Evolutionary Cellular Automata Bonsai

Daniel Ashlock and Carolyn Pugh

Abstract—Cellular automata are known to be capable of Turing-complete computation and yet “programming” them to do particular tasks can be quite daunting. In this paper we use single parent crossover as a means of transferring information between successive evolving populations to create rules for cellular automata that have proscribed shapes. The proscription of regions where the automata are permitted to grow is the reason they are called bonsai automata. This work follows earlier work on apoptotic cellular automata that simply exhibit self-limited growth. The correct choice of single parents permits enormous improvement in the performance of evolutionary algorithms searching for automata that satisfy particular bonsai templates. In this study, we demonstrate that single parent techniques make meeting shape constraints on the growth of CAs possible at all in some cases. This study also introduces range niche specialization to control problems with the cloning of ancestors used for single parent crossover in an evolving population. This study demonstrates that different bonsai shapes have highly variable difficulty. It is also shown that automata evolved to satisfy one bonsai template may be needed to enable, via single parent crossover, solutions for another template. The use of bonsai techniques yields many automata not found during studies of apoptotic automata demonstrating that the technique encourages exploration of different parts of the fitness landscape.

I. INTRODUCTION

This study demonstrates how to use the single parent crossover operator to control and significantly enhance evolutionary search for cellular automata (CA) constrained to grow in a particular shape. We call these automata bonsai automata. Cellular automata are a type of discrete model of computation. A cellular automaton has three parts:

1) A collection of cells divided into neighborhoods of each cell,  
2) A set of states that cells can have,  
3) A rule that maps the set of possible cell states of a neighborhood to a new state for the cell with which the neighborhood is associated.

CA are discrete dynamical systems that exhibit self-organizing behavior. A cell population is updated according to local transition rules. The updating may be synchronous, as it is in this study, or asynchronous. Many real world systems are dynamic in nature, and can be modeled as CA. They can be used as models for complex natural systems that contain large numbers of identical components experiencing local interactions[23], [17]. This study builds on and applies material developed in an early study investigating the fitness landscape of the type of CA used here[4].

CA have been applied to the study of a diverse range of topics, such as structure formation[8], heat conduction[9], language recognition[16], traffic dynamics[13] and cryptography[2], to name a few. CA have also been used for more aesthetic purposes, such as image and sound generation. Serquera and Miranda of the Interdisciplinary Center for Computer Music Research, UK, have published many works on the use of CA for sound synthesis[18], [1]. Much of their work consists of mapping the histogram sequence of a CA evolution onto a sound spectrogram, which produces spectral structures evolving in time. It is claimed that the mapping produces a “natural” behavior, and can replicate acoustic instruments[19].

CA have also been applied in the arts. They have been used to produce artistic images[5], [15], and their use has been extended to the fields of architecture and urban design[20], [11]. An interesting application has been the use of CA in simulating the emergence of the complex architectural features found in ancient Indonesian structures, such as the Borobudur Temple[21]. Ashlock and Tsang[5] produced evolved art using 1-dimensional CA rules. CA rules were evolved using a string representation. The CA either underwent slow persistent growth, or planned senescence. The resulting fitness landscapes were conjectured to be rugged with many local optima. These systems produced aesthetically pleasing images. This study revisits these fitness functions with a more general space of cellular automata rules, focusing on understanding the fitness landscape.

In this study we examine the use of single parent techniques, shown to be effective at enhancing fitness when evolving CA[4], to accomplish difficult constrained growth tasks for cellular automata. The goal of this study is to locate automata that automatically grow only inside specified shapes. This gives an artist or game designer the ability to generate multiple image elements or decorations that must fit in a particular shape with a possibly complex boundary.

The remainder of this study is structured as follows. In Section II we carefully define the cellular automata we are using, define the single parent operator, and give the design of the evolutionary algorithm. In Section III we give the design of experiments. In section IV we present and discuss results. In Section V we draw conclusions and outline potential next steps.

II. TECHNICAL DEFINITIONS

The CA used in this study are specializations of the apoptotic automata used in the earlier study. The cell states for the automata are the numbers \{0, 1, 2, 3, 4, 5, 6, 7\}. The automata are one-dimensional arrays of 201 or 401 cells that wrap at the ends. The neighborhood of each cell consists of
the cell itself and two cells to the left and right of it, yielding a neighborhood consisting of five adjacent cells. The rules for neighborhood updating are the target of evolution. They are represented as arrays of 36 cell states. Rules are computed as follows - the numbers in the five cells comprising a neighborhood are summed, yielding a number in the range 0-35. This number is used as an index to look up the new state in the array. State zero is designated at the quiescent state of the automata and the first entry is always forced to be zero. This means in an all zero neighborhood, the rule updated the cell to zero in the next time step. The one-dimensional automata generates a two-dimensional picture by rendering a time history on successive rows of a raster graphics picture.

**Definition 1:** A cellular automata rule is apoptotic if after some finite number of updatings all cells are mapped to the zero state. The automata is \( n \)-apoptotic if the all zero state arrives before the \( n \)th updating.

**Definition 2:** A shape \( S \) is a selection of a connected group of cells for a range of updatings (time-steps). The shape selects a region of the area a cellular automata might grow in over time.

**Definition 3:** A cellular automata rule is a bonsai rule for a shape \( S \) if all non-zero cells in a rendering of the automata are pixels selected by \( S \).

Notice that if \( S \) is finite then bonsai cellular automata are necessarily apoptotic. Two examples of shapes, called the U-bend and trident are given, together with examples of bonsai cellular automata, in Figure 1. The goal of this study is to locate automata that automatically grow only inside specified shapes. All three bonsai shapes are given in Figure 2.

![Fig. 1. Examples of bonsai cellular automata for two different shapes. The shapes are given by gray shading.](image)

**III. DESIGN OF EXPERIMENTS**

All experiments in this study use an evolutionary algorithm derived from (and with parameters set in) Ashlock and McNicholas [4]. It uses a population of 1000 cellular automata rules. The algorithm is steady state [22] and selection is performed with single tournament selection of size seven [3]. This model of evolution chooses a single group of chromosomes and then runs reproduction in which the two best members of the group replace the two worst members of the group. The rate of single parent crossover is set to 50% with the usual two point crossover used otherwise. This means that when a tournament of seven is selected, the two most fit rules are copied and then each copy either selects an ancestor independently and uniformly at random and undergoes crossover with the selected ancestor or the copies engage in standard two point crossover. New rules are also subjected to 1-3 mutations with the number of mutations selected uniformly at random. Standard algorithms are run for 40,000 selection events with summary fitness statistics saved every 400 selection events. Each experiment consists of 30 independent runs of the evolutionary algorithm and yields 30 best-of-run cellular automata rules. These rules are available to be ancestors in subsequent experiments.

There are two factors that distinguish experiments: the choice of shape to control the bonsai behavior of the algorithm and the choice of the ancestor set. Control experiments are also run in which the rate of single parent crossover is set to zero but the single parents are still present. This is needed as part of a cloning prevention technique. In [4] it was found that some rules use only a fraction of their loci during fitness evaluation. It is easy for SPC to create functional duplicates - differing only in unused loci - of these ancestors. It is also possible to simply copy an ancestor with an inopportune SPC. Niche specialization [10], [12] reduces the fitness of a
population member in proportion to the number of similar population members to encourage diversity and so avoid local optima. We solve the problem of functional cloning of ancestors with a variation of niche specialization, described in the next section.

A. Fitness Evaluation

For apoptotic cellular automata, fitness was evaluated as the number of non-quiescent cells, except that a fitness of zero was awarded if the automata continued growing too long. The notion of “too long” is a fixed number of time steps set individually for each experiment. The fitness function for cellular automata bonsai is similar. Fitness is the number of non-quiescent cells generated by the automata unless it leaves the shape controlling the bonsai, in which case fitness is zero. This fitness function is also modified to exclude the possibility of cloning an ancestor.

The principle of niche specialization is to reduce the fitness of a member of the population in proportion to the number of other, similar genes. Similarity is typically measured between genotypes. This varies from simple, in the case of real parameter optimization where Euclidean distance may be used to build easy, rapidly computed similarity measures, to domains like genetic programming where similarity is measured with exceedingly difficult edit metrics. Since some positions in a CA rule may be unused, direct comparison of genomes can be almost meaningless. Since we only need to exclude clones, we award a fitness of zero to any rule that produces a raw fitness equal to the fitness of an ancestor. This is a form of niche specialization, albeit one with a very blunt similarity measure. We call it range niche specialization because it takes place in the range of the fitness function. This form of niche specialization excludes the possibility of cloning as well as, potentially, excluding some novel, good results; since the fitness range includes whole numbers up to several tens of thousands, this is not a large problem. To put it another way, range niche specialization is well suited to the cellular automata bonsai problem because is very cheaply excludes clones.

The fitness evaluation of a cellular automata rule requires that initial conditions be specified. All fitness evaluations in this manuscript use a row consisting of zeros with the middle three cells of the state space filled with $A, B, A$, with the exception of 48 experiments designed to provide a rich set of ancestral rules, $A = 1$ and $B = 2$. The ancestral experiments used all possible pairs that are not both zero: 010, 020, . . . , 101, 111, 121, . . . , 767, 777. Three bonsai shapes are used, see Figure 2.

B. The Pattern of Ancestry

The experiments in this study are built on one another, using some or all of the best-of-run results of one experiment and the ancestor set for another. Table I numbers the experiments, gives the bonsai shape, and gives their nonstandard parameters, if any. Figure 3 shows the pattern of choice of ancestors. All best-of-run cellular automata rules at the base of an arrow are available as ancestors in the experiments at the point of the arrow. This diagram does not show five of the experiments which used single rules as ancestor sets. All possible sets of initial conditions were used in an initial block of experiments that used no bonsai shape. They were performed in a 201 by 201 square and rules that attempted to grow beyond the bottom of the square were awarded zero fitness. These experiments provided 1890 potential ancestors.

<table>
<thead>
<tr>
<th>#</th>
<th>Shape</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-48</td>
<td>none</td>
<td>Used to provide large ancestor set.</td>
</tr>
<tr>
<td>49</td>
<td>none</td>
<td>no experiment.</td>
</tr>
<tr>
<td>50</td>
<td>u-bend</td>
<td>400,000 mating events</td>
</tr>
<tr>
<td>51</td>
<td>u-bend</td>
<td>no SPC.</td>
</tr>
<tr>
<td>52</td>
<td>u-bend</td>
<td>400,000 mating events</td>
</tr>
<tr>
<td>53</td>
<td>u-bend</td>
<td>no SPC.</td>
</tr>
<tr>
<td>54</td>
<td>trident</td>
<td>no SPC.</td>
</tr>
<tr>
<td>55</td>
<td>trident</td>
<td>no SPC.</td>
</tr>
<tr>
<td>56</td>
<td>Snowman</td>
<td>no SPC.</td>
</tr>
<tr>
<td>57</td>
<td>Snowman</td>
<td>no SPC.</td>
</tr>
<tr>
<td>58</td>
<td>u-bend</td>
<td>unique ancestor</td>
</tr>
<tr>
<td>59</td>
<td>trident</td>
<td>unique ancestor</td>
</tr>
<tr>
<td>60</td>
<td>Snowman</td>
<td>unique ancestor</td>
</tr>
<tr>
<td>61</td>
<td>u-bend</td>
<td>unique ancestor</td>
</tr>
<tr>
<td>62</td>
<td>u-bend</td>
<td>unique ancestor</td>
</tr>
<tr>
<td>63</td>
<td>u-bend</td>
<td>unique ancestor</td>
</tr>
<tr>
<td>64</td>
<td>trident</td>
<td>unique ancestor</td>
</tr>
<tr>
<td>65</td>
<td>trident</td>
<td>unique ancestor</td>
</tr>
</tbody>
</table>

Fig. 2. The bonsai shapes used in this study. From left to right the u-bend, trident, and snowman.

Fig. 3. Patterns of ancestry for the experiments in this paper, other than the unique single parent experiments.

After the initial set of runs, a block or runs were performed to assess the degree to which SPC could help satisfy
bonsai fitness functions. For each shape, experiments were performed with no SPC and 50% SPC. Even when SPC was not used the ancestor set was still read in and its fitness values used for range niche specialization; this was done to ensure that the experiments remained comparable. For the u-bend shape, runs were performed, both with and without SPC, that were permitted to run for 10-fold more mating events.

To assess the impact of using ancestors already knowledgeable about a given shape, three more experiments were performed in which the results of an earlier bonsai experiment were provided as ancestor for another set of runs. Note that experiment 59 uses u-bend rules for its ancestor set. Finally, five rules were selected and used in unique single parent experiments in which a single rule was used as the entire ancestor set with 50% SPC.

C. Use of a large, diverse ancestor set.

An alert reader will have noticed that the initial block of ancestor-acquisition experiments used 48 different initial conditions while all other experiments used just $A = 1, B = 2$. Examine the automata in Figure 4. When a rule is improved by single parent crossover, one possible mechanism is that a piece of another rule was spliced in that expands a pattern that the current rule exhibits only at its terminus. Figure 4 gives an example of such a tail expansion. Since the terminal pattern is not constrained to have any particular form, the choice of an ancestor set able to use any set of initial conditions enhances the probability that one of the ancestor rules contained the information needed for a tail expansion.

![Fig. 4. A rule from Experiment 52 (left) and a rule produced in an experiment using it as a unique ancestor.](image)

IV. RESULTS AND DISCUSSION

Each experiment generates 30 fitness values, one for each of its best-of-run automata. A $t$-test can then be used for the null hypothesis that the mean of these values is not the same. Application of the $t$-test led to the following conclusions:

1) Additional evolution time, comparing experiments 50 and 53, was helpful to a small but significant degree when SPC was not used for the u-bend bonsai, $p = 0.04$.

2) Additional evolution time, comparing experiments 52 and 53, was helpful to a small but significant degree when SPC was not used for the u-bend bonsai, $p = 0.04$.

3) Use of SPC from the large diverse ancestor set was not helpful for any of the three bonsai shapes, u-bend - $p = 0.51$, trident - $p = 0.13$, snowman $p = 0.62$.

4) Use of SPC with ancestors evolved for the u-bend fitness function was astronomically significantly helpful for the u-bend fitness function, $p = 3.99E - 22$.

5) Use of SPC with ancestors evolved for the snowman fitness function was significantly helpful for the snowman fitness function $p = 0.0079$.

6) Use of SPC with ancestors evolved for the u-bend fitness function was significantly helpful for the trident fitness function $p = 0.021$. Note this experiment crosses fitness functions.

In[4] is was shown that large, diverse ancestor sets are very helpful in improving performance when evolving apoptotic cellular automata. In this study they were unhelpful and, in the case of the u-bend function, counterproductive (fitness was worse with SPC) though with no statistical significance. The success story lies in items 4-6 where having ancestors that had made a start on solving the shape-matching problem granted substantial benefits.

The reader may wonder why the trident-with-trident ancestors experiment was not performed. The reason for this is that both experiments, numbers 56 and 57, were failures. Examine Figure 5. All 60 evolutionary runs in these two experiments failed to use the left and right arms of the trident. When supplied with u-bend ancestors for SPC, however, results like those seen in Figure 6 were obtained.

A. Snowmen

The snowman shape was the easiest to deal with. Unlike the u-bend and trident shapes, most of the large diverse ancestor set have positive fitness for the snowman shape. This means that a search for better shapes begins immediately. Examples of results for the snowman shape are given in Figure 7.

B. Unique ancestor experiments

Experiments 61-65, where SPC was used with a single rule forming the ancestor set, followed on earlier research[4] that showed that the use of a single ancestor strongly localized search in the area of that ancestor. The best results for the trident shape all arise in experiments 64 and 65 which used trident competent ancestors that were themselves evolved using u-bend competent ancestors. The ancestors used are given in Figure 8.

In all five of the unique ancestor experiments, we see that the solutions discovered, while forced to be different from their parents by range niche specialization, have appearances that echo the appearance of their ancestor. These results strongly confirm the earlier result that SPC localizes search.

We now look at these five experiments individually. In doing
so we will see that the localization of search can have several different forms.

1) Experiment 61: This experiment demonstrates a type of negative result: the ancestor rule yields runs with very low fitness. Six examples of the outcomes of runs in this experiment are shown in Figure 10. Of the 30 runs in the experiment one did not have the dwarf fitness. This suggests
that the localization caused by the use of this ancestor is to a local optima that is difficult to escape. These automata are aesthetically pleasing with an elegant color scheme but, in spite of being initially evolved for the u-bend shape they have impaired ability to grow in that environment.

2) Experiment 62: The ancestor chosen for Experiment 62 was picked primarily for its appearance - it is visually distinct from most other rules- but it also does a good job of positioning itself above the arms of the u-bend. The first row of Figure 11 shows four of the outcomes of evolutionary runs in this experiment. The last of these has a different upper structure; it is the only one. All 29 other rules differed only in how they deal with exploiting the space in the arms of the u-bend. None of them do especially well - there is a weaker version of the trapping optima that we observed in Experiment 61.

3) Experiment 63: All 30 runs in this experiment dealt well with the u-bend, having higher fitness than their ancestor. Sadly, 29 of these were identical and one, with a higher fitness, was found once. These are shown in Figure 9. This is a different type of trapping optima. There is something better - and it is very easy to find - but it is masking still more interesting results.

4) Experiment 64: The ancestor used in Experiment 64 was one of the few in experiment 59 that managed to place cells in all three arms of the trident shape. In spite of this 27 of the 30 runs in this experiment ran only down the middle limb of the trident. One of these is shown in the last position of the second row of Figure 11 while the three successful trident results are shown in positions 1-3.

5) Experiment 65: The ancestor used in Experiment 65, like that used in Experiment 64, was one of the few in experiment 59 that managed to place cells in all three arms of the trident shape. Unlike the ancestor from experiment 64, it produced diverse, successful solutions that used all three limbs of the trident. The third row of Figure 11 shows four of the outcomes of evolutionary runs in this experiment. Of the 30 runs, 23 produced results with cells in all three limbs of the trident while 7 produced results with limbs in the part of the trident corresponding to the u-bend.

V. CONCLUSIONS AND NEXT STEPS

This study demonstrates that is is possible to find cellular automata rules that can be bonsai’d; made to stay within given shapes. This is in spite of the fact the rule itself has no access to the shape except through the selection process in the evolutionary algorithm used to select it. The study also highlights that there is an important quality that is not being measured. Given a set of ancestors, what is the potential diversity of populations produced by SPC using that ancestor set? The unique ancestor experiments show that while there is always localization of search, it can be more or less restrictive. In Experiment 61 the ancestor permitted diverse but low-fitness descendants. In Experiment 62 the descendants shared most of their fitness with the ancestor and simply diddled about with details. In experiment 63 there is apparent success from a fitness perspective, but a single solution, better than the ancestor, is located in all but one of 30 evolutionary runs. Experiments 64 and 65 are a contrast - the green ancestor produced spectacular results rarely and the red ancestor produced good results, inferior to the green ancestors best, but it did so routinely.
The greatest disappointment in this study is that the laboriously gathered set of 1890 diverse ancestors from many different initial conditions did not enhance performance more. Experiments with those ancestors not reported in this study demonstrate that they do help non-bonsai apoptotic experiments. These results, when contrasted against experiments 58-65, demonstrate that the choice of the ancestor set is critical to enhancing performance.

We also see that different shapes yield bonsai fitness functions that have very different difficulty. The snowman function was quite easy, the u-bend more difficult, and the trident impossible without the use of ancestors adapted to the u-bend. This latter result is the most interesting in the study. The trident based fitness function has a titanic collection of local optima based on running down the middle limb only. While difficult, the u-bend fitness function lacks these tempting local optima. Rules that have solved the u-bend to some degree are thus prepositioned to add a middle limb and do well on the trident fitness function.

It is worth briefly mentioning range niche specialization. In domains with real-valued fitness or a small number of discrete fitness values, it is clear it would be counter-productive.
In this study is was a simple and excellent expedient to avoid cloning, in part because of the huge range of integer fitnesses. Checking for exact image match would be a better, but far more expensive, way of preventing cloning. The results of experiment 63 suggest that results located during experiments might profitably be added to the list of forbidden fitness numbers driving range niche specialization. We note that most of the cost of exact image matching could be avoided by using fitness numbers as a first check.

A. Next Steps

The experiments in this study use ancestor sets with the following sizes: 1, 60, 120, and 1890. This suggests that there is a neglected middle ground. In particular, a set of two very different ancestors might yield blends of those ancestors. Actually, some sets of two ancestors will yield blends and some will not - the size of the space of rules alone makes this a near-certainty. This, in turn, leads to a notion of compatibility or fertility of pairs or sets of rules that is worth investigating.

An obvious next step is more and more complex shapes. This is tied to another possible area for leveraging work in other disciplines. The ancestry diagram in Figure 3 suggests that techniques from animal breeding, intended to enhance traits like milk production, weight gain, or disease resistance may be profitably mined to come up with breeding programs for cellular automata bonsai.

The techniques used in this study were intended to be generic at the algorithmic level. This means that, beyond specification of the bonsai shape, there was no attempt to specialize fitness functions with domain knowledge. It is likely that much better rules for all three shapes used in this study could be located by giving greater rewards to cells that are difficult to reach, e.g. farther from the center of the growth area. If many experiments were run with a shape, the number of times a given cell position was hit could be tabulated and reward for growing in a cell position be made proportional to past rarity. Such rules, once located, could then be used to enrich the ancestor set.

This leads naturally to the idea of fuzzy fitness functions. All the growth areas used are contained within a rectangle. If each cell of the rectangle has a reward level - corresponding perhaps to a grey-scale picture - then simple image manipulation programs could be used to create complex, sophisticated fitness functions.

This study is part of a research program that started as an evolved art project[5] and then added a theory component when it was noticed that apoptotic cellular automata have very complex behavior and an interesting fitness landscape[4]. This study contributes to both directions of research. We note that automata that would have good fitness on the u-bend and trident fitness functions simply did not appear in the earlier studies. This suggests that the bonsai shapes are steering the algorithm into parts of the fitness landscape where it did not go before.

Finally, we note that the representation for rules used in this study is very simple. All the complexity of shape and form that arise follow from the iterative nature of the production of time histories. It may be worth examining other rule representations for this problem.

REFERENCES


