarse and variable temporal resolution. While the nominal sampling interval was set to $\Delta t \approx 30$ minutes, experimental constraints led to variations in the exact acquisition timing. To ensure the highest fidelity in estimating diffusion parameters, we revoked the assumption of a fixed frame rate.

Instead, exact acquisition timestamps were parsed from the image metadata (filename format: DDdHHhMMm). Let T_k denote the absolute timestamp of frame k converted into total seconds. The elapsed time $t(k)$ for any point in a trajectory relative to the initial inoculation ($t = 0$) is given by

$$t(k) = [T_k - T_0]_{\text{seconds}} \quad (3.1)$$

By processing all temporal data in seconds, we prevent quantization errors and ensure that the MSD calculation shown in 3.2 is accurate even when the lag time τ fluctuates between frames. For biological interpretability, final diffusion coefficients

D derived from the fit $\text{MSD} \propto 4Dt^\alpha$ are reported in both SI units ($\mu\text{m}^2/\text{s}$) and standard biological units ($\mu\text{m}^2/\text{h}$).

Despite this precision, the average interval ($\Delta t \approx 1800$ s) remains significantly larger than the timescale of rapid cellular fluctuations. For a cell migrating at approximately $1.0 \mu\text{m}/\text{min}$, the displacement between frames can exceed $24 \mu\text{m}$. Consequently, a standard search radius of $L_{\text{max}} = 20$ pixels would systematically fail.

3.3.2 Evaluated Tracking Strategies

To address this challenge while maintaining computational feasibility, we implemented and evaluated three distinct tracking strategies:

1. **Standard Euclidean Tracking (Baseline):** L_{max} was conservatively set to 20 pixels. This serves as a control to quantify the "survival bias" introduced by standard parameters found in literature.
2. **Adaptive Large-Displacement Tracking:** L_{max} was expanded to 50 pixels to theoretically capture the fast-moving subpopulation calculated above (> 24 pixels). To manage combinatorial complexity in dense regions, an adaptive subnet solver was employed to iteratively reduce the search radius if candidate links exceeded computational limits.
3. **Velocity-Predictive Tracking (Proposed):** This strategy utilizes a Nearest Velocity Predictor to model directional persistence. It projects a particle's expected position at $t + \Delta t$ based on its prior velocity vector. The search is then performed within a localized radius ($L_{\text{local}} = 20$ pixels) around the *predicted* position. This allows for the linking of large displacements (> 50 pixels) without the combinatorial explosion associated with a global 50-pixel search radius.

3.3.3 Gap Closing and Filtering

To account for transient segmentation failures such as temporary occlusions or under-segmentation errors, a memory parameter of 3 frames was implemented across all strategies. This allows a trajectory to bridge a gap of up to $3\Delta t$, or approximately 1.5 hours, if a suitable candidate reappears within the projected search radius. Finally, to ensure robust statistics for the MSD calculation, any trajectories persisting for fewer than 10 frames were discarded.

3.3.4 Data-Driven Validation of Velocity Persistence

To empirically validate the hypothesis that glioblastoma migration is velocity-driven rather than purely diffusive, we devised a data-driven testing framework. This is done on the three tracking strategies simultaneously to a stratified subset of $N = 96$ experimental replicates.

The performance of the Velocity-Predictive method relative to the Standard baseline serves as a proxy test for persistence. If cell motion were purely Brownian, the Velocity Predictor, which assumes correlation between steps, would yield no improvement or even degrade performance compared to the Standard tracker. Conversely, a significant increase in the number of recovered trajectories using the Velocity Predictor provides empirical evidence of directional persistence in the underlying motility mechanism.

3.4 Feature extraction

The primary objective of this pipeline is to extract quantifiable, biologically descriptive features from the imaging data to bridge the gap between experimental observations and the theoretical models. We focus on three core metrics: mean cell radius, the diffusion coefficient (and its fractional generalization, the Hurst parameter), and the proliferation rate. These features were chosen because they directly correspond to the parameters governing the agent-based models and can be robustly inferred from the spatial configuration of a cell culture. This section details the computational implementation used to empirically extract these features, thereby establishing the ground truths required for downstream inference.

3.4.1 Radius feature from instance masks

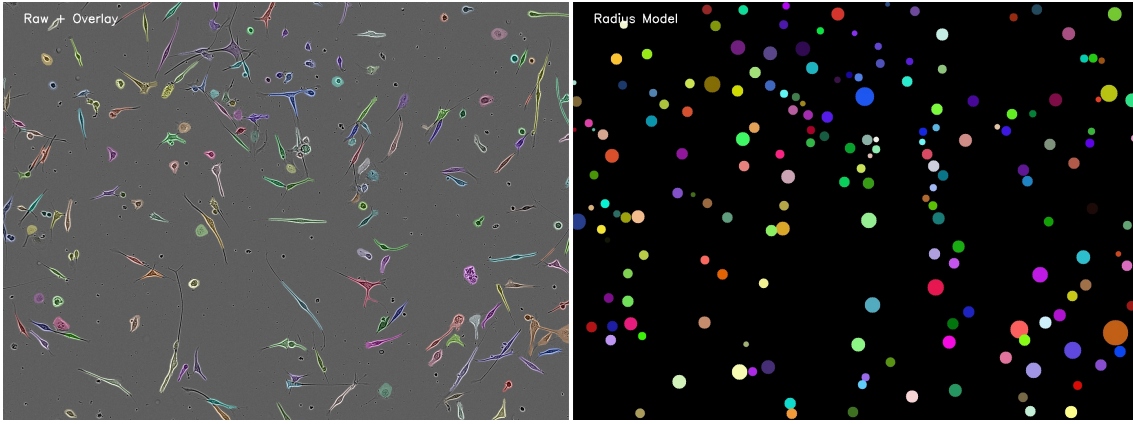
Let $\Omega \subset \mathbb{Z}^2$ denote the pixel grid of a single image frame, and let $M_k \subset \Omega$ be the set of pixels belonging to the k -th segmented cell instance. The pixel area of the instance is simply the cardinality $A_k^{\text{px}} = |M_k|$. Assuming an approximately circular morphology, an equivalent geometric radius is defined as the radius of a circle with the identical area

$$r_k^{\text{px}} = \sqrt{\frac{A_k^{\text{px}}}{\pi}}.$$

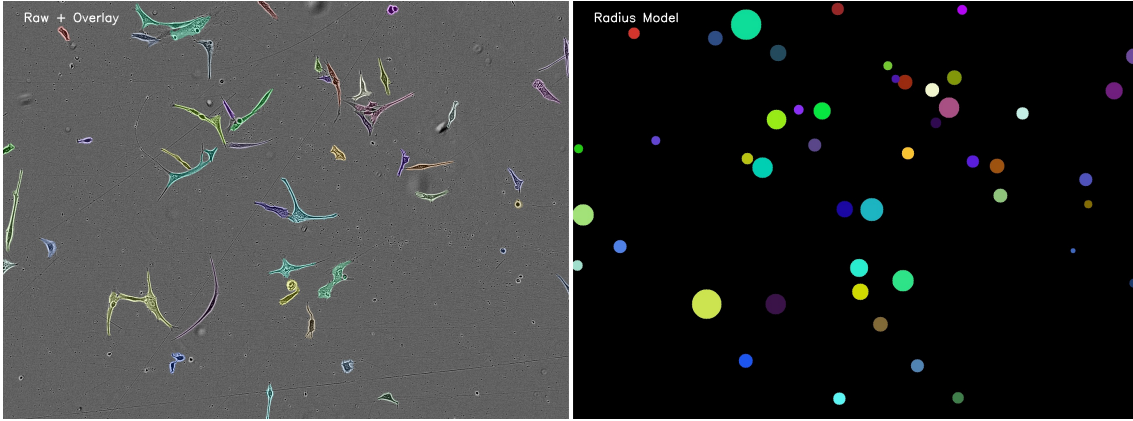
To convert this measurement into physical units, let $\delta > 0$ denote the microscope calibration in microns per pixel ($\mu\text{m}/\text{px}$). The area and radius in physical units are subsequently computed as

$$A_k = A_k^{\text{px}} \delta^2, \quad r_k = r_k^{\text{px}} \delta.$$

For each time point t , we compute the empirical mean radius across all detected instances, alongside its standard deviation as a measure of morphological heterogeneity. This radius feature serves a dual purpose: (i) as a core summary statistic for downstream inference, and (ii) as a temporal sanity check. Systematic upward drift in the mean radius over time is often interpreted as a signature of density-dependent segmentation errors, as un-split cell merges artificially inflate the inferred radius. Figure 3.4 visualises this transformation from pixel-level instance predictions to geometric radius estimations.



(a) Instance segmentation prediction of cell line 558 A1 at time point $t = 00\text{d } 03\text{h } 15\text{m}$ (left) and corresponding geometric cell approximations (right).



(b) Instance segmentation prediction of cell line 558 A10 at time point $t = 04\text{d } 09\text{h } 47\text{m}$ (left) and corresponding geometric cell approximations (right).

Figure 3.4: Comparison of instance segmentation prediction overlays (left) and the resulting transformed geometric radius models (right).

3.4.2 Diffusion feature via mean squared displacement

As established in Section 2.3, pure Brownian diffusion follows the classical scaling identity $\text{MSD}(\tau) = 4D\tau$. In the computational implementation, cell trajectories $\{X_n\}_{n=0}^N$ are obtained by linking segmented centroids across consecutive frames. For a discrete lag ℓ , the empirical ensemble Mean Squared Displacement is computed by averaging the squared displacements over all tracked cells

$$\widehat{\text{MSD}}(\ell\Delta) = \frac{1}{N - \ell + 1} \sum_{n=0}^{N-\ell} \|X_{n+\ell} - X_n\|_2^2 \quad (3.2)$$

where Δ is the temporal resolution of the acquisition interval.

To estimate the diffusion coefficient $\hat{D}$, we perform a linear regression ($\widehat{\text{MSD}}(\tau) \approx a\tau + b$) over an early-lag window where the curve remains empirically linear. Restricting the fit to early lags mitigates bias introduced by long-lag effects such as

spatial confinement, interaction-induced drift, track fragmentation, and cumulative measurement noise, resulting in the estimator $\hat{D} = a/4$.

3.4.3 Directional persistence via the Hurst parameter

For the velocity-driven fractional models, the Hurst parameter H encodes directional persistence. Based on the theoretical scaling law of fractional Brownian motion, the true MSD scales proportionally to τ^{2H} . Consequently, taking the logarithm yields a linear relationship

$$\log \widehat{\text{MSD}}(\ell\Delta) \approx \log C + 2H \log(\ell\Delta) \quad (3.3)$$

In practice, H is estimated by defining a lag set $\mathcal{L} = \{\ell_{\min}, \dots, \ell_{\max}\}$ over which the empirical MSD exhibits approximate power-law behaviour. We actively exclude extremely small lags to avoid localisation noise, and very large lags to avoid finite-sample variance. By fitting a linear model $y_\ell \approx a + bx_\ell$ (where $x_\ell = \log(\ell\Delta)$ and $y_\ell = \log \widehat{\text{MSD}}$) via least squares, the Hurst parameter is extracted as $\widehat{H}_{\text{MSD}} = b/2$.

3.4.3.0.1 Drift correction. If trajectories contain deterministic drift, the squared increments can artificially scale as ℓ^2 , biasing $\widehat{H}$ upwards. To mitigate this, we subtract a globally estimated drift velocity vector prior to calculating the empirical MSD.

3.4.4 Proliferation rate from cell-count time series

Utilising the exponential population growth model shown in Equation (2.3), we estimate the proliferation rate ρ directly from the segmented image sequence. The per-frame cell count $N(t)$ is obtained via connected-component counting post-segmentation, with excessively small components filtered out to suppress debris noise.

The initial population N_0 and proliferation rate ρ (measured in s^{-1}) are estimated by fitting the empirical counts to the exponential curve using nonlinear least squares regression. From this, the culture’s doubling time can be derived as $T_d = \ln(2)/\rho$.

It is important to emphasise that this estimator measures an effective net growth rate of detected instances. Because segmentation errors can artificially skew counts—for example, merges falsely decrease N , while over-segmentation splits falsely increase N , the proliferation estimate must be interpreted in conjunction with the radius-based sanity checks to ensure reliability.

3.5 Temporal consistency of derived quantities

While spatial overlap metrics evaluate per-frame agreement with manual annotations, they do not fully characterize temporal failure modes. A segmentation model may achieve high average spatial overlap while occasionally merging adjacent cells

in highly confluent frames. Such errors, while rare, severely bias downstream time-series statistics. Therefore, prior to deploying the data for parameter inference, two sanity checks are performed to detect density-dependent tracking failures.

3.5.1 Radius stability under increasing density

Biological cells do not undergo continuous, unbounded growth in radius over the modeled time scales; size remains approximately stable, with variability driven strictly by morphology and division events. Consequently, the inferred empirical mean radius time series $\bar{r}(t)$ should not exhibit a systematic upward drift. If such drift is detected, it is a strong quantitative indicator of under-segmentation, as un-split cell merges artificially inflate the predicted mask area and the resulting equivalent radius.

3.5.2 Consistency of mean squared displacement

If segmentation and centroid tracking are temporally stable, the empirical MSD should closely follow the fitted theoretical scaling law over the designated early-lag regime. Large deviations or sudden collapses in the empirical MSD curve indicate severe track fragmentation or identity switches caused by transient segmentation failures.

3.6 Simulation models and numerical algorithms

To execute the inference pipeline, forward simulations were performed utilizing both the baseline overdamped Langevin dynamics and the proposed fractional velocity-driven models. This section outlines the specific numerical schemes and algorithms utilized to computationally evolve the cell trajectories.

3.6.1 Numerical integration via the Euler-Maruyama scheme

The baseline positional model is governed by a system of interacting stochastic differential equations. To simulate these continuous-time dynamics on a discrete computational grid, we employ the Euler-Maruyama numerical scheme.

Let Δt denote the fixed discrete time step, and $t_n = n\Delta t$. The position of cell i at the next time step, $X_i(t_{n+1})$, is updated according to the discretized equation

$$X_i(t_{n+1}) = X_i(t_n) + F_i(X(t_n))\Delta t + \sqrt{2D\Delta t} Z_{i,n} \quad (3.4)$$

where the drift term $F_i(X(t_n))$ aggregates all deterministic intercellular interaction forces acting on cell i at time t_n . The stochastic driving force is simulated via $Z_{i,n}$, which is a standard two-dimensional Gaussian random vector, $Z_{i,n} \sim \mathcal{N}(0, I)$, drawn independently for each cell at each time step. This scheme ensures that the variance of the simulated Brownian increments correctly scales with Δt .

3.6.2 Algorithmic implementation of edge forces

The theoretical formulation of the step-aware boundary edge forces is derived in Chapter 2. In the computational implementation, this domain confinement logic is evaluated synchronously with the Euler-Maruyama update.

At each time step t_n , prior to calculating the displacement, the orthogonal distance between each cell centroid and the four domain boundaries is computed. If a cell enters the designated boundary band, a dynamic repulsive force is calculated. Because this force magnitude is dynamically scaled by the current time step Δt and local cell density, it is safely superimposed onto the intercellular drift vector $F_i(X(t_n))$. This explicit algorithmic integration ensures strict confinement within the simulation bounding box without the need for non-smooth, post-hoc coordinate reflection logic.

3.6.3 Generation of fractional Gaussian noise via the Davies-Harte method

For the velocity-driven models, generating trajectories requires sampling from Fractional Brownian Motion (fBm). Because the increments of fBm—known as fractional Gaussian noise (fGn)—are highly correlated, standard independent Gaussian sampling (as used in the Euler-Maruyama scheme) is invalid. Instead, we implement the Davies-Harte algorithm, which leverages the properties of circulant matrices and the Fast Fourier Transform (FFT) to exactly and efficiently sample these correlated increments in $\mathcal{O}(N \log N)$ time.

The theoretical autocovariance sequence of fGn for a given Hurst parameter H and time step Δt is defined as $\gamma(k)$. Because the covariance matrix of a stationary sequence is Toeplitz, it can be embedded into a larger $2N \times 2N$ circulant matrix. The algorithm proceeds computationally as follows:

By integrating the increments generated by Algorithm 6, we obtain the exact discrete-time path of the fractional velocity, which is subsequently integrated via the Euler scheme to update the cell coordinates.

3.6.3.0.1 Implementation note (NumPy FFT scaling). Many numerical libraries (e.g. NumPy) use a non-unitary FFT convention: IFFT includes the factor $1/M$. In that convention, the unitary inverse transform F^* equals $\sqrt{M}$ IFFT, and (2.20) becomes

$$\mathbf{Y} = \sqrt{M} \text{IFFT}(\sqrt{\lambda} \odot \mathbf{Z}).$$

This is the implemented form.

Algorithm 4 Davies-Harte Method for fGn Generation

Require: Length N , Hurst parameter H , variance σ^2 , time step Δt **Ensure:** A sequence of N exact fractional Gaussian noise increments

- 1: Compute the autocovariance sequence $\gamma(k)$ for $k = 0, 1, \dots, N$
- 2: Construct the first row of the circulant embedding matrix C :

$$c = [\gamma(0), \gamma(1), \dots, \gamma(N-1), \gamma(N), \gamma(N-1), \dots, \gamma(1)]$$

- 3: Compute the discrete Fourier transform of the sequence: $\Lambda = \text{FFT}(c)$
 - 4: **if** any $\Lambda_k < 0$ **then**
 - 5: **return** Error: Circulant matrix is not positive definite (requires larger padding)
 - 6: **end if**
 - 7: Generate two independent standard Gaussian sequences $Z^{(1)}, Z^{(2)} \sim \mathcal{N}(0, 1)$ of length N
 - 8: Construct the complex Gaussian vector V utilizing Λ , $Z^{(1)}$, and $Z^{(2)}$
 - 9: Compute the inverse Fourier transform: $W = \text{IFFT}(V)$
 - 10: **return** The first N elements of W , scaled by $\sigma \Delta t^H$
-

3.7 Summary statistics for Approximate Bayesian Computation

To map the high-dimensional spatial configuration of a cell culture into a low-dimensional representation suitable for ABC, we evaluate two distinct classes of summary statistics:

- **Baseline Summaries:** A combination of the endpoint cell count and the Pair Correlation Function (PCF), directly motivated by the prior work of Rosén et al. [44].
- **Topological Summaries:** Multi-scale geometric features extracted using Topological Data Analysis (TDA).

3.7.1 Baseline endpoint summaries and distance

As introduced in Chapter 1, the baseline framework utilizes two complementary endpoint summaries to capture the system dynamics: the final population size and the spatial arrangement. While the conceptual motivation is established early on, this section details their discrete computational implementation.

The normalized count discrepancy E_N evaluates the net proliferation error. Let N_e denote the number of cells in the experimental endpoint image, and let N_s denote the number of cells in the simulated endpoint configuration. The error is computed as

$$E_N = \frac{N_e - N_s}{1 + N_e} \quad (3.5)$$

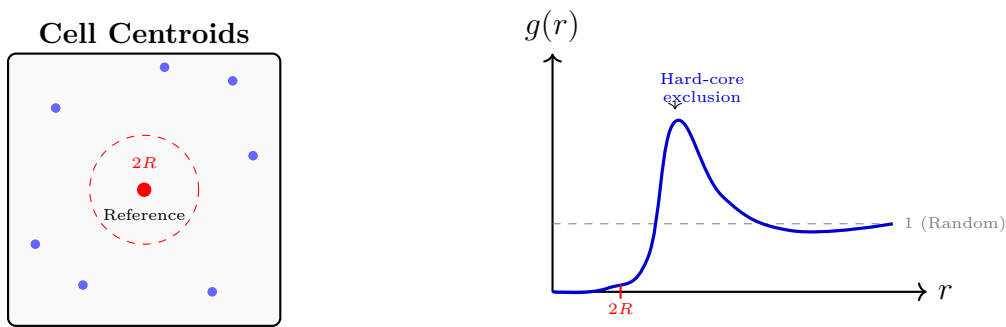


Figure 3.5: Conceptual illustration of the Pair Correlation Function ($g(r)$). **Left:** A simulated centroid pattern showing a reference cell and its steric exclusion zone ($2R$). **Right:** The corresponding continuous $g(r)$ curve, capturing the hard-core exclusion limit at small radii, followed by a local clustering peak before decaying to spatial randomness.

To evaluate spatial clustering computationally, the continuous Pair Correlation Function which captures the relative density of cells as a function of radial distance (Figure 3.5) is discretized into K radial bins.

This yields the numerical arrays $\text{PCF}_e(i)$ and $\text{PCF}_s(i)$ for $i = 1, \dots, K$. The discrepancy is quantified as

$$E_{\text{PCF}} = \frac{1}{K} \sum_{i=1}^K \left(\frac{\text{PCF}_s(i) - \text{PCF}_e(i)}{1 + \text{PCF}_e(i)} \right)^2 \quad (3.6)$$

The denominator $1 + \text{PCF}_e(i)$ is explicitly included to stabilize the contribution of array bins with near-zero densities. During the execution of the ABC, the acceptance kernel requires a single scalar distance. We aggregate these discrepancies using a combined Euclidean metric

$$E_{\text{Baseline}} = \sqrt{E_N^2 + E_{\text{PCF}}^2} \quad (3.7)$$

3.7.2 Advanced topological extraction and hybrid summaries

While localized statistics like the PCF capture immediate cellular neighborhoods, they often fail to encapsulate the macroscopic spatial architecture of the culture. To capture higher-order geometries, the ABC framework was augmented to utilize Topological Data Analysis (TDA).

Upon the completion of a forward stochastic simulation, the continuous spatial coordinates of the living cell centroids are processed using the `gudhi` computational geometry library [50]. Rather than relying on standard proximity-based complexes (such as the Vietoris-Rips complex) which can generate artificial higher-dimensional cliques, the algorithm constructs the exact Delaunay triangulation of the point cloud and filters it to generate the Alpha Complex. This mathematically guarantees that the extracted persistent homology strictly represents physical, non-intersecting ge-

ometries. The birth and death scales of the connected components (H_0) and spatial voids (H_1) are extracted as raw persistence diagrams.

To bypass the algorithmic instability inherent to discrete feature counting such as traditional Betti curves, which act as rigid step-functions and heavily penalize minor stochastic shifts in cell placement during ABC distance evaluation, these diagrams are instead mapped into continuous functional spaces. Using the `persim` topological package [46], we construct normalized Persistence Images. The raw coordinates are linearly transformed into birth-persistence space, where a weighted two-dimensional Gaussian blur is applied. The variance of these distributions inherently smooths out the aforementioned stochastic spatial noise, while a quadratic weight function filters transient topological artifacts caused by temporary cellular overlap.

The continuous surface is uniformly discretized to reduce dimensionality. The H_0 components are integrated into a 1×20 grid (tracking cluster formations), and the H_1 voids are integrated over a 10×10 grid. Crucially, to decouple geometric structure from cell density, these Persistence Images are normalized by their total sum, transforming them into spatial probability distributions. By converting the complex topological features into these fixed-dimensional grid representations, the difference between any two H_0 or H_1 images can be directly and efficiently computed pixel-by-pixel using standard vector metrics, such as the Euclidean distance.

3.7.2.1 Hybrid summary framework

Despite its stability, relying exclusively on global topology introduces parameter unidentifiability. Because TDA is scale-invariant, a highly diffusive region of small cells presents a topological pattern indistinguishable from a stationary region of large cells.

To break this parameter degeneracy, we implement a Hybrid Summary Statistic framework. During endpoint extraction, the algorithm concurrently calculates the absolute PCF and the normalized Persistence Images. The absolute PCF evaluates localized repulsions, perfectly constraining the morphological cell radius parameter. Simultaneously, the Persistence Images capture the global void distributions, isolating the spatial homogenization strictly driven by cellular diffusion.

These statistics are concatenated alongside the absolute cell count into a unified summary vector. Mathematically, the flattened 1×20 grid of H_0 components, the flattened 10×10 grid of H_1 voids, the K bins of the discretized PCF, and the final cell count N are combined into a single one-dimensional array

$$S_{\text{hybrid}} = [N, \text{PCF}_1, \dots, \text{PCF}_K, H_{0,1}, \dots, H_{0,20}, H_{1,1}, \dots, H_{1,100}].$$

This explicit formulation ensures that the inference algorithm simultaneously evaluates both the localized morphological constraints and the global topological patterns without losing any critical spatial information.

3.8 Data-driven prior extraction

To constrain the parameter search space to biologically relevant regimes while avoiding distributional bias, we employed a data-driven prior extraction strategy. Rather than assuming theoretical distributions which might artificially bias the inference towards a preconceived mean, we constructed uniform priors bounded strictly by the empirical extrema observed during the feature extraction phase (Section 3.4).

Let $\Theta = (r, D, \alpha, H)$ denote the parameter vector. The prior distribution $\pi(\Theta)$ is defined as a product of uniform distributions

$$\pi(\Theta) = \mathcal{U}(r_{\min}, r_{\max}) \times \mathcal{U}(\log D_{\min}, \log D_{\max}) \times \mathcal{U}(\log \alpha_{\min}, \log \alpha_{\max}) \times \mathcal{U}(H_{\min}, H_{\max}). \quad (3.8)$$

To mitigate the risk of excluding valid extreme values, the empirically observed lower and upper bounds were artificially expanded by a fixed percentage margin prior to inference.

3.9 Approximate Bayesian Computation implementation

With the prior distributions established and the numerical simulators configured, the ABC algorithms were deployed. To evaluate performance, both standard Rejection ABC and Regression-Adjusted ABC variants were implemented.

To ensure the robustness of the inference pipeline prior to application on *in vitro* data, the ABC framework was benchmarked utilizing synthetic datasets. For a given ground-truth parameter vector θ^* , a synthetic endpoint image was generated, and the ABC algorithm was deployed to approximate the posterior distribution.

To summarize each marginal posterior into a single point estimate, we calculated the marginal posterior mode. For each parameter component, a one-dimensional Kernel Density Estimate (KDE) was fitted to the accepted samples, with the mode serving as the optimal point estimate $\hat{\theta}$.

Finally, to quantify the performance of the inference pipeline across the synthetic dataset, we employ the Pearson correlation coefficient R as our primary evaluation metric. By calculating the linear correlation between the ground-truth parameters θ^* and the estimated posterior modes $\hat{\theta}$, we measure the inferential reliability of the model. A high correlation coefficient, approaching $R = 1$, indicates strong predictive fidelity, confirming that the parameter space is fully identifiable and that the chosen summary statistics successfully break parameter degeneracy.

3.10 Simulation models and numerical algorithms

To execute the inference pipeline, forward simulations were performed utilizing both the baseline overdamped Langevin dynamics and the proposed fractional velocity-driven models. This section outlines the specific numerical schemes, boundary configurations, and algorithms utilized to computationally evolve the cell trajectories.

3.10.1 State space and domain

All models in this work are considered and simulated on the same state space and domain.

First, let $\Omega \subset \mathbb{R}^2$ denote the imaging window, modelled as a rectangle

$$\Omega = [L_x, R_x] \times [B_y, T_y].$$

At time t , the system contains $N(t)$ cells with positions

$$X(t) = (X_1(t), \dots, X_N(t)), \quad X_i(t) \in \Omega.$$

Each cell is modelled as a disk of fixed radius $R > 0$ (consistent with the non-growing assumption used later for the segmentation sanity check). Cell division increases $N(t)$ through a stochastic birth mechanism described in Section 3.10.3.

3.10.2 Baseline agent-based model

This section specifies the baseline model used to generate synthetic endpoint images and trajectory-derived summary statistics for inference. The model is an interacting particle system evolving in continuous space under overdamped stochastic dynamics, with stochastic proliferation events.

3.10.2.1 Overdamped stochastic dynamics

Between proliferation events, each cell i at time t follows an overdamped Langevin stochastic differential equation (SDE),

$$dX_i(t) = F_i(X(t)) dt + \sqrt{2D} dW_i(t), \quad (3.9)$$

where $D > 0$ is the diffusion and $W_i(t)$ are independent 2D standard Brownian motions [39, 23]. The drift F_i aggregates deterministic interaction contributions,

$$F_i(X) = F_i^{\text{cell-cell}}(X) + F_i^{\text{edge}}(X).$$

In the baseline implementation, $F_i^{\text{cell-cell}}$ enforces short-range exclusion to mitigate cell overlapping, while F_i^{edge} prevents escape from the imaging window.

3.10.2.2 Cell–cell exclusion force

Cells are treated as soft disks with a repulsive force that activates when two cells overlap. This is also known as rigid interaction.

Let $r_{ij} = \|X_i - X_j\|$ denote the pairwise distance. For $r_{ij} < 2R$ we define an overlap depth $\delta_{ij} = 2R - r_{ij} > 0$ and apply a linear spring repulsion

$$F_{ij} = \kappa \delta_{ij} \frac{X_i - X_j}{r_{ij}}, \quad r_{ij} < 2R, \quad (3.10)$$

and $F_{ij} = 0$ for $r_{ij} \geq 2R$. The total cell–cell force on cell i is

$$F_i^{\text{cell-cell}}(X) = \sum_{j \neq i} F_{ij}.$$

3.10.2.3 Step-aware edge forces for bounded stochastic motion

The stochastic forcing in Eq.(3.9) implies that, under any explicit time discretisation, a cell may leave the imaging domain in a single step due to a large Brownian increment. Because the domain is an experimental field-of-view rather than a physical container, we require a numerically stable, minimally intrusive boundary treatment that prevents frequent loss of particles without introducing discontinuous "clamping" or hard reflections. We therefore implement soft confinement via an inward drift acting in a thin band adjacent to each boundary wall.

To derive a design rule, we analyse boundary crossing in a local one-dimensional coordinate. We begin by fixing a wall and defining the signed distance $x \geq 0$ measured into the domain along that wall's inward normal. For example, for the left wall $X = L_x$ we use $x = X - L_x$ and for the right wall $X = R_x$ we use $x = R_x - X$. Analogous definitions apply to the bottom and top walls.

Within a thin interior band of width $w > 0$ we apply an inward wall force. We parameterise an arbitrary band force profile by a nonnegative shape function ϕ and strength k

$$F_{\text{edge}}(x) = k \phi(x), \quad 0 \leq x \leq w, \quad \text{and} \quad F_{\text{edge}}(x) = 0 \text{ for } x > w, \quad (3.11)$$

where $\phi(x) \geq 0$ ensures the drift is inward.

Along the inward normal coordinate, the diffusion part of Eq.(3.9) reduces to $\sqrt{2D} dW(t)$, so a single Euler–Maruyama step for the signed distance is

$$x_{n+1} = x_n + F_{\text{edge}}(x_n) \Delta t + \sqrt{2D \Delta t} \xi_n, \quad \xi_n \sim \mathcal{N}(0, 1). \quad (3.12)$$

Crossing the wall in one step corresponds to the event $x_{n+1} < 0$.

To find the probability of crossing a wall, we condition on $x_n = x$. From Eq.(3.12), the random variable x_{n+1} is Gaussian with mean $x + F_{\text{edge}}(x) \Delta t$ and variance $2D \Delta t$.

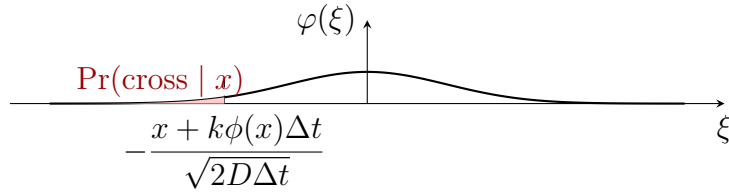


Figure 3.6: The one-step exit probability in (3.13) is a Gaussian left-tail probability induced by the Euler–Maruyama update.

Therefore,

$$\Pr(\text{cross} \mid x) = \Pr\left(\xi_n < -\frac{x + F_{\text{edge}}(x)\Delta t}{\sqrt{2D\Delta t}}\right) = \Phi\left(-\frac{x + k\phi(x)\Delta t}{\sqrt{2D\Delta t}}\right), \quad 0 \leq x \leq w, \quad (3.13)$$

where Φ is the standard normal CDF. Outside the band ($x > w$) we have $F_{\text{edge}}(x) = 0$ and hence

$$\Pr(\text{cross} \mid x) = \Phi\left(-\frac{x}{\sqrt{2D\Delta t}}\right), \quad x > w. \quad (3.14)$$

We now design (w, k) to make boundary crossing rare even in worst-case configurations. Let Δt_{max} be the largest step size used by the integrator, and fix a per-wall, per-step leak tolerance $\varepsilon \in (0, 1)$. Define the corresponding upper Gaussian quantile

$$z_\varepsilon := \Phi^{-1}(1 - \varepsilon).$$

The goal is to enforce bounds on $\Pr(\text{cross} \mid x)$ uniformly over the relevant near-wall region.

The most dangerous location inside the band is contact with the boundary ($x = 0$). Evaluating Eq.(3.13) at $x = 0$ yields

$$\Pr(\text{cross} \mid x = 0) = \Phi\left(-\frac{k\phi(0)\Delta t}{\sqrt{2D\Delta t}}\right).$$

Requiring $\Pr(\text{cross} \mid x = 0) \leq \varepsilon$ for the largest time step Δt_{max} is sufficient if

$$-\frac{k\phi(0)\Delta t_{\text{max}}}{\sqrt{2D\Delta t_{\text{max}}}} \leq \Phi^{-1}(\varepsilon) = -z_\varepsilon,$$

which is equivalent to the strength constraint

$$k \geq \frac{z_\varepsilon}{\phi(0)} \sqrt{\frac{2D}{\Delta t_{\text{max}}}}. \quad (3.15)$$

This ensures that even a particle in contact with the wall crosses with probability at most ε per step (per wall), under the largest step size used.

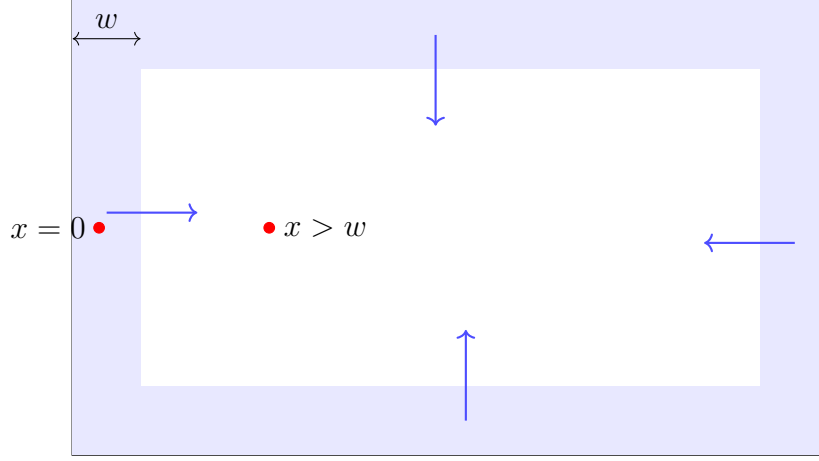


Figure 3.7: Rectangular domain with interior boundary bands (light blue). Inside a band, a constant inward drift is applied along the wall normal; outside the band, no edge force is applied.

A second failure mode is that a particle located outside the band ($x > w$) experiences no edge force, yet may "skip" across the boundary in one step. From Eq.(3.14), the crossing probability is maximised at the smallest x outside the band, i.e. at $x = w$

$$\sup_{x \geq w} \Pr(\text{cross} \mid x) = \Phi\left(-\frac{w}{\sqrt{2D\Delta t}}\right).$$

Enforcing $\Phi\left(-\frac{w}{\sqrt{2D\Delta t_{\max}}}\right) \leq \varepsilon$ gives

$$w \geq z_\varepsilon \sqrt{2D\Delta t_{\max}}. \quad (3.16)$$

Thus (3.16) prevents a particle from crossing in one step while being outside the band, up to the same leak tolerance ε .

Equations (3.15)–(3.16) are step-aware, meaning that they scale explicitly with D and $\Delta t_{\max}$. Larger diffusivity or larger time steps require wider bands and stronger inward drift to achieve the same leak probability.

In our implementation we use the constant shape $\phi(x) \equiv 1$ on $[0, w]$. Then $\phi(0) = 1$ and (3.15)–(3.16) reduce to

$$k \geq z_\varepsilon \sqrt{\frac{2D}{\Delta t_{\max}}}, \quad w \geq z_\varepsilon \sqrt{2D\Delta t_{\max}}. \quad (3.17)$$

In two dimensions the band force is applied independently per wall along inward normals, with additive contributions at corners, where two walls can be active simultaneously, which makes the single-wall bounds conservative.

The algorithm design is summarised in 5.

Algorithm 5 Step-aware sizing and application of constant edge forces in a rectangle

Require: Diffusion $D > 0$; maximum step $\Delta t_{\max}$; tolerance $\varepsilon \in (0, 1)$; domain $\Omega = [L_x, R_x] \times [B_y, T_y]$

- 1: $z_\varepsilon \leftarrow \Phi^{-1}(1 - \varepsilon)$
- 2: $w \leftarrow z_\varepsilon \sqrt{2D\Delta t_{\max}}$
- 3: $k \leftarrow z_\varepsilon \sqrt{2D/\Delta t_{\max}}$
- 4: **for** each time step n with step size $\Delta t \leq \Delta t_{\max}$ **do**
- 5: **for** each cell i **do**
- 6: Compute signed distances: x_L, x_R, x_B, x_T
- 7: $\mathbf{F}_{\text{edge}} \leftarrow (0, 0)$
- 8: **if** $0 \leq x_L \leq w$ **then**
- 9: $\mathbf{F}_{\text{edge}} \leftarrow \mathbf{F}_{\text{edge}} + (k, 0)$
- 10: **end if**
- 11: **if** $0 \leq x_R \leq w$ **then**
- 12: $\mathbf{F}_{\text{edge}} \leftarrow \mathbf{F}_{\text{edge}} + (-k, 0)$
- 13: **end if**
- 14: **if** $0 \leq x_B \leq w$ **then**
- 15: $\mathbf{F}_{\text{edge}} \leftarrow \mathbf{F}_{\text{edge}} + (0, k)$
- 16: **end if**
- 17: **if** $0 \leq x_T \leq w$ **then**
- 18: $\mathbf{F}_{\text{edge}} \leftarrow \mathbf{F}_{\text{edge}} + (0, -k)$
- 19: **end if**
- 20: Use $\mathbf{F}_{\text{edge}}$ as an additive drift term in the Euler–Maruyama update of (X_i, Y_i)
- 21: **end for**
- 22: **end for**

3.10.3 Stochastic proliferation with contact inhibition

Cell division is implemented as a stochastic birth process with state-dependent intensity. Each cell i is assigned a birth propensity

$$b_i(X) = \alpha \exp\left(-\beta S_i(X)\right), \quad (3.18)$$

where $\alpha > 0$ is the baseline birth rate and $\beta > 0$ controls the strength of inhibition. Specifically, as the local spatial crowding $S_i(X)$ around a cell increases, the exponential term reduces the probability of division. This explicitly captures the biologically relevant phenomenon of contact inhibition discussed earlier, ensuring that cell proliferation naturally slows down or halts in dense regions of the culture.

The inhibition signal $S_i(X)$ is computed from nearby neighbours

$$S_i(X) = \sum_{j \neq i: \|X_i - X_j\| \leq r_{\text{inh}}} \exp\left(-\left(\frac{\|X_i - X_j\|}{\eta R}\right)^2\right), \quad r_{\text{inh}} = 6R, \quad (3.19)$$

with $\eta > 0$ setting the length scale of inhibition. Thus, close neighbours contribute strongly to S_i and reduce b_i ; distant neighbours contribute negligibly.

The total birth intensity is

$$B(X) = \sum_{i=1}^{N(t)} b_i(X).$$

A standard continuous-time event mechanism is used, with the steps listed below.

1. Sample the waiting time to the next birth as $\Delta T \sim \text{Exp}(B(X))$.
2. Select the parent cell i with probability $b_i(X)/B(X)$.
3. Place the daughter cell at distance $2R$ from the parent in a uniformly random direction.

This produces a density-regulated proliferation mechanism compatible with high-confluence cultures, while keeping the model analytically and computationally tractable.

3.10.4 Numerical scheme: coupled diffusion and events

Between events, (3.9) is integrated using Euler–Maruyama. On a time step Δt ,

$$X_i(t + \Delta t) = X_i(t) + F_i(X(t)) \Delta t + \sqrt{2D \Delta t} \xi_i, \quad \xi_i \sim \mathcal{N}(0, I). \quad (3.20)$$

To combine event timing with discrete integration, the step size is chosen as

$$\Delta t = \min(\Delta t_{\text{sim}}, t_{\text{event}} - t),$$

so the integrator lands exactly on event times without overshooting. This ensures consistent coupling of continuous stochastic motion and discrete proliferation.

3.10.5 Implementation for computational efficiency

Large-scale inference requires repeated simulation, often 10^4 to 10^6 runs depending on the ABC strategy. For a system with N cells, a naive approach to calculating cell interactions (such as steric repulsion or contact inhibition) requires checking every cell against every other cell. This results in a computational cost that scales quadratically as $O(N^2)$, which quickly becomes a severe bottleneck for large populations. To make large-scale inference feasible, the baseline implementation relies on efficient spatial data structures. By partitioning the simulation domain into a grid, the algorithm ensures computational locality, meaning it only evaluates interactions between cells that are physically adjacent. This approach reduces the neighbour search to a sub-quadratic scaling (significantly faster than $O(N^2)$), which is strictly necessary to simulate dense, high-confluence cultures in a reasonable timeframe.

3.10.6 Neighbour search via KD-trees

Both the repulsion force (3.10) and the inhibition signal (3.19) are local: contributions vanish beyond $2R$ and $6R$, respectively. Instead of evaluating all $O(N^2)$ pairwise distances, we construct a KD-tree for the current positions and query only nearby neighbours:

- **Repulsion:** query overlapping pairs within $2R$ and accumulate forces only for those pairs.
- **Inhibition:** query neighbours within $6R$ and compute S_i using only the returned lists.

This reduces the typical cost per step from quadratic to approximately $O(N \log N + M)$, where M is the number of returned neighbour pairs.

3.10.7 Vectorised force accumulation and memory layout

Force computation is implemented in a vectorised manner over the active neighbour pairs, avoiding Python-level loops over all cell pairs. Positions are stored as contiguous arrays, and inactive entries (after removal or bookkeeping) are masked to keep array shapes stable. This layout is favourable for NumPy operations and reduces overhead in repeated updates.

3.10.8 Parallel simulation of independent replicates

Inference requires many independent simulations for different parameter draws. These runs are noticeably parallel, and are executed concurrently across CPU cores. Simulation outputs (summary statistics, endpoint configurations, and metadata) are written incrementally to disk, so that large sweeps do not require storing all results in memory simultaneously.

3.10.9 Summary

The baseline framework is an interacting particle system with: (i) overdamped stochastic motion (3.9), (ii) short-range exclusion (3.10), (iii) density-regulated proliferation (3.18)–(3.19), and (iv) step-aware edge forces (3.17) to suppress boundary escape. The model is implemented using local interaction cutoffs, KD-tree neighbour queries, and parallel execution to support the repeated simulations required by likelihood-free inference.

3.10.10 Modelling Discrete-time velocity and Davies–Harte simulation

To incorporate directional persistence, velocity-driven models are discretized on a grid $t_n = n\Delta$. Correlated fractional Gaussian noise (fGn) increments are generated computationally via the Davies–Harte method (summarized schematically

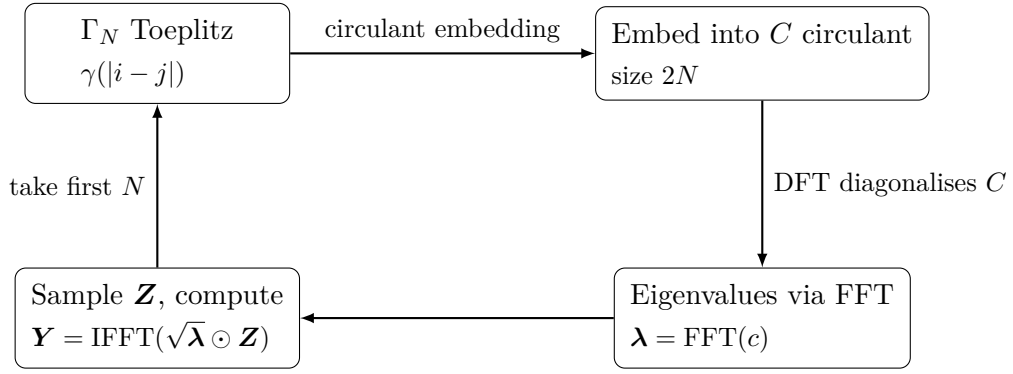


Figure 3.8: Schematic of the Davies–Harte method: embed the Toeplitz covariance in a circulant matrix, use FFT to obtain eigenvalues, sample in the Fourier domain, and invert by FFT to recover correlated increments.

in Figure 3.8 and detailed in Algorithm 6), avoiding the $O(N^3)$ cost of direct Cholesky factorisation by utilizing circulant embedding and fast Fourier transforms in $O(N \log N)$ time.

A generic Euler-type update of our velocity-driven model takes the form

$$\mathbf{V}_{i,n+1} = \mathbf{V}_{i,n} - \left[\mu \mathbf{V}_{i,n} + \frac{1}{m} \mathbf{F}_i(\mathbf{X}_{i,n}) \right] \Delta + \sigma \Delta \mathbf{W}_{i,n}^H, \quad \mathbf{X}_{i,n+1} = \mathbf{X}_{i,n} + \mathbf{V}_{i,n} \Delta, \quad (3.21)$$

where $\{\Delta \mathbf{W}_{i,n}^H\}$ are vector-valued fGn increments with covariance prescribed by Eq.(2.14). When $H > 1/2$, successive increments are positively correlated, which induces directional persistence in velocity and, consequently, smoother trajectories than purely positional Brownian forcing. This mechanism is precisely the modelling feature sought in the present thesis when extending baseline positional dynamics to velocity-driven dynamics.

Algorithm 6 Davies–Harte simulation of fGn and fBm on an equidistant grid

Require: Hurst parameter $H \in (0, 1)$; volatility $\sigma > 0$; step size $\Delta > 0$; number of increments N .

Ensure: fGn increments $(G_0, \dots, G_{N-1})$ and fBm path $(B_0, \dots, B_N)$ at times $t_n = n\Delta$.

- 1: Compute the autocovariance $\gamma(k)$ for $k = 0, \dots, N$ using (2.14).
- 2: Form the length- $2N$ embedding vector $(c_0, \dots, c_{2N-1})$ by

$$c_0 = \gamma(0), \quad c_k = \gamma(k) \quad (1 \leq k \leq N), \quad c_{2N-k} = \gamma(k) \quad (1 \leq k \leq N-1).$$

- 3: Compute eigenvalues $\lambda_j = \text{FFT}(c)_j$ for $j = 0, \dots, 2N-1$.
- 4: **(Check)** If any $\lambda_j < 0$ (numerical negativity), set $\lambda_j \leftarrow \max(\lambda_j, 0)$ or increase embedding size.
- 5: Sample independent Gaussians:

$$Z_0, Z_N \sim \mathcal{N}(0, 1), \quad Z_j^{(1)}, Z_j^{(2)} \sim \mathcal{N}(0, 1) \text{ for } j = 1, \dots, N-1.$$

- 6: Build a conjugate-symmetric complex vector $\mathbf{Z} \in \mathbb{C}^{2N}$ via

$$Z_j := \frac{Z_j^{(1)} + iZ_j^{(2)}}{\sqrt{2}}, \quad Z_{2N-j} := \overline{Z_j}, \quad j = 1, \dots, N-1,$$

with real Z_0, Z_N .

- 7: Compute $\mathbf{Y} \leftarrow \text{IFFT}(\sqrt{\lambda} \odot \mathbf{Z})$ and take $G_n \leftarrow \Re(Y_n)$ for $n = 0, \dots, N-1$.
 - 8: Form fBm by cumulative sum $B_0 \leftarrow 0$ and $B_{n+1} \leftarrow B_n + G_n$ for $n = 0, \dots, N-1$.
-

4

Results

This chapter outlines the empirical results for each stage of the computational pipeline, progressing sequentially from image segmentation and trajectory tracking to feature extraction and Approximate Bayesian Computation (ABC) inference.

4.1 Segmentation results

4.1.1 Quantitative comparison

Figure 4.1 summarises the distribution of Dice Similarity Coefficient (DSC) scores across the evaluated segmentation configurations. Table 4.1 reports the corresponding statistical mean and standard deviation for both the DSC and the Intersection over Union (IoU) metrics on the test set.

The CNN encoder–decoder architectures systematically outperformed the other evaluated methods. The U–Net equipped with a ResNet34 encoder achieved the highest mean DSC (0.8282) and IoU (0.7079), while also showing the lowest variance across heterogeneous test frames. In contrast, the transformer-based SegFormer models achieved competitive, yet systematically lower, overlap scores. Finally, the classical Fogbank baseline yielded the lowest performance metrics (DSC: 0.7042, IoU: 0.5452) alongside the highest variance. Based on the joint criteria of high median overlap and low sensitivity to outliers, the U–Net (ResNet34) configuration was deemed as the best performer and selected as the default segmentation method for all downstream inference.

4.1.2 Density-Dependent Performance and Boundary Preservation

Because downstream trajectory tracking is highly sensitive to object merging, we motivated a deeper analysis into the variance observed in the aggregate metrics (Table 4.1). When analyzing the test set by cellular confluence, a distinct divergence in model performance emerged. In sparse cultures, where cells are physically isolated, all learning-based models—and even the classical Fogbank method—achieved high overlap scores. However, as the cultures reached high density, the performance

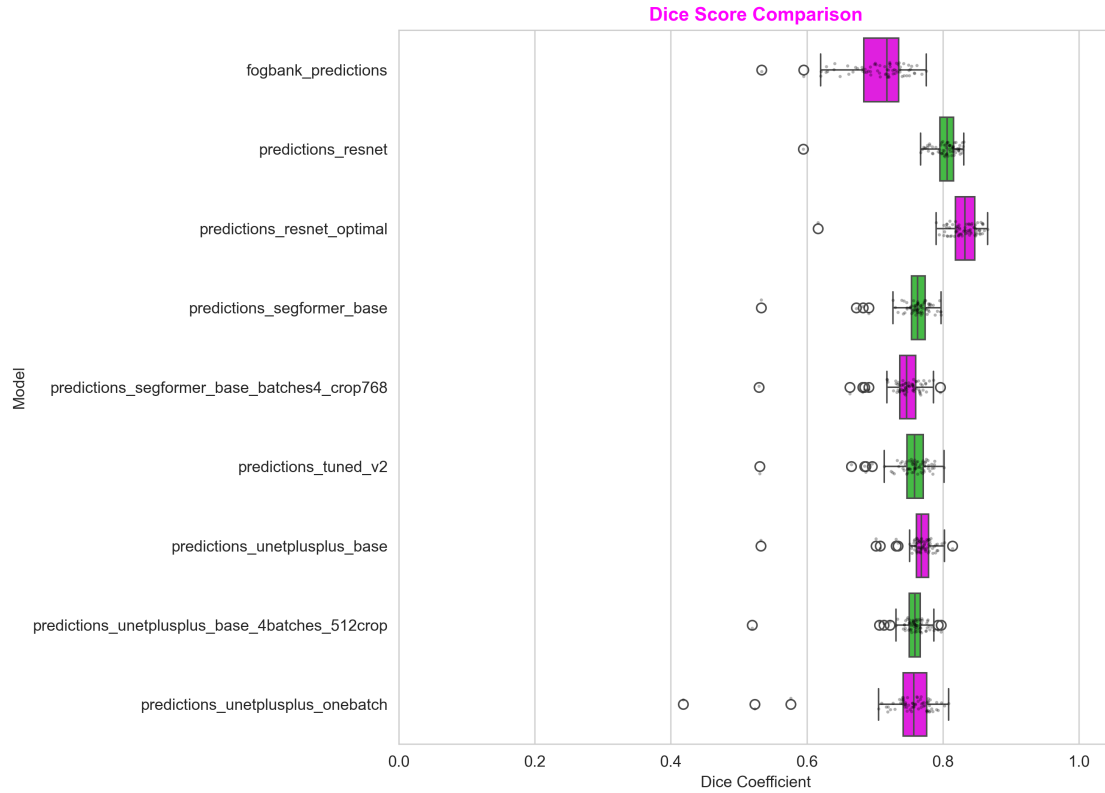


Figure 4.1: Distribution of Dice scores across evaluated segmentation configurations. Each row corresponds to one trained model or baseline method (as encoded by experiment identifiers). The selected model is the one that provides the best trade-off between median overlap and robustness to outliers in dense regimes.

diverged significantly.

The high standard deviation recorded for the classical Fogbank baseline (DSC Std: 0.0450, IoU Std: 0.0517) is directly attributable to its catastrophic failure in dense regimes. Relying strictly on pixel intensity gradients and morphological operations, it was unable to distinguish between the shared boundary of two touching cells and their internal cytoplasm. This resulted in massive under-segmentation, merging multiple cells into single, artificially large super-structures (as visually corroborated by the dense cluster mismatches in Figure 4.2a).

Among the deep learning models, the transformer-based SegFormer occasionally struggled with boundary contrast. While it successfully identified the global presence of cellular mass, its predictions at the immediate cell-cell interfaces were often blurred or softly thresholded, leading to "bridging" artifacts. In contrast, the U-Net (ResNet34) architecture consistently maintained hard, distinct separations between touching cells. Its pixel-level precision at these dense interfaces was the primary driver of its superior overall Jaccard Index and its lower variance, proving its statistical robustness in the highly confluent conditions necessary for accurate spatial modeling.

Table 4.1: Segmentation performance on the test set (sorted by mean Dice).

Model	Dice Similarity (DSC)		Intersection over Union (IoU)	
	Mean	Std	Mean	Std
U-Net (ResNet34) Optimal	0.8282	0.0315	0.7079	0.0415
U-Net (ResNet34) Base	0.8017	0.0299	0.6700	0.0375
U-Net++ (Base)	0.7656	0.0332	0.6212	0.0387
SegFormer (Base)	0.7580	0.0353	0.6114	0.0414
U-Net++ (Crops)	0.7554	0.0325	0.6079	0.0371
SegFormer (Tuned v2)	0.7524	0.0363	0.6043	0.0426
U-Net++ (One Batch)	0.7478	0.0574	0.6000	0.0629
SegFormer (Crops)	0.7437	0.0344	0.5931	0.0399
<i>Fogbank (Traditional)</i>	<i>0.7042</i>	<i>0.0450</i>	<i>0.5452</i>	<i>0.0517</i>

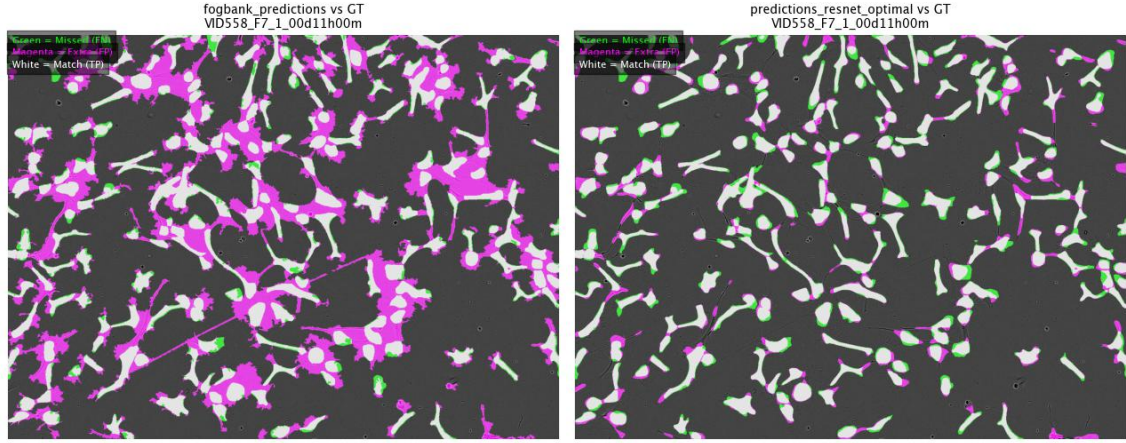
Mean and standard deviation are computed across test images. Dice is an overlap metric; IoU is stricter and therefore numerically smaller for the same predictions.

4.1.3 Qualitative comparison

To contextualize the scalar overlap metrics, qualitative visual inspections were performed using self-developed software. Figure 4.2 illustrates the segmentation mapping on a highly confluent cellular distribution with a multitude of cell morphologies (VID559_F7_1_00d11h00m). The overlay explicitly maps True Positives (white), False Positives (magenta), and False Negatives (green). The comparison visually confirms the discrepancy captured in the metrics: the learning-based U-Net architecture significantly reduces the boundary mismatches and spurious background activations present in the classical Fogbank prediction.

4.1.4 Temporal radius stability and MSD consistency

Prior to parameter inference, the segmentation and tracking outputs were subjected to temporal consistency checks on a representative cell line (A10). Figure 4.3 displays the temporal evolution of the empirical mean cell radius. The radius remains bounded and stable, exhibiting no systematic upward drift over the 5-day observation period. Concurrently, Figure 4.4 plots the empirical Mean Squared Displacement (MSD) against the theoretical diffusion trend, demonstrating tight agreement over the early-lag fitting regime.



(a) Mismatch results for fogbank prediction on VID559_F7_1_00d11h00m. (b) Mismatch results for u-net resnet prediction on VID559_F7_1_00d11h00m.

Figure 4.2: Example of result from mismatch evaluation software, showing False Negatives (green), False Positives (magenta) and True Positives (white).

Table 4.2: Global Comparison of Tracking Strategies ($N = 96$ Replicates)

Metric	Standard	Adaptive	Velocity
Search Radius ($L_{\max}$)	20 px	50 px	≈ 50 px (Predicted)
Total Particles Detected	6,267.5	1,816.2	8,483.4
Valid Tracks (> 10 frames)	612.5	756.5	453.6
Mean Track Duration	26.5 fr	42.6 fr	22.4 fr

4.1.5 Evaluation of Tracking Strategies

The comparative analysis of the three tracking methodologies (Standard, Adaptive, and Velocity-Predictive) across all independent experiments revealed distinct quantitative trade-offs regarding data recovery and track continuity. The aggregated performance metrics are summarized in Table 4.2.

The Velocity-Predictive method recovered the highest volume of data, identifying an average of 8,483.4 unique particles per experiment. The Standard method ($L_{\max} = 20$) recovered significantly fewer, averaging 6,267.5 particles, while exhibiting a mean track duration of 26.5 frames. The Adaptive strategy ($L_{\max} = 50$) experienced severe drop-out, recovering only 1,816.2 particles on average. However, for the small subset of isolated cells it successfully tracked, the Adaptive method yielded the longest mean track duration (42.6 frames). Finally, anomalous diffusion analysis performed on the trajectories recovered by the Velocity method yielded an exponent of $\alpha \approx 1.25$.

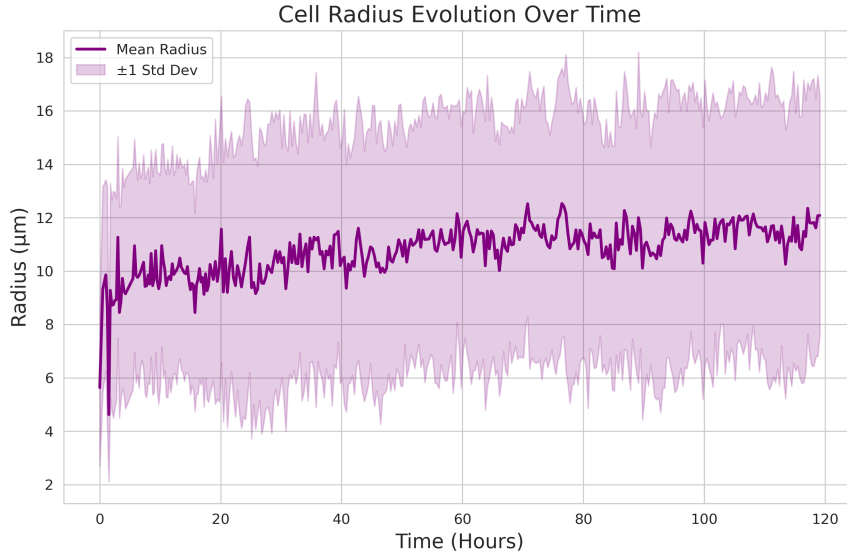


Figure 4.3: Sanity check on A10: radius evolution over time. A stable mean radius without systematic drift supports the assumption that the segmentation model does not progressively merge cells at high density. Details of the radius estimator are provided in the later cell-dynamics section.

4.2 Feature extraction & Characterization of the Prior Search Space

The empirical feature extraction defined the physical search space for the simulation parameters. The bounds for the Uniform prior distributions, derived from the in-vitro dataset, are shown in Figure 4.5 and summarized in Table 4.3.

The cell radius (r) was bounded between $4.72 \mu\text{m}$ and $25.82 \mu\text{m}$. The proliferation rates spanned $\log_{10} \alpha \in [-6.93, -5.20]$, corresponding to highly extended doubling times. The diffusion coefficient (D) exhibited the widest variance, spanning a full order of magnitude from $\log_{10}(D) = -2.07$ to -1.07 . For the fractional velocity model, the Hurst parameter (H) distribution was explicitly bounded between 0.47 and 0.83, with the vast majority of the empirical distribution mass located above the $H = 0.5$ threshold.

4.3 Inference results

4.3.1 Performance on Synthetic Data

The inference pipeline was initially benchmarked in a controlled environment using synthetic datasets generated from the baseline Langevin model. Figure 4.6 visualizes the parameter recovery for the three ABC variants. The baseline Rejection ABC approach exhibited noticeable dispersion around the identity line. Regression-Adjusted

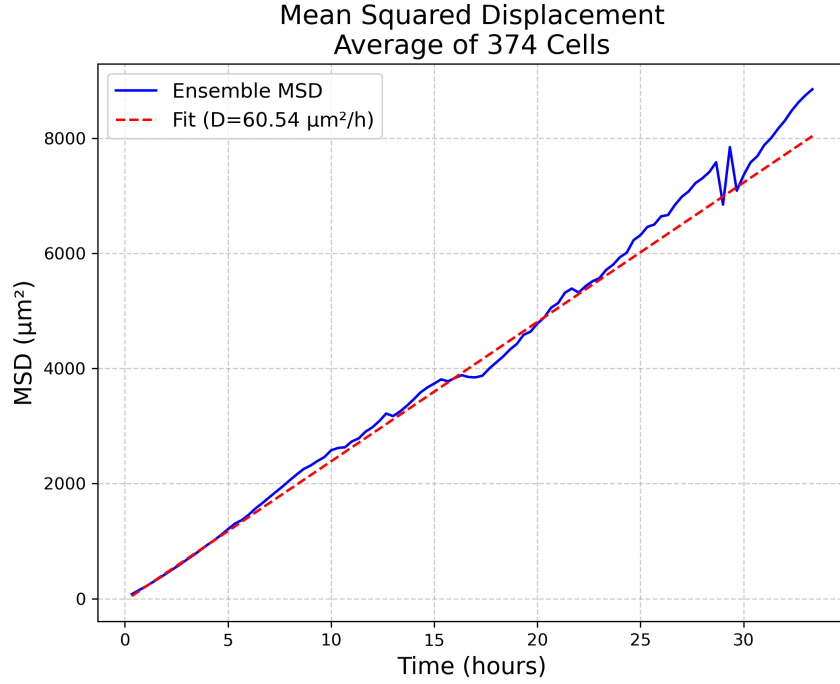


Figure 4.4: Sanity check on A10: mean squared displacement (MSD) curve and fitted diffusion trend over the fitting regime. Agreement supports that segmentation and tracking yield stable displacement statistics. The diffusion estimator is described in the later cell-dynamics methodology section.

ABC tightened this variance, while the Sequential Monte Carlo (SMC-ABC) implementation demonstrated the highest global fidelity in recovering the ground-truth parameters.

4.3.2 Parameter Inference on Experimental Data

Table 4.4 details the Pearson Correlation Coefficients (R) achieved when the inference pipeline was deployed on the true *in vitro* HGCC datasets. Using the baseline Langevin model, Rejection ABC yielded correlations of 0.81 (α), 0.48 (D), and 0.69 (r). Regression-Adjusted ABC elevated these to 0.82, 0.66, and 0.80, respec-

Table 4.3: Data-Driven Bounds for Simulation Parameters & Characterization

Parameter	Description	Lower Bound	Upper Bound
R (μm)	Cell Radius	4.72	25.82
$\log_{10}(D)$	Diffusion Coeff. ($\mu m^2/s$)	-2.07	-1.07
$\log_{10}(\alpha)$	Proliferation Rate (s^{-1})	-6.93	-5.20
H	Hurst Parameter (Persistence)	0.47	0.83

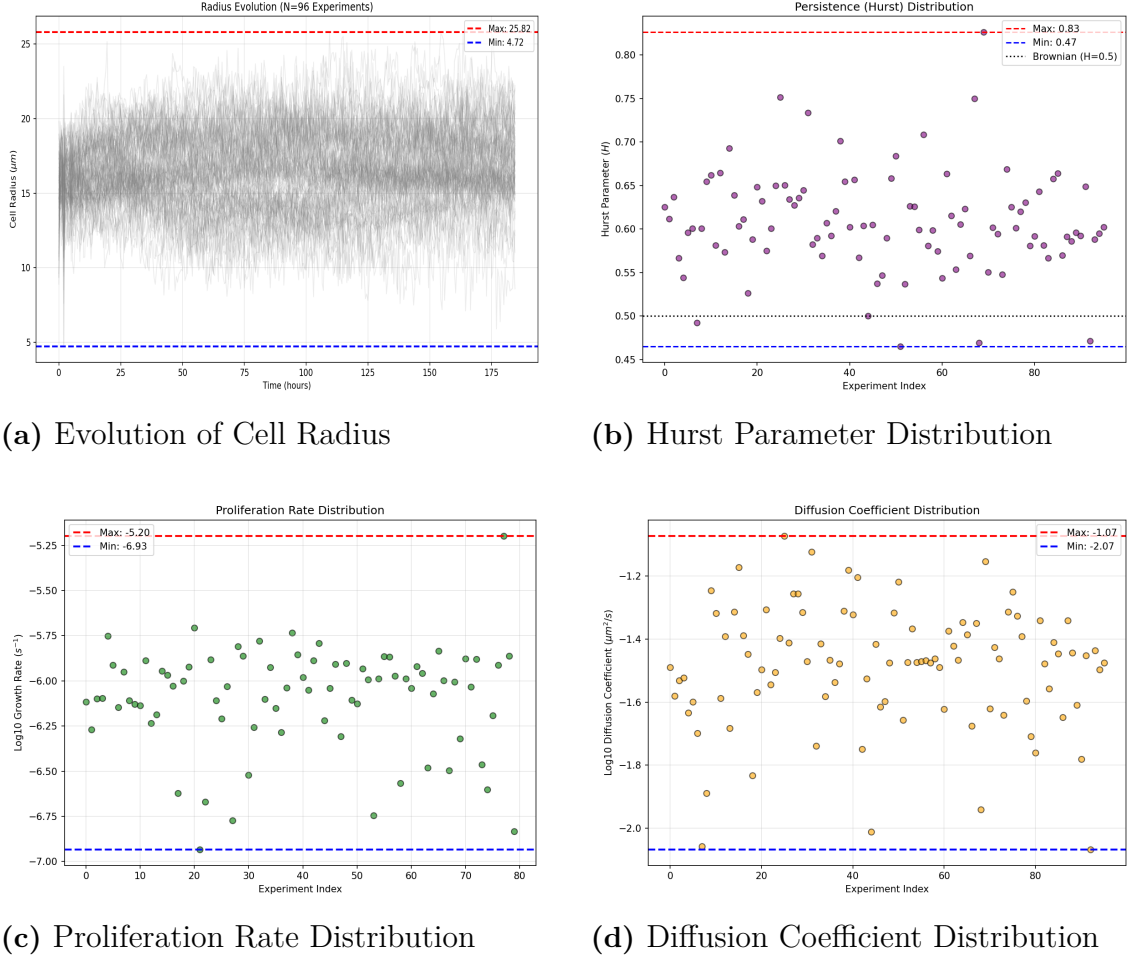
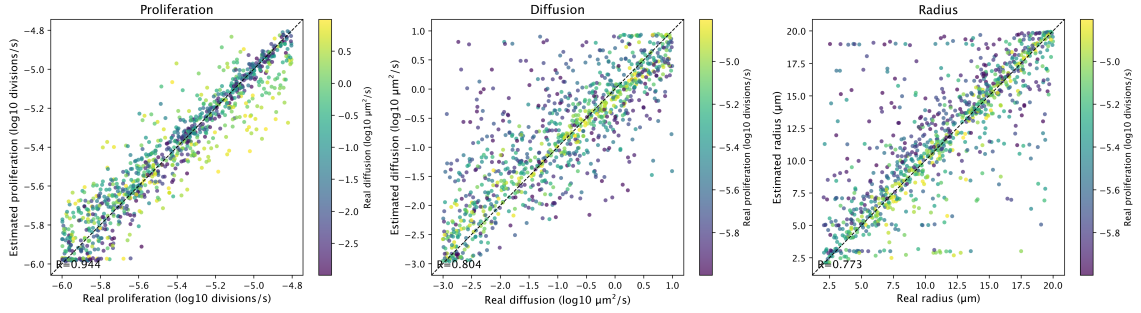
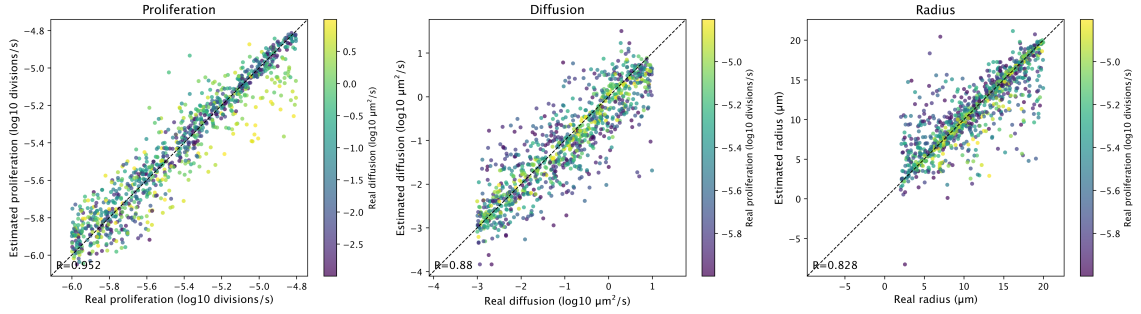


Figure 4.5: Data-Driven Definition of Simulation Priors. (a) Time evolution of the mean cell radius, showing stability over 4 days. (b) Distribution of growth rates ($\log_{10} \alpha_{birth}$), indicating a slow-cycling population. (c) Distribution of Diffusion Coefficients ($\log_{10} D$), spanning an order of magnitude. (d) Distribution of the Hurst Parameter (H). The dashed black line at $H = 0.5$ represents random Brownian motion; the observed shift to higher values ($H \in [0.47, 0.83]$) indicates super-diffusive persistence.

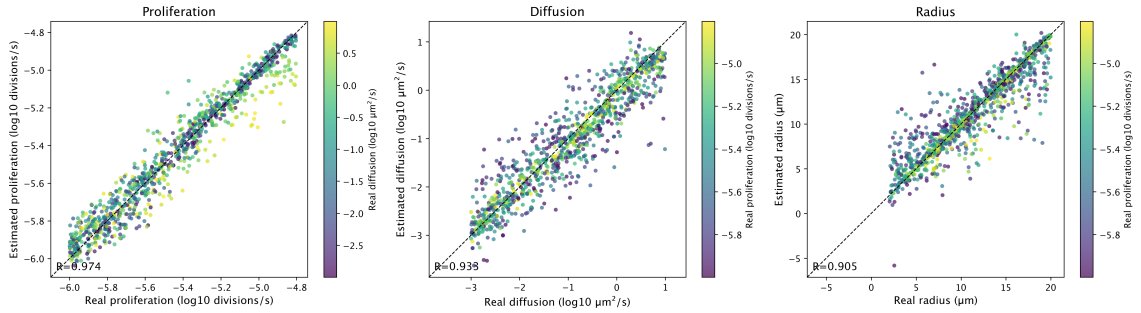
4. Results



(a) Rejection



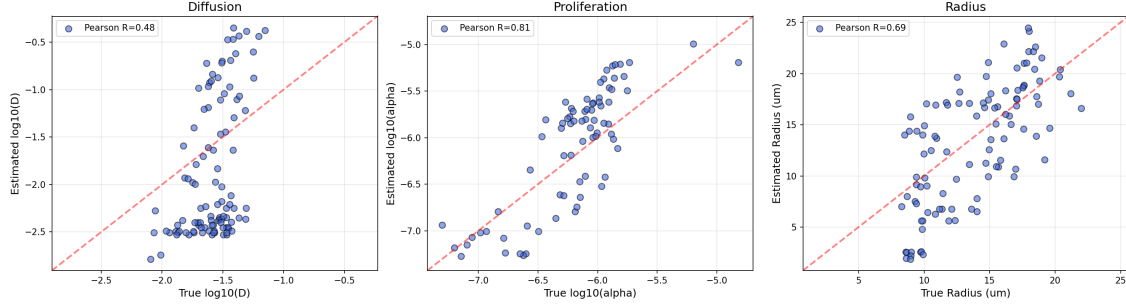
(b) Regression



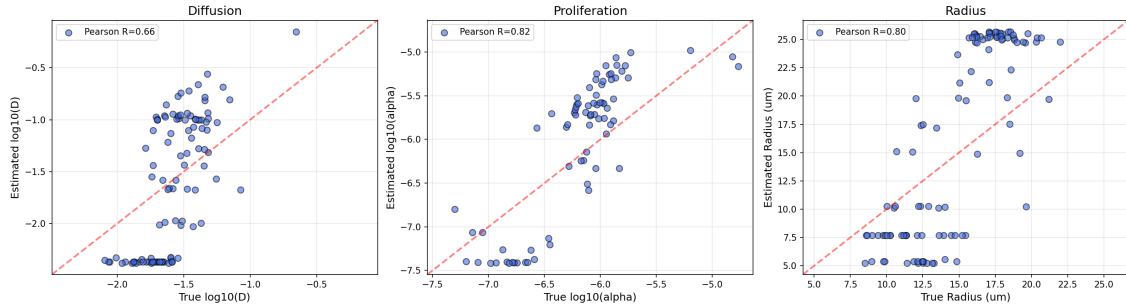
(c) SMC

Figure 4.6: Scatter plot of estimated parameter values against true values in the collection $\theta = (\log_{10} \alpha, \log_{10} D, r)$ generated by the synthetic data set. Visual tightening around the identity line demonstrates the correction of tolerance-induced bias.

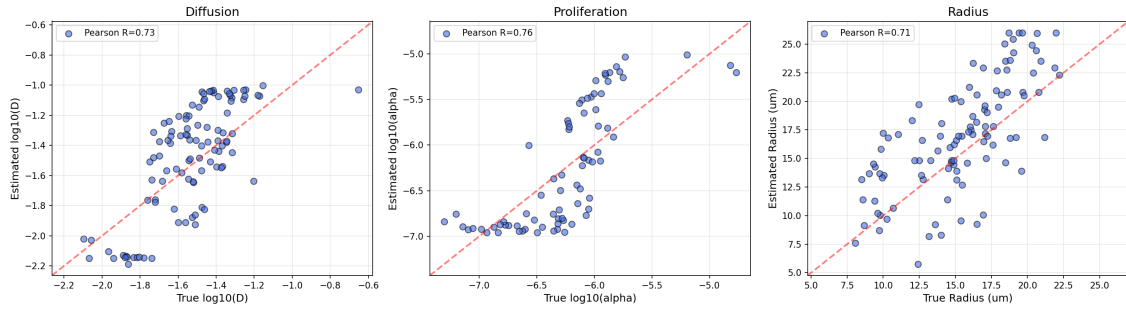
tively. SMC-ABC achieved highly balanced recovery (0.76, 0.73, and 0.71). Figure 4.7 confirms that advanced Monte Carlo adjustments successfully resolved the severe estimation difficulties observed for the diffusion parameter in the Rejection baseline.



(a) Rejection



(b) Regression



(c) SMC

Figure 4.7: Scatter plot of estimated parameter values against true values in the collection $\theta = (\log_{10} \alpha, \log_{10} D, r)$ for the experimental datasets using the Langevin model.

4.3.3 Inference Incorporating Fractional Brownian Motion

When the pipeline was expanded to evaluate the fBm model on the in-vitro data, Rejection ABC recovered the Hurst parameter (H) with a correlation of 0.53, while marginally improving the diffusion recovery to 0.72 (Table 4.4). However, applying Regression-Adjusted ABC to the fBm model caused a severe degradation in motility inference with R correlation coefficients of 0.53, 0.07 for diffusion D and the

Table 4.4: Pearson Correlation Coefficients (R) for parameter inference across models and ABC methods on in-vitro data.

Method	Proliferation (α)	Diffusion (D)	Radius (r)	Hurst (H)
<i>Model: Positional Langevin SDE</i>				
Rejection ABC	0.81	0.48	0.69	–
Regression-Adjusted ABC	0.82	0.66	0.80	–
SMC-ABC	0.76	0.73	0.71	–
<i>Model: Fractional Brownian Motion (fBm)</i>				
Rejection ABC	0.79	0.72	0.62	0.53
Regression-Adjusted ABC	0.73	0.53	0.84	0.07
SMC-ABC	0.67	0.77	0.53	0.77

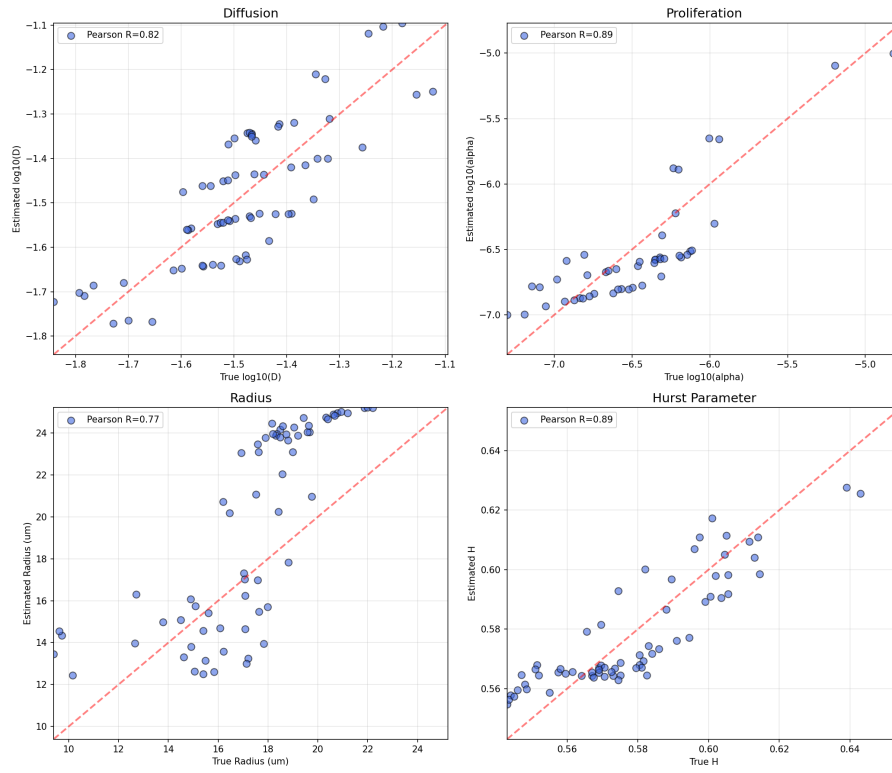
hurst parameter H respectively. SMC-ABC successfully stabilized the fBm inference, achieving correlations of 0.67, 0.77, 0.53, 0.77 for the parameters α , D , r and H respectively, as visualized by the clustered alignment in Figure 4.8.

4.3.4 Model Capacity and Parameter Identifiability

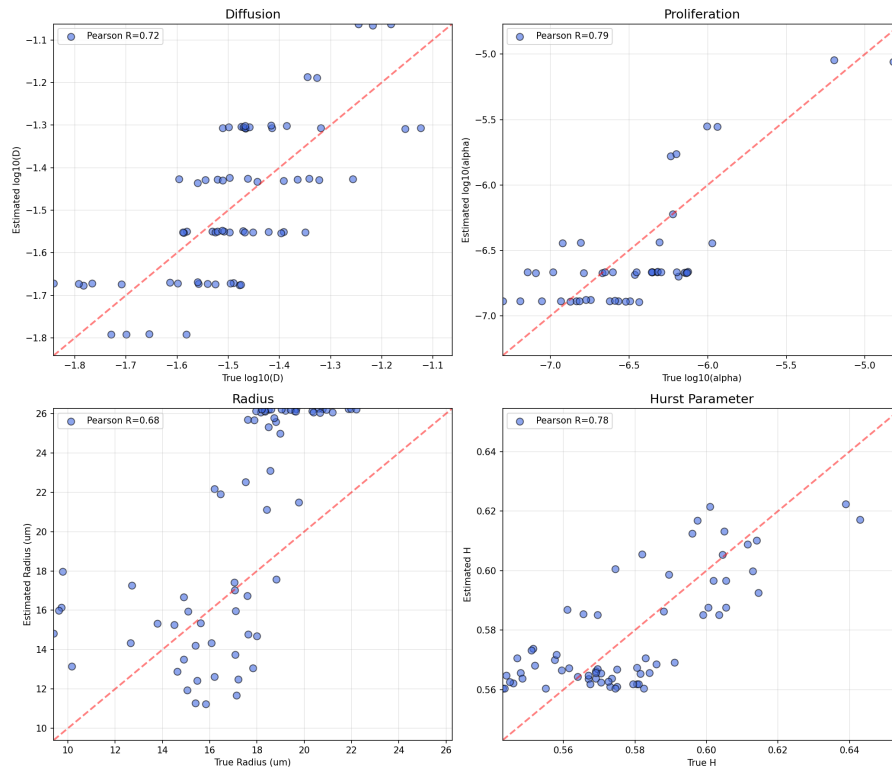
Beyond the scalar correlation metrics (Table 4.4), analyzing the dispersion in the estimated versus true scatter plots (Figure 4.8) reveals critical insights into the underlying mechanics of the velocity-driven model. While SMC-ABC successfully pulls the estimates closer to the identity line ($y = x$) compared to the Rejection baseline, there is noticeably more scatter (variance) in the fBm predictions compared to the simpler Langevin model.

This increased dispersion visually represents a drop in parameter identifiability, a direct consequence of increasing the model's capacity. Because the ABC algorithm evaluates the spatial structure at a single endpoint, it suffers from "unidentifiability" when multiple parameters govern similar physical traits. For instance, the simulation can achieve highly similar spatial expansion targets by either utilizing a low baseline diffusion (D) coupled with highly persistent, straight-line motion (high H), or by utilizing a high baseline diffusion with erratic, random-walk behaviour ($H \approx 0.5$).

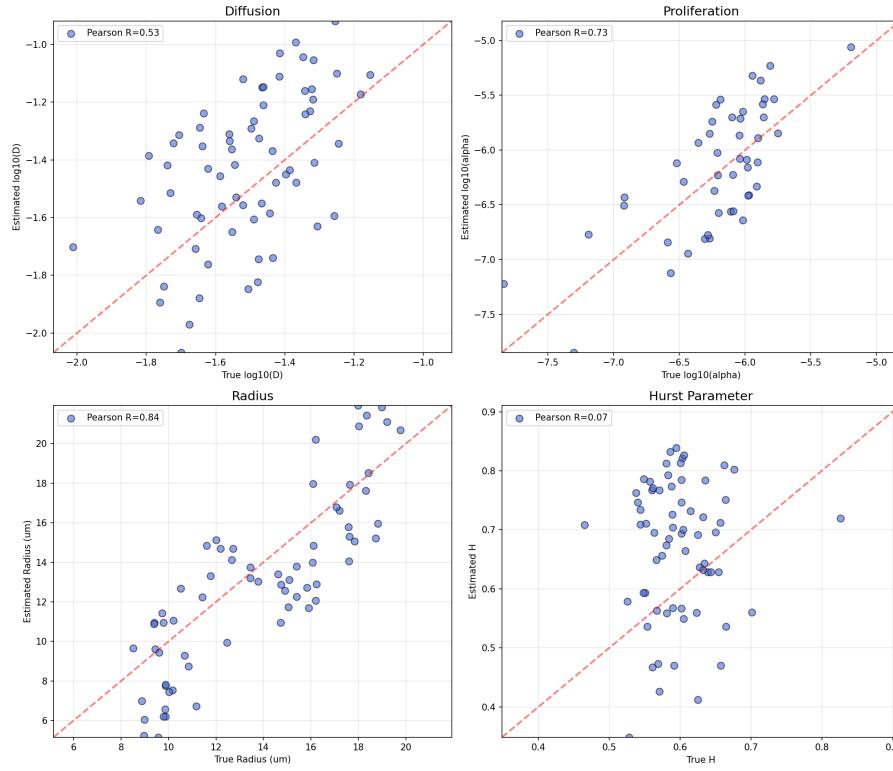
Because both parameters dictate cellular spreading, the algorithm struggles to definitively assign credit to one mechanism over the other based strictly on an endpoint snapshot. Consequently, the algorithm occasionally estimates a slightly higher D and a lower H than the ground truth (or vice versa), which manifests as the vertical and horizontal spread away from the perfect identity line in Figure 4.8. This empirically demonstrates that while adding physical realism (via Fractional Brownian Motion) improves the theoretical validity of the model, it inherently reduces the identifiability of individual parameters unless the summary statistics are enriched to completely decouple them.



(a) SMC



(b) Rejection



(c) Regression

Figure 4.8: Scatter plot of estimated parameter values against true values in the collection $\theta = (\log_{10} \alpha, \log_{10} D, r, H)$ for the Fractional Brownian Motion model.

4.3.5 Topological Feature Extraction and Synthetic Performance

To evaluate the efficacy of the proposed Topological Data Analysis (TDA) summaries in breaking parameter degeneracy, we first analyzed the feature profiles extracted from the *in vitro* simulations. Figure 4.9 illustrates the multimodal feature extraction profile for a representative simulation, with each sub-panel capturing a distinct physical trait. The Pair Correlation Function (PCF) isolates local demographic spacing in the left panel. The discrete Betti curves in the middle panel capture the hard physical packing limits by recording the exact integer count of geometric features across the filtration radius.

Most importantly, the right panel displays the pure TDA Persistence Image Vector. As established in the hybrid summary framework, the 2D H_0 and H_1 persistence grids are flattened and concatenated into a continuous 1D summary vector for the inference algorithm. The sharp peaks in this normalized intensity profile correspond to the dominant spatial scales of the topological features. Specifically, they highlight the characteristic proximity at which individual cells merge into connected tissue complexes (H_0) and the primary spatial scales of the steric voids trapped between cells (H_1). On ideal synthetic data (Figure 4.10), these structures provide clean, mathematically precise geometric signals well separated from background noise.

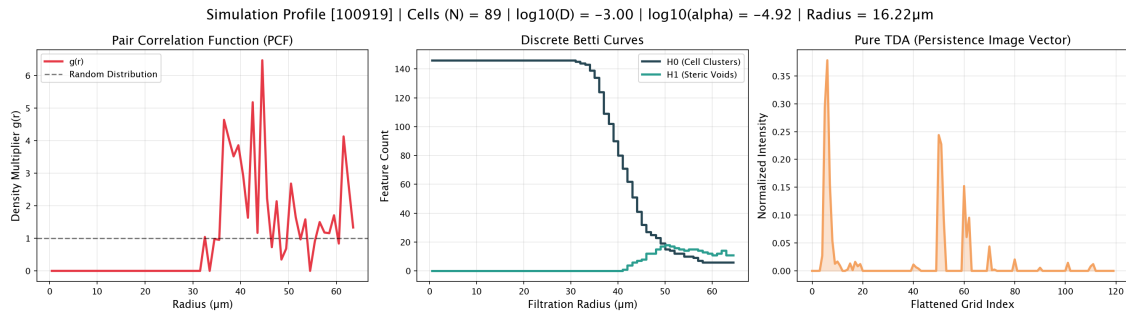


Figure 4.9: Multimodal feature extraction for a single simulation, showing local spatial mechanics (PCF), discrete topological features (Betti curves), and continuous topological robustness (Persistence Images).

However, the theoretical purity of TDA breaks down when exposed to real-world biological noise. Figure 4.11 visualizes the Alpha Complex and persistence diagram extracted from a true *in vitro* frame (VID559_F7_1_00d11h00m). The biological tissue is inherently messy, containing segmentation artifacts, highly diffuse cellular boundaries, and debris. A single segmentation error bridging two cells breaks down the underlying Alpha Complex. Furthermore, this topological fragility is exacerbated by true biological morphology. While the in-silico simulator models cells as perfect rigid spheres, real glioblastoma cells possess deformable membranes and lamellipodial protrusions. This innate "squishiness" naturally violates rigid geometric packing rules, creating irregular cellular intersections. Consequently, the topological voids seen in the synthetic data are erased. As seen in the right panel of Figure 4.11, the persistence diagram becomes overwhelmingly saturated with "topological dust", a

massive cluster of short-lived H_0 and H_1 features crammed against the birth-death diagonal, making it extremely difficult to isolate the true macroscopic biological structures.

To evaluate algorithmic robustness, performance was stratified by cellular diffusion ($\log_{10}(D) = -0.926$). As shown in Figure 4.12, the Mean Absolute Error (MAE) for proliferation and cell radius increased significantly in the high-diffusion regime across almost all models. This occurs because high diffusion prevents cells from reaching confluence. Without tight cellular packing, closed H_1 topological loops rarely form, depriving the Betti-1 curve of the steric exclusion data required to accurately estimate the cell radius. Despite this challenge, the discrete **Betti-Curves** model yielded the lowest MAE for radius estimation under synthetic conditions, while **PCF** exhibited the highest error across synthetic regimes.

However, when evaluated on real in-vitro data, the performance hierarchy inverted (Figure 4.13). **PCF** emerged as the strongest predictor for the proliferation rate ($R = 0.83$). The discrete **Betti-Curves** maintained strong performance for morphological radius ($R = 0.82$). In contrast, the continuous TDA models (**Pure-TDA** and **Hybrid-Continuous**) exhibited severe metric collapse, generating identical, horizontal parameter predictions across disparate biological targets.

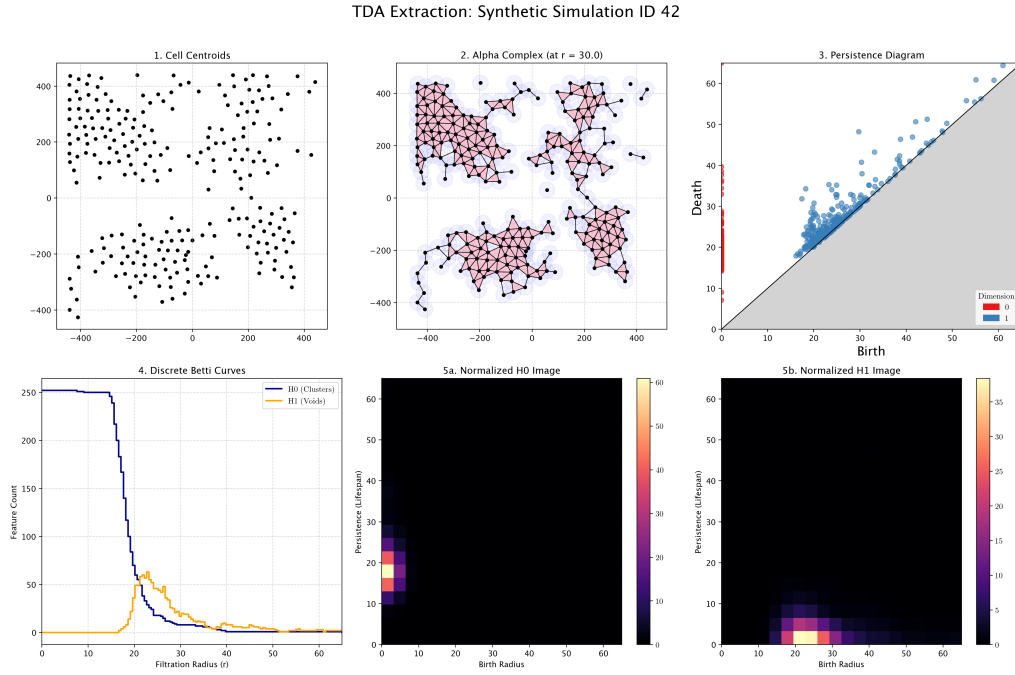


Figure 4.10: Alpha complex and persistence diagram for synthetic data. The strict physical rules of the simulator generate clean, mathematically perfect voids (H_1) that are easily captured by TDA.

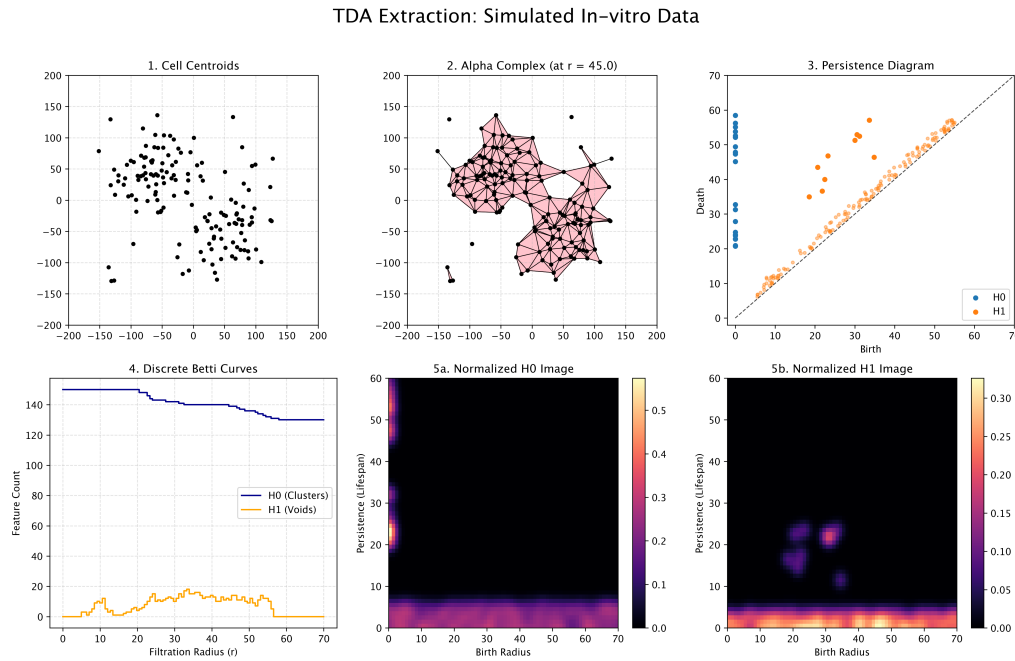


Figure 4.11: Alpha complex and persistence diagram extracted from real in-vitro data. Segmentation errors and diffuse boundaries shatter the pristine topology, generating massive amounts of low-persistence "topological dust" (features clustered near the diagonal line).

4. Results

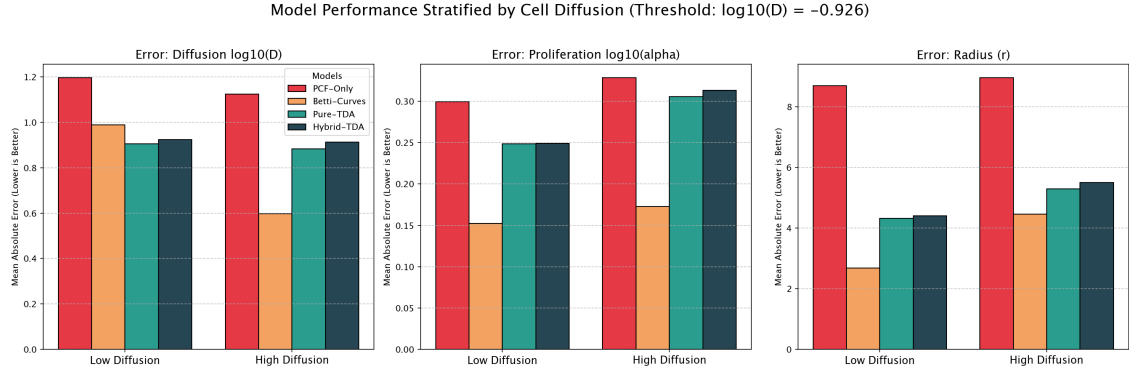


Figure 4.12: Model performance (Mean Absolute Error) stratified by cellular diffusion. Lower values indicate better parameter recovery. High diffusion generally decreases the accuracy of proliferation and radius estimates.

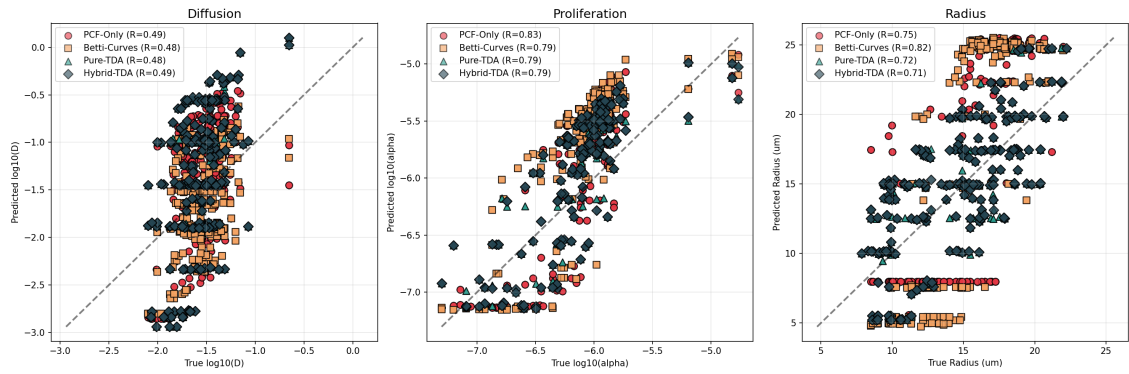


Figure 4.13: Predicted vs. True parameters for in-vitro experimental data. The horizontal banding indicates metric collapse in the continuous TDA models, while discrete topology (Betti) and spatial mechanics (PCF) demonstrate greater robustness to real-world morphological and proliferative noise, respectively.

5

Discussion

The results of this thesis demonstrate that inferring complex, interacting cell dynamics from single time-point images is both mathematically viable and biologically highly informative. However, the progression from basic positional models to advanced velocity-driven frameworks exposed profound methodological trade-offs. This chapter interprets these findings, explicitly identifying the optimal configurations for models, algorithms, and summary statistics, while dissecting the inherent limitations of high-capacity inference in noisy biological environments.

5.1 Segmentation Mechanics: Inductive Bias and Boundary Preservation

The quantitative triumph of the U-Net (ResNet34) over both classical methods and modern Vision Transformers (SegFormer) highlights an important intersection between deep learning theory and biological image analysis. The segmentation of high-density phase-contrast microscopy is uniquely challenging because the visual features separating two touching cells—often just a faint, one-pixel-wide diffraction halo—are highly localized.

The U-Net architecture possesses a strong "convolutional inductive bias." Convolutional Neural Networks (CNNs) inherently process images through local receptive fields, forcing the network to prioritize immediate spatial neighborhoods and high-frequency edge gradients. This hardwired locality makes the U-Net exceptionally adept at detecting the contrast-heavy, localized boundary lines required to separate confluent cells.

In contrast, the SegFormer architecture relies on a self-attention mechanism, which globally evaluates the relationships between all patches of an image simultaneously. While self-attention excels at capturing macroscopic context (e.g., identifying that a large region of the image is a tumor mass), it lacks the strict local inductive bias of a CNN. Without being trained on a massive, scale-invariant dataset, the transformer struggles to resolve the microscopic, high-frequency boundary details. Therefore, in the context of phase-contrast instance segmentation where a single misclassified pixel can merge two trajectories and corrupt the downstream mean

squared displacement, the local precision of the CNN strictly outweighs the global awareness of the Transformer.

Furthermore, the success of the U-Net was highly dependent on the boundary-aware probability mapping (Eq. 2.1) implemented during post-processing. Training the network to predict an explicit, third "boundary" class provided a dedicated mathematical channel for separation. By actively subtracting the boundary probability from the cell-interior probability, as seen in (2.1), we forced the continuous probability landscape to trench at cell interfaces. This probabilistic suppression proved mandatory for generating the discrete, biologically accurate geometries required for the downstream Topological Data Analysis.

5.2 Segmentation and Tracking: Overcoming Survival Bias

The quantitative results from the segmentation and tracking modules explicitly highlight the fragility of classical computer vision techniques in dynamic biological environments. While the transformer-based SegFormer architectures hold theoretical promise, the U-Net (ResNet34) configuration proved strictly superior in this context. This is largely due to the convolutional inductive bias, which actively preserves local boundary evidence, a critical requirement for preventing under-segmentation in highly confluent regions. The temporal sanity checks further validated this choice, proving that boundary-aware probability mapping successfully halts artificial radius inflation.

More profoundly, the tracking results expose a severe survival bias inherent to standard proximity-based algorithms. The Standard tracker ($L_{\max} = 20$) recovered roughly 2,200 fewer particles than the Velocity-Predictive tracker. This indicates that approximately 35% of the population, specifically the highly motile, invasive phenotypes capable of large spatial displacements is entirely invisible to standard parameter constraints. If not corrected, this dropout systematically skews the population-level diffusion estimates towards stationary behavior. Furthermore, while the Adaptive tracker failed globally due to combinatorial explosions in dense clusters, its ability to yield the longest continuous tracks for isolated cells proves that trajectory fragmentation is an algorithmic artifact of restricted search radii, rather than biological disappearance.

5.3 Model Selection and the Cost of Capacity (fBm)

By transitioning from the memoryless Langevin model to the Fractional Brownian Motion (fBm) model, this thesis sought to validate the hypothesis that directional persistence is a fundamental driver of glioblastoma motility. The empirical recovery of the Hurst parameter skewed heavily into the super-diffusive regime ($H \in [0.47, 0.83]$) provides definitive, data-driven proof that these cells possess mi-

gratory memory, invalidating pure Brownian motion for the cell lines studied.

While the fBm model demonstrates a clear improvement in the biological representation of cellular motility, this realism comes at a steep statistical cost, illustrating a classic manifestation of the curse of dimensionality.

As model capacity increases (from three parameters in Langevin to four in fBm), the potential for information loss and parameter degeneracy amplifies. As shown by the posterior collinearity (Section 4.5.4), the dynamics of diffusion (D) and persistence (H) are deeply coupled. When we increased the model capacity to include H , the accuracy of the isolated motion parameters improved, but the identifiability of the non-motion parameters (radius and proliferation) inherently degraded. Two dynamical parameters are vastly more difficult to untangle from a static snapshot than one. This confirms that adding physical realism to an endpoint-inference framework inadvertently introduces parameter degeneracy unless the constraining power of the summary statistics is proportionally scaled.

5.4 ABC Algorithmic Hierarchy

The evaluation of the ABC algorithms established a clear empirical hierarchy required to handle this expanded model capacity: SMC-ABC outperforms Regression-Adjusted ABC, which strictly outperforms pure Rejection ABC.

While Regression-Adjusted ABC successfully projected accepted samples toward the exact observation for the simpler Langevin model, its severe failure on the fBm model with R correlation coefficients of 0.07 and 0.53 for H and D respectively, highlights a critical mathematical limitation. Regression ABC assumes a locally linear relationship between the summary statistics and the parameter space within the tolerance threshold. However, the coupled dynamic between diffusion and the Hurst parameter creates a highly nonlinear, irregular mapping manifold. As explicitly documented in the broader Approximate Bayesian Computation literature, when this local linearity assumption is violated, linear regression adjustments can severely distort the posterior and even project adjusted samples outside the boundaries of the prior [8, 47]. Consequently, when linear regression attempts to project samples across the complex space generated by the fBm model, it systematically distorts the inferred distributions.

SMC-ABC circumvents this by employing adaptive sequential tolerances, strict particle reweighting, and perturbation kernels. It iteratively herds particles into high-posterior regions without relying on linear assumptions. To fully understand the necessity of this approach, it is instructive to observe how the algorithmic performance shifted across different data domains and summary statistics. On ideal synthetic data, SMC-ABC already exhibited significantly tighter posterior contraction compared to standard rejection methods. When transitioning to real biological data, this performance gap was amplified. The inherent noise and segmentation artifacts in the true *in vitro* images caused standard rejection methods to suffer from

extreme acceptance failure, whereas the adaptive tolerance schedule of SMC-ABC successfully maintained a viable particle population. Furthermore, the algorithmic hierarchy was actively disrupted when swapping the underlying summary statistics. While Regression ABC provided marginal improvements over basic rejection when utilizing the simple Pair Correlation Function, introducing the Topological Data Analysis summaries completely inverted this relationship. As discussed previously, the highly nonlinear geometry of the topological space caused Regression ABC to systematically distort the posterior, causing it to perform worse than basic rejection sampling. Therefore, SMC-ABC is the mandatory algorithmic choice when deploying high-capacity models coupled with advanced topological summaries.

5.5 The Sim-to-Real Gap and Topological Fragility

The results of this study reveal a profound paradox in Simulation-Based Inference: the summary statistics that mathematically dominate in an idealized synthetic environment often collapse in noisy in-vitro imagery.

The severe metric collapse observed in the continuous TDA models (Pure-TDA and Hybrid-TDA) on real data is a symptom of "Topological Fragility." In synthetic simulations, cells are perfect geometric entities that obey strict repulsive physics, generating pristine topological voids (H_1). However, real biological tissue is inherently messy, containing segmentation artifacts, diffuse boundaries, and debris. A single segmentation error bridging two cells completely shatters the underlying Alpha Complex, causing a massive, discontinuous penalty in the topological distance metric.

Because this topological distance between the real target and the synthetic simulation is universally massive, the ABC acceptance kernel effectively discards the TDA vector and relies entirely on the cell count (N). This forces the posterior to degenerate, returning identical parameter estimates for any experiments that share similar population sizes.

Conversely, the Pair Correlation Function (PCF) exhibits "Statistical Robustness." Because PCF averages the radial distances between all neighbors, local segmentation errors barely shift the global aggregate. Thus, PCF emerged as the strongest predictor on real data not because it is mathematically superior, but because it is blunt enough to ignore the biological noise that destroys the fragile continuous TDA signal.

5.5.1 The Information Limit of Endpoint Assays: Integrals vs. Suprema

The drastic performance inversion between synthetic and experimental data highlights a fundamental mathematical difference in how summary statistics process biological noise. The Pair Correlation Function (PCF) survived the transition to real-world data because it is fundamentally an integral statistic. PCF calculates

the radial density by averaging pairwise distances across the entire domain. If the segmentation algorithm generates a false positive (e.g., classifying a piece of debris as a cell), the impact on the PCF is merely a perturbation of magnitude $\mathcal{O}(1/N^2)$. The global average effectively absorbs the local error.

In stark contrast, Topological Data Analysis relies on the exact geometric construction of the Alpha Complex, which is dictated by Voronoi boundaries and local suprema. Simplicial homology is exquisitely sensitive to topological "short circuits." A single false-positive segmentation can act as an artificial bridge between two previously disconnected cellular clusters. Mathematically, this instantaneous topological merge destroys a Betti-0 feature (a connected component) and often creates a false Betti-1 feature (a spatial loop).

Because the Persistence Image maps these discrete topological features onto a continuous grid, a single segmentation error does not merely add noise to the image; it fundamentally deletes a high-persistence macroscopic feature and replaces it with low-persistence topological dust. When the ABC algorithm calculates the Mean Squared Error (MSE) between a pristine synthetic heatmap and a dust-saturated real heatmap, the distance penalty explodes. Faced with universally massive topological distances across all simulated targets, the ABC acceptance kernel effectively discards the TDA vector and falls back entirely on the raw cell count (N). This mathematical breakdown forces the posterior to degenerate, resulting in the identical, horizontal bands of parameter estimates observed during inference. Continuous TDA extracts higher-order geometric information that PCF misses, but it requires a level of absolute coordinate fidelity that current phase-contrast segmentation networks simply cannot guarantee.

5.6 The Mechanism of the Hybrid Summary Statistic

Given the mathematical vulnerabilities of continuous TDA and the over-simplicity of the PCF, the **Hybrid-Discrete** configuration (PCF combined with discrete Betti-0 and Betti-1 curves) emerged as the strictly optimal metric. The success of this specific combination lies in its decoupling of distinct physical forces.

The PCF acts as the "soft" demographic anchor. It successfully captures the long-range spatial homogenization driven by diffusion, as well as the overall clustering driven by proliferation. However, PCF struggles to resolve the exact physical boundaries of the cells, which is why it performed poorly at estimating the cell radius (r) in isolation.

The discrete Betti curves serve as the "hard" geometric anchor. Because Betti curves count the exact number of topological features alive at a specific proximity radius, they act as a strict detector for steric exclusion. As the filtration radius ϵ expands, the precise moment that the Betti-0 curve plunges (cells merging into complexes)

directly corresponds to the physical packing limit of the cell culture. By concatenating these two statistics, the ABC acceptance kernel forces the simulation to simultaneously satisfy the broad demographic spreading (via PCF) and the strict physical packing geometry (via Betti curves), breaking the parameter degeneracy that otherwise plagues high-capacity velocity models.

5.7 Limitations

A significant constraint during the deep learning evaluation was hardware limitations, specifically the 8 GB VRAM capacity of the NVIDIA Quadro P4000 GPU. This precluded the use of massive foundational models such as Segment Anything (SAM) and Micro-SAM. Utilizing the ViT-H (Huge) encoder requires up to 9 GB of VRAM solely for intermediate activation maps during 1024×1024 image embeddings, resulting in Out-of-Memory errors. Consequently, the study was restricted to more computationally efficient architectures (like SegFormer and U-Net).

Furthermore, due to computational time constraints, fine-tuning techniques for the Regression and SMC variants on the computationally heavy fBm model were limited. ABC-MCMC was also excluded from the framework, as its reliance on single Markov chains risks entrapment in local modes, making ABC-SMC the strictly preferred alternative for robust posterior exploration.

Additionally fundamental biological and modelling limitation of the current study is its restricted empirical scope. The inference pipeline and the underlying deep learning segmentation models have been evaluated exclusively on a single biological cell line. While the proposed topological framework is designed to be highly generalizable, the specific motility profiles, contact inhibition thresholds, and spatial clustering behaviors observed may be idiosyncratic to this specific cell type. Additionally, the *in-silico* simulator relies on a fixed set of mathematical representations for cellular dynamics. We have not systematically explored alternative generative models, such as utilizing different inter-cellular force functions (e.g., varying the repulsive interaction potentials) or implementing alternative mathematical forms for density-dependent cell division. Validating the pipeline across diverse cell lines and comparing a broader class of mechanobiological models remains an essential avenue for future research.

5.8 Future Work

5.8.1 Bridging the Sim-to-Real Inference Gap

To unlock the theoretical power of TDA for real-world microscopy, future frameworks must explicitly address topological fragility. This could be achieved via Forward Modeling, wherein the simulator passes its output through a synthetic optical pipeline, adding blur and segmentation artifacts before the TDA extraction occurs. Alternatively, applying strict lifespan thresholding to the persistence diagrams could

algorithmically filter out the "topological dust" caused by segmentation errors, preserving only the macroscopic tissue structures.

5.8.2 Preliminary Exploration of Gradient-Based Contour Extraction (The Franzeg Algorithm)

Given that downstream topological inference is heavily bottlenecked by binary segmentation errors, future work in computer vision must focus on highly precise, artifact-free contour extraction. During this thesis, a novel scale-space contour algorithm (termed the "Franzeg Algorithm") was developed and preliminarily tested to address cells with complex lighting halos.

The algorithm utilizes scale-normalized Gaussian derivatives to stabilize gradient computations against high-frequency microscopy noise. The image I is convolved with a Gaussian kernel G_σ , and the maximum scale-normalized gradient response is computed across a discrete set of scales s

$$\mathcal{G}(x) = \max_{s \in \mathcal{S}} s \cdot \left\| \nabla(G_s * I)(x) \right\| \quad (5.1)$$

This normalisation ensures that both sharp, thin boundaries and broad, faint halos exhibit comparable edge strengths. Candidate contours are subsequently extracted by sweeping level sets across the intensity distribution and extracting polygonal curves via the marching squares algorithm.

To filter spurious artifacts, the algorithm employs hysteresis thresholding on the multi-scale gradient map, coupled with strict directional intensity tests. By computing the orthogonal normal vector $\mathbf{n}(x)$ along the contour, the algorithm validates that the exterior intensity S_{out} is strictly brighter than the interior intensity S_{in}

$$\text{median}_{x \in P} \left(S(x + d\mathbf{n}) - S(x - d\mathbf{n}) \right) \geq \gamma_{\min} \quad (5.2)$$

Finally, geometric constraints ensure the closed contour encircles a region satisfying a minimum dark-area threshold relative to the heavily blurred macroscopic background.

In preliminary testing on densely populated cell lines (e.g., A1), the Franzeg algorithm achieved an exceptional DSC score of 0.98. However, its current reliance on strict gradient thresholds limits its generalization to cells with highly diffuse, invisible leading edges, preventing its deployment as the primary engine in this thesis. Future research dedicated to adaptive, localized pre-processing could finalize this algorithm, potentially eliminating the binary segmentation artifacts that currently cripple Topological Data Analysis.

6

Conclusion

6.1 Segmentation and Feature Extraction

The foundation of this thesis relied on the accurate extraction of cell morphology and spatial distributions from high-density *in-vitro* imagery. To achieve this, we implemented a robust segmentation pipeline combining a three-class probabilistic prediction model with boundary-aware probability shaping. By rigorously benchmarking multiple model families, we determined that the CNN-based U-Net architecture with a ResNet34 backbone provided the optimal balance of quantitative accuracy and qualitative robustness. Temporal sanity checks confirmed that this boundary-penalizing approach effectively mitigated instance merging in confluent cultures, producing stable radius and displacement statistics.

Crucially, our evaluation of trajectory tracking strategies revealed a severe survival bias in standard methodologies, which systematically fail to capture highly motile, invasive phenotypes. The implementation of a Velocity-Predictive tracking algorithm proved mandatory for recovering these trajectories, ensuring that the downstream Bayesian inference was fed biologically representative, unbiased data.

6.2 Parameter Estimation and Mechanistic Modelling

Building upon these extracted spatial features, this thesis demonstrated that the inference of interacting cell dynamics from single time-point images can be substantially advanced through algorithmic sophistication.

Regarding model selection, we conclude that the velocity-driven Fractional Brownian Motion (fBm) model is the optimal mechanistic representation of glioblastoma motility. The empirical recovery of the Hurst parameter provided definitive evidence of directional persistence, invalidating standard memoryless Brownian models for our cell lines. However, this thesis highlights an important caveat regarding model capacity: as we increased the parameter space to capture this persistence, we encountered the curse of dimensionality. The deeply coupled dynamics of diffusion

and persistence create parameter degeneracy at a single temporal endpoint. Adding capacity improves motility realism, but leads to information loss regarding morphology, as two dynamical parameters are vastly more difficult to untangle from a static snapshot than one.

To navigate this non-linear, high-capacity parameter space, advanced inference algorithms were required. While Regression-Adjusted ABC sufficed for simpler models, it failed completely on the complex fBm manifold. We conclude that Sequential Monte Carlo (SMC-ABC) is the strictly optimal algorithm for this framework, as its adaptive, sequential tolerance schedule successfully mitigated tolerance-induced bias without relying on local linear assumptions.

6.3 The Efficacy of Summary Statistics

A major finding of this work is the observed "Sim-to-Real Gap" in spatial summary statistics. While continuous Topological Data Analysis (TDA) via Persistence Images provided flawless inference on synthetic data, it suffered from catastrophic topological fragility when exposed to the noise and segmentation artifacts of real *in-vitro* imagery.

Consequently, we conclude that a hybrid approach utilizing the Pair Correlation Function (PCF) alongside discrete Betti Curves is the optimal summary statistic configuration for real-world endpoint inference. The robust, averaging nature of the PCF successfully isolated proliferation and density dynamics, while the discrete geometric counting of Betti-0 and Betti-1 curves perfectly constrained the hard physical packing limits governing cell radius.

6.4 Final Remarks

Ultimately, this work proves that single-image inference is a highly viable, scalable approach for quantifying complex cell population dynamics. However, successfully extracting advanced trajectory memory and interaction dynamics from static snapshots requires a careful balancing act. The pursuit of biological realism through higher-capacity mathematical models must be met with equally robust summary statistics and advanced, non-linear Bayesian algorithms (SMC-ABC). Moving forward, extending the data collection to include multi-time endpoints or simulating synthetic optical noise before topological extraction will fully decouple these motion signatures, paving the way for high-throughput, non-invasive profiling of invasive diseases such as glioblastoma.

Bibliography

- [1] Henry Adams, Tegan Emerson, Michael Kirby, Rachel Neville, Chris Peterson, Patrick Shipman, Sofya Chepushtanova, Eric Hanson, Francis Motta, and Lori Ziegelmeier. Persistence images: A stable vector representation of persistent homology. *Journal of Machine Learning Research*, 18(8):1–35, 2017.
- [2] Bruce Alberts, Alexander Johnson, Julian Lewis, David Morgan, Martin Raff, Keith Roberts, and Peter Walter. *Molecular Biology of the Cell*. Garland Science, 6 edition, 2014.
- [3] Daniel B. Allan, Thomas Caswell, Nathan C. Keim, and Casper M. van der Wel. soft-matter/trackpy: Trackpy v0.6.1, 2023. URL <https://doi.org/10.5281/zenodo.34028>.
- [4] Mark A. Beaumont. Approximate bayesian computation. *Annual Review of Statistics and Its Application*, 2019. doi: 10.1146/annurev-statistics-030718-105212.
- [5] Mark A. Beaumont, Wenyang Zhang, and David J. Balding. Approximate bayesian computation in population genetics. *Genetics*, 162(4):2025–2035, 2002.
- [6] Jan Beran. *Statistics for long-memory processes*. CRC press, 1994.
- [7] F. Biagini, Y. Hu, B. Øksendal, and T. Zhang. *Stochastic Calculus for Fractional Brownian Motion and Applications*. Springer, 2008.
- [8] Michael GB Blum and Olivier François. Non-linear regression models for approximate bayesian computation. *Statistics and Computing*, 20(1):63–73, 2010.
- [9] Alexander P Browning, Scott W McCue, Rachelle N Binny, Michael J Plank, Esha T Shah, and Matthew J Simpson. Quantitative analysis of collective cell migration using approximate bayesian computation. *PLOS Computational Biology*, 14(1):e1005911, 2018.
- [10] Gunnar Carlsson. Topology and data. *Bulletin of the American Mathematical Society*, 46(2):255–308, 2009.
- [11] Joe Chalfoun, Michael Majurski, Alden Dima, Christian Stuelten, Adele Peskin, and Mary Brady. Fogbank: a single cell segmentation across multiple cell lines

- and image modalities. *BMC Bioinformatics*, 15(1):431, 2014. doi: 10.1186/s12859-014-0431-x. URL <https://doi.org/10.1186/s12859-014-0431-x>.
- [12] Caitlin E. Chan and David J. Odde. Traction dynamics of filopodia on compliant substrates. *Cellular and Molecular Bioengineering*, 1:327–338, 2008. doi: 10.1007/s12195-008-0035-5.
 - [13] David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Stability of persistence diagrams. *Discrete & computational geometry*, 37(1):103–120, 2007.
 - [14] John C Crocker and David G Grier. Methods of digital video microscopy for colloidal studies. *Journal of Colloid and Interface Science*, 179(1):298–310, 1996.
 - [15] R. B. Davies and D. S. Harte. Tests for Hurst effect. *Biometrika*, 74(1):95–101, 1987.
 - [16] Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. An adaptive sequential monte carlo method for approximate bayesian computation. *Statistics and Computing*, 22:1009–1020, 2012. doi: 10.1007/s11222-011-9271-y.
 - [17] Lee R. Dice. Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302, 1945.
 - [18] T. Dieker. Simulation of fractional Brownian motion. Master’s thesis, University of Twente, 2004. Technical report / master’s thesis.
 - [19] C. R. Dietrich and G. N. Newsam. Fast and exact simulation of stationary Gaussian processes through circulant embedding of the covariance matrix. *SIAM Journal on Scientific Computing*, 18(4):1088–1107, 1997.
 - [20] Herbert Edelsbrunner and John Harer. *Computational topology: an introduction*. American Mathematical Society, 2010.
 - [21] Albert Einstein. On the movement of small particles suspended in stationary liquids required by the molecular-kinetic theory of heat. *Annalen der Physik*, 1905.
 - [22] Peter Friedl and Katarina Wolf. Plasticity of cell migration: a multiscale tuning model. *The Journal of Cell Biology*, 188(1):11–19, 2010. doi: 10.1083/jcb.200909003.
 - [23] Crispin W. Gardiner. *Handbook of Stochastic Methods*. Springer, 4 edition, 2009.
 - [24] Benjamin Geiger, Joachim P. Spatz, and Alexander D. Bershadsky. Environmental sensing through focal adhesions. *Nature Reviews Molecular Cell Biology*, 10:21–33, 2009. doi: 10.1038/nrm2593.
 - [25] Robert W Ghrist. *Elementary applied topology*. Createspace Seattle, 2014.

-
- [26] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.
 - [27] Allen Hatcher. *Algebraic topology*. Cambridge University Press, 2002.
 - [28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. doi: 10.1109/CVPR.2016.90.
 - [29] Paul Jaccard. The distribution of the flora in the alpine zone. *New Phytologist*, 11:37–50, 1912.
 - [30] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. doi: 10.1109/5.726791.
 - [31] Gustav Lindwall. *Statistical inference on interacting particle systems with applications to cancer biology*. Doctoral thesis, Chalmers University of Technology, 2023.
 - [32] Jarno Lintusaari, Michael U. Gutmann, Ritabrata Dutta, Samuel Kaski, and Jukka Corander. Fundamentals and recent developments in approximate bayesian computation. *Systematic Biology*, 66(1):e66–e82, 2017. doi: 10.1093/sysbio/syw077.
 - [33] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. doi: 10.1109/CVPR.2015.7298965.
 - [34] Adam Malik and Philip Gerlee. The effect of substrate stiffness on migration efficiency of mesenchymal cells. *Journal of Mathematical Biology*, 2019.
 - [35] B. B. Mandelbrot and J. W. Van Ness. Fractional brownian motions, fractional noises and applications. *SIAM Review*, 10(4):422–437, 1968.
 - [36] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *3DV*, 2016. doi: 10.1109/3DV.2016.79.
 - [37] James R Munkres. *Topology*, volume 2. Prentice Hall Englewood Cliffs, NJ, 2000.
 - [38] napari contributors. napari: a multi-dimensional image viewer for Python, 2019. URL <https://doi.org/10.5281/zenodo.3555620>.
 - [39] Bernt Øksendal. *Stochastic Differential Equations: An Introduction with Applications*. Springer, 6 edition, 2003.
 - [40] Igor Podlubny. *Fractional differential equations*. Academic press, 1999.
 - [41] Jonathan K. Pritchard, Mark T. Seielstad, Anna Perez-Lezaun, and Marcus W. Feldman. Population growth of human y chromosomes: A study of y chromosome microsatellites. *Molecular Biology and Evolution*, 16(12):1791–1798, 1999.

- [42] Anne J. Ridley, Martin A. Schwartz, Keith Burridge, Richard A. Firtel, Mark H. Ginsberg, Gary Borisy, J. Thomas Parsons, and Alan R. Horwitz. Cell migration: Integrating signals from front to back. *Science*, 302(5651):1704–1709, 2003. doi: 10.1126/science.1092053.
- [43] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. doi: 10.1007/978-3-319-24574-4_28.
- [44] Emil Rosén, Hitesh Bhagavanbhai Mangukiya, Ludmila Elfineh, Rebecka Stockgard, Cecilia Krona, Philip Gerlee, and Sven Nelander. Inference of glioblastoma migration and proliferation rates using single time-point images. *Communications Biology*, 6(1):402, 2023.
- [45] Gennady Samorodnitsky and Murad S Taqqu. *Stable non-Gaussian random processes: stochastic models with infinite variance*. Chapman and Hall/CRC, 1994.
- [46] Nathaniel Saul and Chris Tralie. Scikit-tda: Topological data analysis for python, 2019. URL <https://doi.org/10.5281/zenodo.2533369>.
- [47] Scott A. Sisson, Yanan Fan, and Mark A. Beaumont, editors. *Handbook of Approximate Bayesian Computation*. Chapman and Hall/CRC, 2018.
- [48] Abdel Aziz Taha and Allan Hanbury. Metrics for evaluating 3d medical image segmentation: analysis, selection, and tool. *BMC Medical Imaging*, 15:29, 2015. doi: 10.1186/s12880-015-0068-x.
- [49] Köm M Taute, Stephan Gude, Sander J Tans, and Thomas S Shimizu. Unraveling the shared stochastic strategy of behavioral variability in e. coli. *Nature Communications*, 6(1):8776, 2015.
- [50] The GUDHI Project. *GUDHI User and Reference Manual*. GUDHI Editorial Board, 2026. URL <https://gudhi.inria.fr/doc/latest/>.
- [51] Tina Toni, David Welch, Natalja Strelkowa, Andreas Ipsen, and Michael P. H. Stumpf. Approximate bayesian computation scheme for parameter inference and model selection in dynamical systems. *Journal of the Royal Society Interface*, 6(31):187–202, 2009. doi: 10.1098/rsif.2008.0172.
- [52] Xavier Trepât and Erik Sahai. Mesoscale physical principles of collective cell organization. *Nature Physics*, 14:671–682, 2018. doi: 10.1038/s41567-018-0194-9.
- [53] Ashish Vaswani et al. Attention is all you need. In *NeurIPS*, 2017.
- [54] Binh N Vo, Christopher C Drovandi, Anthony N Pettitt, and Matthew J Simpson. Quantifying the uncertainty in spatial models of cell migration using approximate bayesian computation. *Mathematical Biosciences*, 269:55–67, 2015.

- [55] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *arXiv preprint arXiv:2105.15203*, 2021.
- [56] Y. Xie et al. The human glioblastoma cell culture (HGCC) database. *The Human Glioblastoma Cell Culture Consortium*, 2015. Available at: <https://www.hgcc.se/>.
- [57] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: A nested u-net architecture for medical image segmentation. *arXiv preprint arXiv:1807.10165*, 2018.

