Evolving Novel Cellular Automaton Seeds Using Compositional Pattern Producing Networks (CPPN)

Joshuah Wolper
Swarthmore College
Deptartment of Computer Science
500 College Avenue
Swarthmore, PA 19081
joshwolper@gmail.com

George Abraham
Swarthmore College
Deptartment of Computer Science
500 College Avenue
Swarthmore, PA 19081
gabraha1@swarthmore.edu

ABSTRACT

The aim of this study is to evolve novel seeds for John Conway's Game of Life cellular automaton (CA) with Compositional Pattern Producing Networks (CPPNs), a variation of artificial neural networks known to evolve organic patterns when used to process visual data. CPPNs were evolved using both objective search (implemented with NeuroEvolution of Augmenting Topologies) and novelty search, which focuses on finding novel solutions rather than objectively "fitter" solutions. Objective search quickly evolved game of life solutions that converged to trivial combinations of previously known solutions. However, novelty search produced non-trivial symmetries and complex high period oscillators such as the period 15 pentadecathlon. Regardless, neither approach evolved purely novel or undocumented seeds. Despite this failure, the complex evolved solutions demonstrate that CPPNs can serve as a powerful encoding for cellular automata seeds. As such, these results stand as the first baseline for further exploration into encoding cellular automata using CPPN.

1. INTRODUCTION

Recent studies have suggested the importance of cellular automata (CA) in modeling large, or inherently stochastic data sets [4]. However, depending on the resolution of the CA being studied, machine learning algorithms are shown to out-perform traditional computational analysis methods [1]. Conway's Game of Life is a zero-player CA that takes in an input set of dead and alive cells (called a "seed"), and performs a series of updates until the board is completely dead (i.e. empty). Given that many sample solutions to this tend to be symmetric, Compositional Pattern Producing Networks (CPPNs) show promise in evolving novel Game of Life solutions due to the inherent symmetries of the CPPN basis fitness functions [6]. Although artificial neural networks have been applied to CAs before [1], this study is the first to apply CPPNs.

2. EXPERIMENTAL SETUP

The Game of Life board implementation was implemented with a toroidal topology in order to constrain all behavior to a small visible space. Then, for each cell of the board, four values were given to the CPPN to process: x and y values, Euclidean distance from center, and a bias value of 1 (Figure 1).

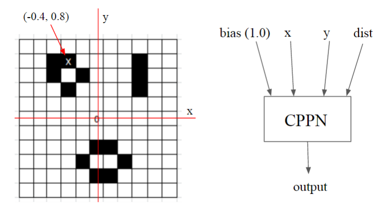


Figure 1: Example Game of Life seed illustrating the underlying coordinate system as well as the inputs and outputs to the CPPN

Neuroevolution of Augmenting Topologies (NEAT) was used to evolve a population of 100 CPPNs to seek novel seed patterns. This required the development of a fitness function in order to allow NEAT to maximize objective fitness. Three fitness functions were developed for this purpose: rewarding solution lifetime, rewarding mass and lifetime, and rewarding lifetime while punishing mass. Lifetime was defined as the number of simulation steps before a trivial solution arose (still lifes, or low period oscillators), while mass was defined as the number of living cells at the end of simulation time.

Although NEAT has demonstrated success as an evolutionary computation algorithm, a major criticism of NEAT is that it tends to converge to sub–optimal solutions when confronted with deceptive tasks. In a previous study [3], when a robot was confronted with a deceptive maze problem that required the robot to initially travel away from the desired goal state, a computational mechanism known as novelty search outperformed NEAT by evolving a wide variety of solutions that broke out of a local maximum. Novelty search is a variation of NEAT in which the CPPNs in the evolving population were rewarded based on novelty or uniqueness, rather than objective fitness models such as those outlined for NEAT. Hence, novelty search was chosen here to evolve a diverse set of CA seeds, given that NEAT may evolve previously known solutions.

The main difficulty in using novelty search is simply defining the sparseness metric, or how different one seed pattern is from another previously seen seed. Sparseness is calculated using a system of "behaviors" that are compared to one another. Each seed has a behavior associated with it: a tuple with a length equal to the number of grid spaces in the seed patterns. Each element of the behavior tuple is either (0,0) indicating that the given grid space has a dead cell, or it has the coordinate pair associated with the grid space to

GECCO '16 July 20-24, 2016, Denver, CO, USA

ACM ISBN 978-1-4503-4323-7/16/07.
DOI: http://dx.doi.org/10.1145/2908961.2908964

indicate a living cell. The k nearest neighbors are then compared by finding the Euclidean distance between each pair of behavior elements for each pairing with the neighbors.

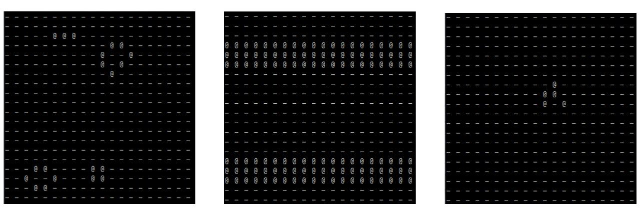
It is important to note that the behavior tuples are formed from the initial seed pattern generated by the CPPN, rather than the final state of the game board after the game has been played. This decision was made to avoid a subtle pitfall of this experimental setup: a solution that has identical end behavior to another solution may receive a high sparseness metric if it is ever so slightly out of phase from the other solution. Thus any number of identical end solutions would all have a high sparseness metric given that they are all out of phase from one another when the simulation ends.

3. RESULTS

Despite initial expectations, many of the fitness functions used in our objective search experiments led to extremely similar end behavior. The fitness function rewarding the seed pattern's lifetime until a still life solution tended to produce similar, somewhat trivial end behavior in every generation: simple patterns of still lives, and occasionally, oscillators of period two called "blinkers" (Figure 2a).

The fitness function rewarding both lifetime until a pure still life solution and the total mass of end behavior solution scarcely ever produced anything save for high period oscillators that were named "wave oscillators" (Figure 2b), as they have bands that move up and down the board in wave like patterns. This was marked as a trivial solution despite its high period since its existence is only possible due to the toroidal nature of the board.

The fitness function rewarding seeds that live longer before converging to a period two oscillator solution and punishing high end state mass predominantly evolved gliders (Figure 2c), small locomotive masses. However, none of the objective fitness functions were able to evolve high period symmetrical oscillators as hoped.



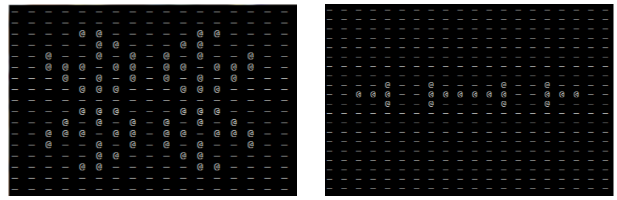
(a) Blinker (b) Wave Oscillator (c) Glider

Figure 2: Examples of solutions generated by fitness functions 1, 2, and 3 respectively

When evolving the CPPNs using novelty search and the sparseness metric, two new types of end behaviors emerged. Two unique high period oscillators were evolved: one of period three named the "pulsar" (Fig. 3a) and one of period 15 named the "pentadecathlon" (Fig. 3b). See Section 5 for links to videos showcasing a visualization of both the pulsar and the pentadecathlon. Although these tests failed to produce purely novel or undocumented solutions, the evolutionary computation produced seeds that had a significantly higher lifetime than random seeds.

4. CONCLUSION

In this work evolving cellular automata seeds using CPPNs was explored, expanding on extensive work done on finding organic symmetries and repetitions using CPPNs for image grid processing by Cheney, MacCurdy, and Clune et. al [2]. Unique high mass high period oscillators were evolved, which is impressive in that this stems from a machine learning extension of a game that already attempts to model organic phenomena.



(a) Pulsar, period 3 (b) Pentadecathlon, period 15

Figure 3: Complex, high period oscillators, moved to the center of the board for ease of viewing

Regardless of the lack of novel and undocumented evolved solutions, the results demonstrate that CPPNs can be used to richly encode complex cellular automata seeds. Perhaps the best direction for future study based on this finding is applying "quality diversity" algorithms to try and evolve CPPNs in order to combine objective and novelty search [5].

5. VIDEO LINKS

Video	Link
Fitness Function 1 Example	https://goo.gl/2Ouzw2
Fitness Function 2 Example	https://goo.gl/YDAiGl
Fitness Function 3 Example	https://goo.gl/ipta7L
Actual Pulsar Run	https://goo.gl/P3ooPH
Actual Pentadecathlon Run	https://goo.gl/KtkxXM
Normalized Pulsar	https://goo.gl/5HW3hU
Normalized Pentadecathlon Run	https://goo.gl/uPJZM9

Table 1: Video Links

6. REFERENCES

[1] O. Charif, R.-M. Basse, H. Omrani, and P. Trigano. Cellular automata model based on machine learning methods for simulating land use change. In O. Rose and A. M. Uhrmacher, editors, *Winter Simulation Conference*, pages 163:1–163:12. WSC, 2012.

[2] N. Cheney, R. MacCurdy, J. Clune, and H. Lipson. Unshackling evolution: Evolving soft robots with multiple materials and a powerful generative encoding. *SIGEVOlution*, 7(1):11–23, Aug. 2014.

[3] J. Lehman and K. O. Stanley. Abandoning objectives: Evolution through the search for novelty alone. *Evol. Comput.*, 19(2):189–223, June 2011.

[4] S. J. Louis and G. L. Raines. Genetic algorithm calibration of probabilistic cellular automata for modeling mining permit activity. In *ICTAI*, pages 515–519. IEEE Computer Society, 2003.

[5] J. K. Pugh, L. B. Soros, P. A. Szerlip, and K. O. Stanley. Confronting the challenge of quality diversity. In *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation*, GECCO '15, pages 967–974, New York, NY, USA, 2015. ACM.

[6] K. O. Stanley. Compositional pattern producing networks: A novel abstraction of development. *Genetic Programming and Evolvable Machines*, 8(2):131–162, June 2007.