Adversarial Reprogramming of Neural Cellular Automata

A robustness investigation.

AUTHORS
Ettore Randazzo
Alexander Mordvintsev
Eyvind Niklasson
Michael Levin

AFFILIATIONS
Google
Google
Google
Allen Discovery Center at Tufts University

PUBLISHED
May 6, 2021

DOI
10.23915/distill.00027.004

Contents

This article is part of the Differentiable Self-organizing Systems Thread, an experimental format collecting invited short articles delving into differentiable self-organizing systems, interspersed with critical commentary from several experts in adjacent fields.

← PREVIOUS ARTICLE
Self-Organising Textures

This article makes strong use of colors in figures and demos. Click here to adjust the color palette.

In a complex system, whether biological, technological, or social, how can we discover signaling events that will alter system-level behavior in desired ways? Even when the rules governing the individual components of these complex systems are known, the inverse problem - going from desired behaviour to system design - is at the heart of many barriers for the advance of biomedicine, robotics, and other fields of importance to society.

Biology, specifically, is transitioning from a focus on mechanism (what is required for the system to work) to a focus on information (what algorithm is sufficient to implement adaptive behavior). Advances in machine learning represent an exciting and largely untapped source of inspiration and tooling to assist the biological sciences. Growing Neural Cellular Automata [1] and Self-classifying MNIST Digits [2] introduced the Neural Cellular Automata (Neural CA) model and demonstrated how tasks requiring self-organisation, such as pattern growth and self-classification of digits, can be trained in an end-to-end, differentiable fashion. The resulting models were robust to various kinds of perturbations: the growing CA expressed regenerative capabilities when damaged; the MNIST CA were responsive to changes in the underlying digits, triggering reclassification whenever necessary. These computational frameworks represent quantitative models with which to understand important biological phenomena, such as scaling of single cell behavior rules into reliable organ-level anatomies. The latter is a kind of anatomical homeostasis, achieved by feedback loops that must recognize deviations from a correct target morphology and progressively reduce anatomical error.

In this work, we *train adversaries* whose goal is to reprogram CA into doing something other than what they were trained to do. In order to understand what kinds of lower-level signals alter system-level behavior of our CA, it is important to understand how these CA are constructed and where local versus global information resides.

The system-level behavior of Neural CA is affected by:

- **Individual cell states.** States store information which is used for both diversification among cell behaviours and for communication with neighbouring cells.
- **The model parameters.** These describe the input/output behavior of a cell and are shared by every cell of the same family. The model parameters can be seen as *the way the system works*.
- **The perceptive field.** This is how cells perceive their environment. In Neural CA, we always restrict the perceptive field to be the eight nearest neighbors and the cell itself. The way cells are perceived by each other is different between the Growing CA and MNIST CA. The Growing CA perceptive field is a set of weights fixed both during training and inference, while the MNIST CA perceptive field is learned as part of the model parameters.

Perturbing any of these components will result in system-level behavioural changes.

We will explore two kinds of adversarial attacks: 1) injecting a few adversarial cells into an existing grid running a pretrained model; and 2) perturbing the global state of all cells on a grid.

For the first type of adversarial attacks we train a new CA model that, when placed in an environment running one of the original models described in the previous articles, is able to hijack the behavior of the collective mix of adversarial and non-adversarial CA. This is an example of injecting CA with differing *model parameters* into the system. In biology, numerous forms of hijacking are known, including viruses that take over genetic and biochemical information flow [3], bacteria that take over physiological control mechanisms [4] and even regenerative morphology of whole bodies [5], and fungi and toxoplasma that modulate host behavior [6]. Especially fascinating are the many cases of non-cell-autonomous signaling developmental biology and cancer, showing that some cell behaviors can significantly alter host properties both locally and at long range. For example, bioelectrically-abnormal cells can trigger metastatic conversion in an otherwise normal body (with no genetic defects) [7], while management of bioelectrical state in one area of the body can suppress tumorigenesis on the other side of the organism [8]. Similarly, amputation damage in one leg initiates changes to ionic properties of cells in the contralateral leg [9], while the size of the developing brain is in part dictated by the activity of ventral gut cells [10]. All of these phenomena underlie the importance of understanding how cell groups make collective decisions, and how those tissue-level decisions can be subverted by the activity of a small number of cells. It is essential to develop quantitative models of such dynamics, in order to drive meaningful progress in regenerative medicine that controls system-level outcomes top-down, where cell- or molecular-level micromanagement is infeasible [11].

The second type of adversarial attacks interact with previously trained growing CA models by *perturbing the states within cells*. We apply a global state perturbation to all living cells. This can be seen as inhibiting or enhancing combinations of state values, in turn hijacking proper communications among cells and within the cell's own states. Models like this represent not only ways of thinking about adversarial relationships in nature (such as parasitism and evolutionary arms races of genetic and physiological mechanisms), but also a roadmap for the development of regenerative medicine strategies. Next-generation biomedicine will need computational tools for inferring minimal, least-effort interventions that can be applied to biological systems to predictively change their large-scale anatomical and behavioral properties.

Adversarial MNIST CA

TRY IN A NOTEBOOK

Recall how the Self-classifying MNIST digits task consisted of placing CA cells on a plane forming the shape of an MNIST digit. The cells then had to communicate among themselves in order to come to a complete consensus as to which digit they formed.

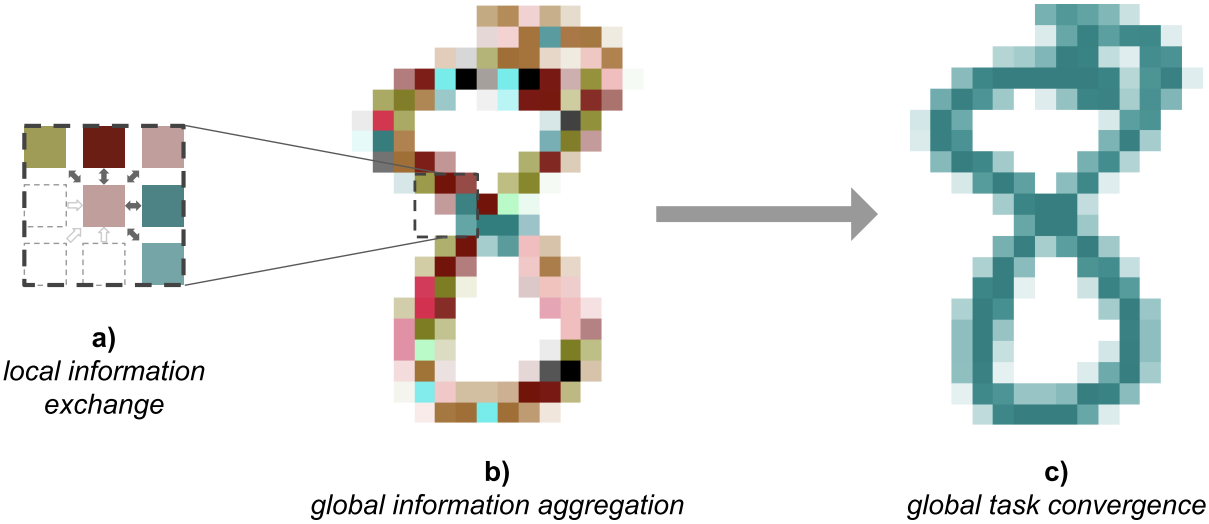


Figure 1: Diagram showing the local vs. global information available in the cell collective.
(a) Local information neighbourhood - each cell can only observe itself and its neighbors' states, or the absence of neighbours.
(b) Globally, the cell collective aggregates information from all parts of itself.
(c) It is able to distinguish certain shapes that compose a specific digit (3 in the example).

Below we show examples of classifications made by the model trained in Self-classifying MNIST Digits.

Figure 2: The original model behavior on unseen data. Classification mistakes have a red background.

In this experiment, **the goal is to create adversarial CA that can hijack the cell collective's classification consensus to always classify an eight**. We use the CA model from [2] and freeze its parameters. We then train a new CA whose model architecture is identical to the frozen model but is randomly initialized. The training regime also closely approximates that of self-classifying MNIST digits CA. There are three important differences:

- Regardless of what the actual digit is, we consider *the correct classification to always be an eight*.
- For each batch and each pixel, the CA is randomly chosen to be either the pretrained model or the new adversarial one. The adversarial CA is used 10% of the time, and the pre-trained, frozen, model the rest of the time.
- Only the adversarial CA parameters are trained, the parameters of the pretrained model are kept frozen.

The adversarial attack as defined here only modifies a small percentage of the overall system, but the goal is to propagate signals that affect all the living cells. Therefore, these adversaries have to somehow learn to communicate deceiving information that causes wrong classifications in their neighbours and further cascades in the propagation of deceiving information by ‘unaware’ cells. The unaware cells’ parameters cannot be changed so the only means of attack by the adversaries is to cause a change in the cells’ states. Cells’ states are responsible for communication and diversification.

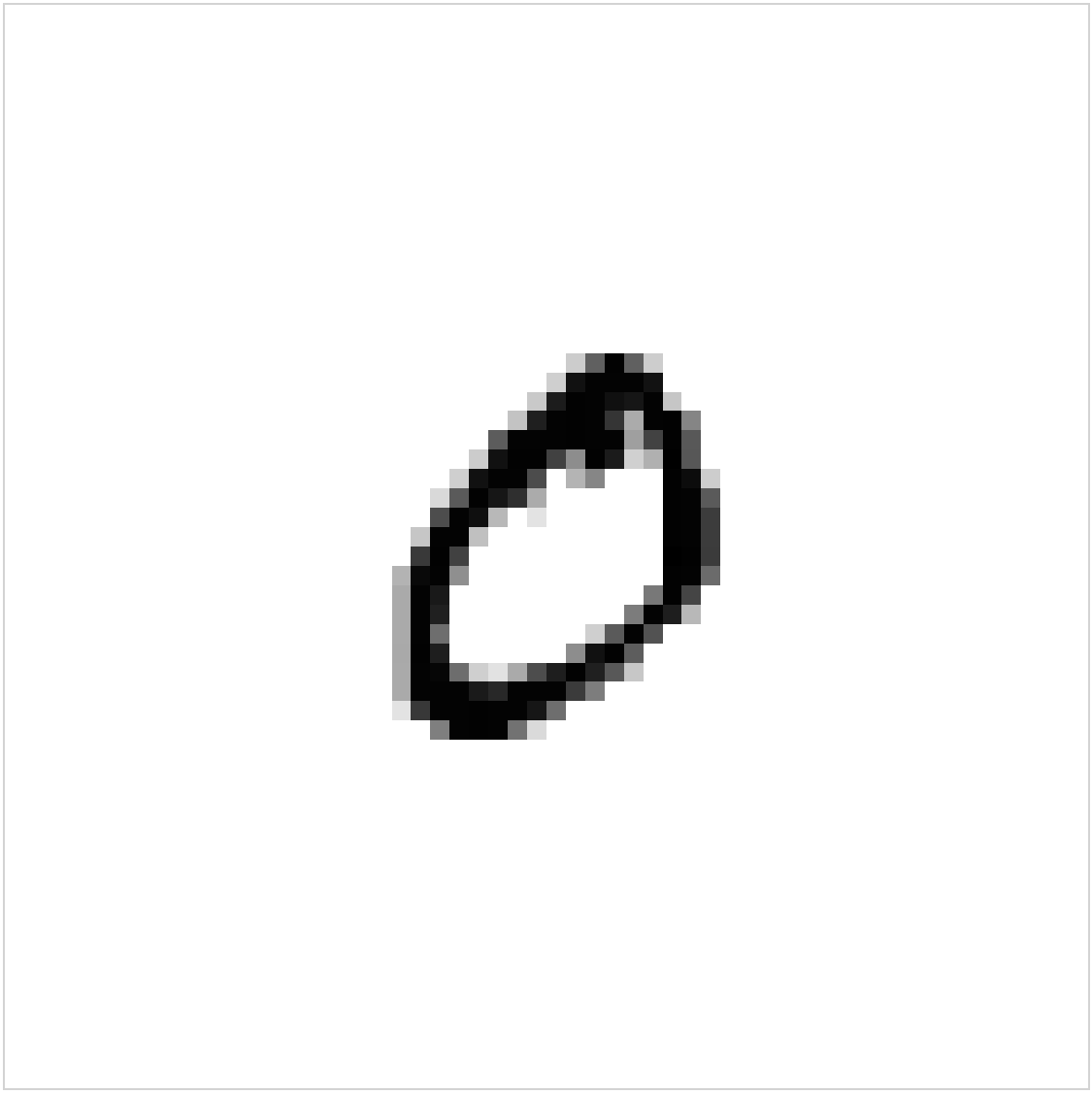
The task is remarkably simple to optimize, reaching convergence in as little as 2000 training steps (as opposed to the two orders of magnitude more steps needed to construct the original MNIST CA). By visualising what happens when we remove the adversaries, we observe that the adversaries must be constantly communicating with their non-adversarial neighbours to keep them convinced of the malicious classification. While some digits don’t recover after the removal of adversaries, most of them self-correct to the right classification. Below we show examples where we introduce the adversaries at 200 steps and remove them after a further 200 steps.

Figure 3: We introduce the adversaries (red pixels) after 200 steps and remove them after 200 more steps. Most digits recover, but not all. We highlight mistakes in classification with a red background.

While we trained the adversaries with a 10-to-90% split of adversarial vs. non-adversarial cells, we observe that often significantly fewer adversaries are needed to succeed in the deception. Below we evaluate the experiment with just one percent of cells being adversaries.

Figure 4: Adversaries constituting up 1% of the cell collective (red pixels). We highlight mistakes in classification with a red background.

We created a demo playground where the reader can draw digits and place adversaries with surgical precision. We encourage the reader to play with the demo to get a sense of how easily non-adversarial cells are swayed towards the wrong classification.



Summary. Each pixel is analogous to a biological cell. It decides its own color and communicates with its immediate neighbors. The goal of the cell population as a whole is to come to an agreement about what their global shape represents. The goal of the adversaries is to steer the classification towards an "8", regardless of the actual shape.

Usage. Interact with the cells by clicking or tapping on the canvas above. Press different digits to load or resample them. Press the bin to clear the canvas. Toggle the Draw Adversary box to draw adversaries as opposed to the original CAs. Adversaries can be drawn surgically (one pixel at a time).

Adversarial Injections for Growing CA

TRY IN A NOTEBOOK

The natural follow up question is whether these adversarial attacks work on Growing CA, too. The Growing CA goal is to be able to grow a complex image from a single cell, and having its result be persistent over time and robust to perturbations. In this article, we focus on the lizard pattern model from Growing CA.

0:03 / 0:06

Figure 5: The target CA to hijack.

The goal is to have some adversarial cells change the global configuration of all the cells. We choose two new targets we would like the adversarial cells to try and morph the lizard into: a tailless lizard and a red lizard.

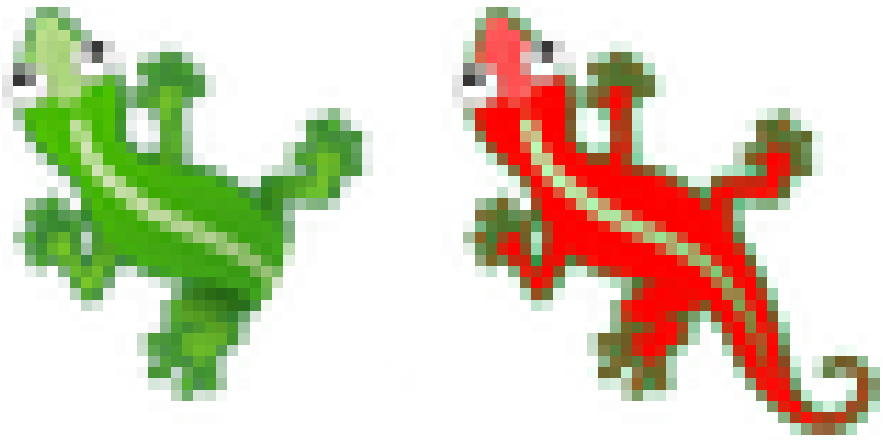


Figure 6: The desired mutations we want to apply.

These targets have different properties:

- **Red lizard:** converting a lizard from green to red would show a global change in the behaviour of the cell collective. This behavior is not present in the dynamics observed by the original model. The adversaries are thus tasked with fooling other cells into doing things they have never done before (create the lizard shape as before, but now colored in red).

- **Tailless lizard:** having a severed tail is a more localized change that only requires some cells to be fooled into behaving in the wrong way: the cells at the base of the tail need to be convinced they constitute the edge or silhouette of the lizard, instead of proceeding to grow a tail as before.

Just like in the previous experiment, our adversaries can only indirectly affect the states of the original cells.

We first train adversaries for the tailless target with a 10% chance for any given cell to be an adversary. We prohibit cells to be adversaries if they are outside the target pattern; i.e. the tail contains no adversaries.

Figure 7: 10% of the cells are adversarial.

The video above shows six different instances of the same model with differing stochastic placement of the adversaries. The results vary considerably: sometimes the adversaries succeed in removing the tail, sometimes the tail is only shrunk but not completely removed, and other times the pattern becomes unstable. Training these adversaries required many more gradient steps to achieve convergence, and the pattern converged to is qualitatively worse than what was achieved for the adversarial MNIST CA experiment.

The red lizard pattern fares even worse. Using only 10% adversarial cells results in a complete failure: the original cells are unaffected by the adversaries. Some readers may wonder whether the original pretrained CA has the requisite skill, or 'subroutine' of producing a red output at all, since there are no red regions in the original target, and may suspect this was an impossible task to begin with. Therefore, we increased the proportion of adversarial cells until we managed to find a successful adversarial CA, if any were possible.

Figure 8: Adversaries are 60% of the cells. At step 500, we stop the image and show only cells that are from the original model.

In the video above we can see how, at least in the first stages of morphogenesis, 60% of adversaries are capable of coloring the lizard red. Take particular notice of the “step 500” [1], where we hide the adversarial cells and show only the original cells. There, we see how a handful of original cells are colored in red. This is proof that the adversaries successfully managed to steer neighboring cells to color themselves red, where needed.

However, the model is very unstable when iterated for periods of time longer than seen during training. Moreover, the learned adversarial attack is dependent on a majority of cells being adversaries. For instance, when using fewer adversaries on the order of 20-30%, the configuration is unstable.

In comparison to the results of the previous experiment, the Growing CA model shows a greater resistance to adversarial perturbation than those of the MNIST CA. A notable difference between the two models is that the MNIST CA cells have to always be ready and able to change an opinion (a classification) based on information propagated through several neighbors. This is a necessary requirement for that model because at any time the underlying digit may change, but most of the cells would not observe any change in their neighbors' placements. For instance, imagine the case of a one turning into a seven where the lower stroke of each overlap perfectly. From the point of view of the cells in the lower stroke of the digit, there is no change, yet the digit formed is now a seven. We therefore hypothesise MNIST CA are more reliant and 'trusting' of continuous long-distance communication than Growing CA, where cells never have to reconfigure themselves to generate something different to before.

We suspect that more general-purpose Growing CA that have learned a variety of target patterns during training are more likely to be susceptible to adversarial attacks.

Perturbing the states of Growing CA

TRY IN A NOTEBOOK

We observed that it is hard to fool Growing CA into changing their morphology by placing adversarial cells inside the cell collective. These adversaries had to devise complex local behaviors that would cause the non-adversarial cells nearby, and ultimately globally throughout the image, to change their overall morphology.

In this section, we explore an alternative approach: perturbing the global state of all cells without changing the model parameters of any cell.

As before, we base our experiments on the Growing CA model trained to produce a lizard. Every cell of a Growing CA has an internal state vector with 16 elements. Some of them are phenotypical elements (the RGBA states) and the remaining 12 serve arbitrary purposes, used for storing and communicating information. We can perturb the states of these cells to hijack the overall system in certain ways (the discovery of such perturbation strategies is a key goal of biomedicine and synthetic morphology). There are a variety of ways we can perform state perturbations. We will focus on *global state perturbations*, defined as perturbations that are applied on every living cell at every time step (analogous to "systemic" biomedical interventions, that are given to the whole organism (e.g., a chemical taken internally), as opposed to highly localized delivery systems). The new goal is to discover a certain type of global state perturbation that results in a stable new pattern.

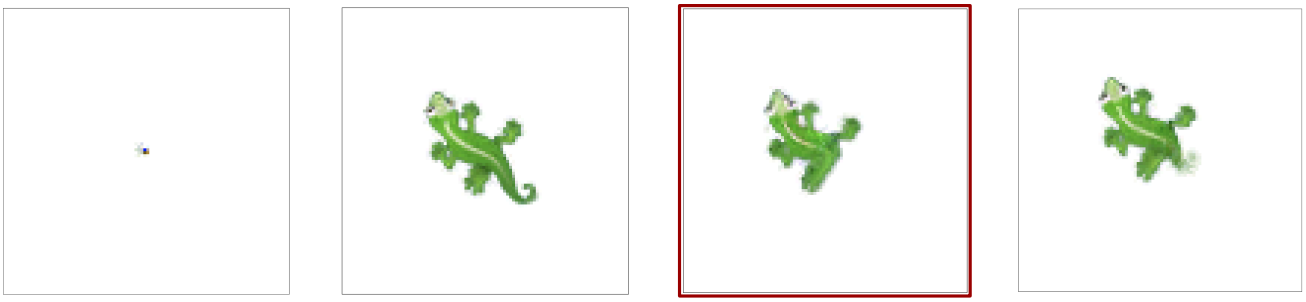


Figure 9: Diagram showing some possible stages for perturbing a lizard pattern. (a) We start from a seed that grows into a lizard (b) Fully converged lizard. (c) We apply a global state perturbation at every step. As a result, the lizard loses its tail. (d) We stop perturbing the state. We observe the lizard immediately grows back its tail.

We show 6 target patterns: the tailless and red lizard from the previous experiment, plus a blue lizard and lizards with various severed limbs and severed head.

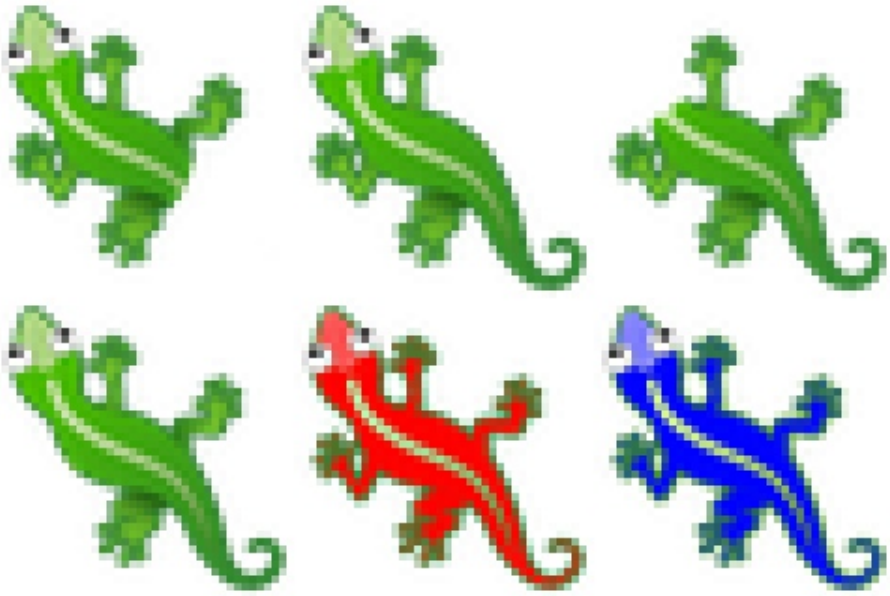


Figure 10: Mosaic of the desired mutations we want to apply.

We decided to experiment with a simple type of global state perturbation: applying a symmetric 16×16 matrix multiplication A to every living cell at every step [2]. To give insight on why we chose this, an even simpler "state addition" mutation (a mutation consisting only of the addition of a vector to every state) would be insufficient because the value of the states of our models are unbounded, and often we would want to suppress something by setting it to zero. The latter is generally impossible with constant state additions, as a constant addition or subtraction of a value would generally lead to infinity, except for some fortunate cases where the natural residual updates of the cells would cancel out with the constant addition at precisely state value zero. However, matrix multiplications have the possibility of amplifying/suppressing combinations of elements in the states: multiplying a state value repeatedly for a constant value less than one can easily suppress a state value to zero. We constrain the matrix to be symmetric for reasons that will become clear in the following section.

We initialize A with the identity matrix I and train A just as we would train the original Growing CA, albeit with the following differences:

- We perform a global state perturbation as described above, using A, at every step.
- The underlying CA parameters are frozen and we only train A.
- We consider the set of initial image configurations to be both the seed state and the state with a fully grown lizard (as opposed to the Growing CA article, where initial configurations consisted of the seed state only).

Figure 11: Effect of applying the trained perturbations.

The video above shows the model successfully discovering global state perturbations able to change a target pattern to a desired variation. We show what happens when we stop perturbing the states (an out-of-training situation) at step 500 through step 1000, then reapplying the mutation. This demonstrates the ability of our perturbations to achieve the desired result both when starting from a seed, and when starting from a fully grown pattern. Furthermore it demonstrates that the original CA easily recover from these state perturbations once it goes away. This last result is perhaps not surprising given how robust growing CA models are in general.

Not all perturbations are equally effective. In particular, the headless perturbation is the least successful as it results in a loss of other details across the whole lizard pattern such as the white coloring on its back. We hypothesize that the best perturbation our training regime managed to find, due to the simplicity of the perturbation, was suppressing a "structure" that contained both the morphology of the head and the white colouring. This may be related to the concept of differentiation and distinction of biological organs. Predicting what kinds of perturbations would be harder or impossible to be done, before trying them out empirically, is still an open research question in biology. On the other hand, a variant of this kind of synthetic analysis might help with defining higher order structures within biological and synthetic systems.

Directions and compositionality of perturbations

Our choice of using a symmetric matrix for representing global state perturbations is justified by a desire to have compositionality. Every complex symmetric matrix A can be diagonalized as follows:

$$A = Q\Lambda Q^{\intercal}$$

where Λ is the diagonal eigenvalues matrix and Q is the unitary matrix of its eigenvectors. Another way of seeing this is applying a change of basis transformation, scaling each component proportional to the eigenvalues, and then changing back to the original basis. This should also give a clearer intuition on the ease of suppressing or amplifying combinations of states. Moreover, we can now infer what would happen if all the eigenvalues were to be one. In that case, we would naturally have $QIQ^{\intercal} = I$ resulting in a no-op (no change): the lizard would grow as if no perturbation was performed. We can now decompose $Q\Lambda Q^{\intercal} = Q(D + I)Q^{\intercal}$ where D is the *perturbation direction* ($\Lambda - I$) in the "eigenvalue space". Suppose we use a coefficient k to scale D: $A_k = Q(kD + I)Q^{\intercal}$. If $k = 1$, we are left with the original perturbation A and when $k = 0$, we have the no-op I. Naturally, one question would be whether we can explore other values for k and discover meaningful perturbations. Since

$$A_k = Q(kD + I)Q^{\intercal} = kA + (1 - k)I$$

we do not even have to compute eigenvalues and eigenvectors and we can simply scale A and I accordingly.

Let us then take the tailless perturbation and see what happens as we vary k:

Figure 12: Effect of the interpolation between an identity matrix and the 'perturbation direction of the tail perturbation.

As we change $k = 1$ to $k = 0$ we can observe the tail becoming more complete. Surprisingly, if we make k negative, the lizard grows a longer tail. Unfortunately, the further away we go, the more unstable the system becomes and eventually the lizard pattern grows in an unbounded fashion. This behaviour likely stems from that perturbations applied on the states also affect the homeostatic regulation of the system, making some cells die out or grow in different ways than before, resulting in a behavior akin to "cancer" in biological systems.

Can we perform multiple, individually trained, perturbations at the same time?

Suppose we have two perturbations A and B and their eigenvectors are the same (or, more realistically, sufficiently similar). Then, $A_k = Q(k_A D_A + I)Q^\intercal$ and $B_k = Q(k_B D_B + I)Q^\intercal$.

In that case,

$$comb(A_k, B_k) = Q(k_A D_A + k_B D_B + I)Q^\intercal = k_A A + k_B B + (1 - k_A - k_B)I$$

would result in something meaningful. At the very least, if $A = B$, setting $k_A = k_B = 0.5$ would result in exactly the same perturbation.

We note that D_A and D_B are effectively a displacement from the identity I and we have empirically observed how given any trained displacement D_A, for $0 \leq k_A \leq 1$ adding $k_A D_A$ results in a stable perturbation. We then hypothesize that as long as we have two perturbations whose positive directions k are $k_A + k_B \leq 1$, this could result in a stable perturbation. An intuitive understanding of this is interpolating stable perturbations using the direction coefficients.

In practice, however, the eigenvectors are also different, so the results of the combination will likely be worse the more different the respective eigenvector bases are.

Below, we interpolate the direction coefficients, while keeping their sum to be one, of two types of perturbations: tailless and no-leg lizards.

Figure 13: Effect of composing two trained perturbations while keeping the sum of ks as 1.

While it largely achieves what we expect, we observe some unintended effects such as the whole pattern starting to traverse vertically in the grid. Similar results happen with other combinations of perturbations. What happens if we remove the restriction of the sum of ks being equal to one, and instead add both perturbations in their entirety? We know that if the two perturbations were the same, we would end twice as far away from the identity perturbation, and in general we expect the variance of these perturbations to increase. Effectively, this means going further and further away from the stable perturbations discovered during training. We would expect more unintended effects that may disrupt the CA as the sum of ks increases.

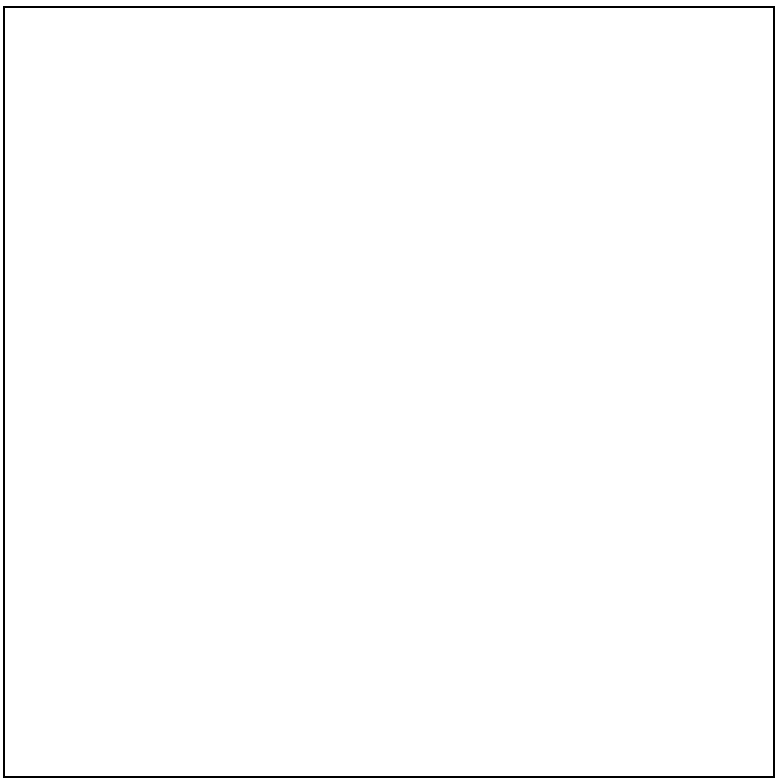
Below, we demonstrate what happens when we combine the tailless and the no-leg lizard perturbations at their fullest. Note that when we set both ks to one, the resulting perturbation is equal to the sum of the two perturbations minus an identity matrix.

0:03 / 0:18

Figure 14: Effect of composing two perturbations.

Surprisingly, the resulting pattern is almost as desired. However, it also suffers from the vertical movement of the pattern observed while interpolating ks.

This framework can be generalized to any arbitrary number of perturbations. Below, we have created a small playground that allows the reader to input their desired combinations. Empirically, we were surprised by how many of these combinations result in the intended perturbations and qualitatively it appears that bounding k to one results in generally more stable patterns. We also observed how exploring negative k values is usually more unstable.



Related work

This work is inspired by Generative Adversarial Networks (GANs) [12]. While with GANs it is typical to cotrain pairs of models, in this work we froze the original CA and trained the adversaries only. This setup is to the greatest degree inspired by the seminal work *Adversarial Reprogramming of Neural Networks* [13].

The kinds of state perturbations performed in this article can be seen as targeted latent state manipulations. Word2vec [14] shows how latent vector representations can have compositional properties and Fader Networks [15] show similar behaviors for image processing. Both of these works and their related work were of inspiration to us.

Influence maximization

Adversarial cellular automata have parallels to the field of influence maximization. Influence maximization involves determining the optimal nodes to influence in order to maximize influence over an entire graph, commonly a social graph, with the property that nodes can in turn influence their neighbours. Such models are used to model a wide variety of real-world applications involving information spread in a graph. [16] [17] [18] A common setting is that each vertex in a graph has a binary state, which will change if and only if a sufficient fraction of its neighbours' states switch. Examples of such models are social influence maximization (maximally spreading an idea in a network of people), contagion outbreak modelling [19] (usually to minimize the spread of a disease in a network of people) and cascade modeling [20] (when small perturbations to a system bring about a larger 'phase change'). At the time of writing this article, for instance, contagion minimization is a model of particular interest. NCA are a graph - each cell is a vertex and has edges to its eight neighbours, through which it can pass information. This graph and message structure is significantly more complex than the typical graph underlying much of the research in influence maximization, because NCA cells pass vector-valued messages and have a complex update rules for their internal states, whereas graphs in influence maximization research typically consist of more simple binary cells states and threshold functions on edges determining whether a node has switched states. Many concepts from the field could be applied and are of interest, however.

For example, in this work, we have made an assumption that our adversaries can be positioned anywhere in a structure to achieve a desired behaviour. A common focus of investigation in influence maximization problems is deciding which nodes in a graph will result in maximal influence on the graph, referred to as target set selection [21]. This problem isn't always tractable, often NP-hard, and solutions frequently involve simulations. Future work on adversarial NCA may involve applying techniques from influence maximization in order to find the optimal placement of adversarial cells.

Discussion

This article showed two different kinds of adversarial attacks on Neural CA.

Injections of adversarial CA in a pretrained Self-classifying MNIST CA showed how an existing system of cells that are heavily reliant on the passing of information among each other is easily swayed by deceitful signaling. This problem is routinely faced by biological systems, which face hijacking of behavioral, physiological, and morphological regulatory mechanisms by parasites and other agents in the biosphere with which they compete. Future work in this field of computer technology can benefit from research on biological communication mechanisms to understand how cells maximize reliability and fidelity of inter- and intra-cellular messages required to implement adaptive outcomes.

The adversarial injection attack was much less effective against Growing CA and resulted in overall unstable CA. This dynamic is also of importance to the scaling of control mechanisms (swarm robotics and nested architectures): a key step in "multicellularity" (joining together to form larger systems from sub-agents [22]) is informational fusion, which makes it difficult to identify the source of signals and memory engrams. An optimal architecture would need to balance the need for validating control messages with a possibility of flexible merging of subunits, which wipes out metadata about the specific source of informational signals. Likewise, the ability to respond successfully to novel environmental challenges is an important goal for autonomous artificial systems, which may import from biology strategies that optimize tradeoff between maintaining a specific set of signals and being flexible enough to establish novel signaling regimes when needed.

The global state perturbation experiment on Growing CA shows how it is still possible to hijack these CA towards stable out-of-training configurations and how these kinds of attacks are somewhat composable in a similar way to how embedding spaces are manipulable in the natural language processing and computer vision fields [14, 15]. However, this experiment failed to discover stable out-of-training configurations that persist *after the perturbation was lifted*. We hypothesize that this is partially due to the regenerative capabilities of the pretrained CA, and that other models may be less capable of recovery from arbitrary perturbations.

Acknowledgments

We thank Hananel Hazan and Nick Moran for their valuable conversations and feedback.

Author Contributions

Research: Ettore designed and performed the experiments in this article. Alexander and Michael gave advisorship throughout the process.

Demos: Ettore, Alexander and Eyvind contributed to the demo.

Writing and Diagrams: Ettore outlined the structure of the article and contributed to the content throughout. Eyvind contributed to the content throughout. Michael made extensive contributions to the article text, providing the biological context for this work.

Footnotes

1. The still-image of the video is on step 500, and the video stops for a bit more than a second on step 500. [↩]

2. In practice, we also clip the state of cells such that they are bounded in $[-3, +3]$. This is a minor detail and it helps stabilise the model. [↩]

References

1. Growing Neural Cellular Automata
Mordvintsev, A., Randazzo, E., Niklasson, E. and Levin, M., 2020. Distill. DOI: 10.23915/distill.00023

2. Self-classifying MNIST Digits
Randazzo, E., Mordvintsev, A., Niklasson, E., Levin, M. and Greydanus, S., 2020. Distill. DOI: 10.23915/distill.00027.002

3. Herpes Simplex Virus: The Hostile Guest That Takes Over Your Home
Banerjee, A., Kulkarni, S. and Mukherjee, A., 2020. Front Microbiol, Vol 11, pp. 733. DOI: 10.3389/fmicb.2020.00733

4. The role of gut microbiota (commensal bacteria) and the mucosal barrier in the pathogenesis of inflammatory and autoimmune diseases and cancer: contribution of germ-free and gnotobiotic animal models of human diseases
Tlaskalová-Hogenová, H., Stěpánková, R., H., K., Hudcovic, T., Vannucci, L., Tučková, L., Rossmann, P., Hrnčíř, T., Kverka, M., Zákostelská, Z., Klimešová, K., Přibylová, J., Bártová, J., Sanchez, D., Fundová, P., Borovská, D., Srůtková, D., Zídek, Z., Schwarzer, M., Drastich, P. and Funda, D.P., 2011. Cell Mol Immunol, Vol 8(2), pp. 110--120. DOI: 10.1038/cmi.2010.67

5. Regulation of axial and head patterning during planarian regeneration by a commensal bacterium
Williams, K.B., Bischof, J., Lee, F.J., Miller, K.A., LaPalme, J.V., Wolfe, B.E. and Levin, M., 2020. Mech Dev, Vol 163, pp. 103614. DOI: 10.1016/j.mod.2020.103614

6. Toxoplasma gondii infection and behavioral outcomes in humans: a systematic review
Martinez, V.O., Mendonça Lima, F.W., Carvalho, C.F. and Menezes-Filho, J.A., 2018. Parasitology research, Vol 117, pp. 3059-3065. DOI: 10.1007/s00436-018-6040-2

7. Resting potential, oncogene-induced tumorigenesis, and metastasis: the bioelectric basis of cancer in vivo
Lobikin, M., Chernet, B., Lobo, D. and Levin, M., 2012. Physical biology, Vol 9. DOI: 10.1088/1478-3975/9/6/065002

8. Transmembrane voltage potential of somatic cells controls oncogene-mediated tumorigenesis at long-range
Chernet, B.T. and Levin, M., 2014. Oncotarget. DOI: 10.18632/oncotarget.1935

9. Cross-limb communication during Xenopus hindlimb regenerative response: non-local bioelectric injury signals
Busse, S.M., McMillen, P.T. and Levin, M., 2018. Development. DOI: 10.1242/dev.164210

10. Local and long-range endogenous resting potential gradients antagonistically regulate apoptosis and proliferation in the embryonic CNS

Pai, V.P., Lemire, J.M., Chen, Y., Lin, G. and Levin, M., 2015. The International journal of developmental biology. DOI: 10.1387/ijdb.150197ml

11. Top-down models in biology: explanation and control of complex living systems above the molecular level
Pezzulo, G. and Levin, M., 2016. Journal of the Royal Society, Interface. DOI: 10.1098/rsif.2016.0555

12. Generative Adversarial Networks
Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y., 2014.

13. Adversarial Reprogramming of Neural Networks
Elsayed, G.F., Goodfellow, I. and Sohl-Dickstein, J., 2018.

14. Efficient Estimation of Word Representations in Vector Space
Mikolov, T., Chen, K., Corrado, G. and Dean, J., 2013.

15. Fader Networks: Manipulating Images by Sliding Attributes
Lample, G., Zeghidour, N., Usunier, N., Bordes, A., Denoyer, L. and Ranzato, M., 2018.

16. Maximizing the spread of influence through a social network [PDF]
Kempe, D., Kleinberg, J. and Tardos, É., 2003. Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '03. ACM Press. DOI: 10.1145/956755.956769

17. The Independent Cascade and Linear Threshold Models [link]
Shakarian, P., Bhatnagar, A., Aleali, A., Shaabani, E. and Guo, R., 2015. Diffusion in Social Networks, pp. 35--48. Springer International Publishing. DOI: 10.1007/978-3-319-23105-1_4

18. A Survey on Influence Maximization in a Social Network [PDF]
Banerjee, S., Jenamani, M. and Pratihar, D.K., 2018. arXiv [cs.SI].

19. Simplicial models of social contagion [link]
Iacopini, I., Petri, G., Barrat, A. and Latora, V., 2019. Nature communications, Vol 10(1), pp. 2485. DOI: 10.1038/s41467-019-10431-6

20. Cascading Behavior in Networks: Algorithmic and Economic Issues [link]
Kleinberg, J., 2007. Algorithmic Game Theory, pp. 613–632. Cambridge University Press. DOI: 10.1017/CBO9780511800481.026

21. On the Approximability of Influence in Social Networks [link]
Chen, N., 2009. SIAM Journal on Discrete Mathematics, Vol 23(3), pp. 1400–1415. Society for Industrial and Applied Mathematics. DOI: 10.1137/08073617X

22. The Computational Boundary of a "Self": Developmental Bioelectricity Drives Multicellularity and Scale-Free Cognition [link]
Levin, M., 2019. Frontiers in Psychology, Vol 10, pp. 2688. DOI: 10.3389/fpsyg.2019.02688

Accessibility

This article relies on using color to demonstrate classification label. If you have trouble distinguishing the colours of lizards or of the digits in the above legend, please try and adjust the slider above to see if there is an alternative colour palette for you. The chosen palette will propagate throughout the article.

Updates and Corrections

If you see mistakes or want to suggest changes, please create an issue on GitHub.

Reuse

Citation

For attribution in academic contexts, please cite this work as

```
Randazzo, et al., "Adversarial Reprogramming of Neural Cellular Automata", Distill, 2021.
```

BibTeX citation

```
@article{randazzo2021adversarial,
  author = {Randazzo, Ettore and Mordvintsev, Alexander and Niklasson, Eyvind and Levin,
Michael},
  title = {Adversarial Reprogramming of Neural Cellular Automata},
  journal = {Distill},
  year = {2021},
  note = {https://distill.pub/selforg/2021/adversarial},
  doi = {10.23915/distill.00027.004}
}
```