

IDP Project Final Presentation

Proliferating Cell Collectives: A Comparison of Hard and Soft Collision Models

Manuel Lerchner
manuel.lerchner@tum.de

Advisor:
Samuel James Newcome

Outline

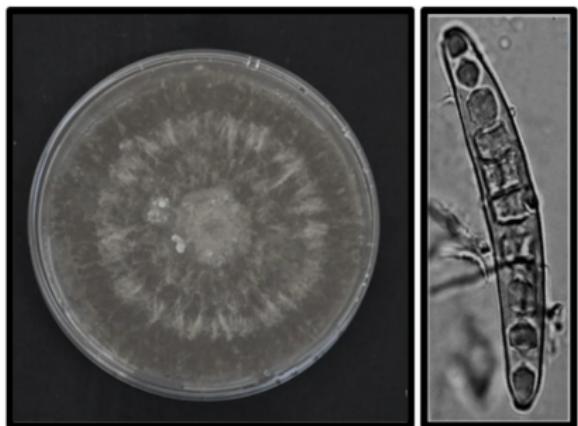
Computational Biology: A Growing Field

Why simulate biological systems?

- Link micro-to macroscopic behavior
- Test hypotheses in silico
- Guide experimental design
- Often just look beautiful

The computational challenge:

- Large system sizes
- Complex interactions
- Long timescales



Real fungal colony showing ring patterns [?].

The Performance Gap in Biological Modeling

Common Practice: Focus on Biological Insight

- Pick a model that *can* reproduce qualitative behavior.
- Tune parameters until the output “looks right”.
- Publish biological findings.
- **Runtime? Scaling?, Source Code?** Often unreported.

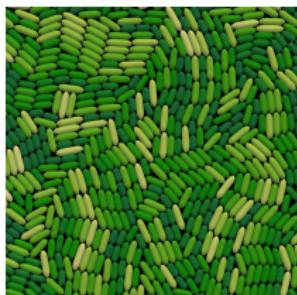
The Neglected Computational Core

- **Algorithmic Efficiency:** Can we optimize further?
- **Performance Scaling:** How does it handle larger systems?
- **Speed vs. Accuracy:** What are the concrete trade-offs?
- **Model Comparisons:** Are different methods equivalent?

This Work: Comparing Collision Models

Research Questions:

1. Do different collision models produce the same biological patterns?
 - Ring formation
 - Microdomain structure
 - Cell density
2. Which model is faster?
3. Which approach scales better?
4. When should each be used?



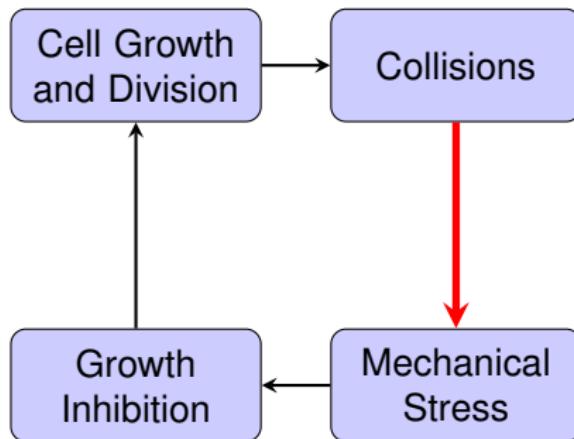
(a) Model 1



(b) Model 2

Outline

Core Biological Mechanism

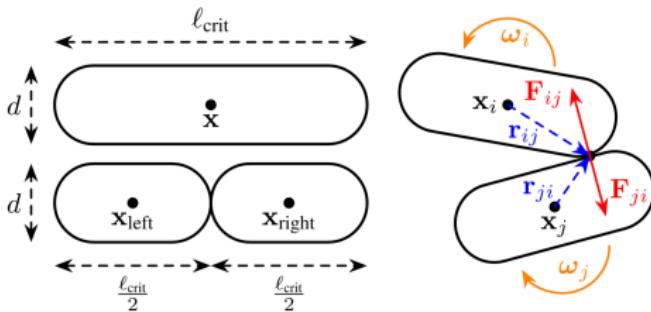


- We compare two methods to turn collisions into forces.

Outline

Cell Representation

- Length $\ell(t)$ grows over time
- Diameter $d = 0.5$ fixed
- Position $\mathbf{x} \in \mathbb{R}^3$, Orientation $\mathbf{q} \in \mathbb{R}^4$
- Divide at $\ell_{\text{crit}} = 2\ell_0$
- Contacts produce forces \mathbf{F}_{ij} and torques



Equations of Motion

Overdamped dynamics ($\text{Re} \ll 1$):

- Viscous forces dominate, inertia negligible
- Cells behave like “swimmers in honey”

Translational motion:

$$\mathbf{u}_i = \frac{1}{\zeta \ell_i} \sum_{j \neq i} \mathbf{F}_{ij}$$

Rotational motion:

$$\omega_i = \frac{12}{\zeta \ell_i^3} \sum_{j \neq i} \mathbf{r}_{ij} \times \mathbf{F}_{ij}$$

Velocity instantly proportional to force

Growth Model

Stress-dependent growth:

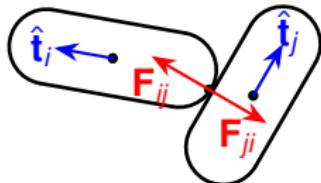
- Exponential growth
- Inhibited by longitudinal stress

Growth rate:

$$\dot{\ell}_i = \frac{\ell_i}{\tau} e^{-\lambda \sigma_i}$$

Longitudinal stress:

$$\sigma_i = \sum_{j \neq i} \frac{1}{2} |\hat{\mathbf{t}}_i \cdot \mathbf{F}_{ij}|$$



Numerical Integration Framework

Colony state vector:

$$\mathcal{C} = [\dots, \mathbf{x}_n^\top, \mathbf{q}_n^\top, \dots]^\top \in \mathbb{R}^{7N}$$

Total Force vector:

$$\mathcal{F} = [\dots, \mathbf{f}_n^\top, \tau_n^\top, \dots]^\top \in \mathbb{R}^{6N}$$

Euler integration:

$$\begin{aligned}\mathcal{C}^{k+1} &= \mathcal{C}^k + \Delta t \mathcal{G}^k \mathcal{M}^k \mathcal{F} \\ \ell^{k+1} &= \ell^k + \Delta t \dot{\ell}(\mathcal{F})\end{aligned}$$

How to compute forces \mathcal{F} ?

Outline

Two Paradigms

Soft Model

Potential-based

- Local pairwise forces
- Allows overlap
- Simple calculation
- Similar to MD simulations

$$\mathbf{F}_{ij} = k\delta^{3/2}\hat{\mathbf{n}}$$

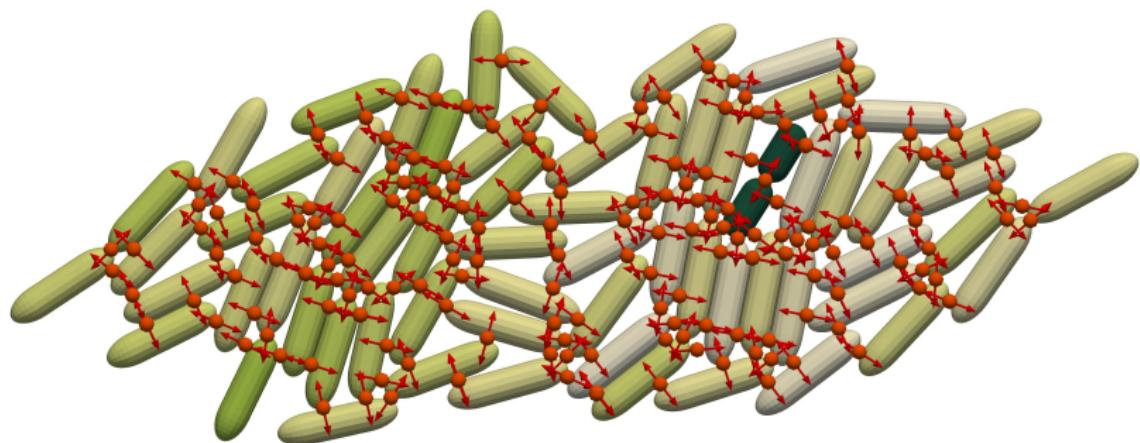
Hard Model [?]

Constraint-based

- Global optimization
- Strict non-overlap
- Complex solver
- Based on Contact Mechanics

$$\mathbf{F}_{ij} = \gamma_{ij}\hat{\mathbf{n}} \\ \text{s.t. } \mathbf{0} \leq \boldsymbol{\gamma} \perp \boldsymbol{\Phi} \geq \mathbf{0}$$

Collisions



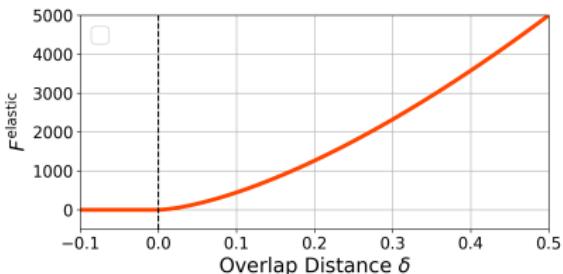
- Models only differ in how they calculate force multipliers.

Soft Model: Hertzian Contact

$$\mathbf{F}_{ij} = k_{cc} \sqrt{d} \delta^{3/2} \hat{\mathbf{n}}$$

Characteristics:

- + Embarassingly parallel
- + Simple implementation
- + Local calculations
- Numerically stiff
- Tiny timesteps (10^{-5})
- Allows overlap



Force increases with overlap

Hard Model: Constraint Framework

Key idea: Introduce non-overlap constraints

- Define gap function for each contact c :

Φ_c^k = "signed distance at contact c at time k "

- Express forces based on lagrange multipliers γ :
 - Contact: $\mathcal{F}(\gamma) = \mathcal{D}\gamma$
 - Stress: $\sigma(\gamma) = \mathcal{L}\gamma$

Physical requirements:

1. Repulsive Forces: $\gamma \geq \mathbf{0}$
2. No overlap in next step: $\Phi^{k+1} \geq \mathbf{0}$
3. No force at distance: $\gamma^\top \Phi^{k+1} = 0$

Hard Model: Mathematical Formulation

Update equations with constraints:

$$\begin{aligned}\mathcal{C}^{k+1} &= \mathcal{C}^k + \Delta t \mathcal{G}^k \mathcal{M}^k \mathcal{F}(\gamma) \\ \ell^{k+1} &= \ell^k + \Delta t \dot{\ell}(\gamma) \\ \text{s.t. } \mathbf{0} \leq \gamma \perp \Phi^{k+1}(\gamma) &\geq \mathbf{0}.\end{aligned}\tag{1}$$

Reformulate as energy minimization:

$$\min_{\gamma \geq \mathbf{0}} E(\gamma) = \min_{\gamma \geq \mathbf{0}} \gamma^\top \Phi^k + \frac{\Delta t}{2} \gamma^\top \mathcal{D}^\top \mathcal{M}^k \mathcal{D} \gamma + \mathbf{1}^\top \frac{\Delta t}{\lambda} \left(\frac{\ell}{\tau} \odot e^{-\lambda \mathcal{L}} \gamma \right)$$

Solve via convex optimization:

- Solve for γ^* that minimizes $E(\gamma^*)$
- Guarantees satisfaction of all constraints
- (See Paper for derivation)

Hard Model: Solver Pipeline

1. BBPGD Solver: Projected gradient descent

- Iteratively minimize $E(\gamma)$
- Typically ~ 2500 iterations
- Converges to tolerance $\epsilon = 10^{-3}$

2. ReLCP Refinement: Remove residual overlaps

- Detect remaining violations
- Add new constraints
- Re-solve (typically 6 iterations)

Characteristics:

- + Larger timesteps ($3 \cdot 10^{-4}$)
- + Strict non-overlap
- High per-step cost
- Complex implementation

Outline

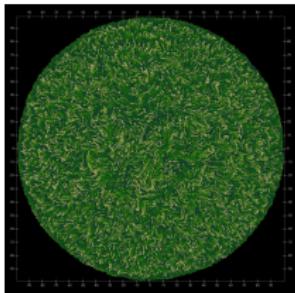
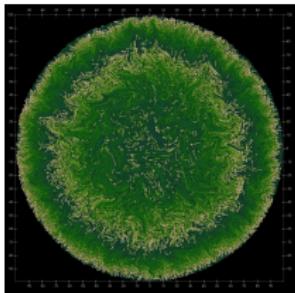
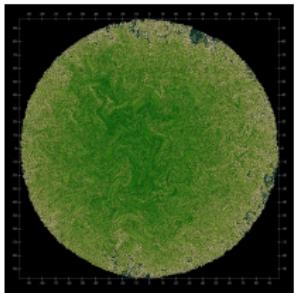
Concentric Rings: Qualitative Comparison

$$\lambda = 10^{-4}$$

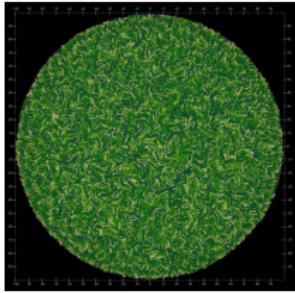
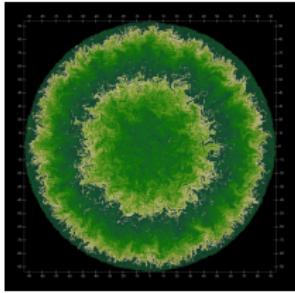
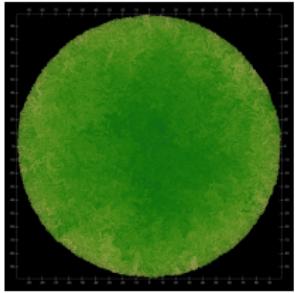
$$\lambda = 10^{-3}$$

$$\lambda = 10^{-2}$$

Hard



Soft

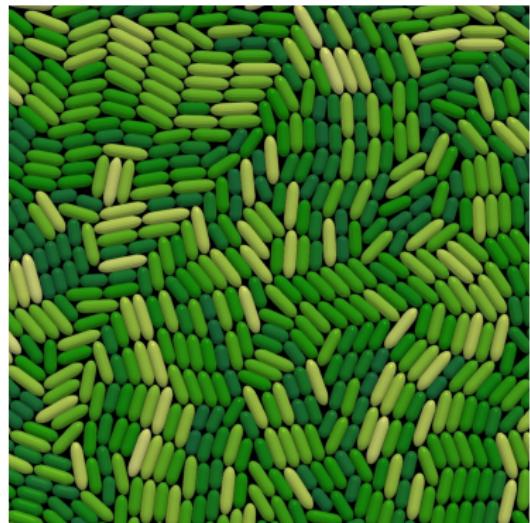


No rings

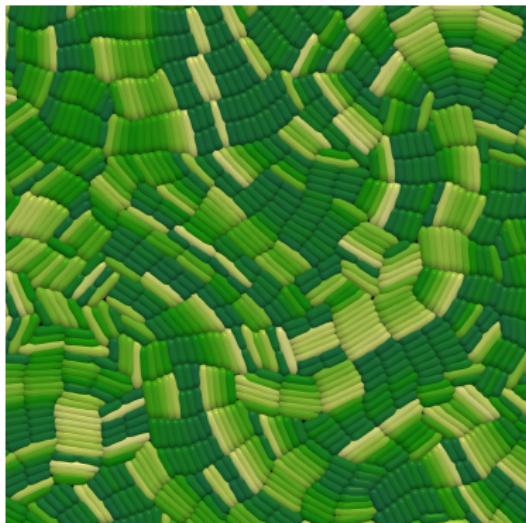
Clear rings

Dissipating

The Critical Difference: Packing Density

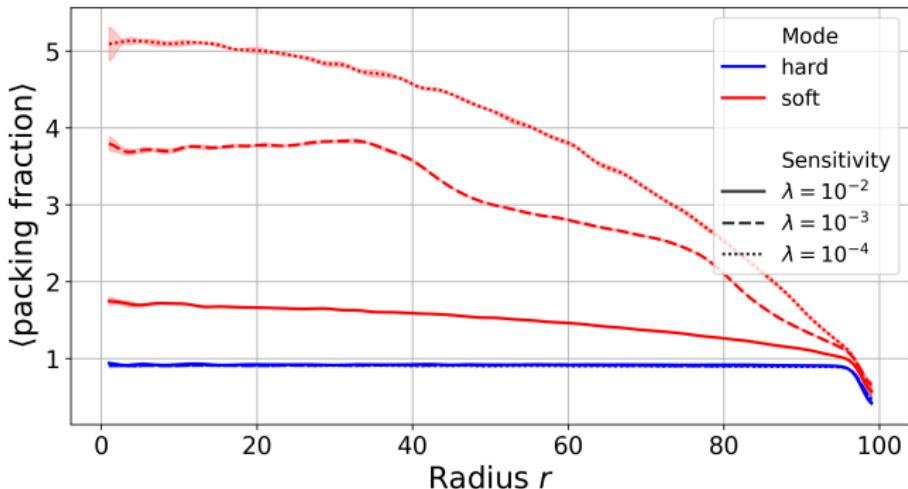


Hard Model



Soft Model

Quantifying the Packing Problem

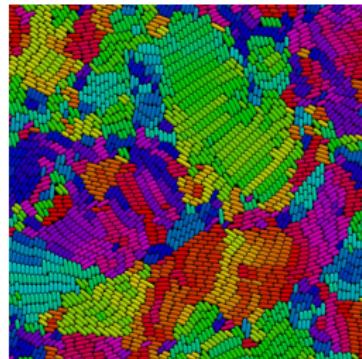


- Soft model: extreme center overcrowding
- Gets worse at low λ (weak inhibition)
- Hard model: constant realistic packing

Microdomain Formation

Small colonies ($R \approx 50$):

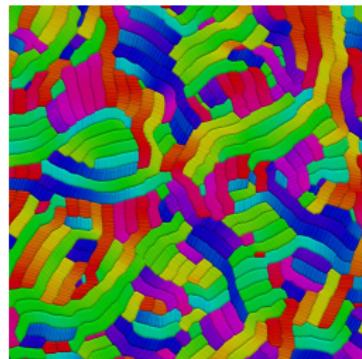
- Both work well
- Similar patch structure
- Matches experiments



Hard: realistic patches

Large colonies:

- Hard: consistent patches
- Soft: elongated bundles
- Artifact from overcrowding



Soft: elongated bundles

Outline

Adaptive Timestepping Strategy

CFL-based criterion:

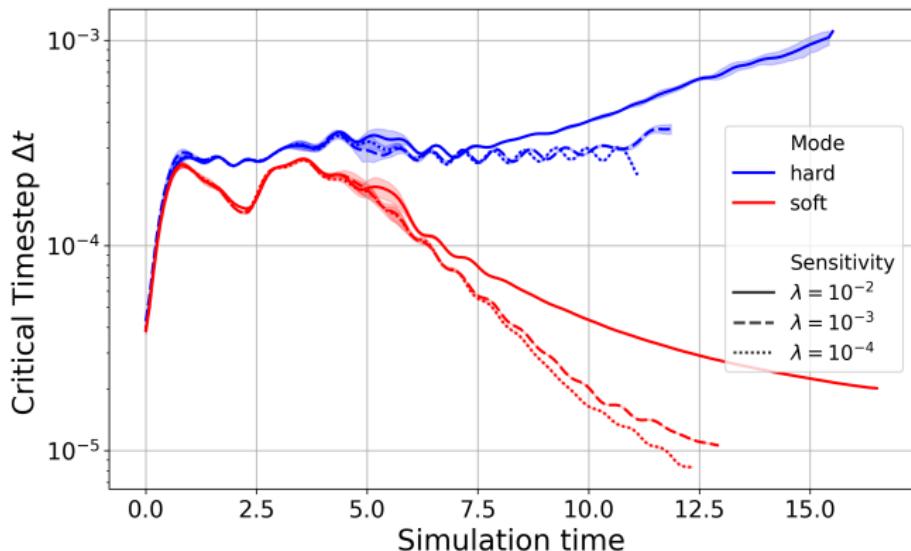
$$\Delta t = \frac{0.5 \varepsilon}{u_m}$$

where u_m = median cell velocity, $\varepsilon = 10^{-3}$ = tolerance

Benefits:

- Prevents excessive motion per step
- Adapts to colony dynamics
- Maintains numerical stability
- Essential for both models

The Timestep Divide



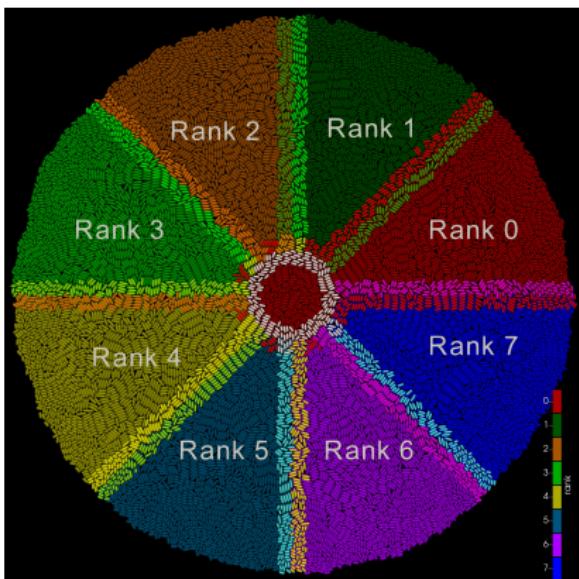
Parallel Architecture

Implementation:

- MPI + PETSc framework
- Distributed vectors/matrices
- Angular sector decomposition
- Ghost particle exchange

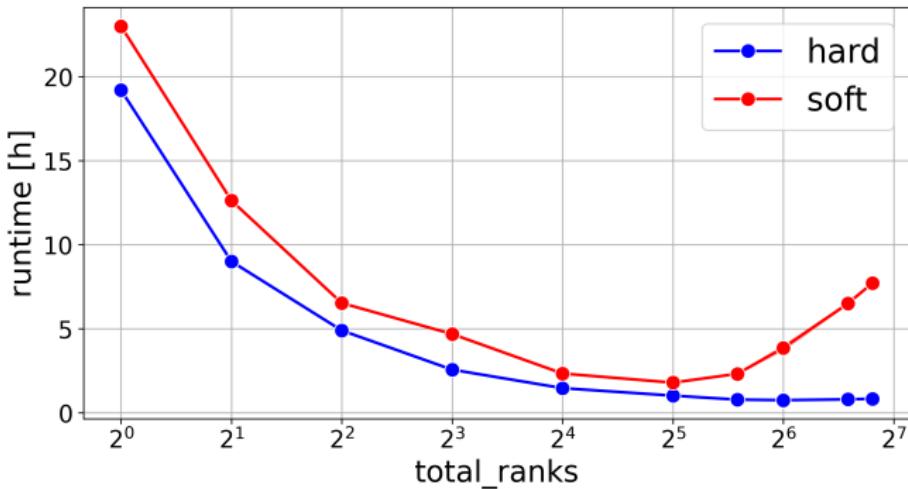
Benchmarking:

- CoolMUC-4 cluster
- Up to 112 cores



Domain decomposition with 8 MPI ranks

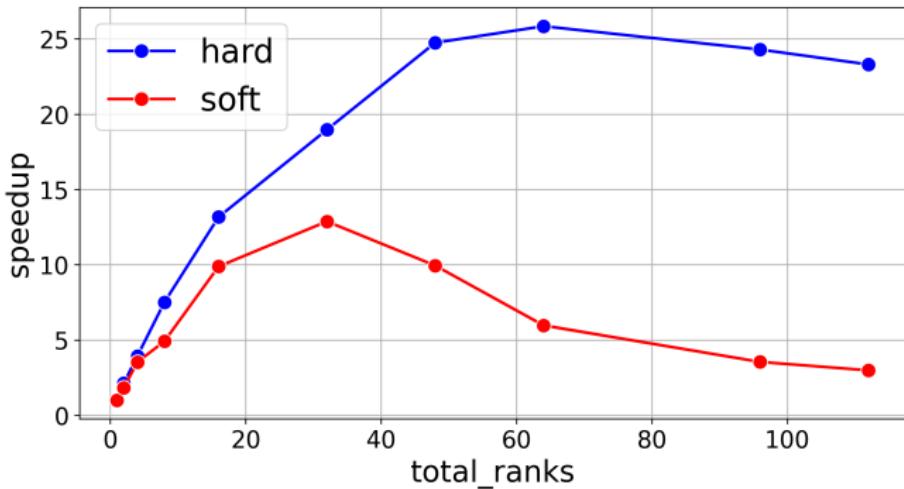
Strong Scaling: Runtime to reach $R = 100$



Key results:

- Hard model is always faster!
- Both model slow down at high core counts

Strong Scaling: Speedup



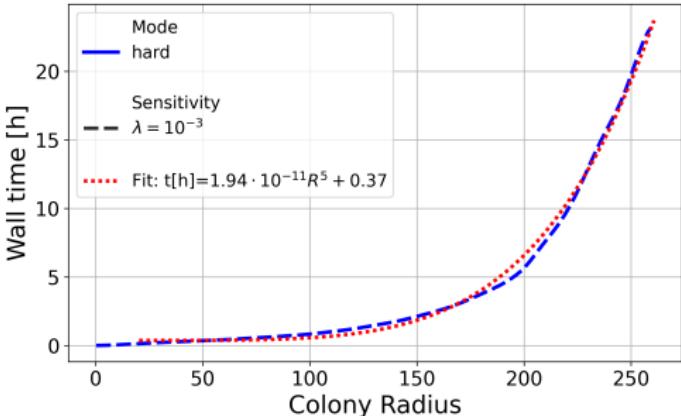
Scaling limits:

- Communication overhead dominates at high ranks
- Larger timesteps amortize communication costs

Maximum Colony Size (24h budget, 112 cores)

Hard Model:

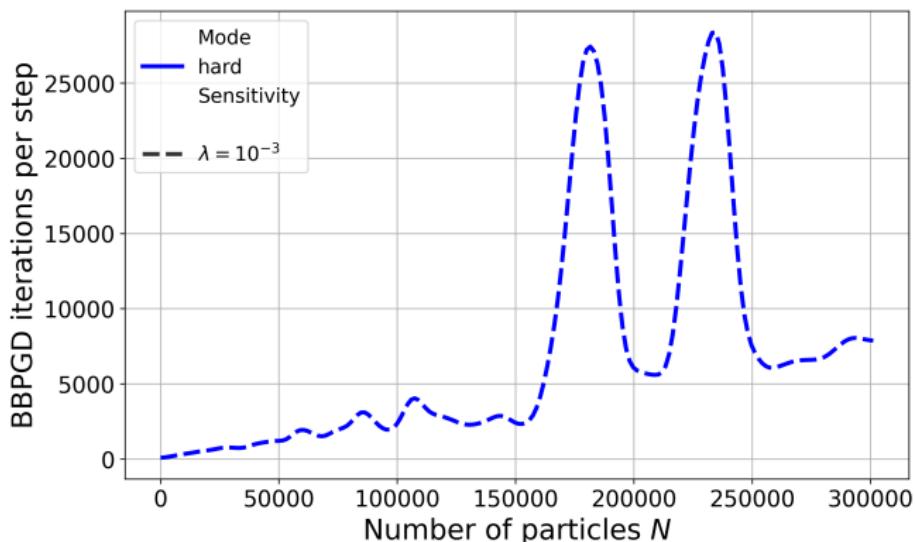
- $R_{\max} = 260$
- 301,116 cells
- Maintains accuracy
- Larger accessible scales



Scaling:

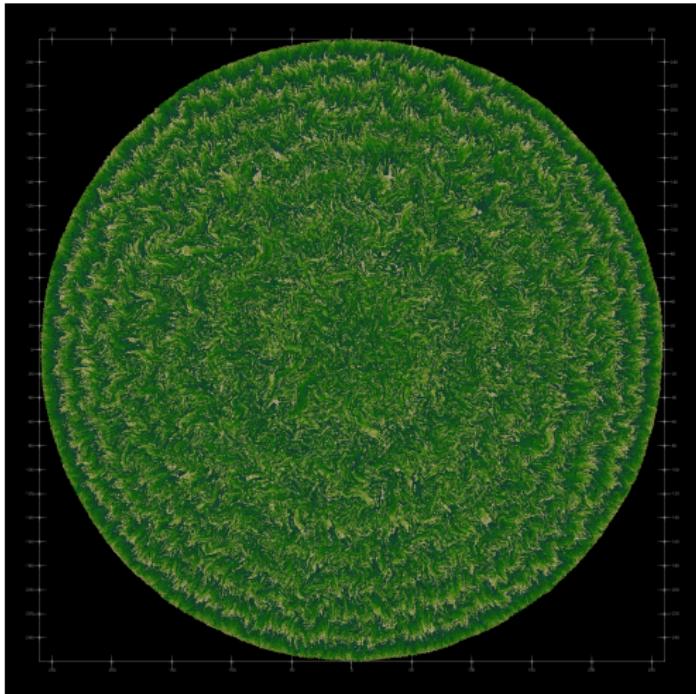
- $T[h] \propto R^5 \propto N^{2.5}$
- $\mathcal{O}(N)$ BBPGD iterations per step
- Many sparse matrix-vector products

BBPGD Solver Performance



- Dominates runtime (90% of computation)
- Scales roughly linearly with particle count

Maximum Colony Visualization ($R = 260$, 301k cells)



Outline

Model Comparison Summary

Criterion	Soft		Hard	
	Small Colonies	Large Colonies	Small Colonies	Large Colonies
<i>Biological Fidelity:</i>				
Ring formation	✓	✓	✓	✓
Packing density	✓	✗	✓	✓
Microdomain quality	✓	✗	✓	✓
<i>Performance:</i>				
Runtime	✓✓	✗✗*	✓	✓✓
Parallel scalability	✗	✓✓*	✗	✓✓
Implementation complexity	✓✓	✓✓	skull	skullskull

* Actual soft model performance degrades significantly due to timestep reduction.

When to use each Model?

Hard Model (Recommended for most applications)

Use when:

- Accurate stress distributions needed
- Cell-scale phenomena matter
- Realistic packing required

Advantages: Superior performance, physical accuracy

Soft Model (Limited use cases)

Only consider when:

- Preliminary exploration of small systems ($R \lesssim 50$)
- Only macroscopic patterns of interest

Warning: Artifacts limit applicability

Outline

Future Work

1. Hard Model Solver Improvements

- Leverage PETSc GPU support
- Warm-start BBPGD with previous solution

2. Soft Model Enhancements

- Consider overlap, in adaptive timestepping.
- Consider alternatives to prevent overlap
- Reduce communication overhead

3. More Applications

- Other cell shapes (soft bodies?)
- More complex models (Nutrient fields?, Outside forces?)
- Utilize 3D capabilities

Thank you for your attention!

Questions?

manuel.lerchner@tum.de

Code:

<https://github.com/manuellerchner/MicrobeGrowthSim-IDP>

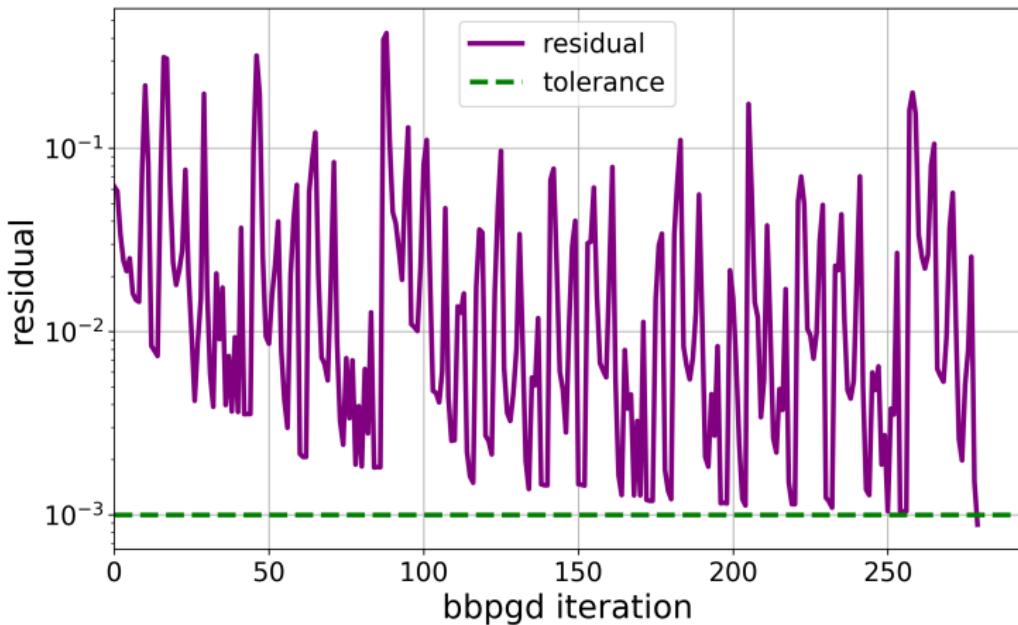
Supplementary materials:

<https://home.cit.tum.de/~ler/bacteria/>

References I

-  Bankole, F. A., Badu-Apraku, B., Salami, A. O., Falade, T. D. O., Bandyopadhyay, R., and Ortega-Beltran, A. (2023). Variation in the morphology and effector profiles of exserohilum turcicum isolates associated with the northern corn leaf blight of maize in nigeria. *BMC Plant Biology*, 23(1):386.
-  Weady, S., Palmer, B., Lamson, A., Kim, T., Farhadifar, R., and Shelley, M. J. (2024). Mechanics and morphology of proliferating cell collectives with self-inhibiting growth. *Phys. Rev. Lett.*, 133:158402.

Backup: BBPGD Convergence



Backup: BBPGD Energy Evolution

