A Distributed and Stochastic Algorithmic Framework for Active Matter

Sarah Cannon[1]
Joshua Daymude[2]
Daniel Goldman[3]
Shengkai Li[3]
Dana Randall[1]
Andrea Richa[2]
Will Savoie[3]

[1] Georgia Institute of Technology, CS
[2] Arizona State University, CS
[3] Georgia Institute of Technology, Physics

The Team

Dana Randall
CS, GT

Marta Arroyo

Sarah Cannon

Andrea Richa
CS, ASU

Josh Daymude

Dan Goldman
Physics, GT

Will Savoie

Shengkai Li

Active Matter

“Condensed matter in a fundamentally new nonequilibrium regime:

- The energy input takes place directly at the scale of each active particle and is thus homogeneously distributed through the bulk of the system, unlike sheared fluids or three-dimensional bulk granular matter, where the forcing is applied at the boundaries.
- Self-propelled motion, unlike sedimentation, is force free: The forces that the particle and fluid exert on each other cancel.
- The direction of self-propelled motion is set by the orientation of the particle itself, not fixed by an external field”

[S. Ramaswamy. The mechanics and statistics of active matter. Annual Review of Condensed Matter Physics, 1(1):323–345, 2010]

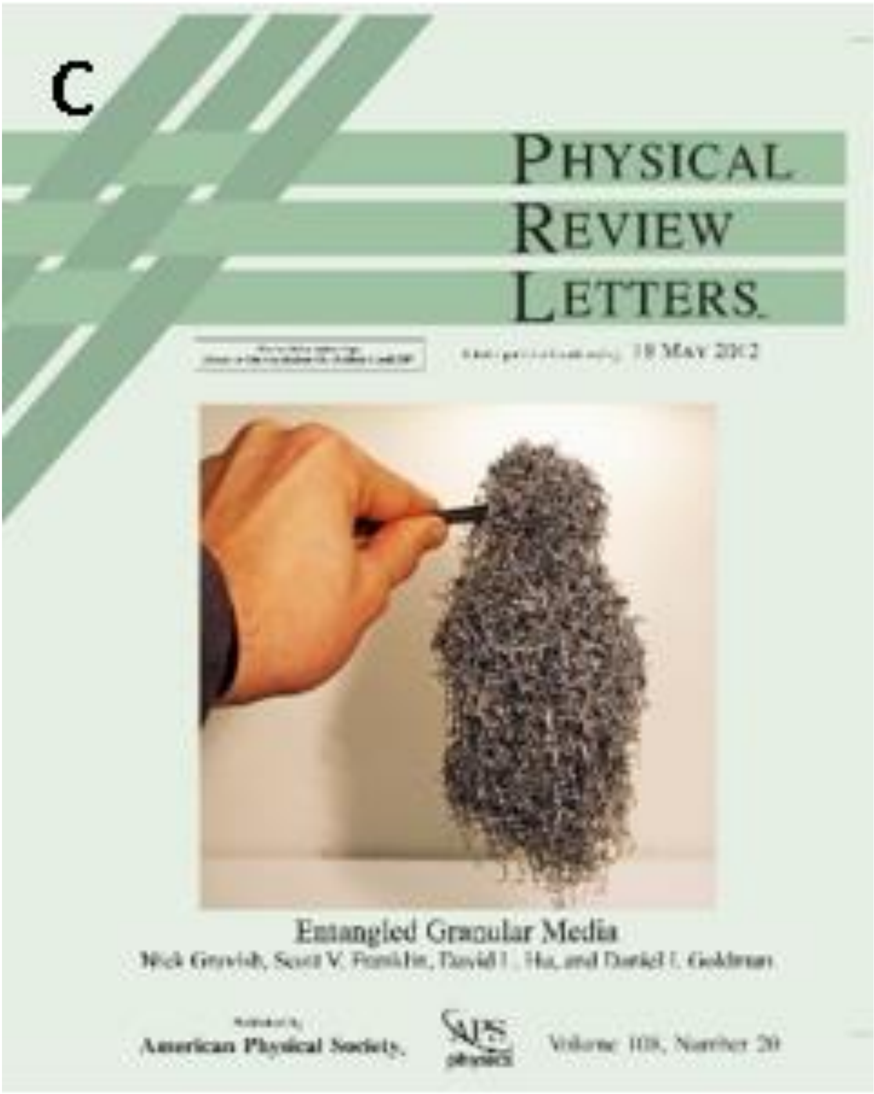
Programmable Active Matter

- Swarm robotics systems of programmable particles (smarticles) with close analogies to physical systems.
  - Smarticles are small in scale, ranging in size from millimeters to centimeters,
  - crowded (i.e., dense) environments
  - behave as active matter

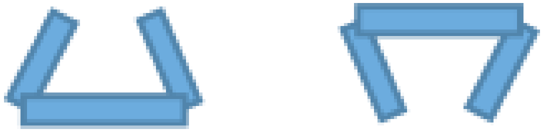
- "task-oriented" approach:
  - start from desired macroscopic emergent collective behavior, and develop the distributed and stochastic algorithmic underpinnings that each robot (smarticle) will run

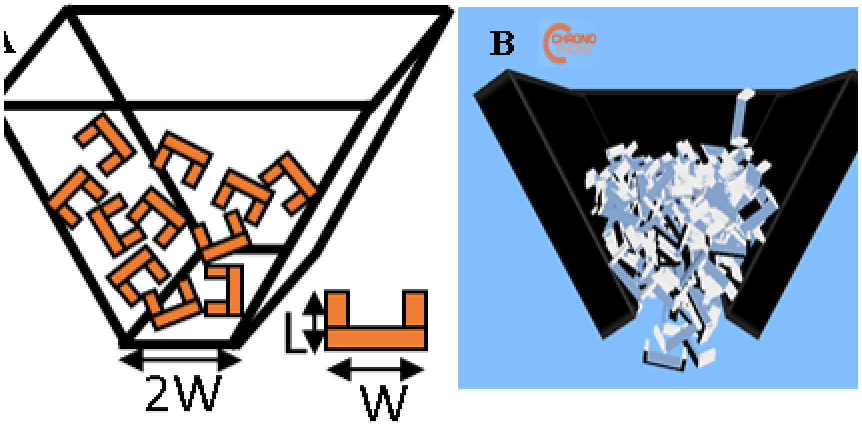
- provide the understanding for yet unexplored collective and emergent systems.

U-shaped Smarticles



Goals: control entanglement or jamming by varying angles of the "U"



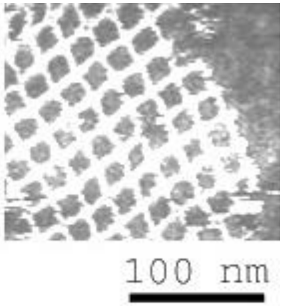


Other Programmable Matter

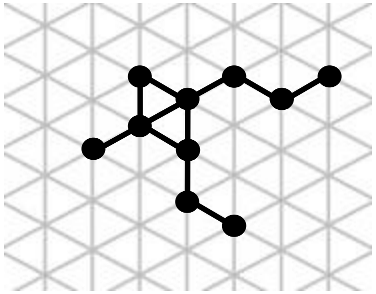
- Modular and swarm robotics
- DNA computing: not self-propelled
- Smart materials

Kilobots:

DNA self-assembly:



- **Self-organizing particle systems:** Collection of simple computational elements that self-organize to solve system-wide problems of movement, configuration, and coordination
  - constant memory
  - fully distributed, local algorithms
  - Amoebot model



[Derakhshandeh, Gmyr, R, Schedeiler, Strothman]

AitF Collaboration: Goals

- Understand minimal computational requirements for certain "tasks"
- Learn to program the ensemble to control emergent collective behavior
- Remove centralized control by having the particles locally respond to their environment
- Provide a **stochastic distributed algorithmic framework** for (programmable) active matter

Collective Behaviors

1. Compression

2. Bridging

3. Alignment

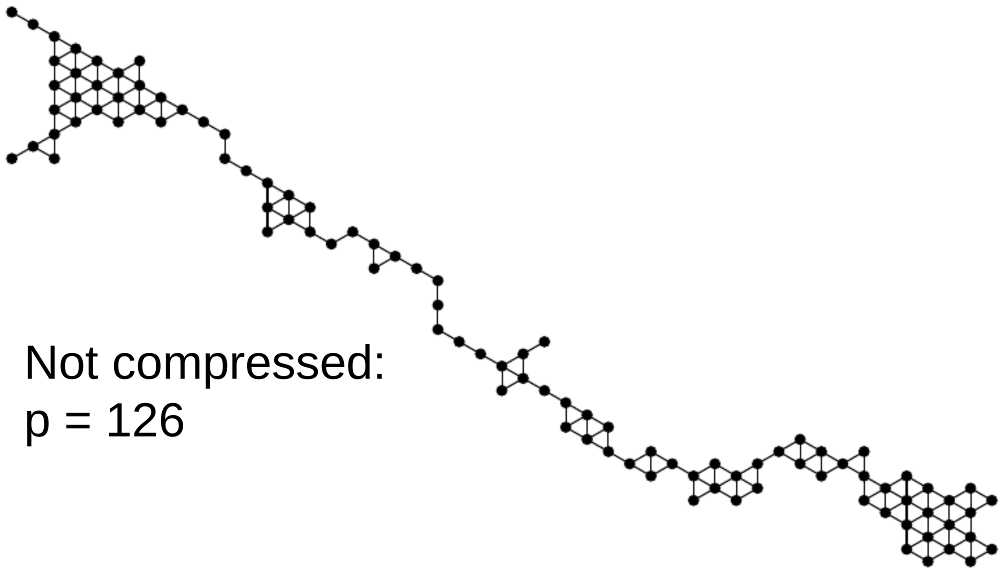
4. Locomotion

Action 1: Compression

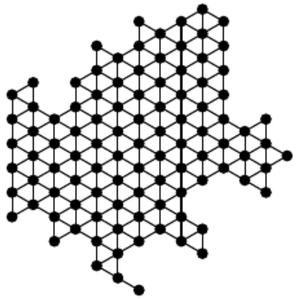
<u>**Q**</u>: Under local, distributed rules, can a connected set of particles "gather" or "compress" to reduce their perimeter?

<u>**Def'n**</u>: A particle configuration is **α-compressed** if its perimeter is at most α times the **minimum perimeter** for these particles.

$$p(\sigma) = 3n - e(\sigma) - 3$$



Not compressed:
p = 126



Compressed:
p = 51

Compression Algorithm

[Cannon, Daymude, Randall, Richa, PODC '16]:

A distributed, stochastic algorithm based on the amoebot model that:

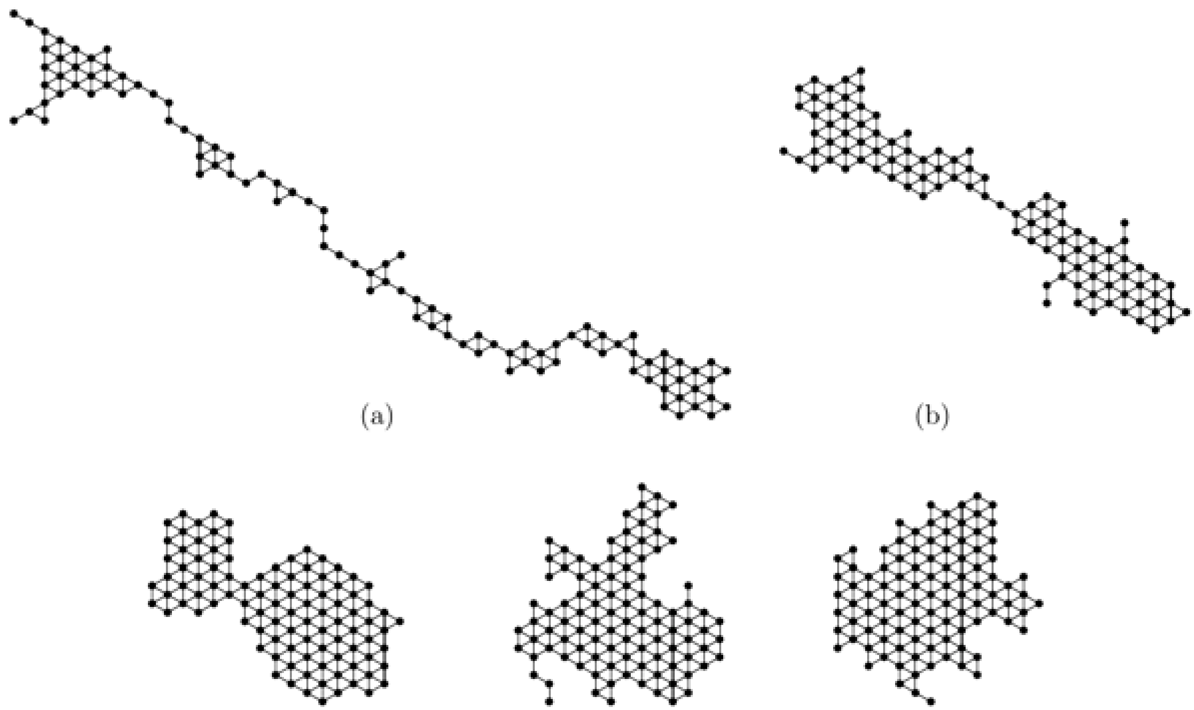
1. Maintains simply connected configurations in the triangular lattice
2. Uses Poisson clocks to find potential moves asynchronously
3. Accepts moves with Metropolis prob. to converge to $\pi(\sigma) = \lambda^{e(\sigma)} / Z$

$\lambda = 4$

100 particles after:

a) 1 million
b) 2 million
c) 3 million
d) 4 million
e) 5 million

iterations.



(a) (b) (c) (d) (e)

Compression Algorithm

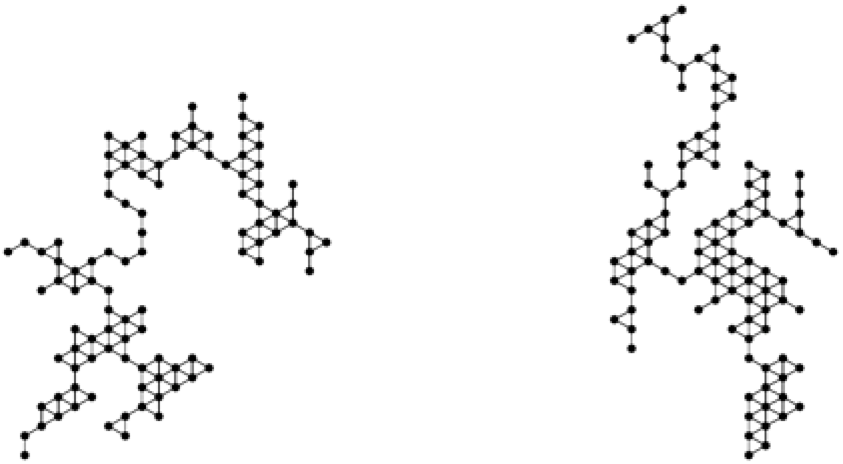
[Cannon, Daymude, Randall, Richa, PODC '16]:

A distributed, stochastic algorithm based on the amoebot model that:

1. Maintains simply connected configurations in the triangular lattice
2. Uses Poisson clocks to find potential moves asynchronously
3. Accepts moves with Metropolis probs to converge to $\pi(\sigma) = \lambda^{e(\sigma)} / Z$

$\lambda = 2$

100 particles after:
a) 10 million
b) 20 million
iterations.



(a) (b)

No compression.

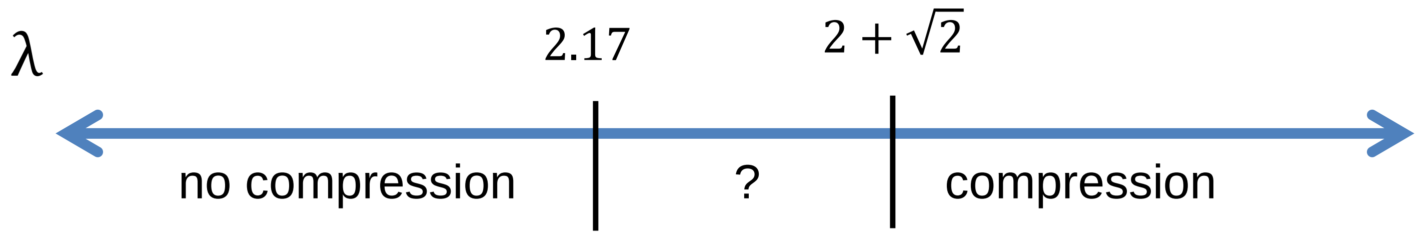
Compression: Theorems

[CDRR'16]

<u>Def'n:</u> A particle configuration is **α-compressed** if its perimeter is at most α times the **minimum perimeter** for these particles.

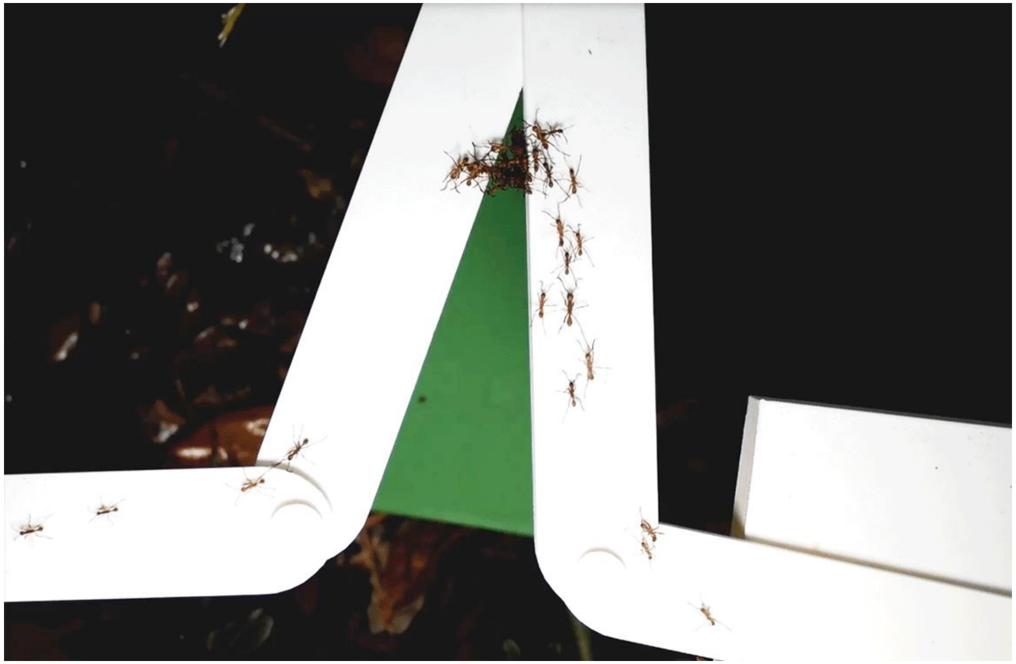
<u>Thm:</u> When $\lambda > 2 + \sqrt{2}$, there exists $\alpha = \alpha(\lambda)$ s.t. particles are α-compressed at stationarity with all but an exp. small probability. (When $\lambda = 4$, $\alpha = 9$.)

<u>Thm:</u> When $\lambda < 2.17$, for *any* $\alpha > 1$, the probability that particles are α-compressed at stationarity is exponentially small.



Note: Expansion works similarly for smaller λ. [CDRR'16]

Action 2: Bridging



[Lutz and Reid '15]

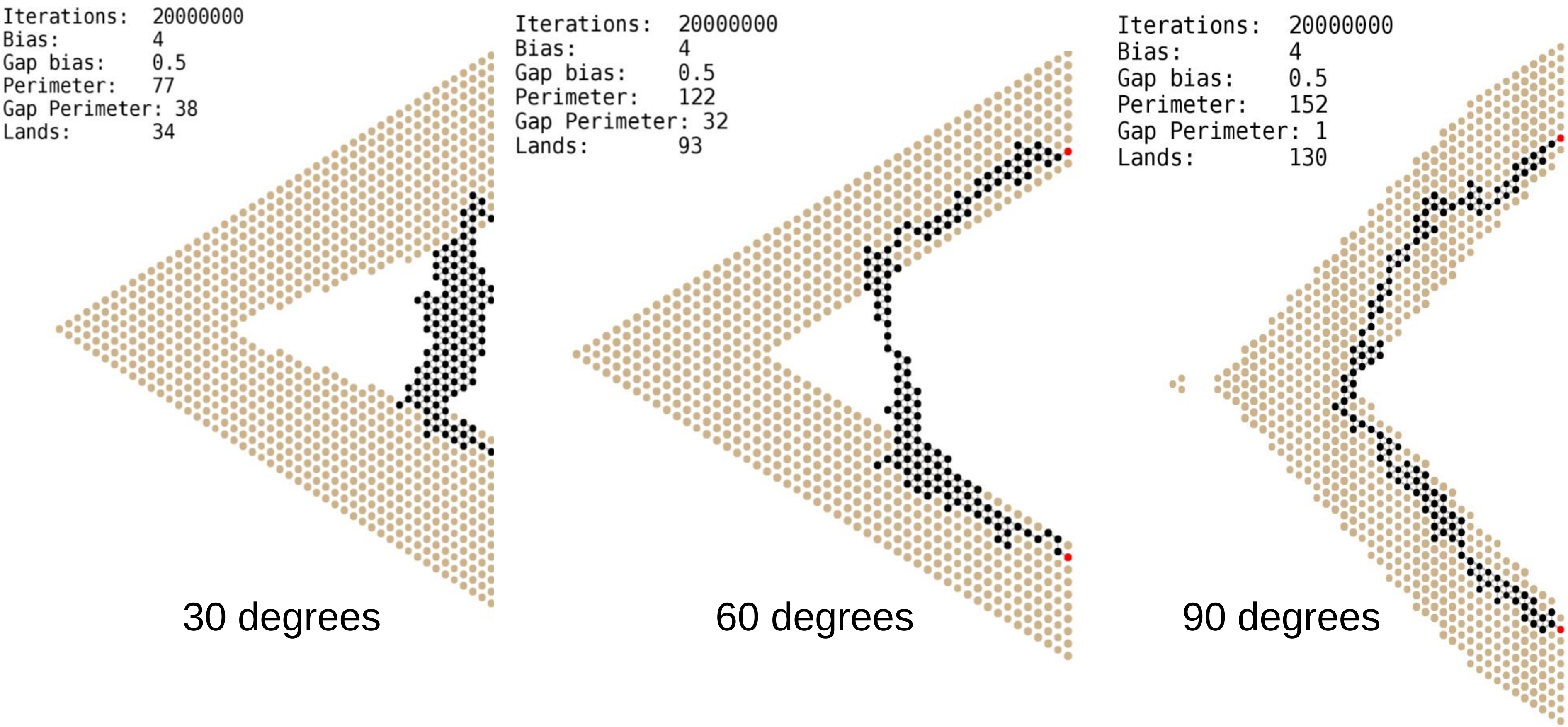
- Army ants construct living bridges to minimize the number of nonworking members of the colony.
- Long bridges are more precarious.

[Arroyo, Cannon, Daymude, Randall, Richa '17]

- Use similar local compression rules favoring neighbors.
- Penalize particles in the gap on the perimeter (for poorer stability).

Bridging

For a fixed angle, the thickness and position of the bridge depends on the clustering and gap parameters:



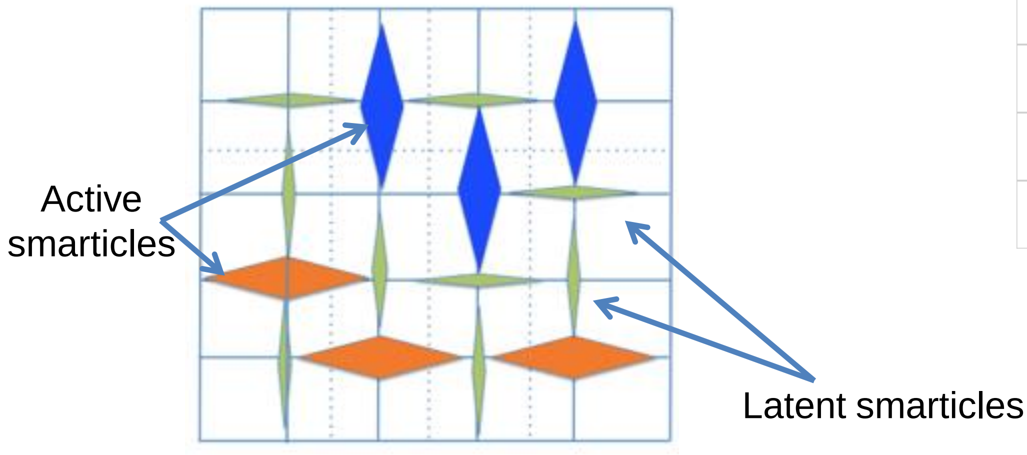
30 degrees

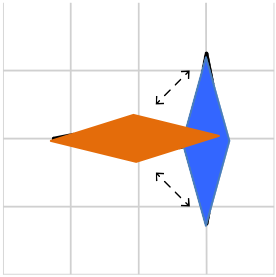
60 degrees

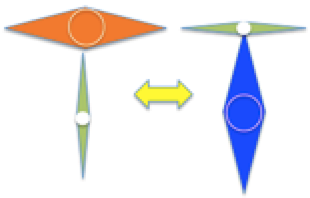
90 degrees

Action 3: Alignment

Smarticles confined to $\mathbf{Z}^2$ that elongate or flatten as they move.





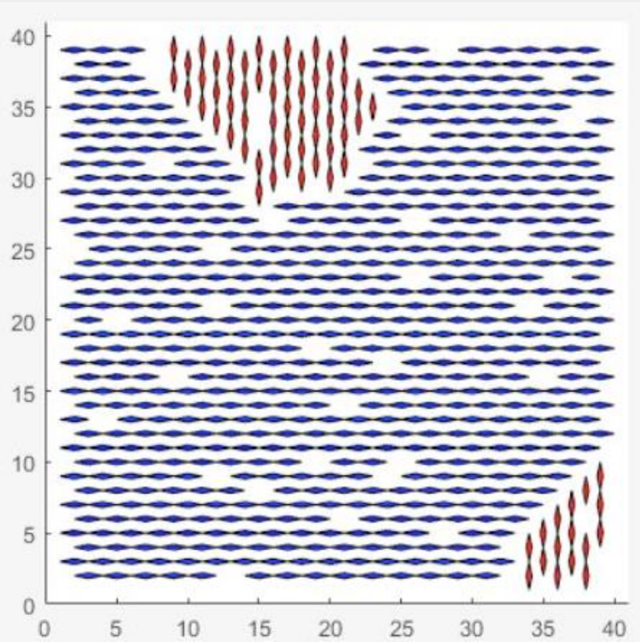


<u>Thm</u>: We get large regions of alignment with mostly vertical or horizontal smarticles.

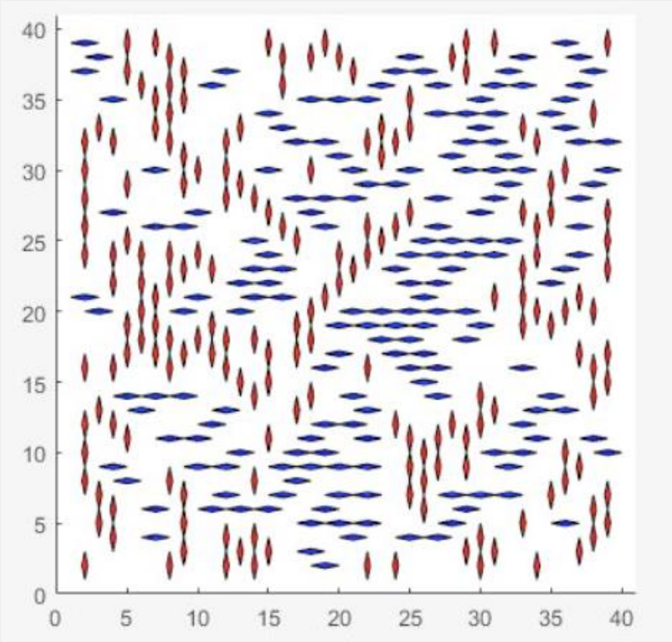
<u>Conj</u>: Also get alignment with limited latent smarticles.

(**Partial proofs*)

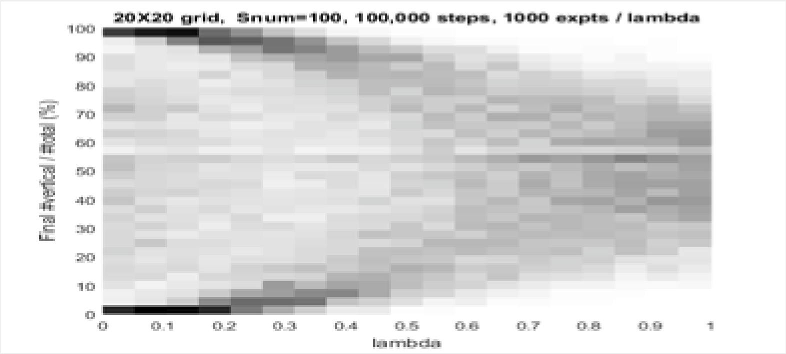
Alignment



Large λ



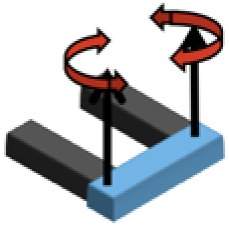
Small λ

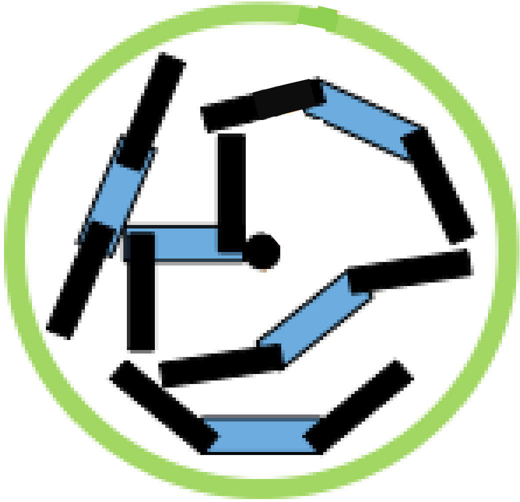


Action 4: Locomotion

[GLS], [CDGLRRS] *(*in progress)*

A robot made of robots





SuperSmarticles

Confine several smarticles in a ring.

- One smarticle: no locomotion
- Allow them to interact through movements: Brownian motion
- Allow interaction, with one inactive smarticle: Brownian motion w/ drift (directed toward inactive smarticle)

Locomotion

Next Steps

1. Composite algorithms that automate transitions in response to the environment.

   Ex. **Foraging**:
   - Use compression around a food source until it's depleted;
   - Transition to expansion when depleted to find a new source;
   - Repeat

2. Build prototypes to refine algorithms (alignment, compression, bridging)
   - New challenges: real space, imperfect interactions, etc.
   - Refines types of interactions between particles.

3. Explore algorithmic foundations underlying:
   - Locomotion
   - "Jamming"
   - Entanglement

Thank you!