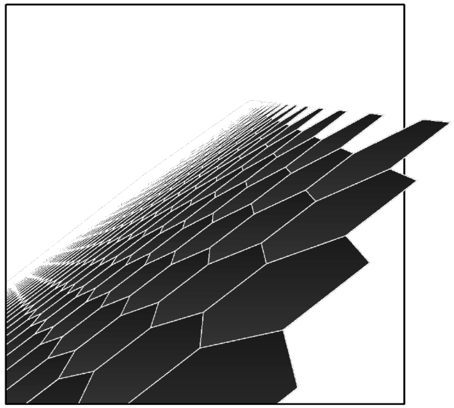
# Towards self-reproducing robots

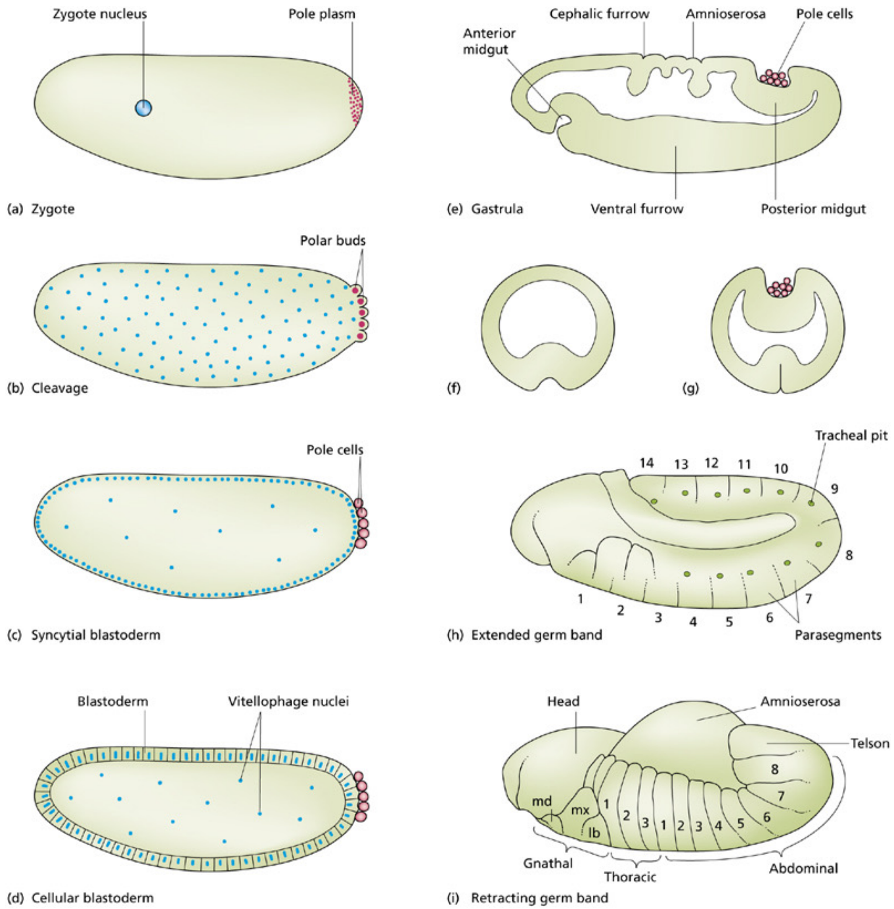


# What you will learn in this class

- Types of self-reproduction
- Self-assembly by mobile robots
- Programmable self-assembly
- 2D multi-cellular robots: *in silico* evolution and hardware assembly
- 3D multi-cellular robots: hardware design and assembly
- Artificial ontogenesis *in silico*
- *In silico* evolution, *in vivo* self-assembly of multicellular organisms
- *In vivo* kinematic self-replication

# Self-reproduction by growth

Organisms self-reproduce by a mechanism of cell division, specialization, and migration




Early development of *Drosophila* [Slack 2006]

# Self-reproduction by self-assembly

At sub-cellular level, self-replication happens by self-assembly of existing materials (see first lecture on “From DNA to Proteins”)

Von Neumann (1966), *Theory of self-reproducing automata*, A.E. Burks (Editor), University of Illinois Press

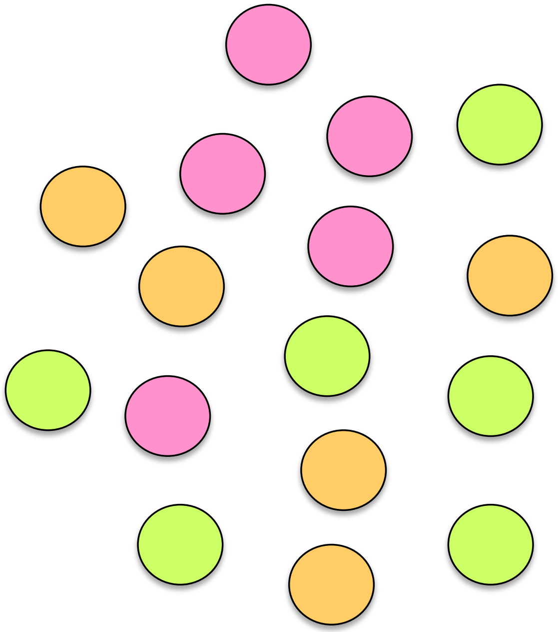
“Self-reproducing robots by self-assembly are possible if a reservoir of specialized cells is available in the environment”

He considered a floating environment with millions of elementary “cells” of approximately 20 types:

- sensor cell
- muscle cell
- cutting cell
- fusing cell
- neuron-like cell
- …

## *2 requirements for self-assembly*

A population
of diverse cells

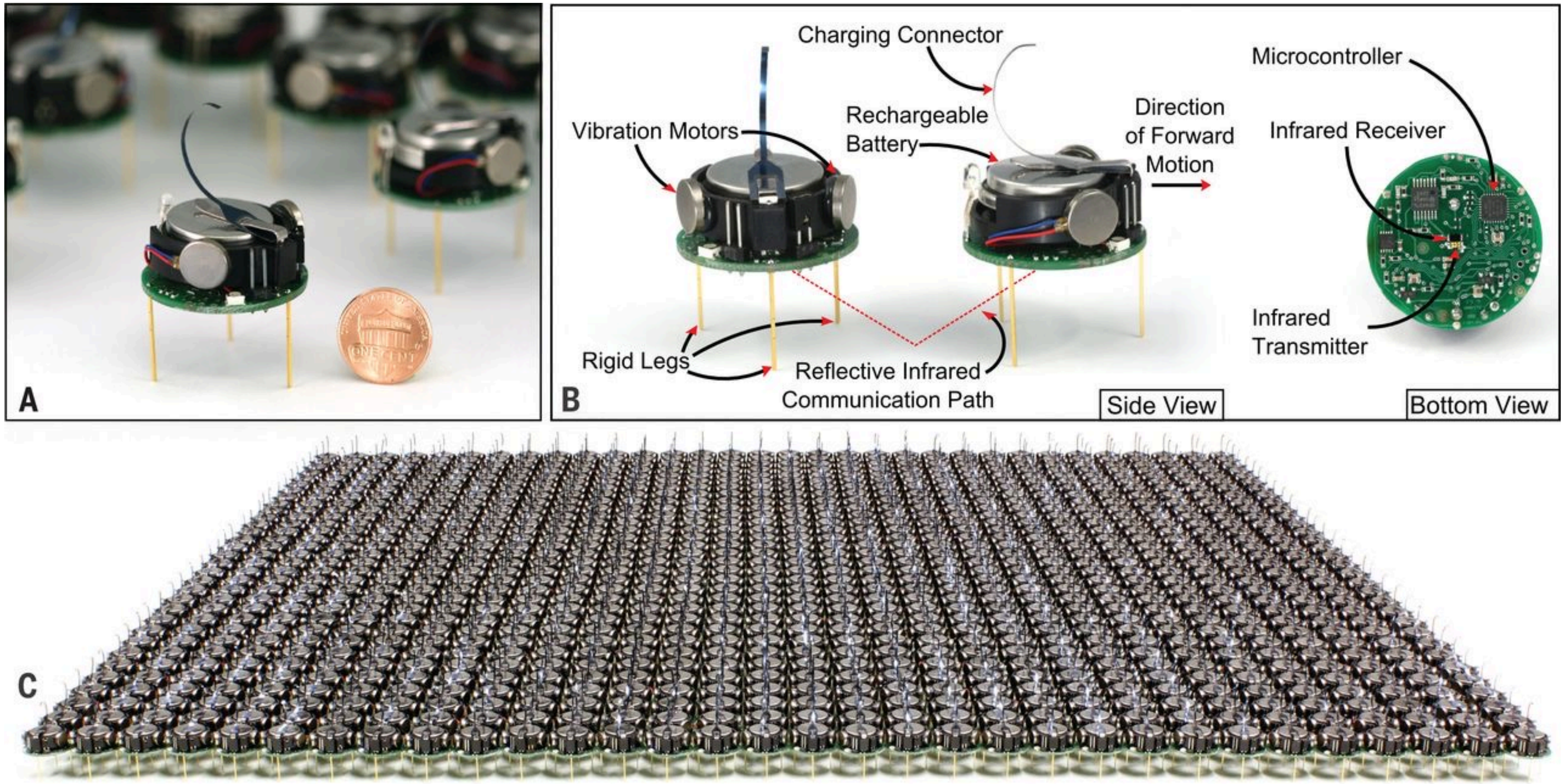


Intrinsic and/or extrinsic
energy potential

# *The impact of morphological diversity*

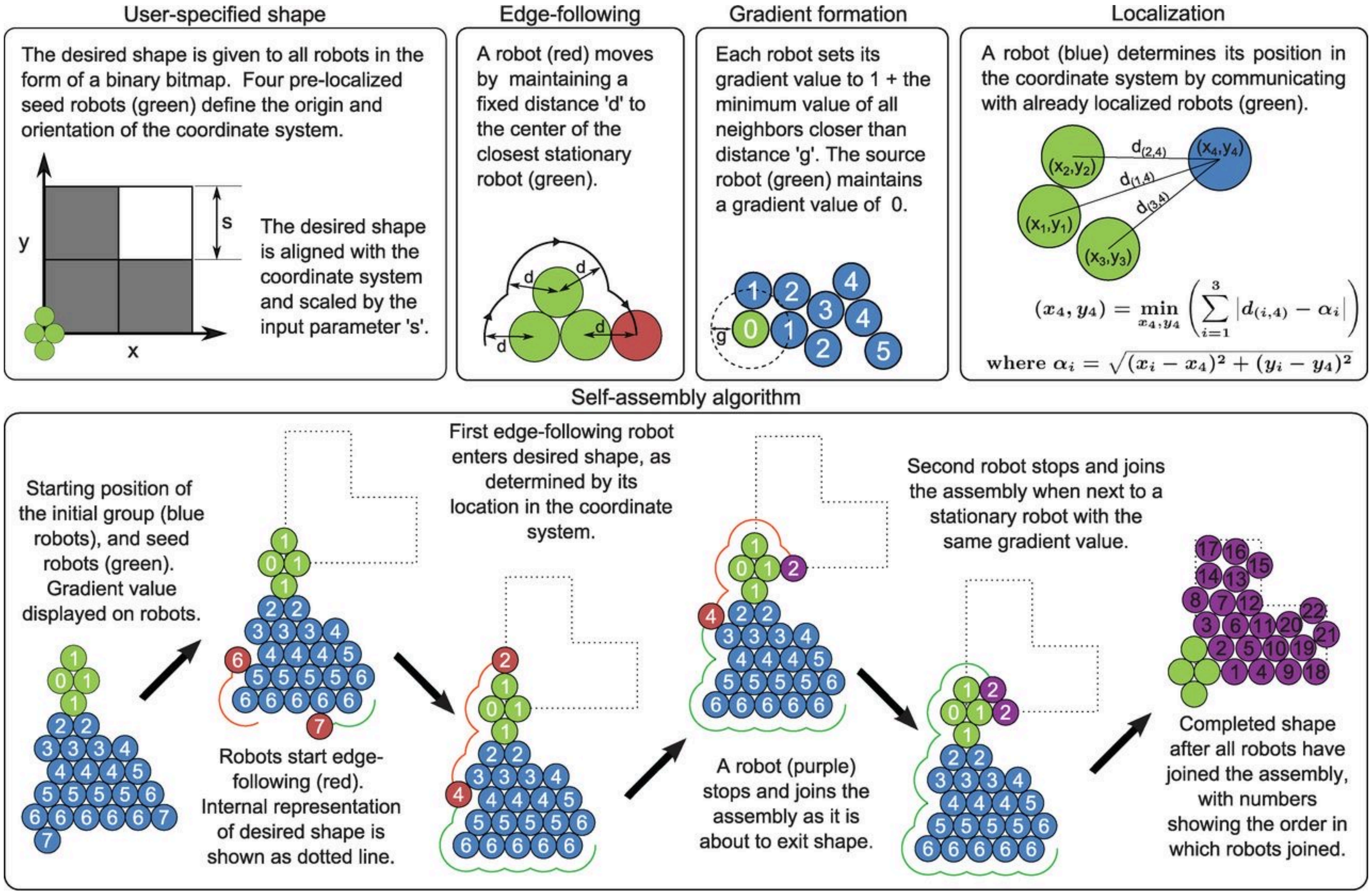

Penrose, L. S. & Penrose, R. *Nature* **179**, 1183 (1957).

# *Self-assembling Kilobots*




Rubenstein et al. (2014) *Science*

# Self-assembly Algorithm

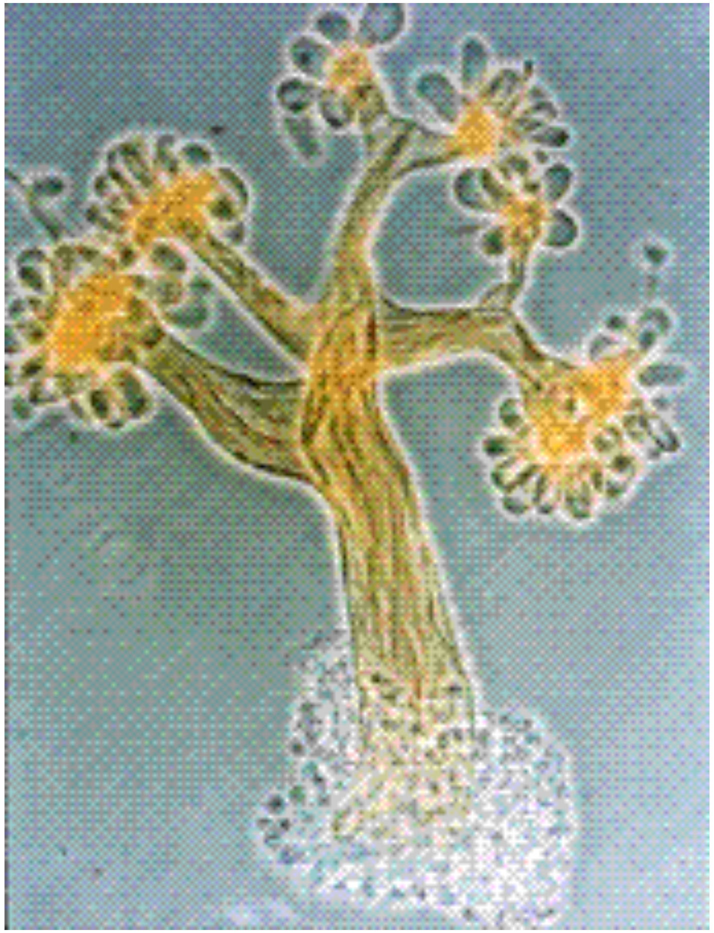


# Self-Organizing Systems Research Group

Harvard University
School of Engineering and Applied Sciences
Wyss Institute for Biologically Inspired Engineering

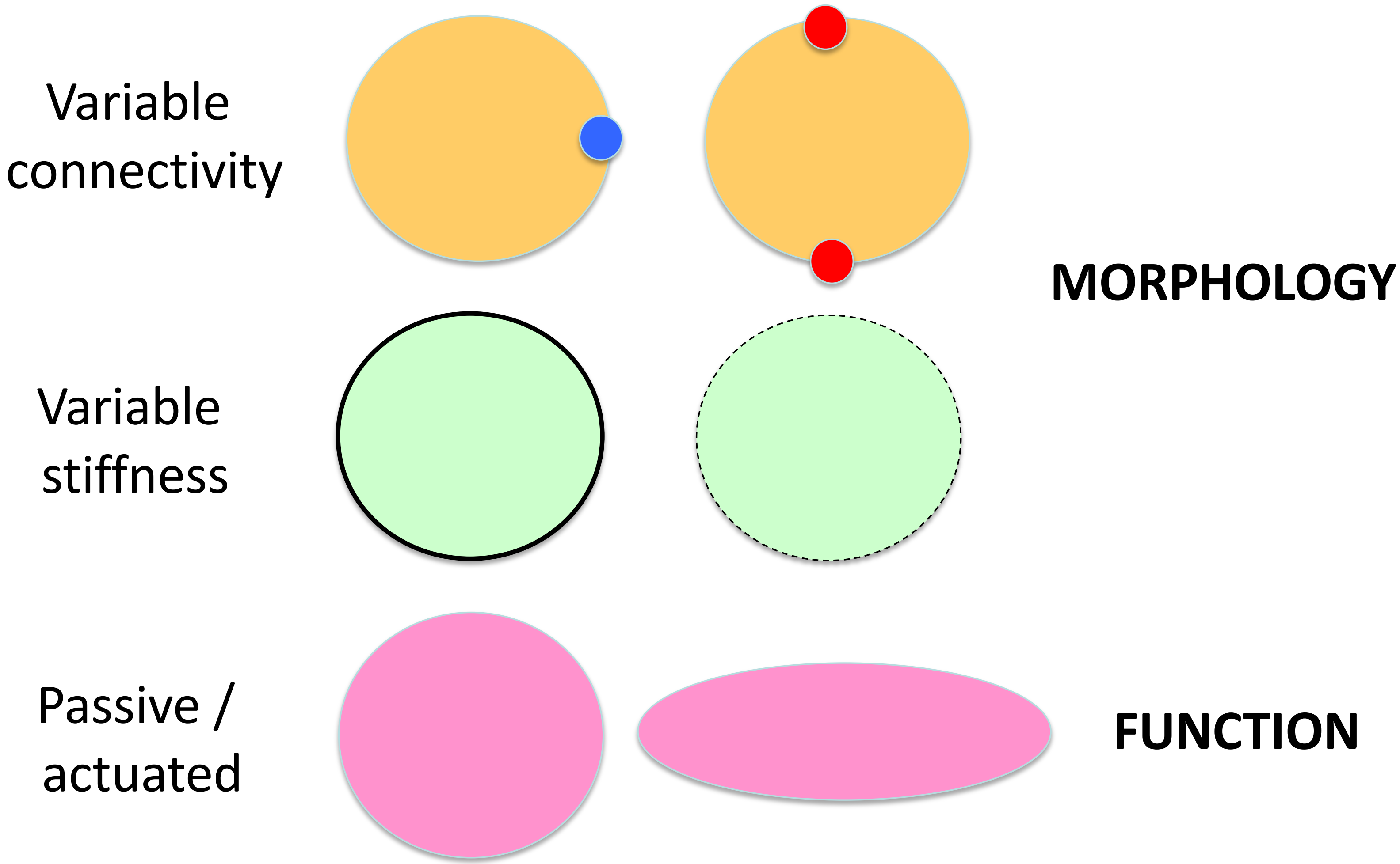
## *Multicellular Biological Organisms*


Cyanobacteria

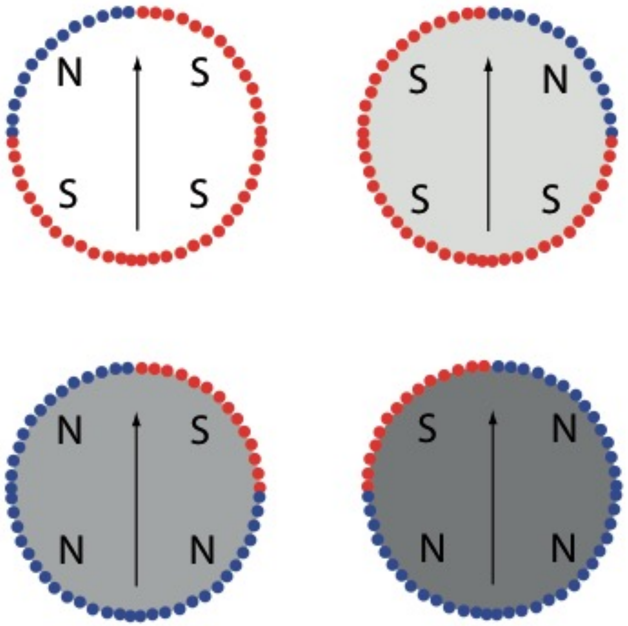


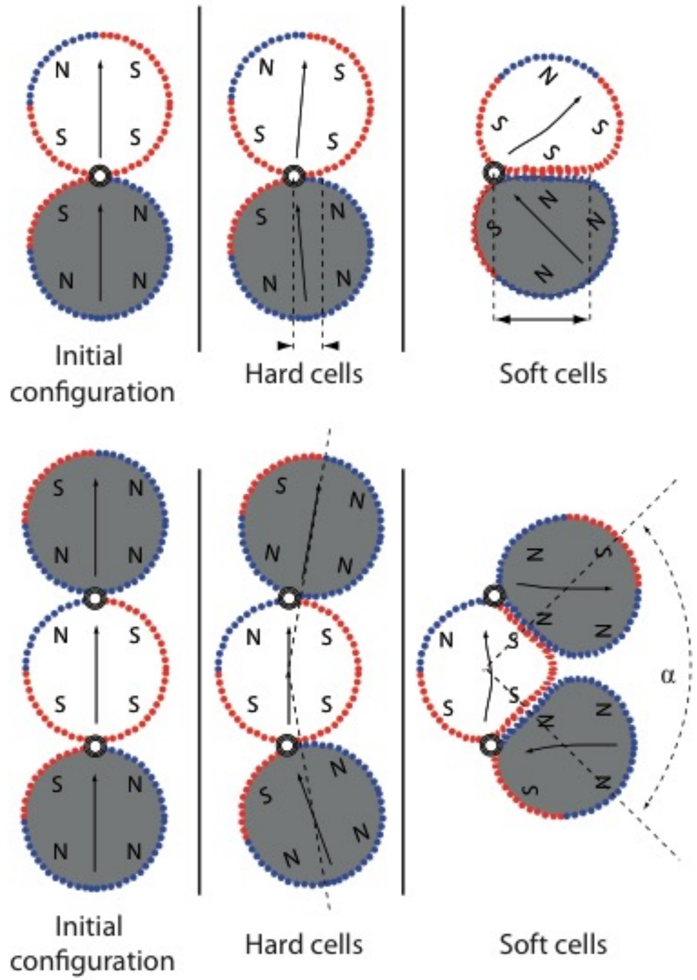
Myxobacteria

# *Cell diversity*



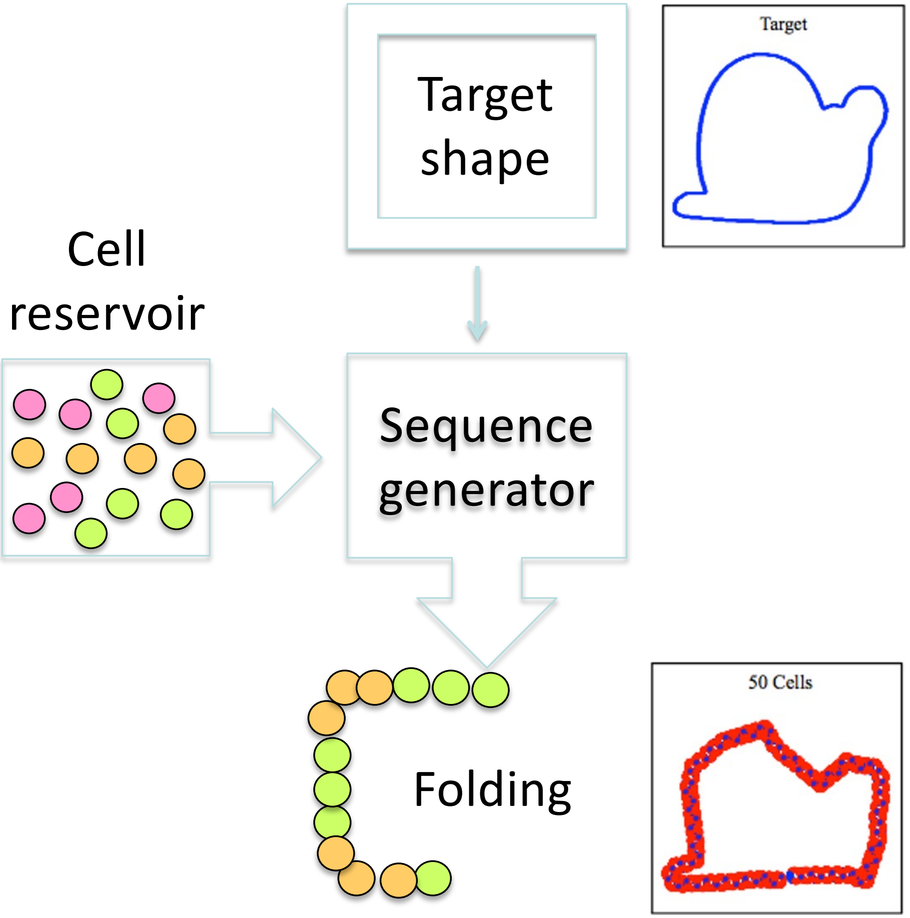
# *Softness affects folding angle*



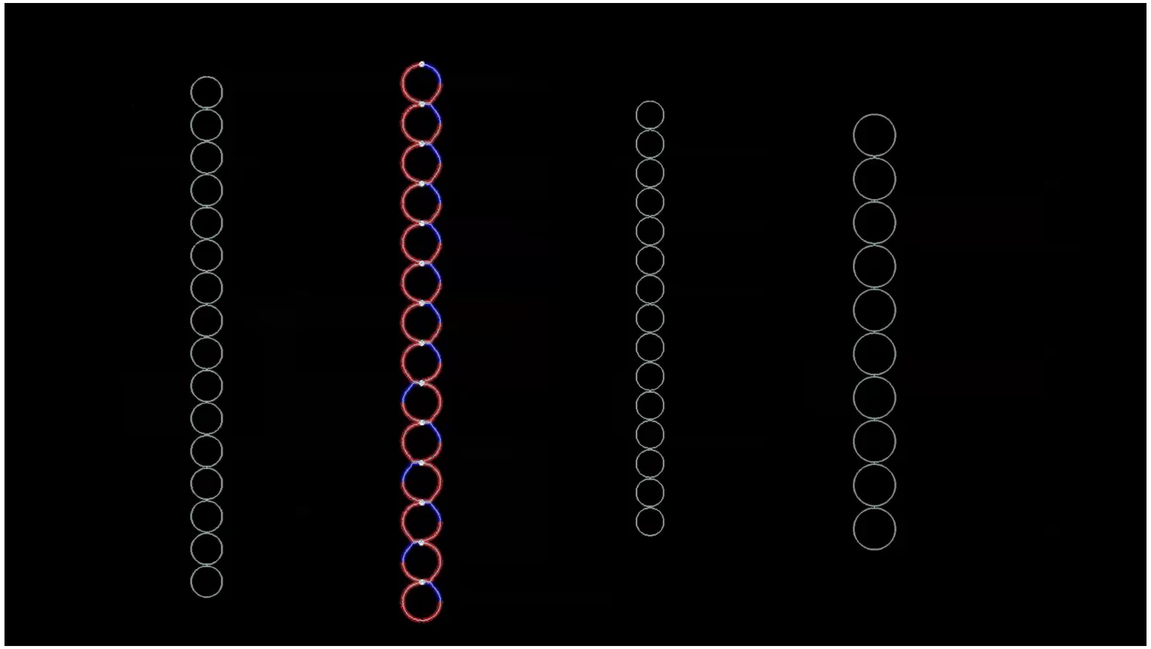

Germann et al (2014)
*Soft Robotics*

# Programmable Self-Assembly



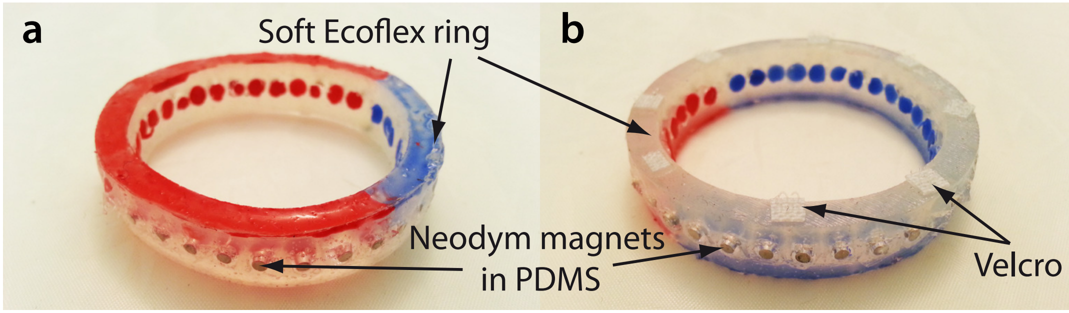

S. Griffith (2004), Growing Machines,
MIT PhD thesis

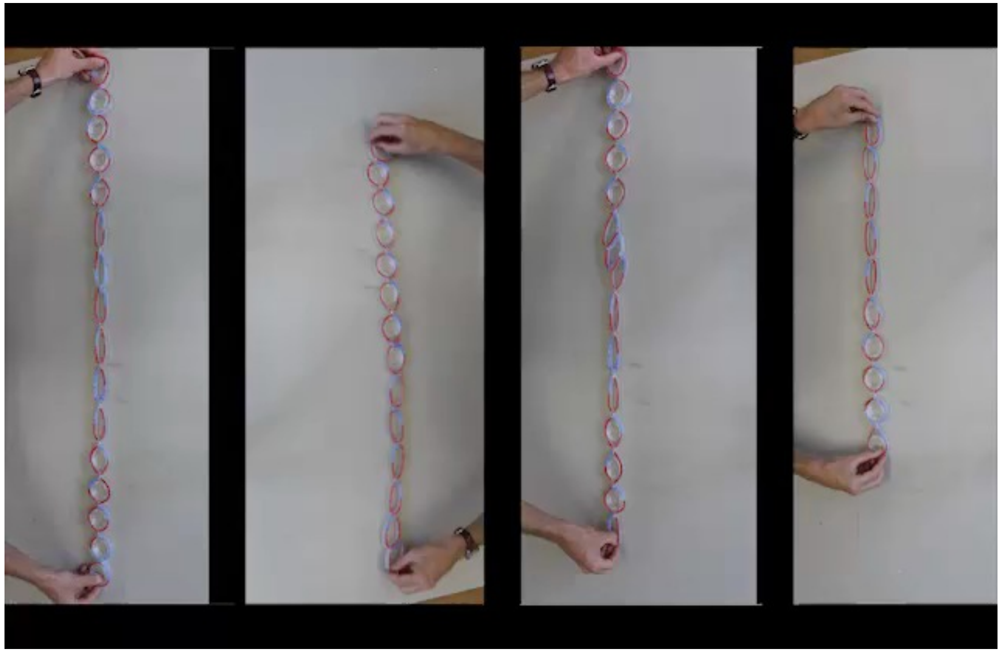
# Programmable self-assembly in silico



Germann et al (2014) *Soft Robotics*

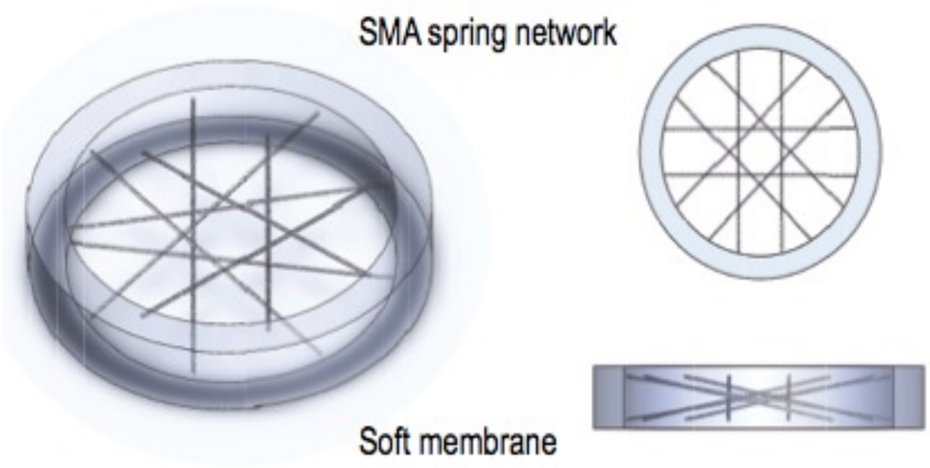
# Programmable self-assembly in hardware

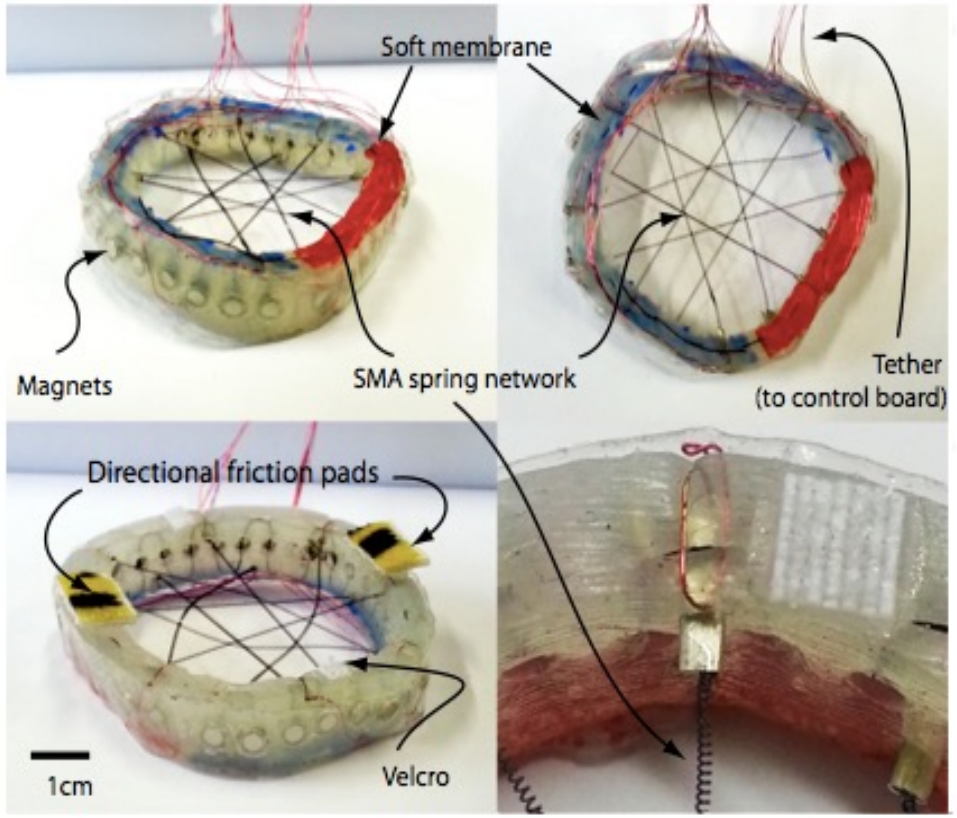


Germann et al (2014) *Soft Robotics*

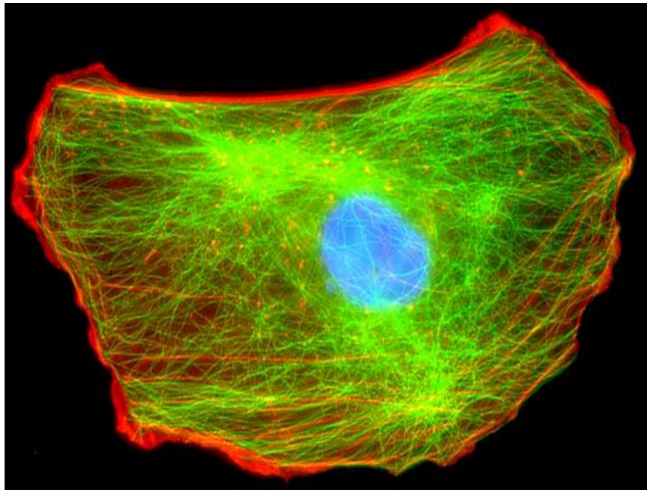
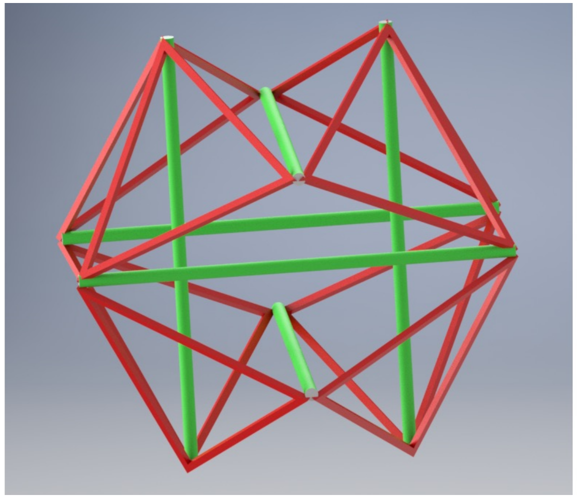
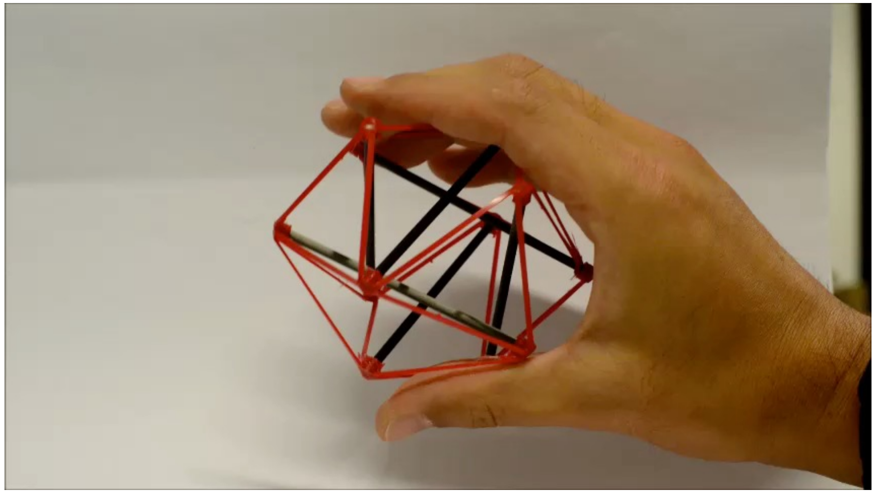
# Muscle cells



# 2D multicellular worm

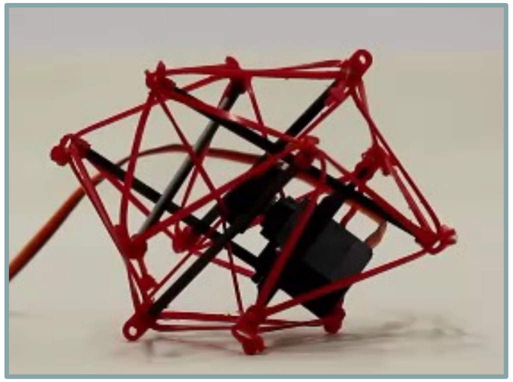
# From 2D to 3D: Tensegrity robotic cells

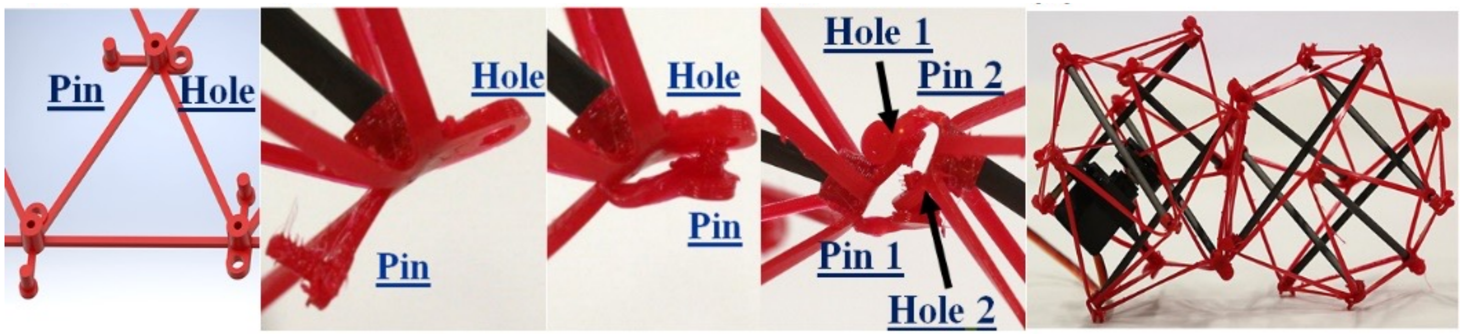
| | | |
|---|---|---|
| Actin filaments |  | Pre-stretched cables |
| Microtubules |  | Struts |

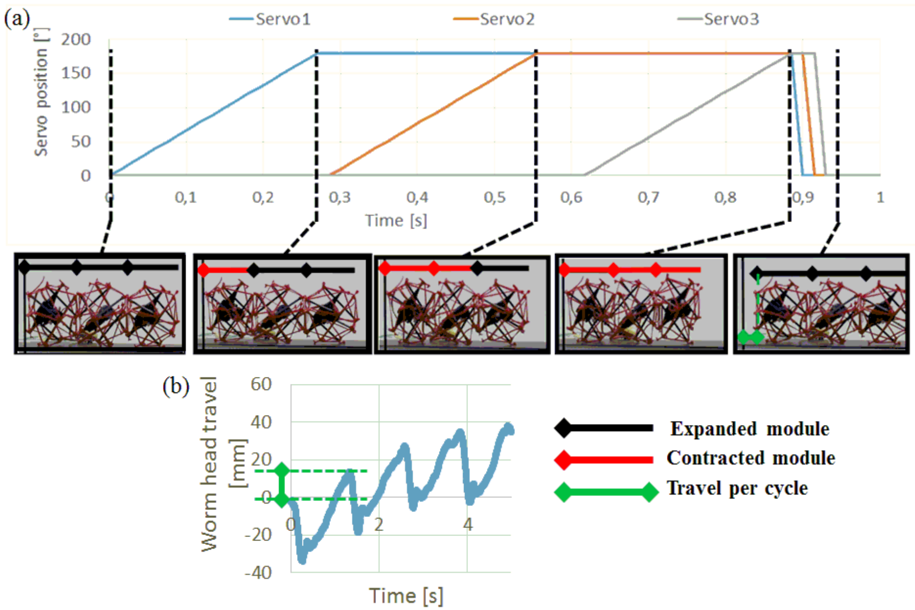
# 3D multicellular worm

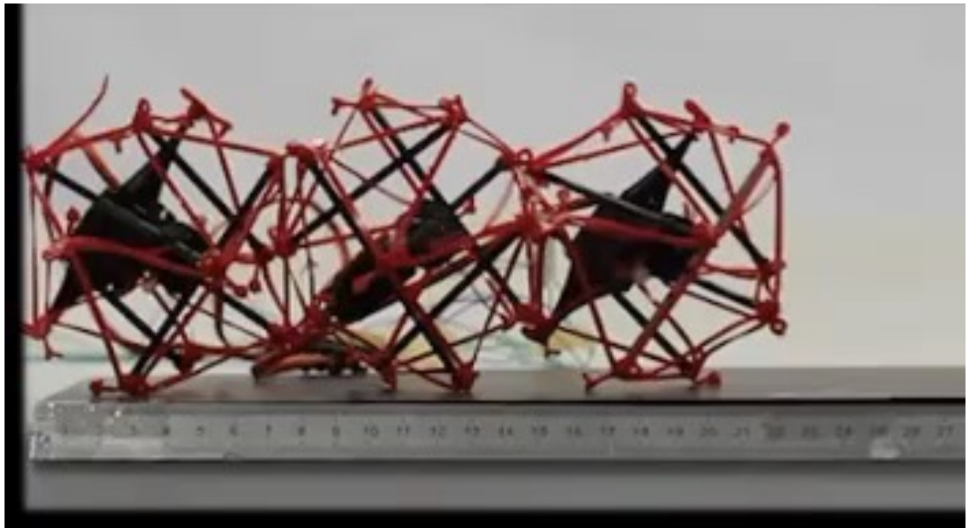
A contracting module



3D printed hole-pin latching

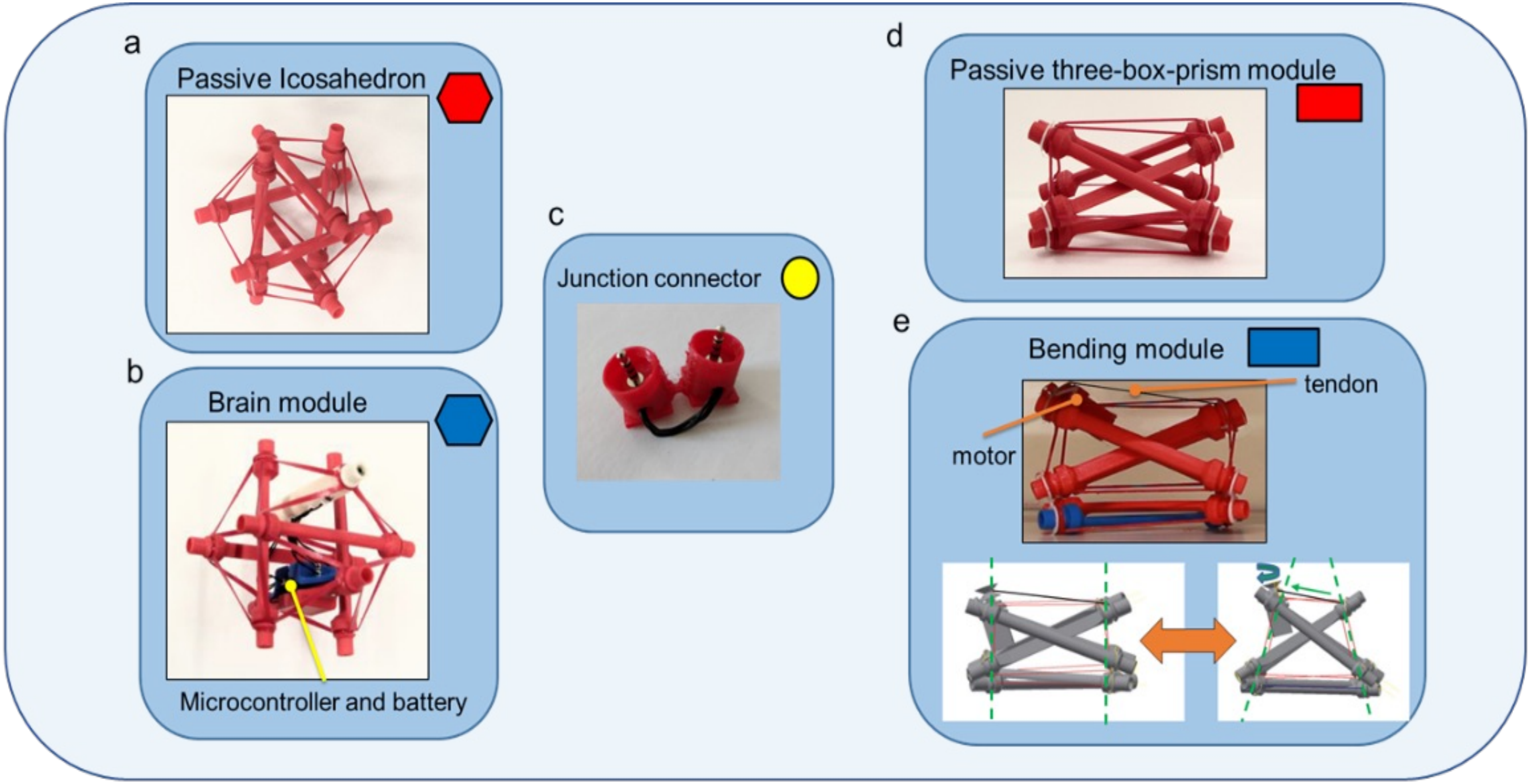


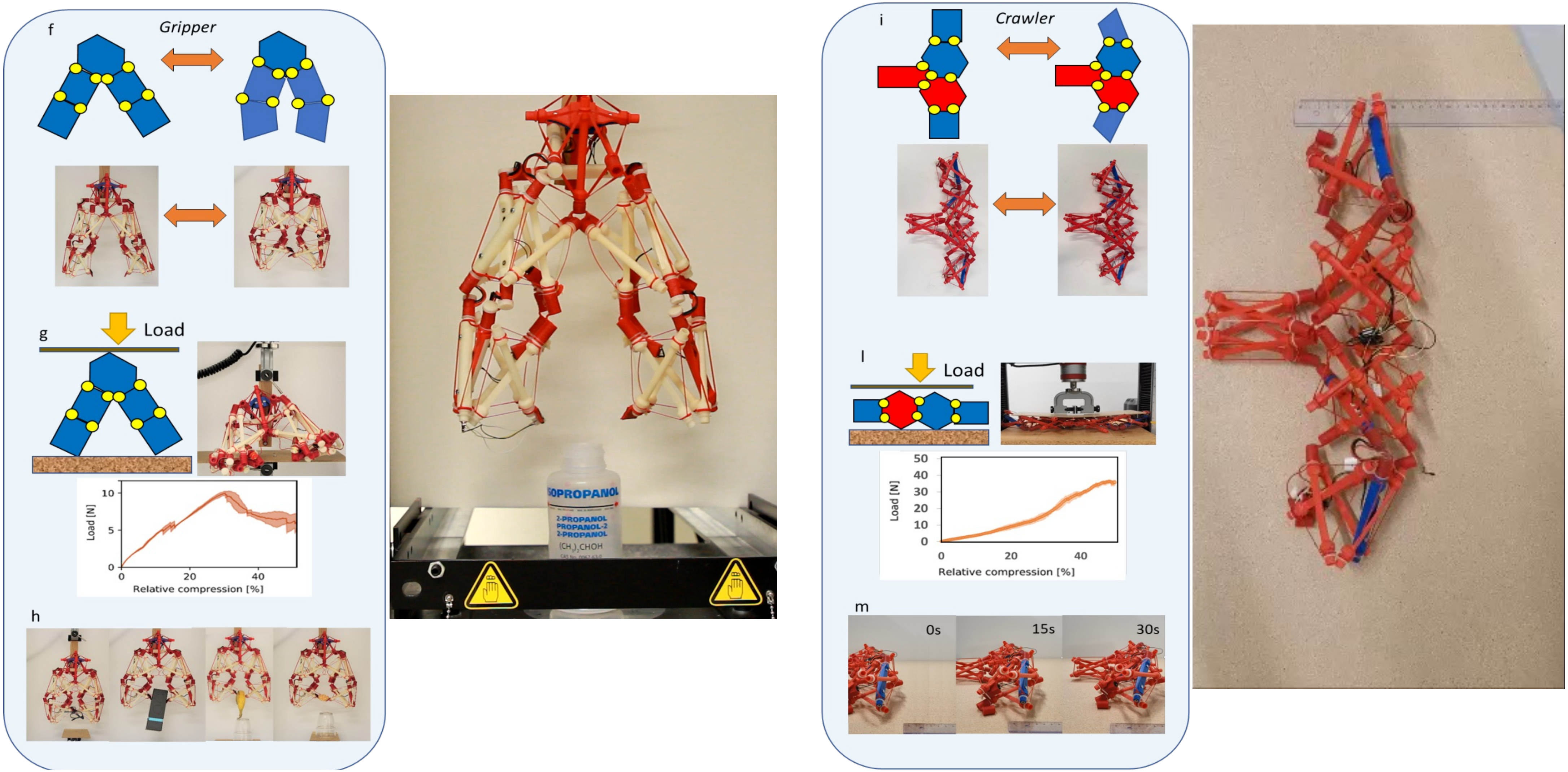



D. Zappetti, S. Mintchev, J. Shintake, e D. Floreano (2017) «Bio-inspired Tensegrity Soft Modular Robots», in *Biomimetic and Biohybrid Systems*, 497–508

# Different types of tensegrity robotic cells

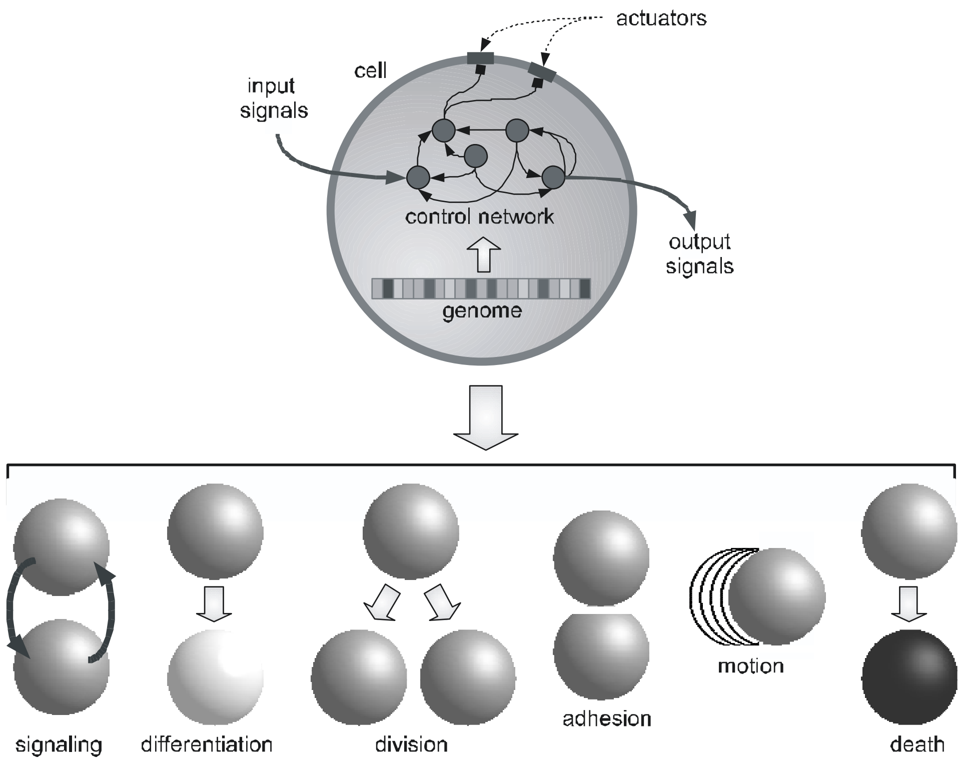


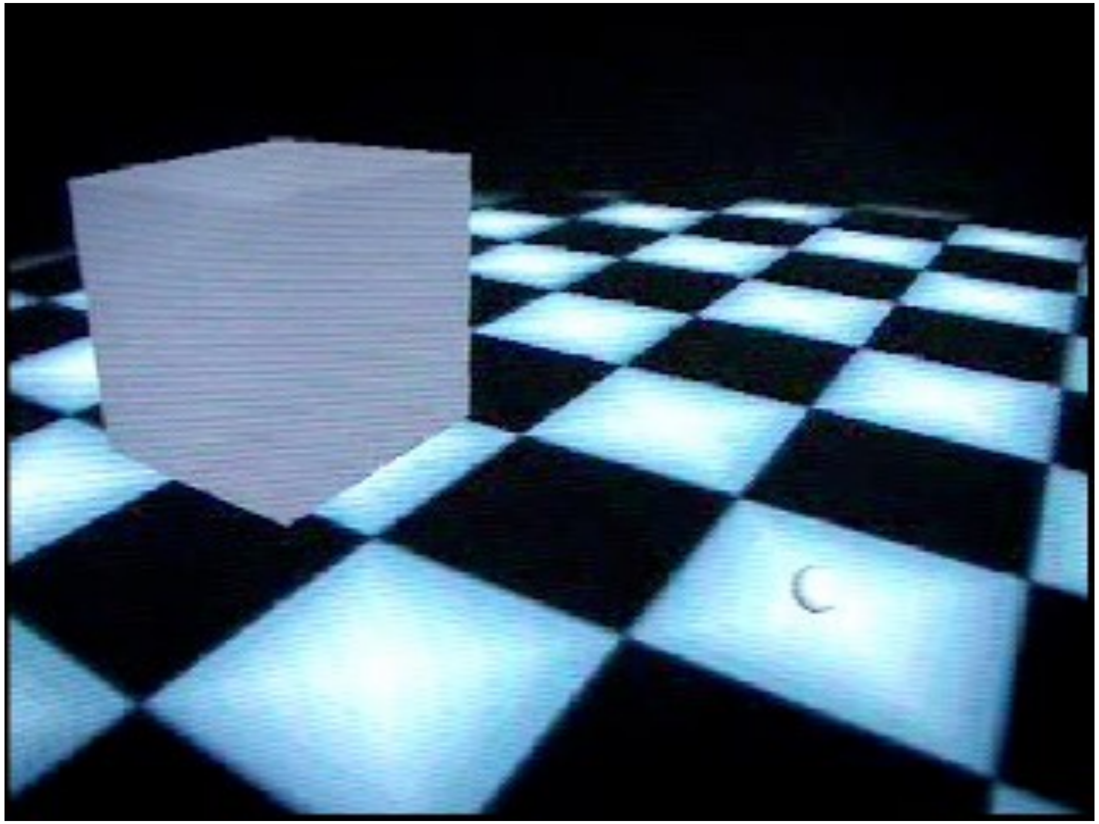
# Multicellular tensegrity robots



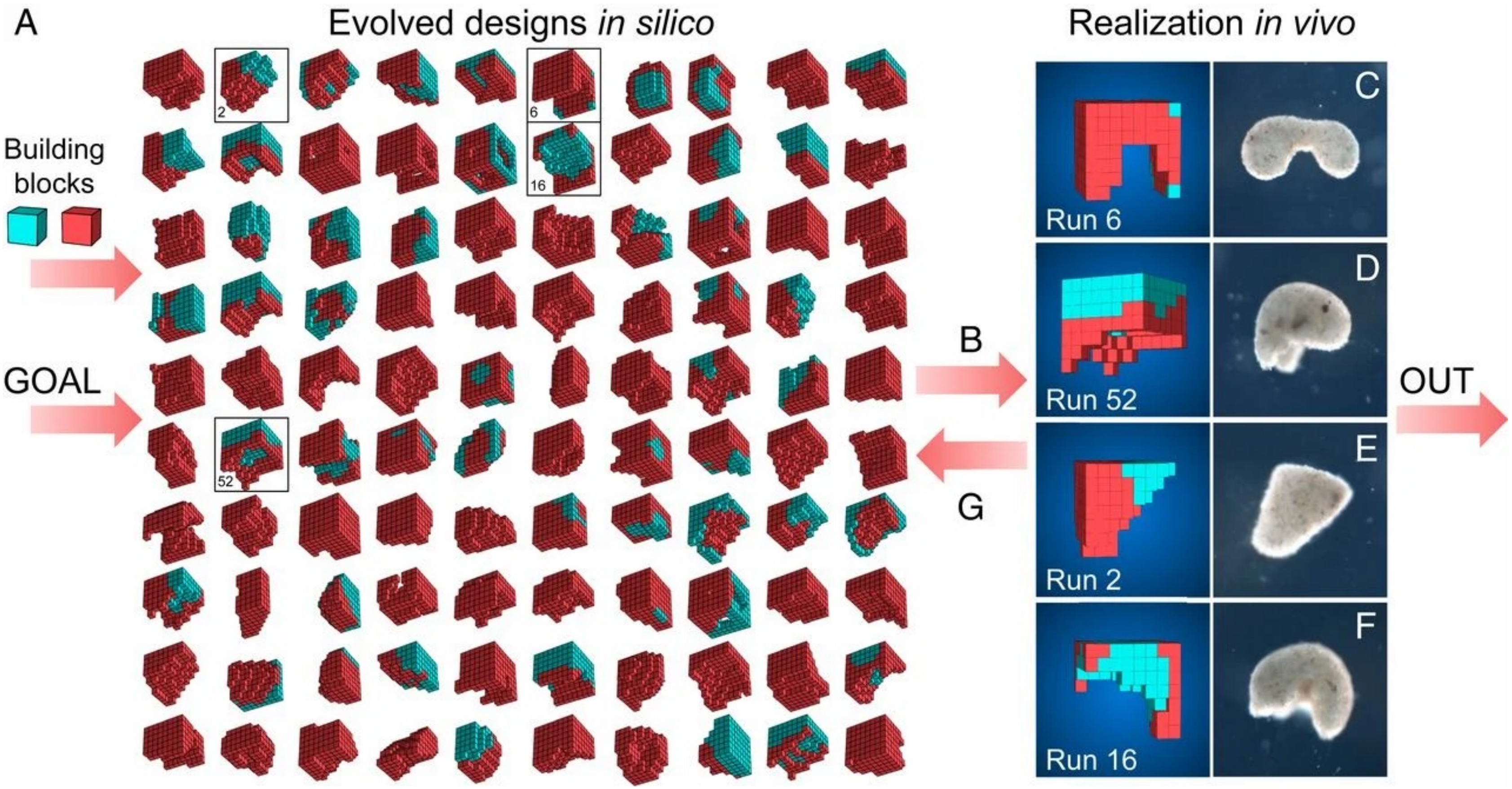
# Artificial Ontogeny (Bongard and Pfeifer, 2001)

E*volutionary developmental process* to synthesize artificial multicellular "creatures"



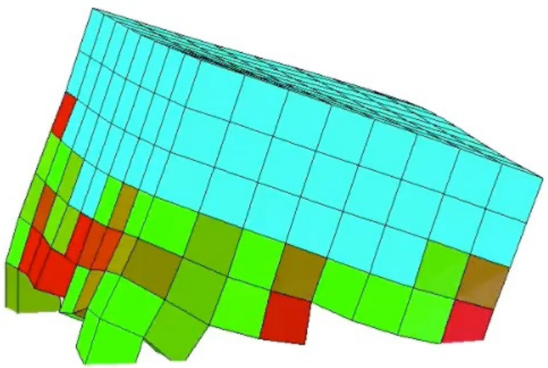
# Xenobots: Evolved *in silico*, self-assembled *in vivo*



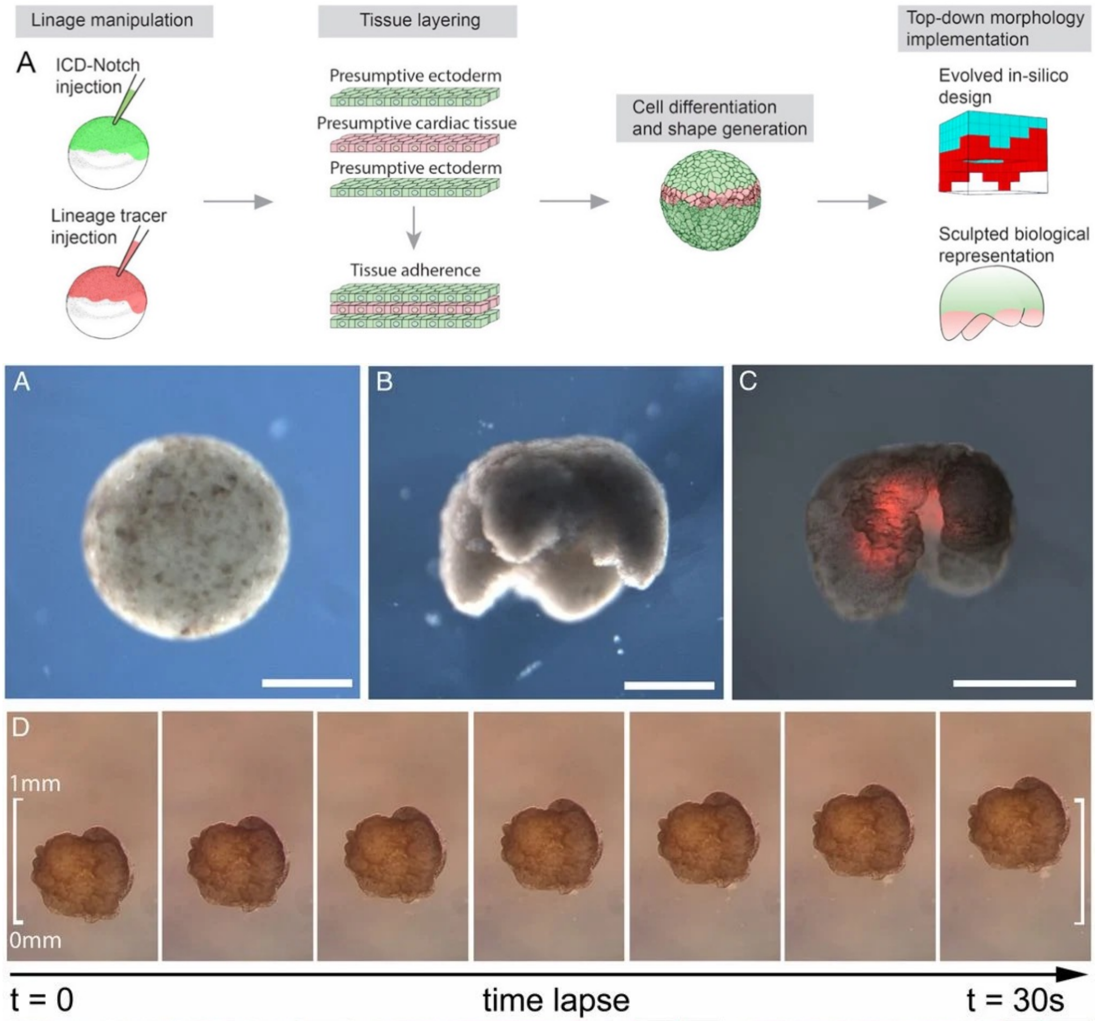

Kriegman, Sam, Douglas Blackiston, Michael Levin, and Josh Bongard (2020) A Scalable Pipeline for Designing Reconfigurable Organisms.” *Proceedings of the National Academy of Sciences* 117(4) : 1853–59. https://doi.org/10.1073/pnas.1910837117.

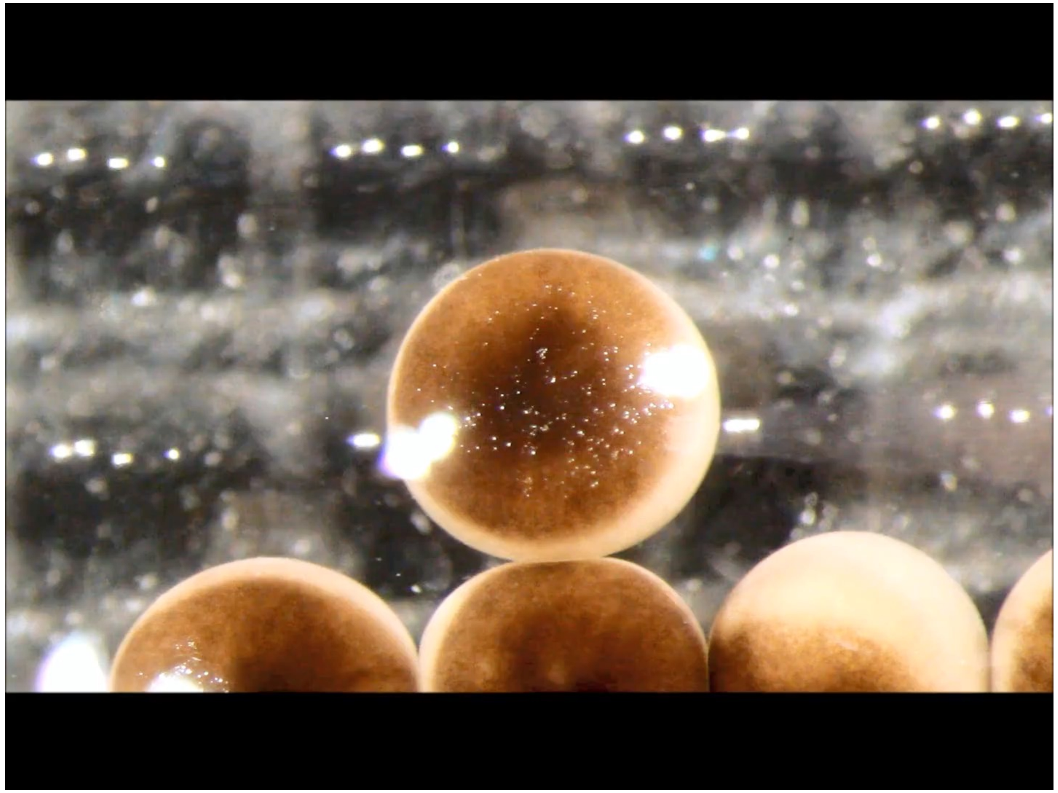
# A scalable pipeline for designing reconfigurable organisms.

Sam Kriegman, Douglas Blackiston, Michael Levin, Josh Bongard
University of Vermont, Tufts University.



# Manufacturing of self-assembling organism

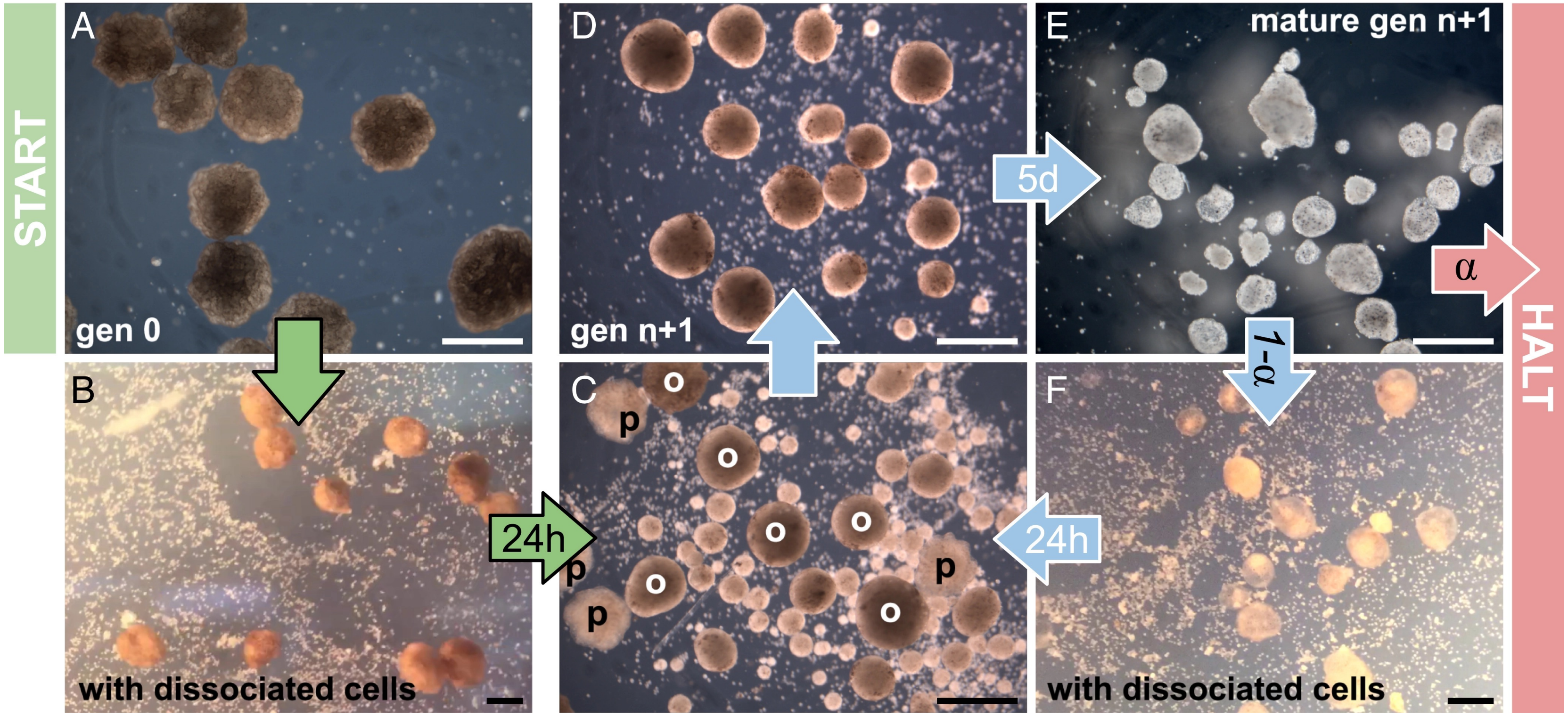


Caution: layering of cardiac cells within ectodermal cells is not shown in the video

# Assembly of Xenobots by frog cells

Spontaneous motion of frog cells assemble clusters of ectodermal stem cells that become Xenobots

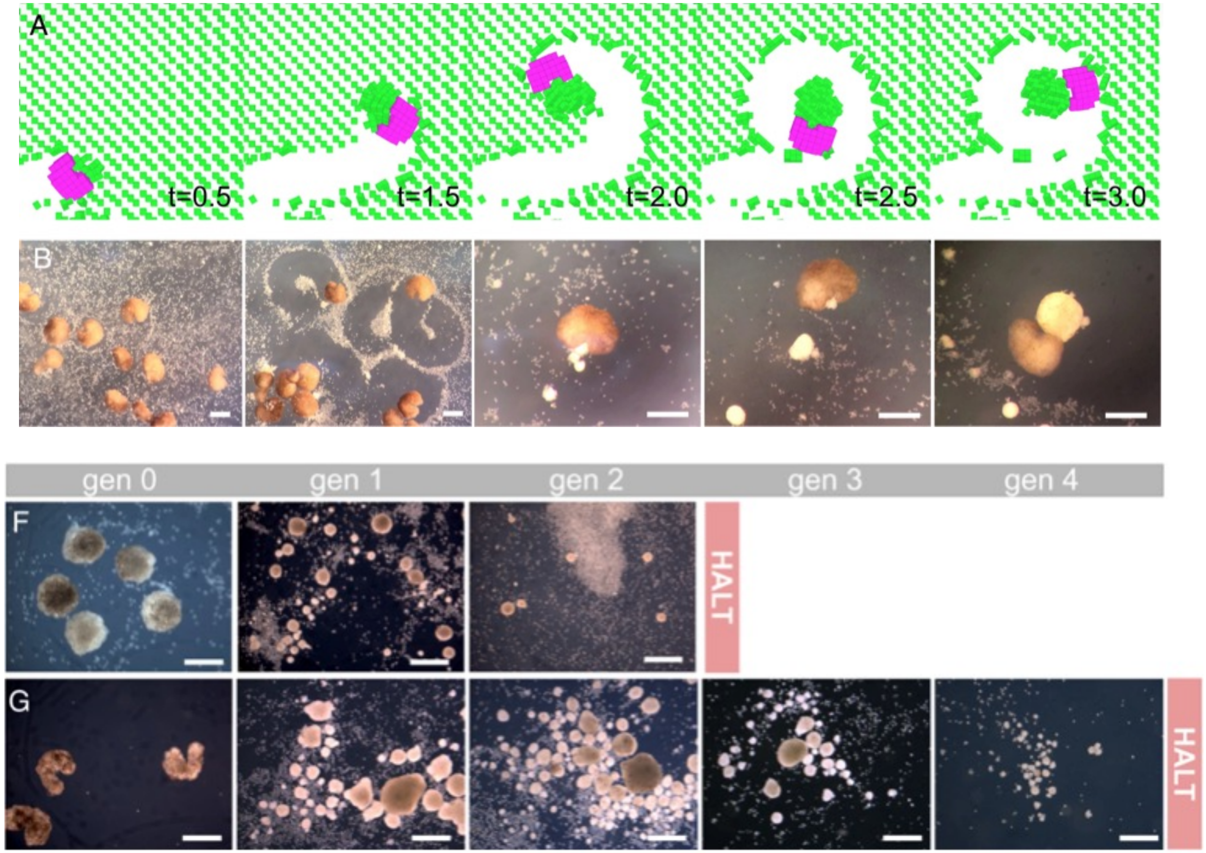



…but Xenobots do not self-replicate

Kriegman, Sam, Douglas Blackiston, Michael Levin, and Josh Bongard (2021) Kinematic Self-Replication in Reconfigurable Organisms. *Proceedings of the National Academy of Sciences* 118(49) https://doi.org/10.1073/pnas.2112672118.

# Kinematic self-replication of Xenobots

*In silico* evolution designs Xenobots that self-replicate for more generations



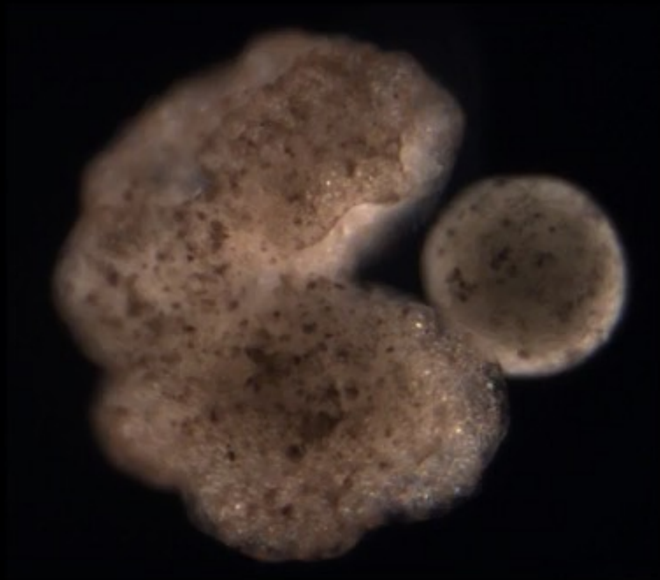

Spheroid Xenobots

Evolved Xenobots

Kriegman, Sam, Douglas Blackiston, Michael Levin, and Josh Bongard (2021) Kinematic Self-Replication in Reconfigurable Organisms. *Proceedings of the National Academy of Sciences* 118(49) https://doi.org/10.1073/pnas.2112672118.

# Kinematic self replication in reconfigurable organisms.

Sam Kriegman[1,2] Douglas Blackiston[1,2] Michael Levin[1,2] & Josh Bongard[3,*]



[1] Allen Discovery Center, Tufts University
[2] Wyss Institute for Biologically Inspired Engineering, Harvard University
[3] Department of Computer Science, University of Vermont
* jbongard@uvm.edu

# Robogen Presentation Timeline

| | |
|---|---|
| 31-May-2022 | Submission of<br>• presentation (.pptx and pdf) and<br>• ALL simulation files (scenario, best robot description file, arena, etc)<br>should be uploaded in a zip file. **Submission portal located in the “2 June 2022” section on Moodle.** |
| 2-June-2022 | Group presentations |

# Robogen Presentation Schedule (2nd June)

| Group | From | To | |
|---|---|---|---|
| **1** | 09:15 | 09:25 | AM |
| **2** | 09:25 | 09:35 | AM |
| **3** | 09:35 | 09:45 | AM |
| **4** | 09:45 | 09:55 | AM |
| **Break** | 09:55 | 10:10 | AM |
| **5** | 10:10 | 10:20 | AM |
| **6** | 10:20 | 10:30 | AM |
| **7** | 10:30 | 10:40 | AM |
| **8** | 10:40 | 10:50 | AM |
| **Break** | 10:50 | 11:05 | AM |
| **9** | 11:05 | 11:15 | AM |
| **10** | 11:15 | 11:25 | AM |
| **11** | 11:25 | 11:35 | AM |
| Presentation (8 mins) + questions (2 mins) | | | |

**Note**:

- Grand Challenge Presentation template.pptx on Moodle in the “2 June 2022” section.
- Your presentation should include a video of the physical robot that you have built (the performance will not be graded).

# Robogen Grading Criteria

| | | Dario Floreano | | | | | Krishna Manaswi Digumarti | | | | | Euan Judd | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Teams | | Method | Clarity | Completeness | **Grade** | Notes | Method | Clarity | Completeness | **Grade** | Notes | Method | Clarity | Completeness | **Grade** | Notes | **Average Grade** |
| | 1 | | | | | | | | | | | | | | | | |
| | 2 | | | | | | | | | | | | | | | | |
| | 3 | | | | | | | | | | | | | | | | |
| | 4 | | | | | | | | | | | | | | | | |
| | 5 | | | | | | | | | | | | | | | | |
| | 6 | | | | | | | | | | | | | | | | |
| | 7 | | | | | | | | | | | | | | | | |
| | 8 | | | | | | | | | | | | | | | | |
| | 9 | | | | | | | | | | | | | | | | |
| | 10 | | | | | | | | | | | | | | | | |
| | 11 | | | | | | | | | | | | | | | | |

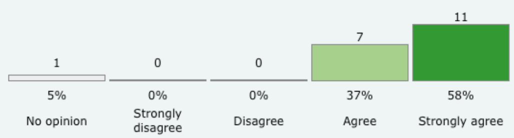
| |
|---|
| Method [50%] |
| Clarity [25%] |
| Completeness [25%] |

**Method:** The method includes describing the problem, your fitness function and how this relates to the problem, and description of the parameters that were used. It also includes your creativity (i.e. for the scenario) and your scientific approach.
**Clarity:** The clarity of your presentation includes clear and concise slides and description of your study.
**Completeness:** The completeness includes evidence of investigating the effects of changing parameters, different fitness functions, generalisability of your solution, and whether both the brain and the body have been evolved.

# Your evaluation of the course

| | |
|---|---|
| **Year** | 2021-2022 |
| **Course** | Evolutionary robotics |
| **Questionnaire** | Indicative feedback of teaching (since 2020-2021) |
| **Nb Registered** | 50 |
| **Nb Answered** | 19 |

**Overall, I think this course is good.**




**Remarks** Consult them 10 remarque(s) in **Html** or in **Excel**

great course. helpful assistants. The practical part on robogen is also great but the computation times make it hard to follow the suggested rythm. It gets much better when teaming up but still is hard.

---------------------------

Material is interesting, however I don't find the exercices helpful to understand the course. They take up a lot of time on slow laptops (which is what the majority of us are stuck with), robogen is bug-ridden and slow, and most of the time spent on the project and the exercices is dedicated to understand how to trick the software into doing what we want.

---------------------------

prendre 10 min a la fin du cours c'est pas top

---------------------------

Prof. Floreano is really a master at giving engaging lectures, and presents the subject with such passion and interest. However, the recorded videos were not the best in quality (I have to watch them due to course conflicts). The Robogen deadline to 3D print is a bit tight and stressful, especially when we have at least 4 other projects to work on. I find the software to be a bit of a blackbox with grid search, but that could well change.

---------------------------

So far all good. It seems like we will be spending a lot of time doing manufacturing (in the DLL building), which seems to me to be a little outside the scope of the course.

---------------------------

The course is very interesting. The matter is well taught but I feel like it would be beneficial to give more concrete example of applications of the concepts we see in class. The exercises and the projects are good as it gives an example of what it's possible to do with evolutionary algorithm, but it give very little knowledge on how to make the evolutionary algorithm.

---------------------------

This class is very well structured. The use of Robogen is very beneficial, makes the course even more interesting and actually gives a concrete example of evolutionary robotics. Nothing to change in my opinion.

---------------------------

Very dedicated teacher and good material (Robogen, 3D-printing of own robots, ...). However, the first Python Assignment was not suitable for Linux-users since it only worked with a specific Python version which I found tricky on my PC.

---------------------------

Very interesting

---------------------------

very interesting but please let us our 15 min breaks + the project begins too late in the semester

---------------------------

<u>Areas of improvement:</u>

Some lectures exceed the 45 minutes

Show more applications of ER

Exercises could show how to program an EA

Robogen has bugs and is slow

Python version does not work on some Linux

<u>In-depth evaluation will be organized by SAC and sent you in June: thank you in advance!</u>