

The Wayback Machine - [https://web.archive.org/web/20220610185431/https://rivesunder.github.io/yuca/step\\_size](https://web.archive.org/web/20220610185431/https://rivesunder.github.io/yuca/step_size)

## yuca

# Step Size is a Consequential Parameter in Continuous Cellular Automata

Experiment with varying step size in an interactive [notebook](#) on: [mybinder](#) -> [launch](#) [binder](#) or in [colab](#) -> [Open in Colab](#)

## Introduction

Cellular automata (CA) dynamics with continuously-valued states and time steps can be generically written as<sup>1</sup>:

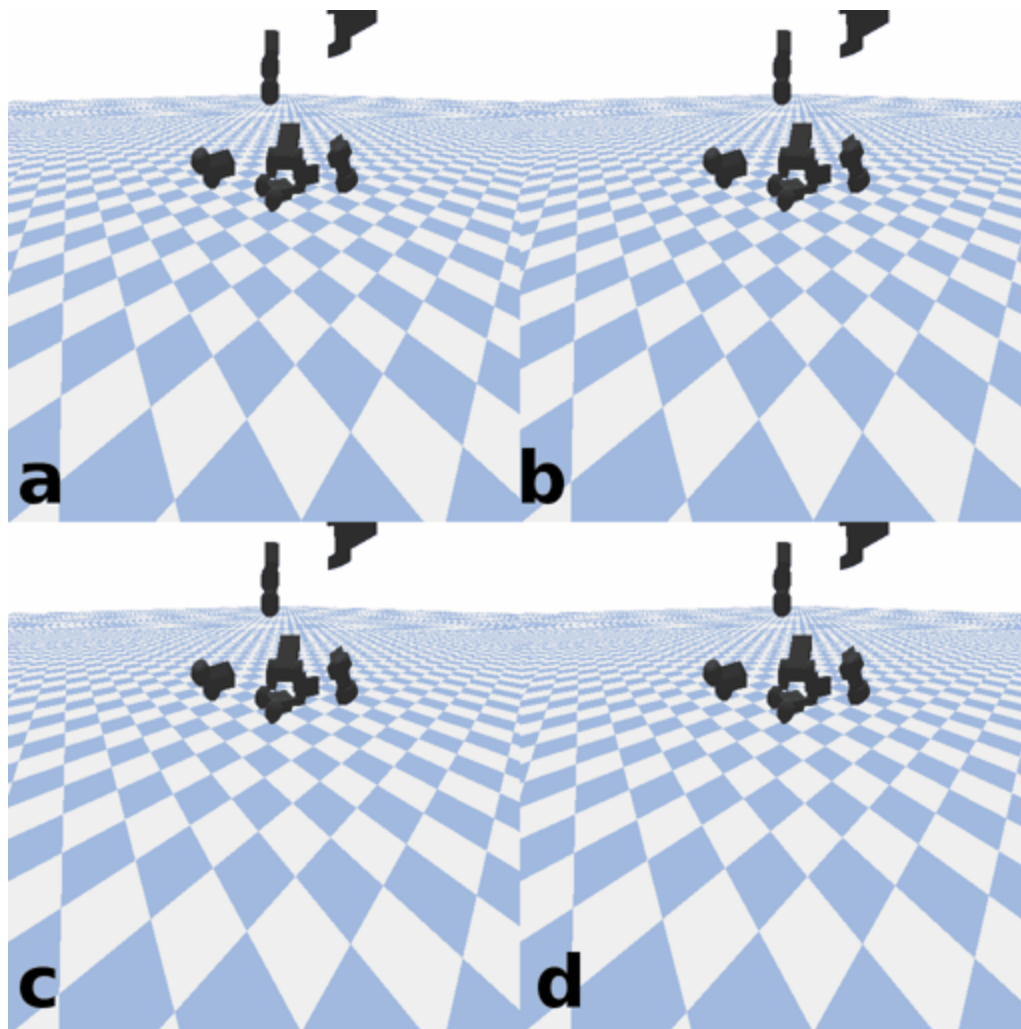
$$A^{t+dt} = A^t + f(A^t)dt$$

In particular the equation above describes the Lenia framework for continuous CA, and as noted previously in that work<sup>2</sup>, the equation above has the same form as Euler's numerical method for solving differential equations, *i.e.* estimating CA dynamics if they are described by a differential equation  $\frac{dA^t}{dt} = f(A^t)$ . CA updates under the Lenia framework are more particularly written as:

$$A^{t+dt} = A^t + dt \cdot G(K * A^t)$$

Where  $A^{t+dt}$  is the grid of cell states at time  $t + dt$ ,  $G(\cdot)$  is the growth function,  $K * A^t$  is the 2d spatial convolution of neighborhood kernel  $K$  with cell states  $A^t$  at time  $t$ , and  $dt$  is the step size.

For numerical estimation of differential equations with [Euler's method](#), error depends on step size. For the most part, smaller step sizes lead to more accurate solutions. Those familiar with working with numerical methods for differential equations and/or physical simulations based on them will likely anticipate that a step size that is too large leads to unstable solutions and, often, catastrophic behavior. As an example, we can consider the following simulation of a robot drop in PyBullet<sup>3</sup> at varying step sizes. In the animation below, the bottom left corner (c) represents the default step size of 1/240 seconds, a reasonable trade-off between execution speed and simulation accuracy/stability. The top row a and b are based on the same initial conditions with step sizes 100 and 10 times smaller than c, respectively. The animation in d used a step size 100 times longer than the default step size in c.

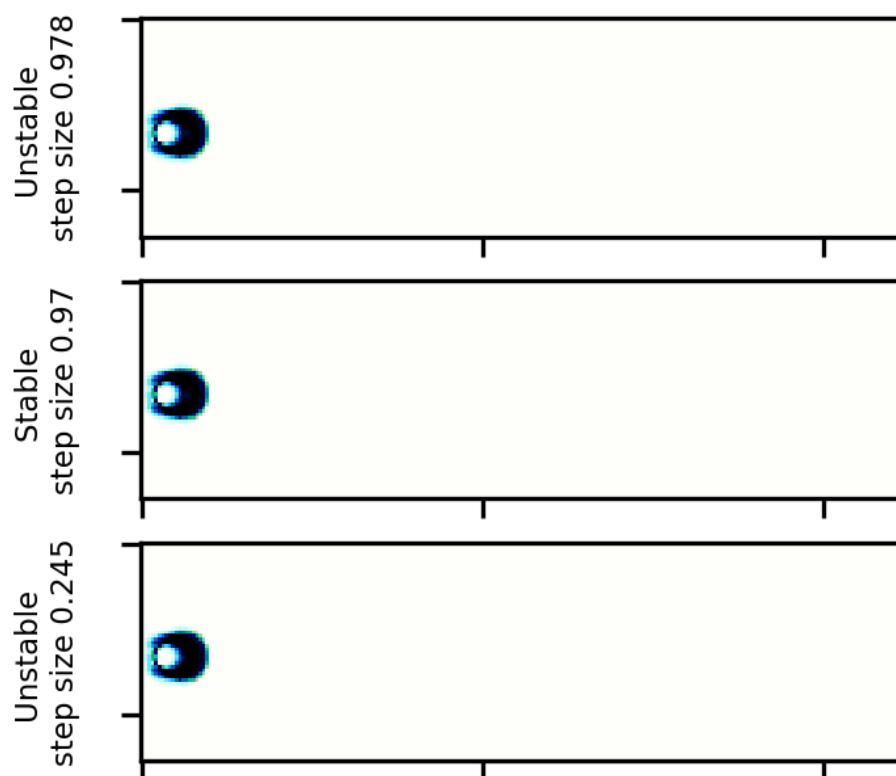


**Modulating time\_step parameter in PyBullet. a:  $dt = 1/24000$  s, b:  $dt = 1/2400$  s, c:  $dt = 1/240$  s, d:  $dt = 1/2.4$  s.  $1/240$  seconds is the default and recommended step size in PyBullet.**

Only the final condition with a step size of  $\sim 0.4166$  seconds displays noticeably non-physical and unstable behavior: the two-wheeled robots plunge through the ground plane before rising slowly like so many spring flowers. Small step sizes behave qualitatively similar to the default, though specific outcomes are noticeably different. In continuous CA, similar differences in step size leads to compromised stability at too small a step size as well as too large. Specific patterns under otherwise identical CA rules occupy different ranges of stable step size. Patterns may also exhibit qualitatively different behaviors at different step sizes. Thus step size choice in continuous CA leads to more interesting differences in outcomes than we see in the PyBullet example and typically expect in numerical physics simulations in general. There are many other physics-based models that support self-organizing solitons, such as chemical reaction-diffusion models (*e.g.* <sup>4</sup>), particle swarms (*e.g.* <sup>5</sup>), or other n-body problems, and these may also exhibit interesting behavioral diversity over a range of step sizes.

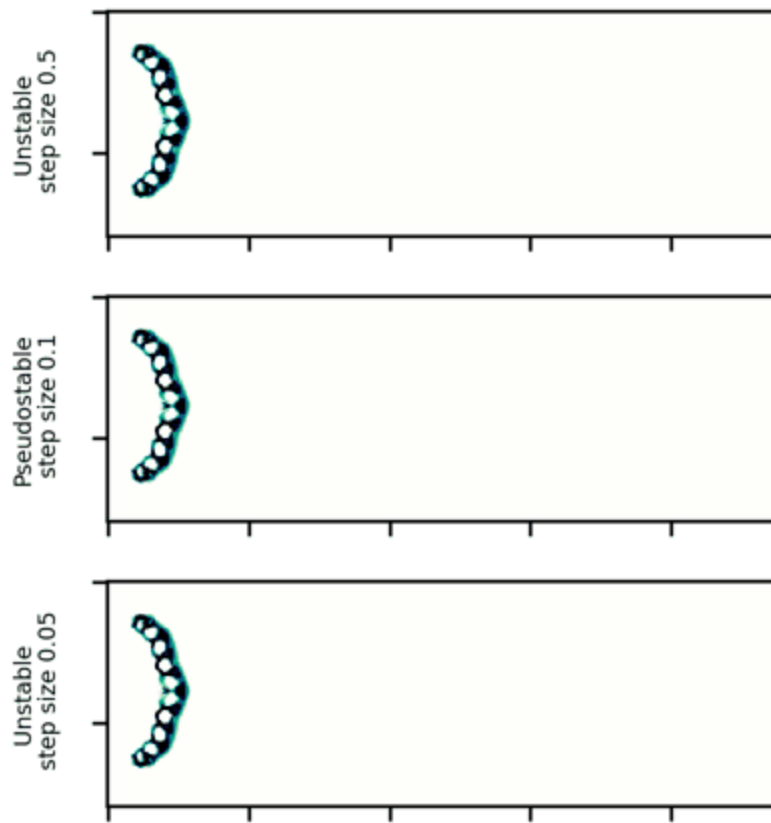
## Pattern stability depends on step size

A minimal glider in the style of the SmoothLife glider<sup>6</sup> and implemented in the *Scutium gravidus* CA under the Lenia framework is only stable in a range of step sizes from about 0.25 to 0.97. A choice of  $dt$  outside this range results in a vanishing glider.



**A minimal glider in Lenia's *Scutium gravidus* rule set<sup>2</sup>, similar to the SmoothLife glider<sup>6</sup>, is unstable at step sizes below about 0.25 and above about 0.97.**

A wide glider in the same CA rule set is typically stable for over 2000 steps at a  $dt$  of 0.1, but disappears at step sizes of 0.05 or below and is also unstable at a step size of 0.5 or above, usually exhibiting unconstrained growth at large step sizes.



**A wide glider in *Scutium gravidus*.** Unlike the narrow glider, this glider is pseudo-stable at a moderate step size of 0.1 and unstable for large and small step sizes above and below about 0.5 and 0.05, respectively.

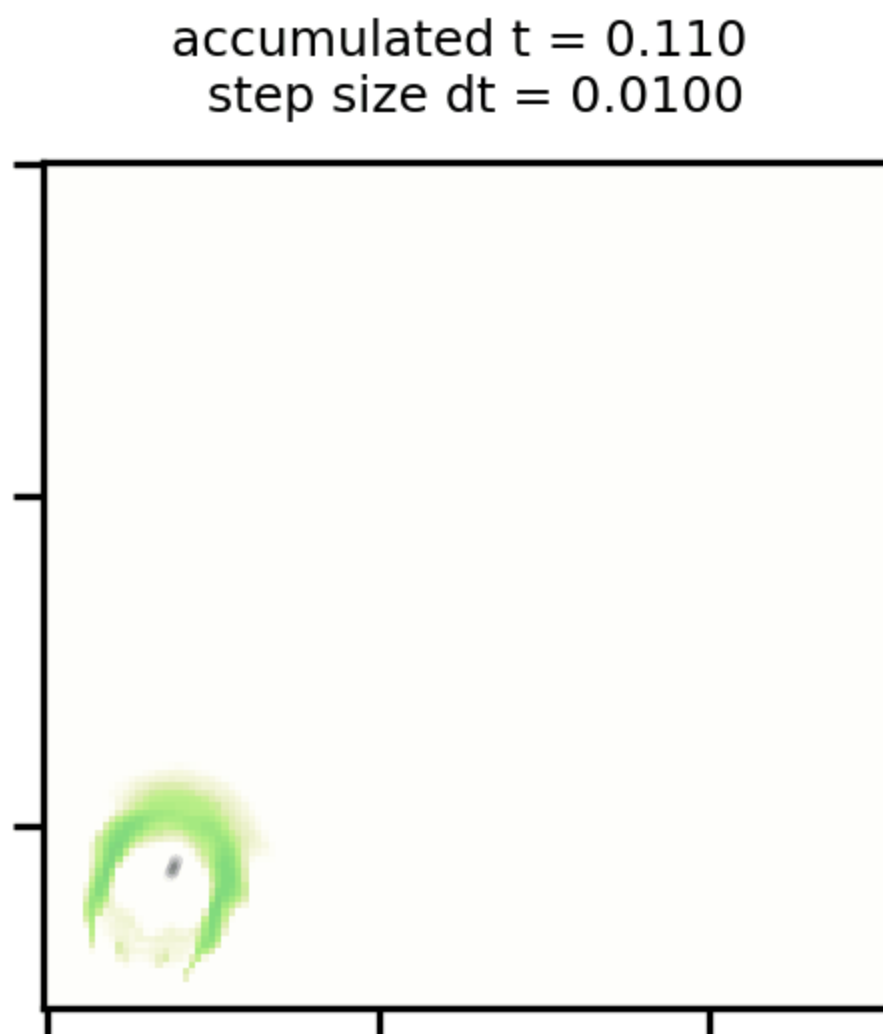
## Behavior of individual patterns can vary qualitatively at different step sizes

A more striking consequence of step size is qualitatively different behavior at different step sizes. The following example is a “frog” pattern implemented in an extension of Lenia called Glaberish <sup>7</sup>. In Lenia, the growth function  $G(\cdot)$  depends only on the results of a 2D convolution of the neighborhood kernel and the cell state grid  $K * A^t$ . Glaberish splits this growth function into *persistence* ( $P(\cdot)$ ) and *genesis* ( $G_{gen}(\cdot)$ ) functions, each contributes to the overall change in cell state according to the current grid states.

$$A^{t+dt} = A^t + dt \left[ (1 - A^t) G_{gen}(K * A^t) + A^t P(K * A^t) \right]$$

Glaberish CA dynamics reinstate the dependence on cell state found in SmoothLife, Conway’s Life <sup>8</sup>, as well as other Life-like CA, while maintaining the flexibility of Lenia’s growth function. The following frog pattern can be found in a Glaberish CA with evolved persistence and genesis parameters called s613 (see <sup>9</sup> for details on how this CA was evolved). While the narrow and wide gliders in Lenia’s *Scutium gravidus* CA occupy particular ranges of  $dt$ , the s613 frog pattern

exhibits qualitatively different behavior across a range of step sizes from about 0.01 to about 0.13.



**For the frog pattern in Glaberish CA s613, varying step size leads to qualitatively different behaviors.**

## Conclusions

This work demonstrates the consequences of varying step size in continuous CA. Patterns simulated in Lenia's *Scutium gravidus* CA are unstable at either too large or too small step size, and different patterns occupy different step size ranges in otherwise identical CA rules. In the Glaberish CA s613, the frog pattern exhibits qualitatively different behavior at step sizes from 0.01 to 0.15, ranging from corkscrewing, meandering, hopping, surging, and finally bursting and disappearing.

The results we have observed for these patterns contrasts sharply with previous remarks

concerning the similarity of continuous CA to Euler's method for solving ODEs with regard to step size<sup>2</sup>. Observations of the mobile *Orbium* pattern in Lenia were consistent with the premise that decreasing step size asymptotically approaches an ideal simulation of the *Orbium* pattern<sup>2</sup>, but for gliders in *Scutium gravidus* and s613 we have shown that the relationship between CA dynamics and step size is not that simple in general. This work demonstrates that for several patterns a lower step size does not entail a more accurate simulation, but different behavior or potential patterns entirely. Given the evidence presented in this work, it follows that step size should be given due consideration when searching for bioreminiscent patterns<sup>2 10</sup>, and for optimization and learning with CA, for example in training patterns to have the agency to negotiate obstacles<sup>11</sup>, or for training neural CA for a variety of tasks such as growing patterns<sup>12</sup>, classifying pixels<sup>13</sup>, learning to generate textures<sup>14</sup>, and control<sup>15</sup>.

Experiment with varying step size in an interactive [notebook](#) on: [mybinder](#) ->  launch binder or in [colab](#) ->  [Open in Colab](#)

**This post has not been peer-reviewed itself, but provides supporting information for the following short article accepted to the 2022 Conference on Artificial Life:**

- Davis, Q, Tyrell and Bongard, Josh. "Step Size is a Consequential Parameter in Continuous Cellular Automata". Accepted to The 2022 Conference on Artificial Life. (2022).

## References and Footnotes

1. **Note:** The different formulation for the original, discrete, SmoothLife, which had a discrete time-step (*i.e.* cell states were replaced at each time step):  $A^{t+dt} = s(K * A^t, A^t)$  ↩
2. Chan, Bert Wang-Chak. "Lenia - Biology of Artificial Life." *Complex Syst.* 28 (2019): <https://arxiv.org/abs/1812.05433>. ↩ ↩<sup>2</sup> ↩<sup>3</sup> ↩<sup>4</sup> ↩<sup>5</sup>
3. [<https://pybullet.org/>][<https://pybullet.org/>] ↩
4. Munafo, Robert. "Stable localized moving patterns in the 2-D Gray-Scott model." *arXiv: Pattern Formation and Solitons* (2014): <https://arxiv.org/abs/1501.01990>. ↩
5. Schmickl, Thomas et al. "How a life-like system emerges from a simple particle motion law." *Scientific Reports* 6 (2016): <https://www.nature.com/articles/srep37969> ↩
6. Rafler, Stephan. "Generalization of Conway's "Game of Life" to a continuous domain - SmoothLife." *arXiv: Cellular Automata and Lattice Gases* (2011): <https://arxiv.org/abs/1111.1567> ↩ ↩<sup>2</sup>
7. Davis, Q, Tyrell and Bongard, Josh. "Glaberish: Generalizing the Continuously-Valued Lenia

Framework to Arbitrary Life-Like Cellular Automata”. Accepted to The 2022 Conference on Artificial Life. (2022). <https://arxiv.org/abs/2205.10463> ↵

8. Gardener, Master. “Mathematical games: the fantastic combinations of john conway’s new solitaire game ‘life’.” (1970). ↵
9. Davis, Q, Tyrell and Bongard, Josh. “Selecting Continuous Life-Like Cellular Automata for Halting Unpredictability: Evolving for Abiogenesis”. Accepted to Proceedings of the Genetic and Evolutionary Computation Conference Companion (2022): <https://arxiv.org/abs/2204.07541> ↵
10. Chan, Bert Wang-Chak. “Lenia and Expanded Universe.” ALIFE 2020: The 2020 Conference on Artificial Life. MIT Press, (2020). <https://arxiv.org/abs/2005.03742> ↵
11. Hamon, G. and Etcheverry, M. and Chan, B. W.-C. and Moulin-Frier, C. and Oudeyer, P.-Y. Learning sensorimotor agency in cellular automata. Blog post. (2022). <https://developmentalsystems.org/sensorimotor-lenia/>. ↵
12. Mordvintsev, A. and Randazzo, E. and Niklasson, E. and Levin, M. Growing neural cellular automata. Distill. (2020) <https://distill.pub/2020/growing-ca>. ↵
13. Randazzo, E. and Mordvintsev, A. and Niklasson, E. and Levin, M. and Greydanus, S. Self-classifying MNIST digits. Distill. (2020) <https://distill.pub/2020/selforg/mnist>. ↵
14. Niklasson, E. and Mordvintsev, A. and Randazzo, E. and Levin, M. “Self-Organising Textures”, Distill, (2021). <https://distill.pub/selforg/2021/textures/> ↵
15. Variengien, A. and Pontes-Filho and Sidney and Glover, T. and Nichele, S. “Towards self-organized control: Using neural cellular automata to robustly control a cart-pole agent.” Innovations in Machine Intelligence, 1:1–14. (2021). <https://arxiv.org/abs/2106.15240v1> ↵

---

This site is open source. [Improve this page.](#)