“Rule 110 is Universal!”
A Gamin’s Guide To Goldbergean Gadgeteering

Harold V. McIntosh

July 10, 1991

Abstract

Glider collisions specific to the demonstration that Rule 110 may be universal are described.

Contents

1 Introduction

The recent claim that Rule 110 is Universal depends on an intricate series of glider collisions, which at first sight is simply bewildering. On examination, it appears that there may be a chain of reasoning which will lead up to the construction which has ben presented, or possibly even to alternative constructions.

Structures which are technically gliders of many kinds exist in Rule 110, but those which seem to offer the most potential and variety are those which are confined to the ether background, of which around a dozen exist. Due to the cyclic nature of the ether, spatial period 14 and temporal period 7, there are parity rules for dislocations in the ether; the most useful of these is for simple binary parity itself. Thus gliders are either even or odd, combinations of gliders add their parities, and the overall parity must be conserved throughout a collision and ever afterwards.

Collisions between odd gliders either give an even, possibly null, result or produce another pair of odd gliders. It is not excluded that these are the same gliders, simply having changed places, giving an instance of solitons. The existence of solitons suggests the transmission of information from one end of a block of structures to the other so that activities could go on in several places and be related to one another, rather than having to wait for the intervening space to work itself out in a more complicated and unpredictable way.

Experimental evolutions and de Bruijn diagrams reveal the existence of gliders and quantify them according to possible shift periodicities. Further experiments reveal the consequences of glider collisions, although they become tedious due to the number of gliders that have to be included. As more and more gliders are involved in simultaneous collisions, the task becomes unmanageable. For Rule 110 several solitonic combinations have been discovered, involving C, F, and EBar gliders. Many other combinations wherein A and B gliders are absorbed and later reemitted can be taken as solitonic. There may be multipliers, which would produce more gliders than the number which first collided. Pure absorbers would be black holes, pure emitters would be glider guns. So far only one glider gun is known, creating A's and B's in equal quantities.

A recent universality proposal relies extensively on the interactions of static C gliders with travelling EBar gliders. Actually the most viable candidate for a static glider was one (or more) of the C gliders, which are already static. Two velocities, to allow overtaking, seems necessary and zero velocity is not an a priori requirement. Playing EBar's off against F's doesn't seem very promising, whereas C's are more appealing. En's are plausible candidates for the second member of the pair, but the final proposal calls for EBars.

Beyond creating a data flow from one part of an extended structure to another, there should be some way to start and stop the flow, which implies a predicate to make the decision. Indeed, one procedure divides a flow into two parts, from which the predicate deletes one, allowing the survivor to continue on its way. Thus there are two tasks; erasure and the decision as to when to apply it. It also turns out that more complicated solitons are required than just those found by examining binary collisions.

We consider three parts of the possibilities of data movement in turn: erasure of oncoming elements, the predicate determining whether erasure will take place, and the solitonic requirements to cross intervening space. Stopping the flow at the other end is also required, but was already more fully understood.

According to [19], page 1115 (Stephen Wolfram speaking):

> " ... [1991] ... the general outline of what had to be done was fairly clear — but there was an immense number of details to be handled, and I asked a young assistant of mine

named Matthew Cook to investigate them. His initial results were encouraging, but after a few months he became increasingly convinced that rule 110 would never in fact be proved universal. I insisted, however, that he keep on trying, and over the next several years he developed a systematic computer-aided design system for working with structures in rule 110. Using this he was then in 1994 successfully able to find the main elements of the proof. Many details were filled in over the next year, some mistakes were corrected in 1998, and the specific version in the note below was constructed in 2001. Like most proofs of universality, the final proof he found is conceptually quite straightforward, but is filled with many excruciatingly elaborate details. And among these details it is certainly possible that a few errors still remain. But ..."

The eventual solution can be divided into two parts; the first consists in identifying promising glider collisons, the second in getting them to work together in the semblance of a computation. The mechanism chosen for the latter is a Cyclic Tag System, similar to the scheme introduced by Emil Post in the 1920's, but its details need not concern us.

The useful collisions, which are described forthwith in detail, permit erasure, filtratration through a barrier, and the selection of alternatives; a predicate, if you will.

It might be opportune to explain that the word "universal," as applied to computation, has two meanings. The general, expansive, form asserts that "You can compute anything." As such, it is a variant of Church's thesis, that all logical or mathematical reasoning is based on specific procedures. These are the arithmetical operations, tests for equality, and the use of certain rules of inference. Whitehead and Russel's *Principia Mathematica* goes to some length to establish this foundation.

The narrower interpretation is embodied in Turing's construction of a model computer, which procedes in three stages. The first shows how an extremely simple, but nevertheless definite and concrete, device can realize the primitive operations, such as balancing parentheses, adding numbers, copying lists, and so on.

In the second stage, it is shown that there is a description of such a device, using parentheses, lists, comparisons and such like, given by a set of quintuples. Universality consists in showing that there is one special device, admitting its own description via a quintuple set, which can mimic the operation of any other, including itself. It is universal, because it is the only device you will ever need to preform calculations according to the agreed procedure.

The third stage, of course, is the notorious one; such a device cannot even bebug itself!

A certain balance is involved in all this. To be called "computing" a process must have a certain complexity - the ability to do arithmetic. But the enterprise has its limitations, so a criterion for not being universal is certainly not leaving any uncertainty in the results. From there on, the use of the word *universal* is purely a philosophical problem.

The "universal constructor" of John von Neumann and the discovery of the Garden of Eden may illustrate the point. His constructor was implemented via a cellular automaton with certain rules and was capable of building anything for which it had a blueprint, and (in principle; it was complicated) had its own blueprint. Yet there were configurations of the automaton which could not be constructed, because they could only exist at the first step, as initial states.

So bear in mind — universality in the strict sense can sometimes be demonstrated, although establishing an adequate system of collisions for Rule 110 shows how arduous the process can be. The possibility for universality in the general sense, that "you can compute anything," only then becomes a possibility, ranking as an alternative route to Church's thesis.

That it is a possibility only establishes a possible degree of complexity; it does not say that the environment will ever perform a computation, much less some specific one. Such complexity could well be evident on other grounds.

In any event, we are only going to exhibit a series of interesting glider collisions which will confirm the existence of a Cyclic Tag System; their possible utility and eventual application is something else again.

2 Collisions Between A and EBar Gliders

Although collisions with A gliders are not central to the demonstrtation that Rule 110 is universal, their presence is felt in certain critical places,

The most important is the flow of A tetramers coming in from the West, whose purpose is to stop the solitonic EBar gliders which have traversed all the C2 groupings in the static part of the Cyclic Tag System. But the tetramers do allow some spurious EBars to ride off into the sunset as a sort of counterflow to the tetramers. That the two possibilities exist and that their operations are so nicely synchronized is both a marvel and a source of instability.

align	monomer	dimer	trimer	triplet	tetramer	tetrad	pentad
hhi	E2	EBar, 2 B, Atet	D1	C3	EBar, Atet	C2	C1
hi	EBar, A	EBar, 2 A	C3	EBar, 3A	C2	C2	C1
mid	EBar, A	D2	D1	D1	C2	C2	C1
lo	EBar, A	E1	F, BBar, 2 B	EBar, 3A	F, BBar, B	C2	C1
llo	EBar, A	E1	F, BBar, 2 B	EBar, 3A	F, BBar, 2 B	C2	C1
top	E2	E1	D1	C3	C2	C2	C1

Table 1: A's can almost pass EBar's, except that in the two extreme alignments they turn them into E2's and stop. An A tetrad ([four equally spaced A's] *vs.* tetramer [block of four A's]) uniformly yields C2's.

Table 1 summarizes some of the An-EBar collisions. Of course, collisions with A gliders can become quite complicated, given the variability of spacing between them. If they are well enough separated, the groups collide individually and the result is cummulative. Otherwise different combinations are likely to give a variety of results.

In the simplest combination, several T1's stick together; these are called polymers, with prefixes such as monomer, dimer, trimer, ..., designating the number of T1's.

Nearly as simple are arrangements which separates monomers by single ether tiles; they are called polyads, following the sequence monad, dyad, ...

Table 1 contains an additional column labelled "triplet," which is not a generic term but refers instead to one specific combination which forms part of the Cyclic Tag System. For this glider, the spacing between monomers is 2, 5.

On the next three pages, all the six possible collisions at different aspects are shown for the three most important classes of A-EBar collisions participating in the Cyclic Tag System. Of these, not all are realized, but it is useful to know what fraction of collisions was useful, and their exact configuration.

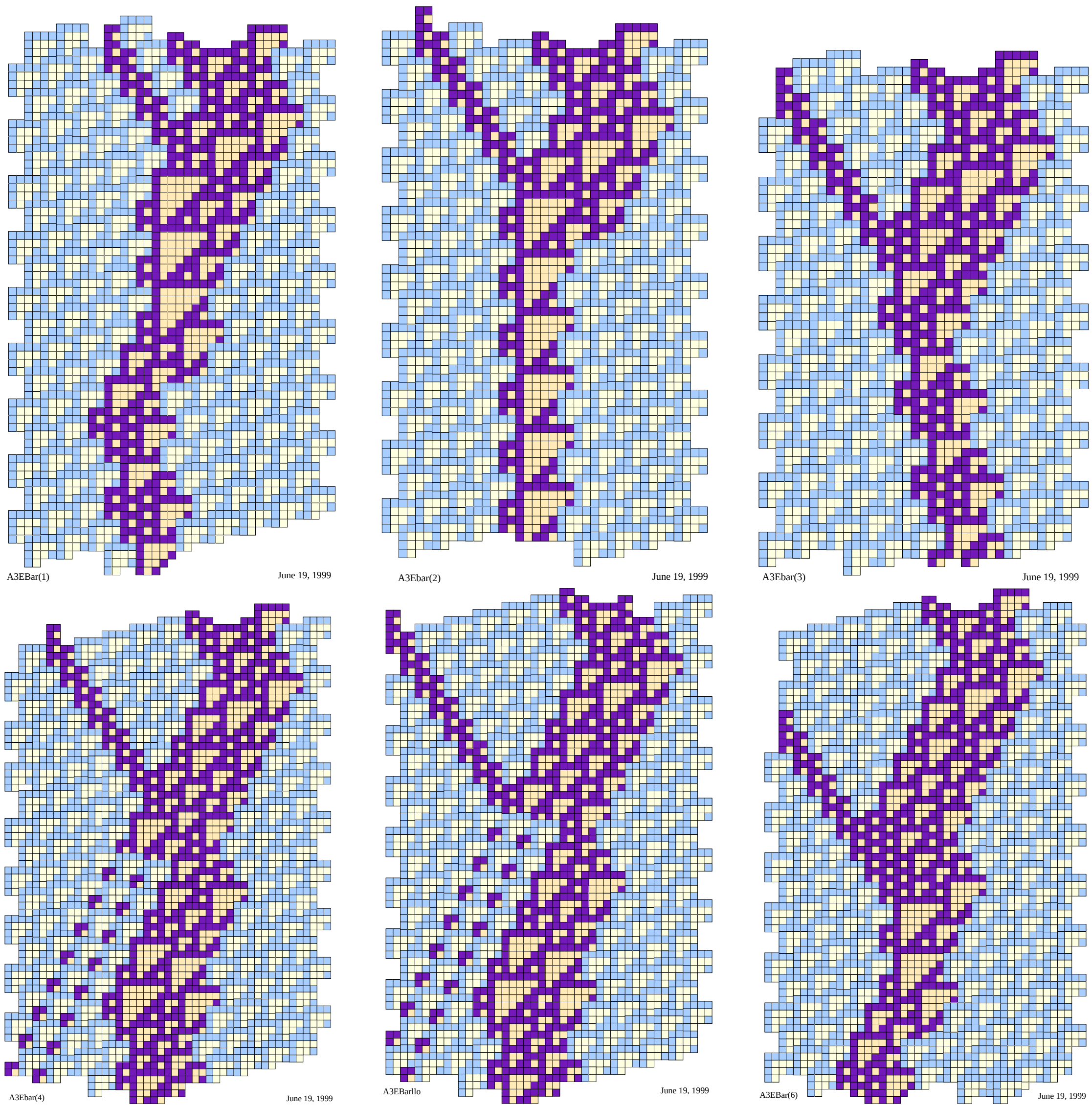

Figure 1: The six (A trimer, EBar) collisions. Three produce D1's, one produces a C3, the other two make leave 2 B's, a BBar, and an F.

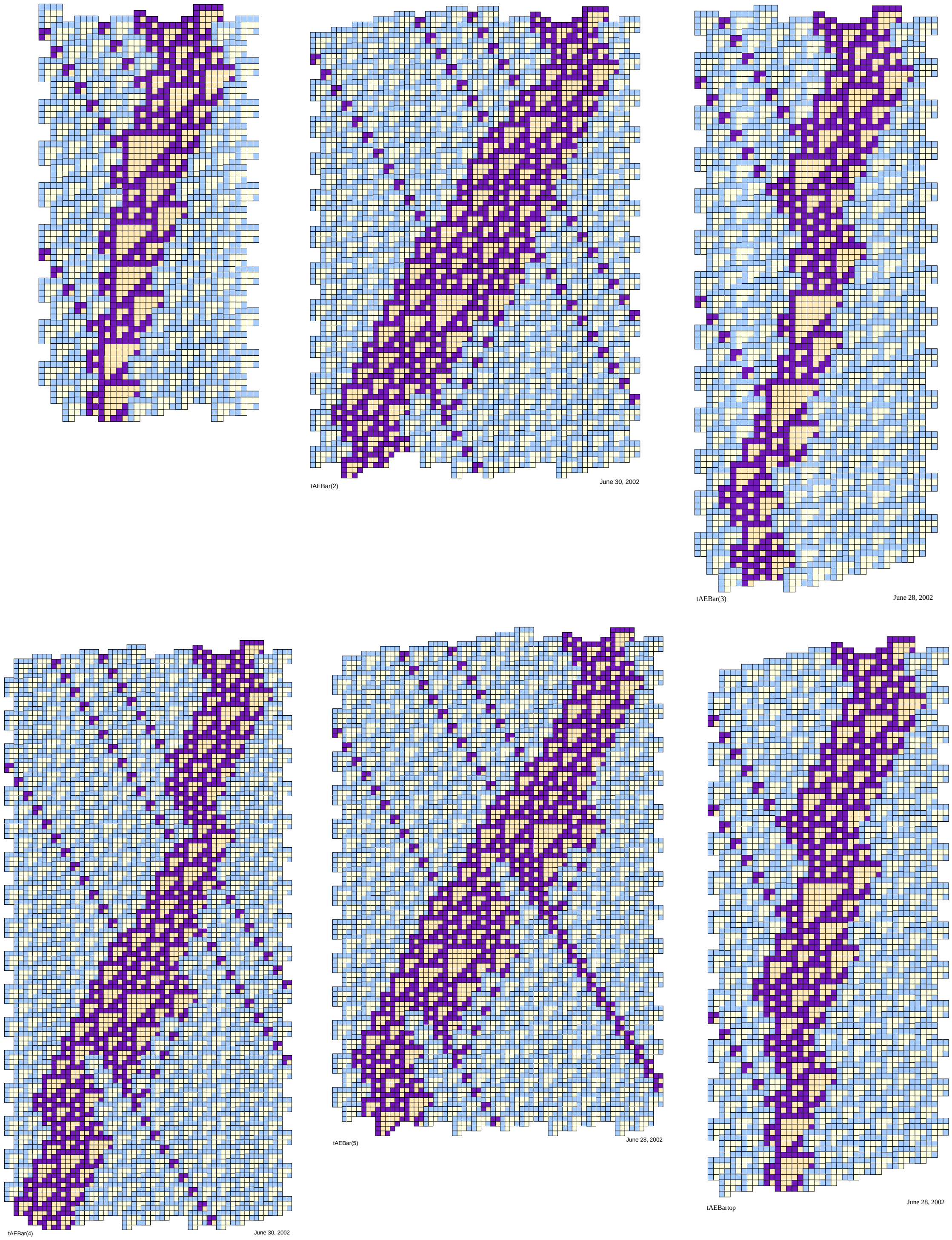

Figure 2: The six A triplet - EBar collisions. Three are solitonic (but not strictly), two yield C3's, and the other one leaves a D1. Some collisions are seperable, others are interdependent.

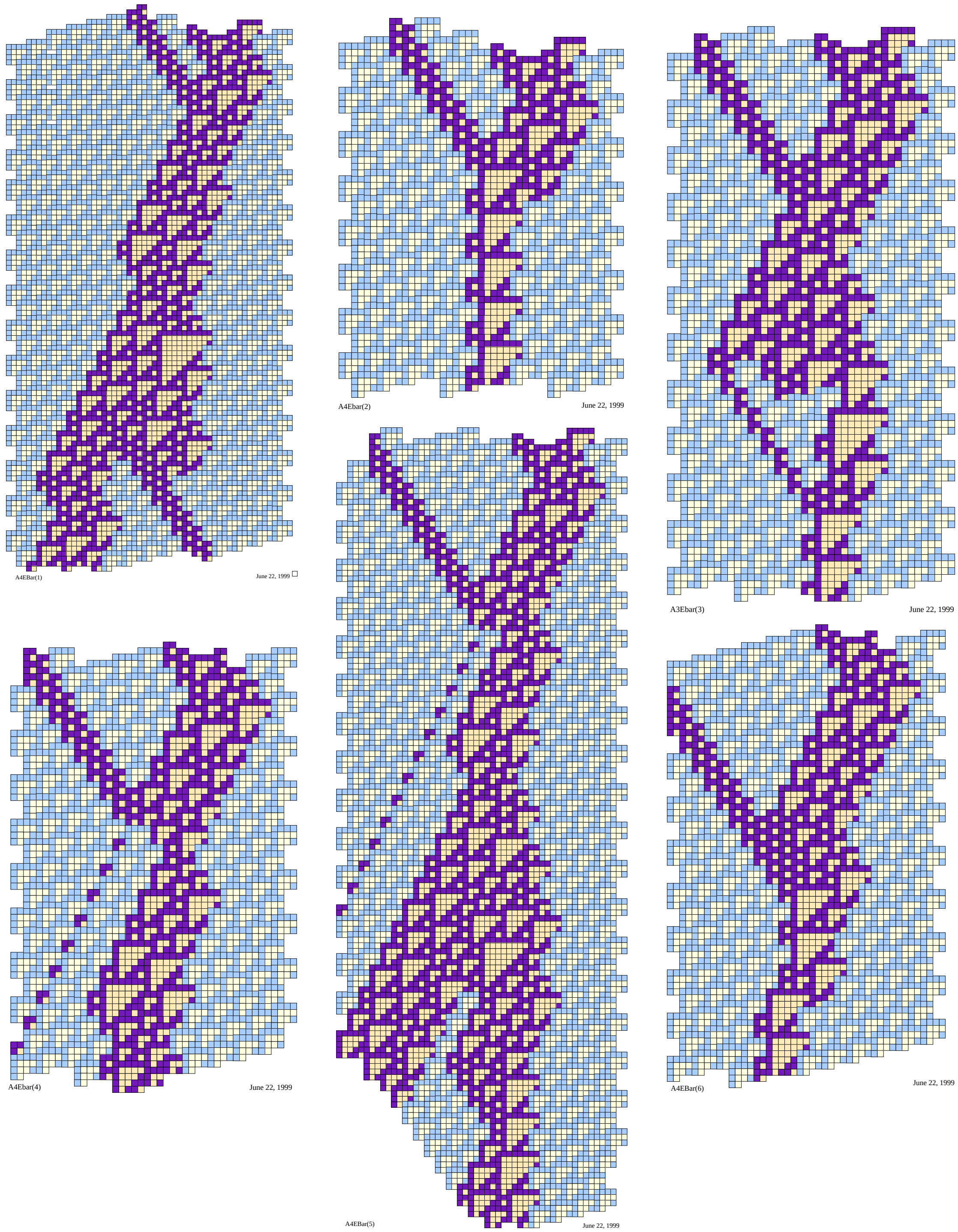

Figure 3: The six (A tetramer, EBar) collisions. One is solitonic, three yield C2's, and the other two eventually result in the combination of an B, BBar5 and F.

3 Collisions Between C and EBar Gliders

It is remarkable, the extent to which the Cyclic Tag System depends on nothing more than interactions between C gliders, especially the C2, and EBar gliders. There are solitonic relations between all the different C's and both EBar and F gliders. Due to the differing velocities of the EBar and F, it is hardly surprising that one or the other would prevail; that it was EBar is a definitive disclosure from the book. Equally interesting is the relegation of C1 and C3 to secondary roles, particularly through the coupling of a *pair* of C1's to a C3.

Before enumerating the C - EBar collisions, it is worth noting that the EBar has a boundary consisting of six ether tiles running in a northeasterly direction, followed by a two tile jog, after which the unit cell repeats. Due to the periodicity of the C gliders, there is only one distinct northeasterly line of approach for the EBar, which is also the direction of approach for B gliders, likewise for BBar's, G's, and even F's.

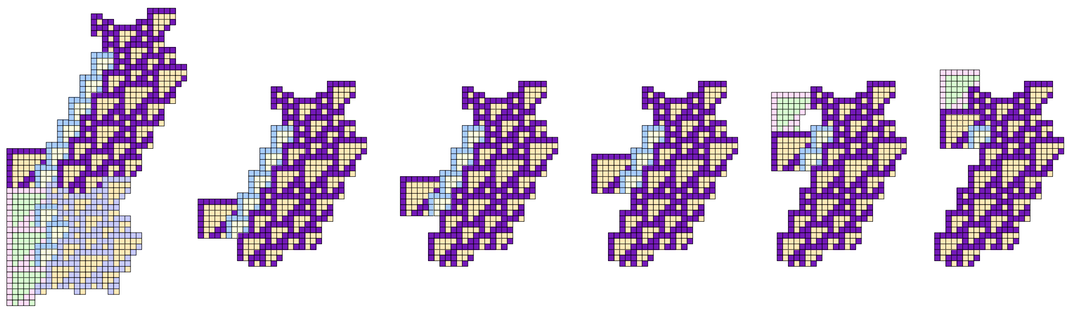

Figure 4: Of six alignments between a C glider and an EBar, one is redundant, one impossible, finally leaving four.

Figure 4 shows a C unit cell (which of the three, C1, C2, C3, doesn't matter because they differ along their left margins far from the site of the collision) positioned along the EBar margin. Due to the jog and resultant overhang, separation by a single ether tile is excluded because the collision would have already taken place. Separation by six tiles is redundant, because the collision delays, and corresponds to a displacement of two tiles when it finally occurs.

Not only do we want to examine the collisions between the three C gliders and EBars, there is a fourth set which is important. Namely, they are the collisions with the C1 pair, which are spaced so closely that the encounters are not separable into individual collisions taking place one at a time in sequence.

alignment	C1	C1 dyad	C2	C3
hi (2)	C1, EBar	D1	C2, EBar	B tetrad
mid (3)	A, B, B dyad	EBar, A triplet	C1, F	C1 dyad
lo (4)	A pentamer	EBar, E trimer	B, B dyad	B, B triad
llo (5)	C1, EBar	EBar, 2 C1	A, BBar, B, B dyad	C1, F

Table 2: Some of the C - EBar collisions are solitonic. Others participate in useful cycles.

3.1 Four collisions between C1 and EBar

Individual C1 collisions do not enter into the Cyclic Tag System, but they are still interesting for the possibilities they may afford for alternative tag systems, as well as for the reasons they may not have been favored in the system actually realized.

Given that the C gliders and the EBar all have odd width, any collision must result in an even overall result. It may consist exclusively of even gliders, such as A's and B's, sometimes even G's, and those may be produced in any quantity.

When there are odd gliders, they must arise in pairs, and may actually consist of the original pair after having changed places. Those are the solitonic collisions, giving hope that similar encounters may be repeated and that one member may end up at the far end of a long chain consisting of instances of the other member.

It is not excluded that the collision of a single odd pair could produce an odd quadruple, but neither that nor any otber amplification has been seen in practice.

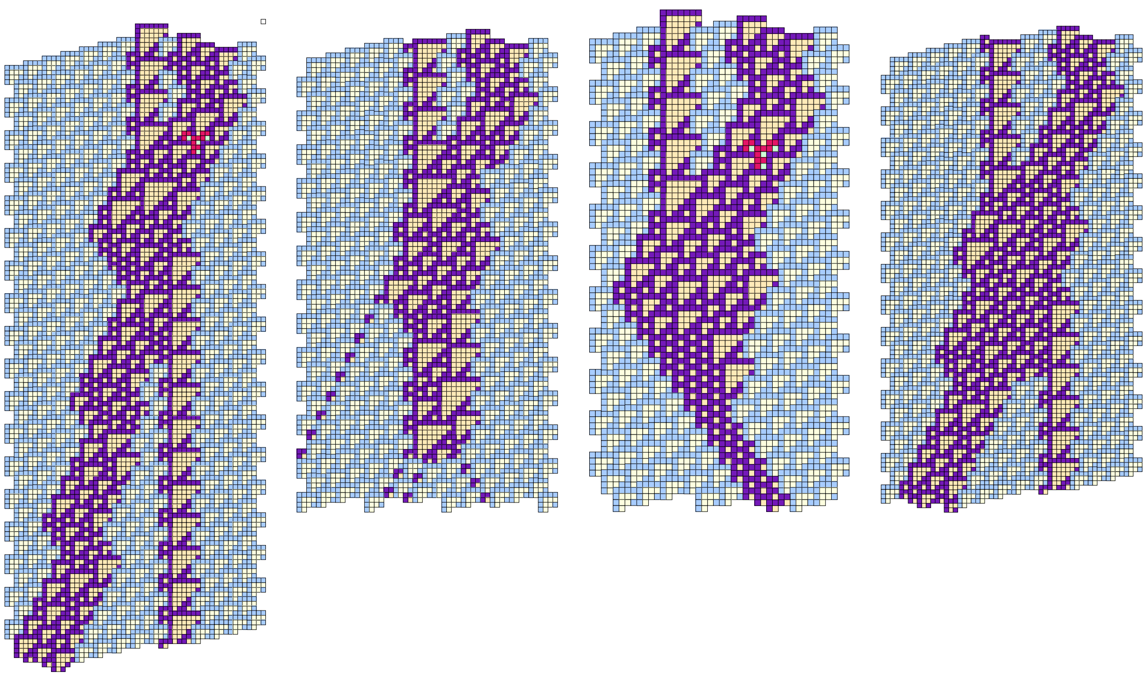

Figure 5: The four C1-EBar collisions; two are solitonic, none are used in the tag system.

Figure 5 displays the four C1-EBar collisions, Two are solitonic, giving hope that they could participate in a tag system. The other two produce only even gliders, the A pentamer collision showing interest because of its interaction with oncoming EBar's or F's, principally in stopping them and converting them into C1's or C2's.

In the actual Cyclic Tag System a similar role is played by tetramers gratuitously arriving from the far west. The tetramers are also more selective, allowing some EBars to pass, while stopping others.

The remaining collision produces an assortment of A's and B's. Although they could have their uses, dealing with both at the same time complicates the situation.

3.2 Four collisions between C2 and EBar

Although much experimentation centered on the possibility of using the choice between a C1 or a C2 glider to represent a bit, the way the Cyclic Tag System has been formulated is to use C2 gliders exclusively, the relative spacing between them playing the role of a bit. Actually, it turns out even more complicated than that; it is the relative spacing between two pairs of C2's, the spacing within pairs being constant.

C2 gliders have both an active and a passive role. The passive form must allow all EBar gliders free passage as they wend their way west. Since the encounter requires contact between the jog in the EBar and the T6 of the C2 (with a marginal displacenment of 2), the spacing of the C2's must conform to this relationship, and can only vary by including one of more extra EBar unit cells.

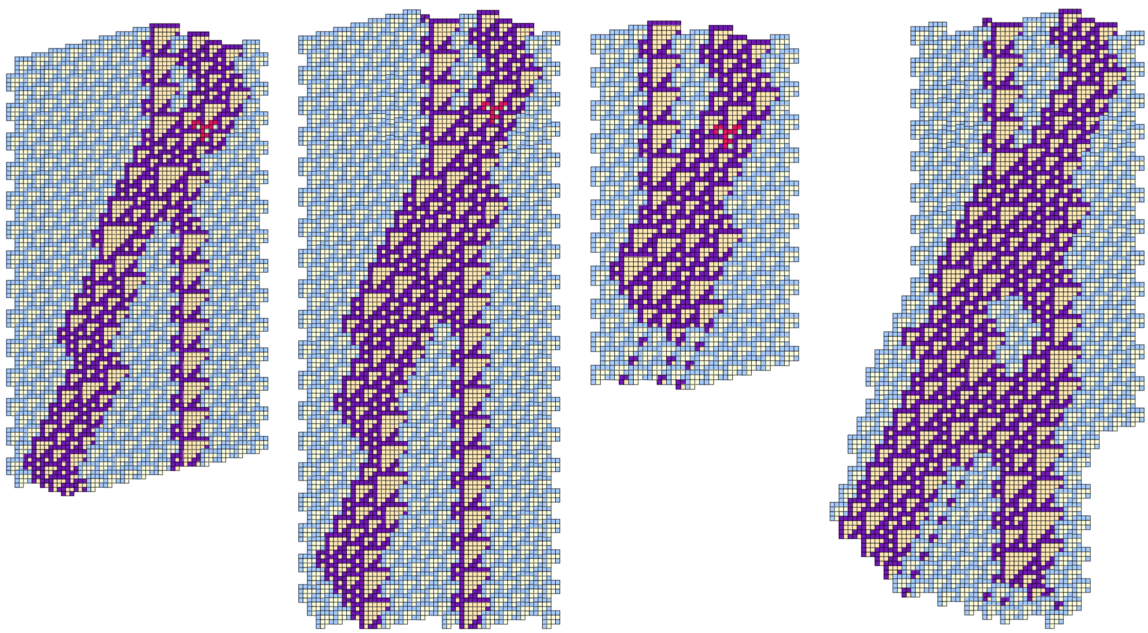

Figure 6: The four C2-EBar collisions. One is solitonic; a defining characteristic of the tag system. Another makes a C3 pair, eventually playing an important role in filtering EBar streams.

Figure 6 shows the four C2-EBar collisions. Besides the soliton, there is another near-soliton, in which a C1 is left behind after the encounter rather than a C2. Additionally, the EBar converts into an F. Had the C1 been more important, this might have been a mechanism for flipping between the two pairs.

The third collision in the figure is typical of C-E collisions, in which a threefold of B gliders is released, but with a spacing typical of such encounters. Its effect on additional C's has been investigated in detail, but it takes no further part in the present discussion.

The remaining collision, complicated though it is, actually constitutes the mechanism by which predicates are implemented in the Cyclic Tag System. The BBar5, on striking the next C2, converts into an EBar of a type which is ignored by the western shower, but which gets it out of the way. The three B's which are released in accompaniment with the BBar play a transient role in this transformation, and so disappear from sight.

It is the surviving A which reacts back with some earlier residue and the forerunner of an EBar packet, in a series of steps, to alternatively generate the D1 - A trimer leapfrog which erases arriving EBar's, or sets up a tricky filter which allows a third of them to pass. They can't be allowed to pass directly because of the parity rules, but it is acceptable to delete a pair of them while passing a third.

3.3 Four collisions between C3 and EBar

Secondary roles are assigned to both the C1 and the C3 in the Cyclic Tag System; nevertheless these roles are important ones. Single C3's alternate with C1 pairs when the transmission of a flotilla of EBars is required, albeit with attrition.

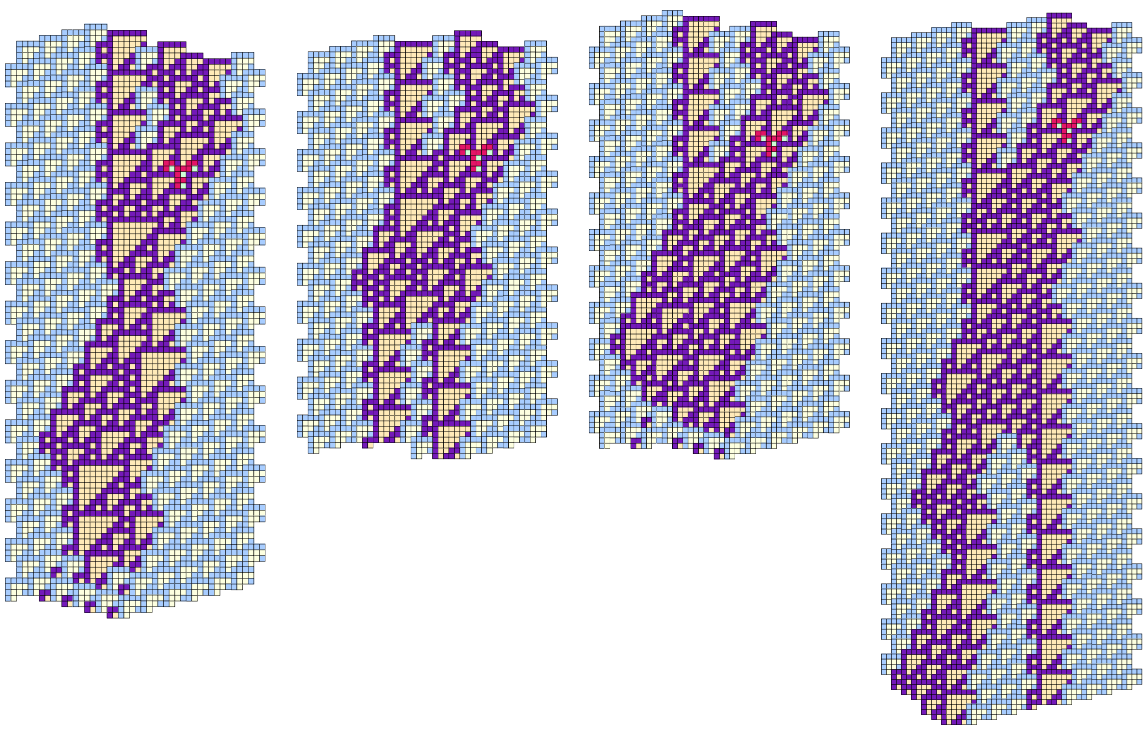

Figure 7: The four C3-EBar collisions. One is solitonoid, and one produces a C1 pair. Unlike the collision with a single C1, the latter combination is essential for the tag system.

Figure 7 shows the four C3-EBar collisions. Two dissipate into B groupings; one a tetrad, the other a common artifact in E collisions. There is also a double exchange, flipping EBar for F alongside flipping C3 for C2. The fourth has a critical place in the Cyclic Tag System, because it alternates the C1 pair with C3's to create a semipermeable membrane which lets a third of the Ebar's through.

3.4 Four collisions between C1-C1 and EBar

An unexpected ingredient in the Cyclic Tag System was the C1-C1 pair, although one of the obvious extensions of the catalog of collisions which had been prepared was to move on to trinary and other multiple collisions. In the case of the C1-C1 pair, this should have been all the more urgent because it was a product of a reaction already in the catalog. The separation was too close to dismiss its collisions as composites of individual C-collisions.

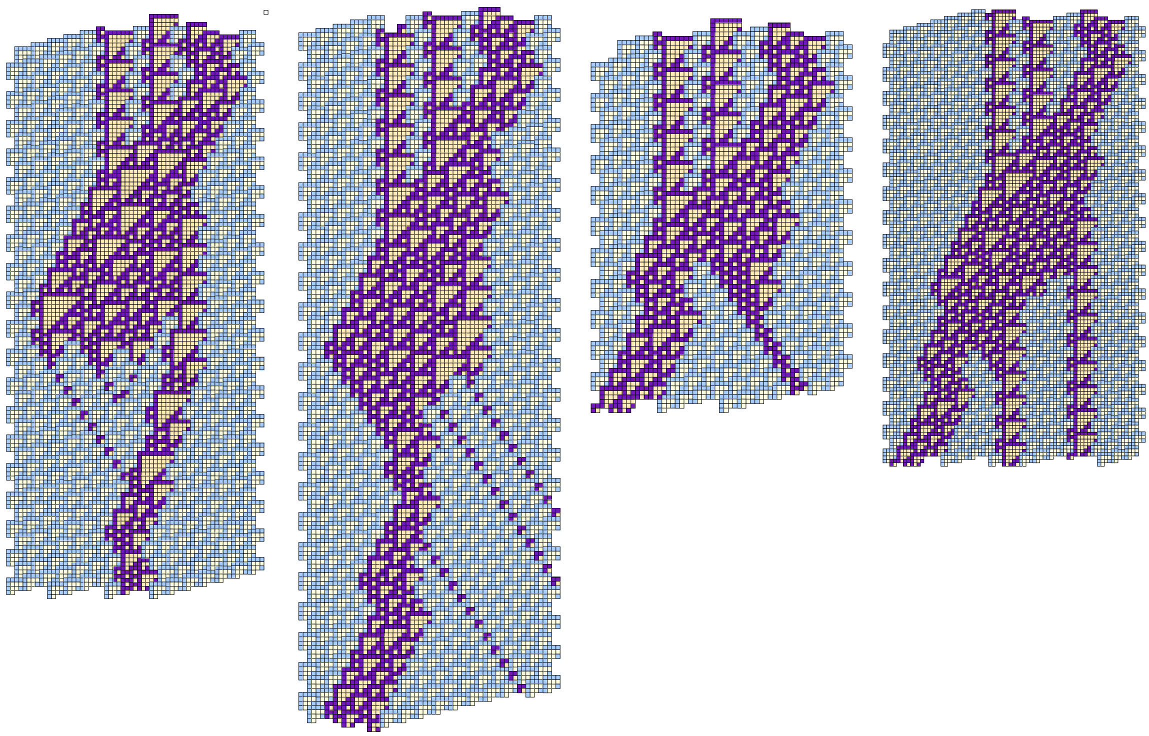

Figure 8: The four C1-C1-EBar collisions.

On the other hand, full generality would have required studying collisions with all the other gliders; knowing a specific application, it suffices to study only EBar collisions, as shown in figure 8.

These collisions have to produce odd residues, thus at least one odd glider. The first is somewhat solitonoid since the EBar passes through leaving the C1's more separated than they were before. It is not evident how to go on increasing the separation, leading perhaps to a binary tree. The second is also solitonoid, passing the EBar and ejecting an A trimer, such as forms part of the (A-trimer,D1) leapfrog. It is also possible to get a single D1 glider, as the third collision shows.

The fourth collision is the important one, because it exploits the last EBar to close the cycle which began with the C3, and prepare the way for the next bit or the stop signal in the EBar stream.

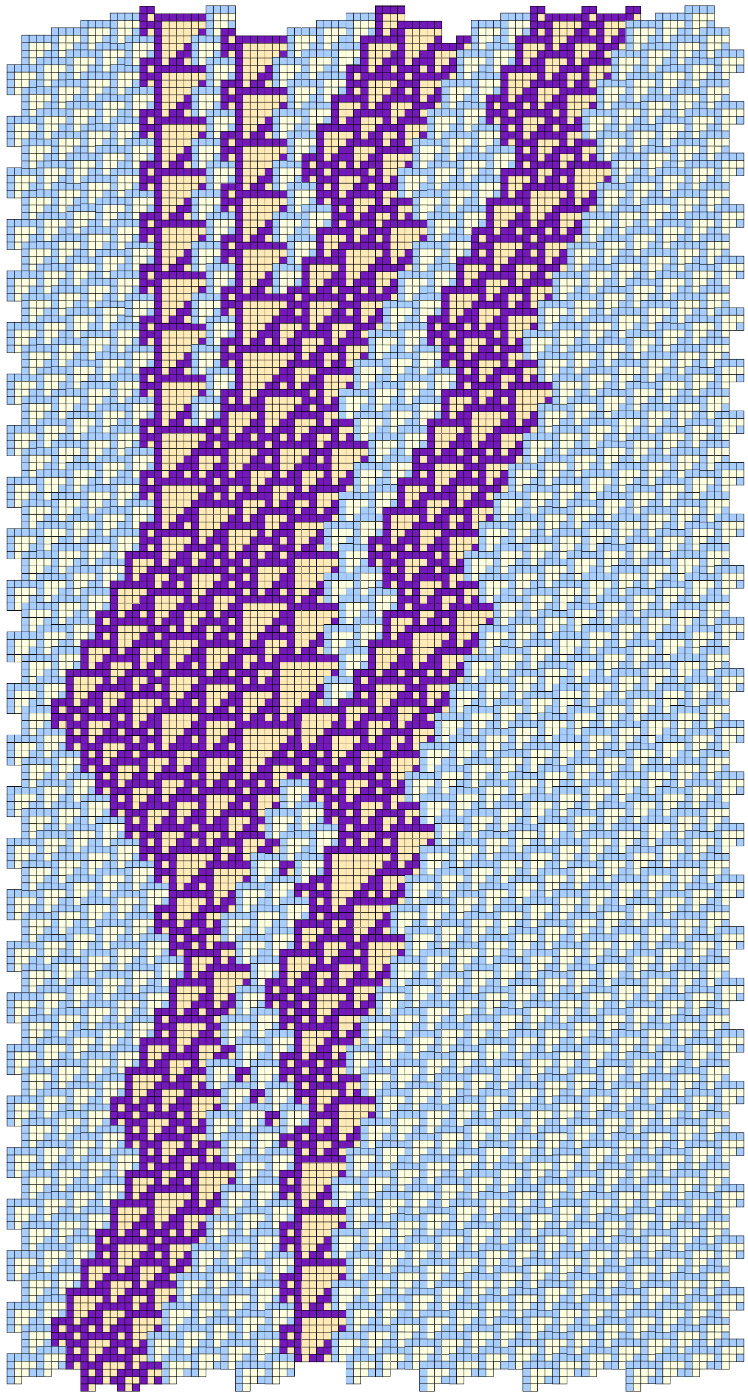

Figure 9: One of the C1-C1-EBar-EBar collisions passes an EBar and leaves a C3, reversing, in a way, the C3-EBar collision which leaves a C1 pair. It is an essential combination for the Cyclic Tag System

4 A Starting Point: Erasing Unwanted Gliders

Parity makes it difficult to selectively remove structures because almost all gliders are odd and will either pass by another odd object in soliton fashion, leave another pair of gliders, or fizzle out in A's and B's. Removing some without a trace while passing others unimpeded is hard or impossible to arrange; an interesting one has an A trimer colliding with an EBar to produce a D, which collides with another EBar to restore the original A trimer.

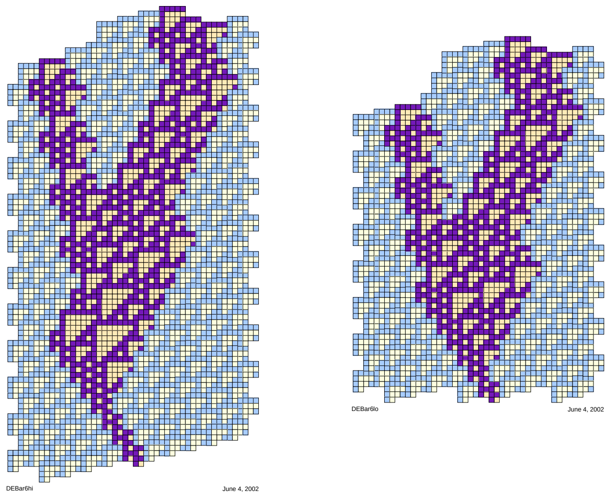

Figure 10: The two (D1, EBar) collisions which might participate in the leapfrog.

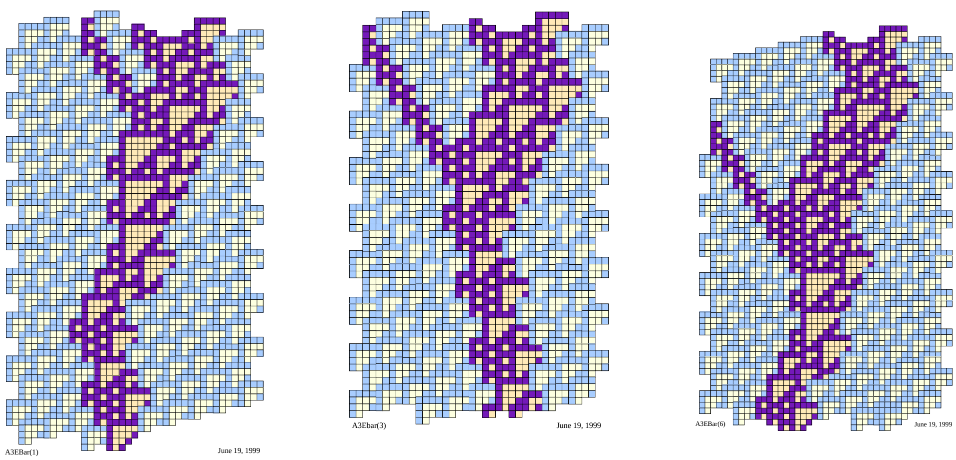

Figure 11: The three (A trimer, EBar) collision candidates for the leapfrog.

Of the eight (D1, EBar) collisions, two lead to A trimers; both arrive at the same position on the EBar (Fig. 10). Of six collisions with the A trimer, three result in D1's (Fig. 11).

Figure 12 shows one leapfrog erasing three successive EBar's.

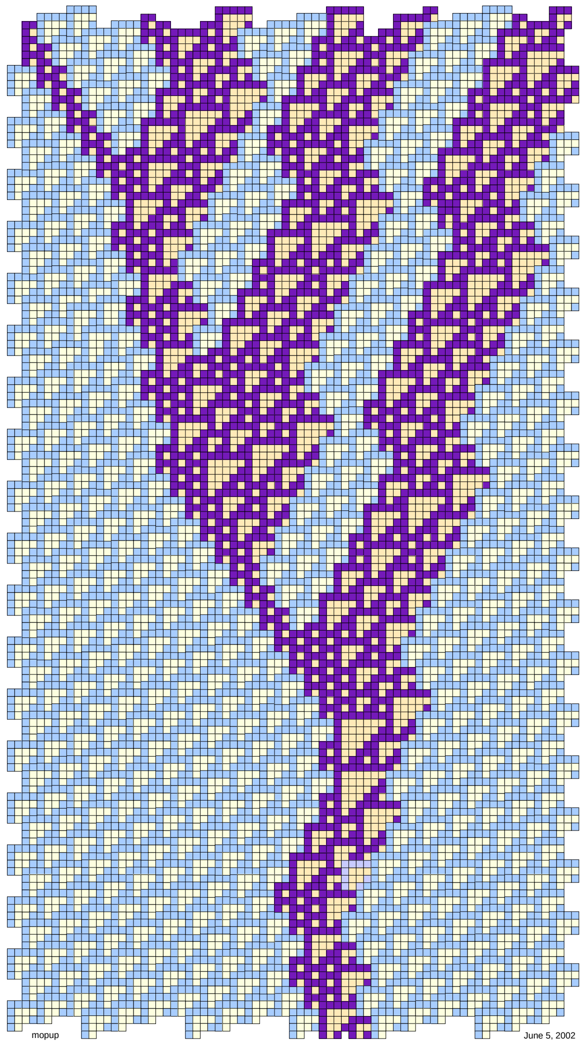

Figure 12: One of the requirements of a tag system is the ability to erase structures, be they oncoming EBar packets, or static C groupings. The (A trimer, D) leapfrog will erase all incoming EBars, so long as they are synchronized with the A-D erasure stream.

Eventually, of course, the leapfrog chain has to be terminated. Since the termination has to stop the other leapfrog, which is the sieve which allows some EBar's to pass, it is better to defer its presentation until the controlling predicate has been analyzed; see Figure 21.

5 The Computation Cycle Begins with a Predicate

The difference between a computer and a calculator is that the latter follows a determined sequence of instructions, even if it is the operator who decides the sequence. A computer is capable of taking decisions, often as not based on the sign of a result, as to which of two alternative courses of action to follow. Nearly always, one of the alternatives will be to go back and repeat a previous sequence, a retrograde step in an otherwise forwardleading sequence.

Going back, in a glider based environment such as Rule 110, would imply countermovement in addition to whatever process was bringing instructions and data together. Skipping backwards is equivalent to skipping forward if the instruction stream is repeated periodically and unwanted instructions can be erased, or ignored, or whatever.

That is the reason that discovery of the (A trimer, EBar, D1) leapfrog has such importance for setting up a computer based on Rule 110. Dilligent search had revealed some simple solitons, built up out of combinations of C, EBar and F gliders. Using both EBar's and F's is unwieldy because of their differing velocities, whereas the C's are stable and could reasonably be taken as program elements.

The other glider could represent data and be passed through the instructions, then stopped at their far end and turned into a new program element through glider collisions. Several candididates had been discovered and cataloged. Stoppage implies a flow of gliders from the left which had to be independent of the data which it had to stop, implying further structure to any prospective computing environment.

Thus the importance of either selectively eliminating gliders arriving from the right, or the rightmost static gliders which they were going to meet. That the (A trimer, D1, EBar) combination was overlooked provides an object lesson in dilligence and careful observation. Consulting the catalog of collisions [8] the paragraph dedicated to (D, EBar) collisions is empty, due to its falling at the end of the alphabet and laziness. A more carefully done Atlas [10] contains the collisions, but by then checking collision sequences wasn't being followed out to the necessary conclusion.

Viewing the drawings in [19] revealed one solution to the problem, from which a consultation of the table of (C, E) collisions (as in Table 2) singles out C2 gliders as the stationary elements. Fortunately, EBar's turn into C2's under many A collisions (see Table 1) which could inhabit the western badlands - the remote left environment.

But now, given that only one kind of glider could be static, and one other kind mobile, the predicate necessary to switch the flow of computation cannot reside in the selection of glider. The spacing between C2's must conform to the phase of the EBar gliders with which they will interact, but variation in terms of the EBar unit cell is still available. So, the spacing between gliders is the next detail that ought to be examined.

The combination finally used by the Cyclic Tag System is intricate indeed, drawing in some ancillary E gliders to absorb stray sparks or provide some of their own. Some nine stages are involved, shown in greater detail on the following pages, beginning with a map of the first few.

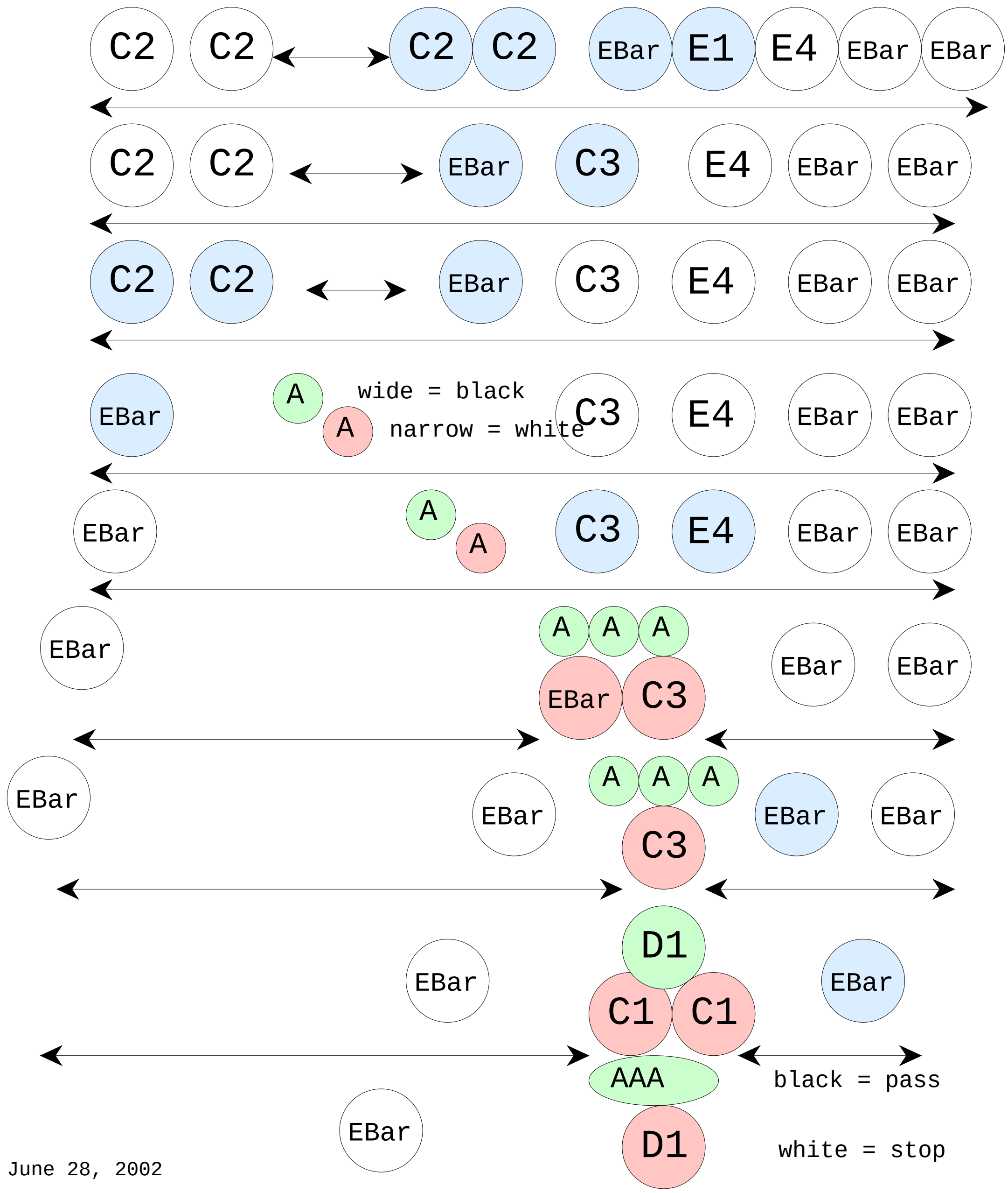

Figure 13: Nine stages participate in reading a predicate and releasing either the D1 glider or the A trimer according to the result. EBars following along after will then filter through the skein of C3's which will be set up, or be erased by the leapfrog.

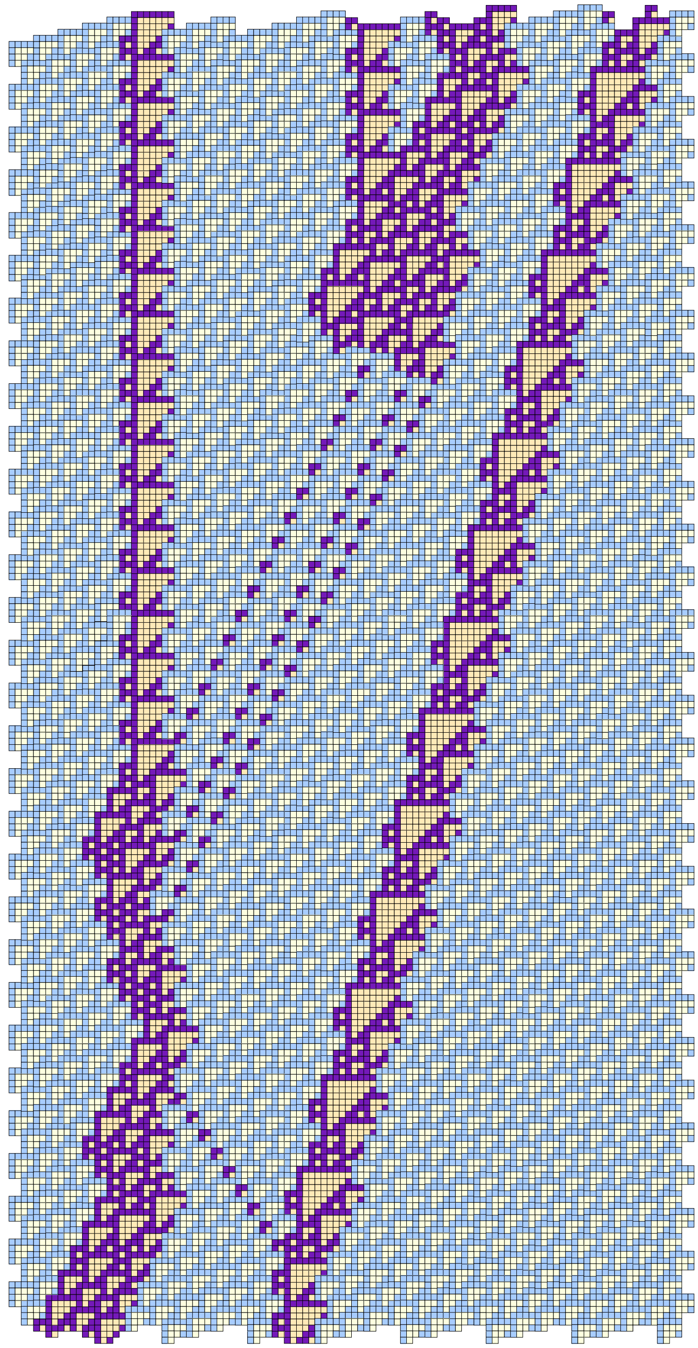

Figure 14: The leapfrog gets started in three stages, the first of which always creates a (C3, EBar) pair. Fifteen ether tiles running northeasterly must separate the two C2's.

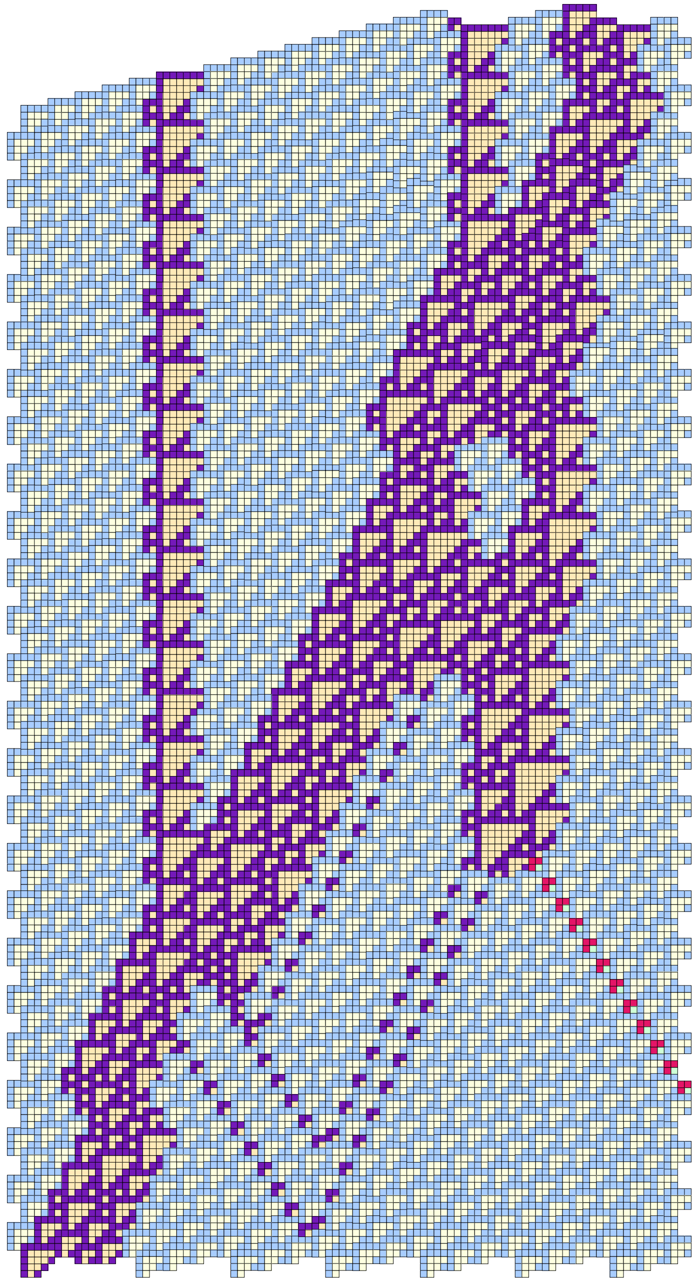

Figure 15: Second of three stages to read a static element, using the third and fourth C2, reading from right to left. A single A glider is emitted now; *where*, exactly, depends on how far the EBar goes to meet the third C2. Also created is a BBar, stopped by the fourth C2, becoming an EBar. The third and fourth C2's have a fixed separation of 19 ether tiles, alterable in multiples of three. Internally, three A's eventually annihilate three B's.

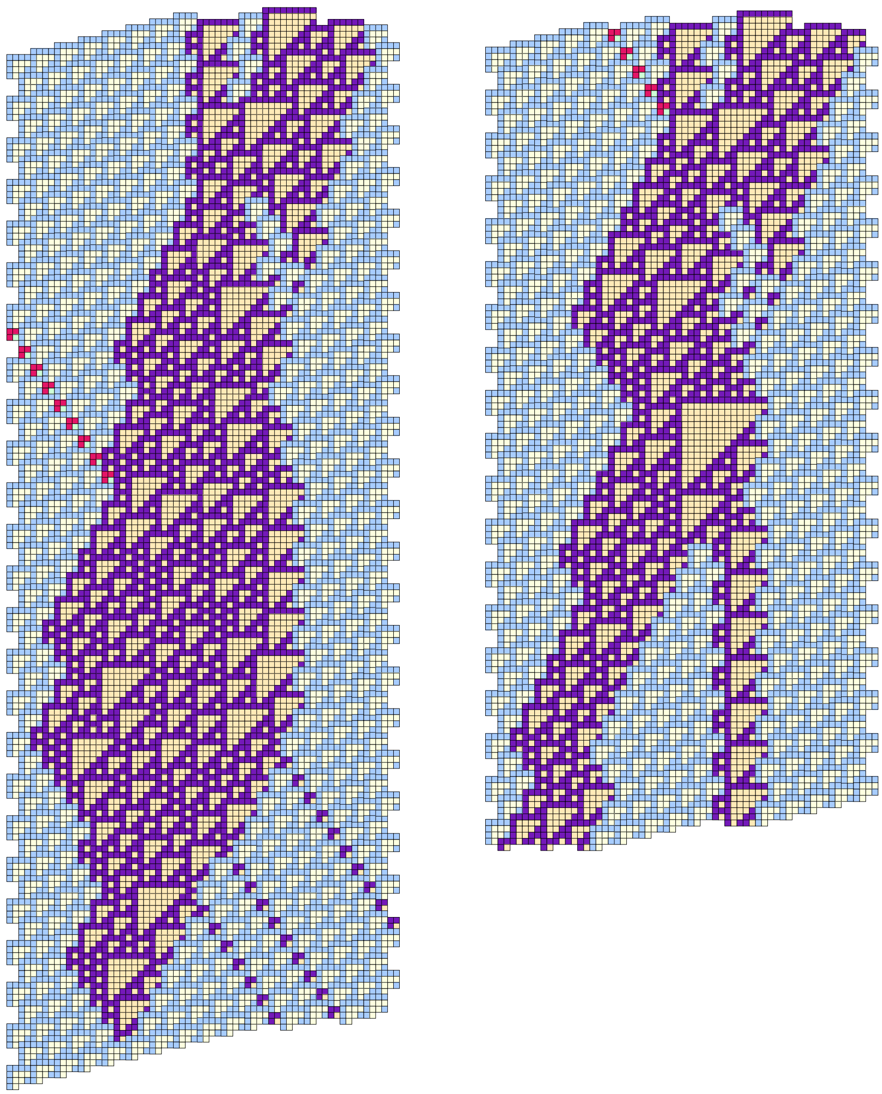

Figure 16: Third stage: Black (left) and white (right) predicate results. They vary because the second stage A glider arrives at different times, thus at different sites along the EBar.

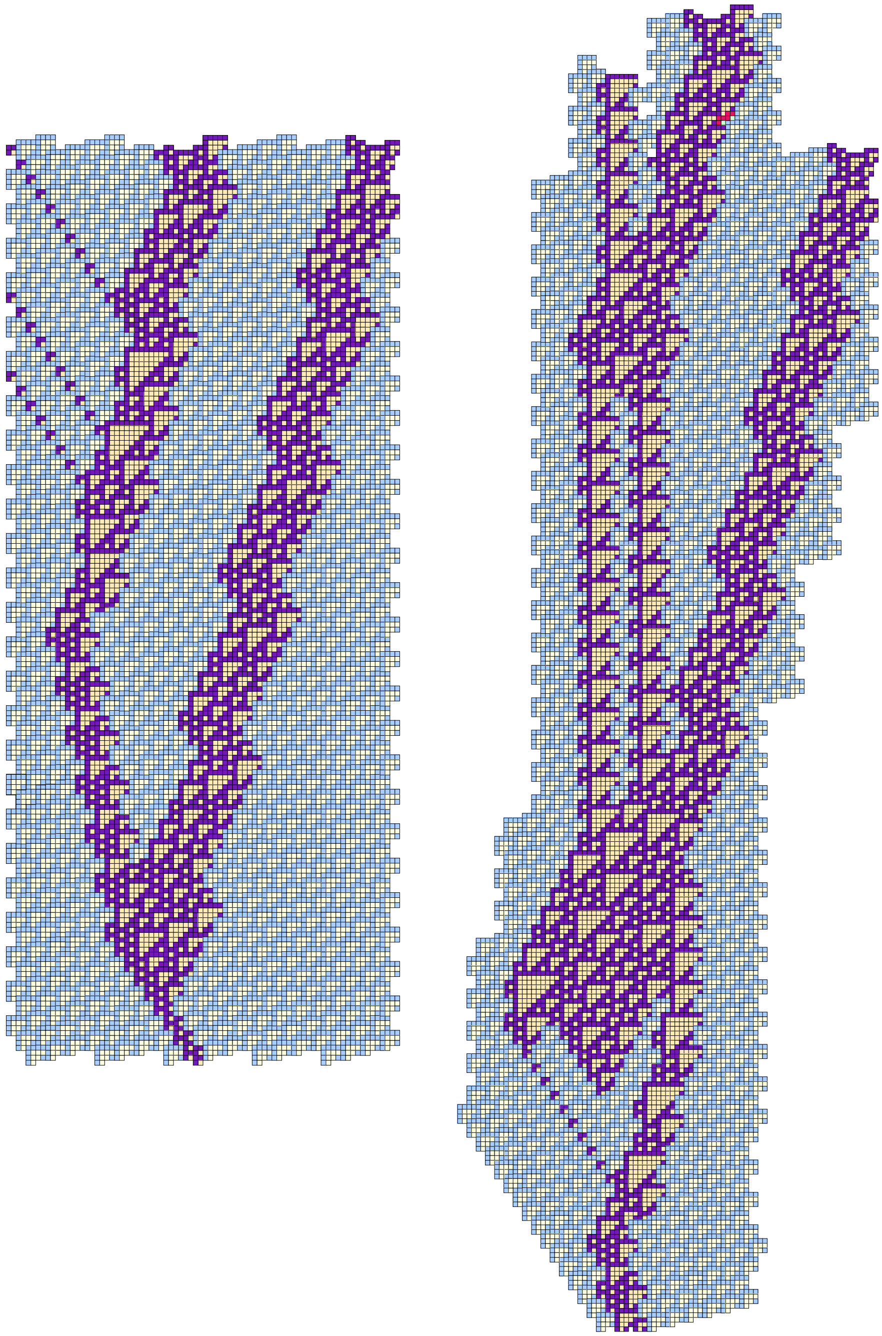

Figure 17: At the fourth stage, two more oncoming EBar's are met by either the A triplet or a C3, leaving respectively an A trimer or a D1 as befits parity conservation. But the story is not yet finished, because these results have to be checked against the oncoming EBar stream. So another pair will be needed, to invert the roles of trimer and D1, and to create the data structure which will allow some of the EBar's to pass.

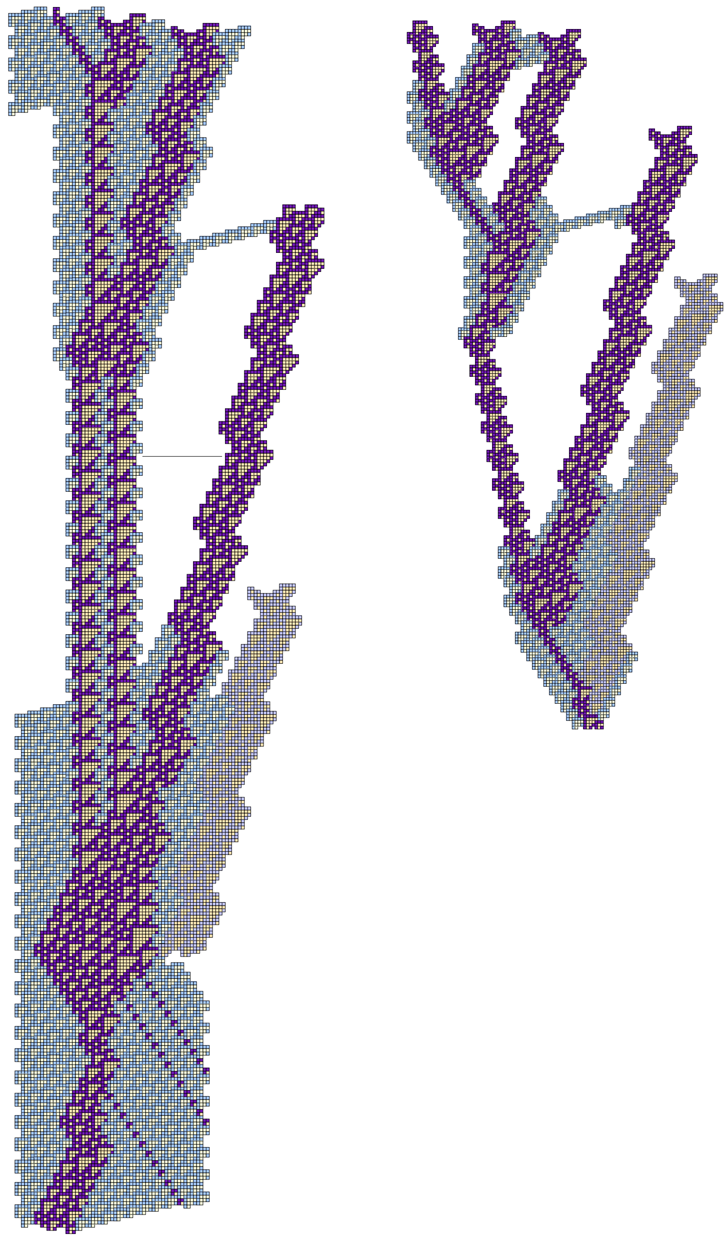

Figure 18: Once the predicate has been activated, a fifth stage of preparation remains. The black predicate must prepare a sieve, whilst the white predicate is ready for action and can begin erasing EBars. Nevertheless, their actions must be synchronized with each other and the oncoming glider packet. Its leading edge, shown in fainter colors, must just barely contact the source of the three A's to produce a C3 rather than the triplet shown. But it is not part of the primer; moreover it is worth noting that similar contact is requred between following packets.

6 EBar Packets

Discovering the white, or erasure, leapfrog would have been a matter of more careful attention to detail in analyzing a mass of glider collision data. The black, or transmission, leapfrog is much more subtle, but not necessarily that much more remote. Although C3's do not figure in the (A trimer, D1, EBar) interaction, they aree one of the primitive gliders, and so their collisions would have been included in any comprehensive list of binary glider collisions.

Odd pairs either produce odd pairs, or dissipate. The products of dissipation being even, only A, B, or G combinations are possible; when collimated they may be useful, otherwise they become single gliders to return to their place amongst binary single glider collisions. Similarly, the separation of the members of an odd pair requires that they be used immediately in a multiple collision, or can be safely relagated to the domain of single collisions.

Noteworthy collimated combinations are the C1 dyad and the C1-C2 pair left over when E3's collide with EBar's. Someone apparently noticed that the combination could be useful; indeed the C1 dyad is essential to the Cyclic Tag System under discussion.

On the other hand, it was an obvious task to study collisions with C phalanxes, and only their overwhelming number impeded a really serious analysis. Many such combinations have proven useful in creating large T's through glider collision.

Whatever the story of its eventual discovery, the (C1 dyad, C3) leapfrog interacts with EBar's arriving from the right, although only one in three succeeds in passing through. Fortunate, indeed, that the same spacing holds in erasure mode! And that the attrition can be compensated by inserting more gliders into the stream.

One esoteric detail has to be respected. Although there can be an infinite cycle of C1 dyad - C3 alternations, it must begin and end precisely where the EBar contact is not separable. The height of the C3 column can be varied, as can the height of the C1 dyad column, with an adjustment in EBar spacing, so those are separable.

Otherwise EBar packets of arbitrary length can be accomodated. However, since the value of the predicate governing their forwarding cannot be known in advance if computation is to take place, programming must take into account that it is *all* transmission, or *all* erasure.

Figure 19, on the next page, shows one cycle, albeit not the phase used in the Cyclic Tag System, of the (C3, C1 dyad) cycle.

The final EBar sextet will release one finalt A triplet to interact with the shim separating packets, as shown in Figure 21

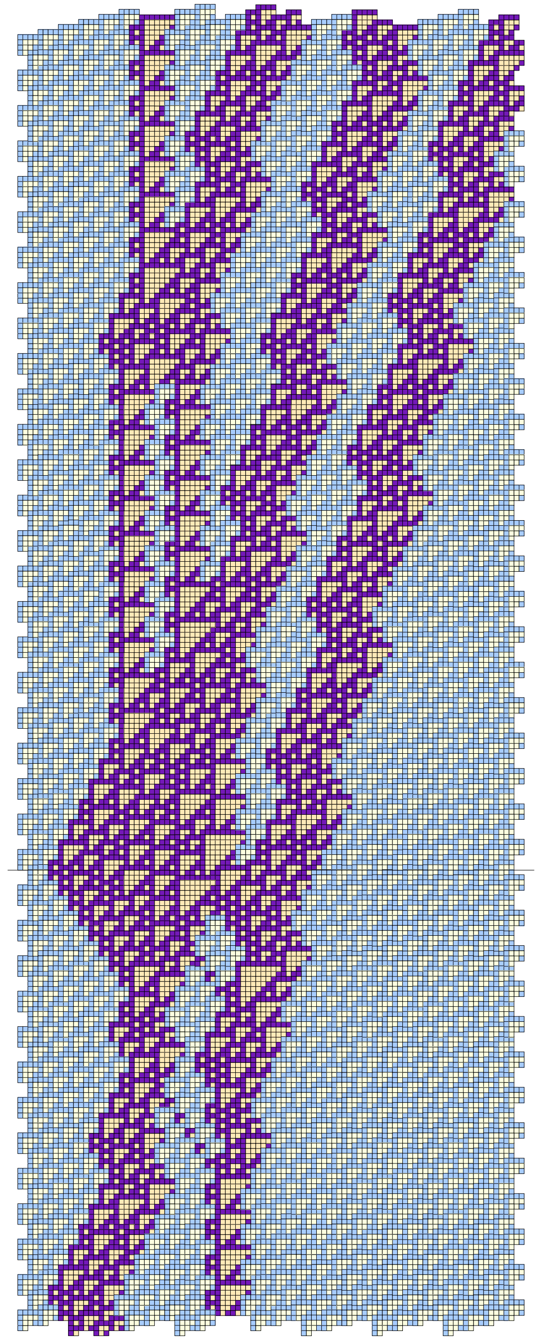

Figure 19: The sieve through which oncoming EBars pass is another leapfrog constructed from C3's and C1 doublets; it consumes two Ebars for every third one passed, starting its cycle on a C3 progenitor about a third of the way up from the bottom of the diagram. The separation of the rightmost EBar's is fixed, but the leading EBar can delay by offsetting in the A direction.

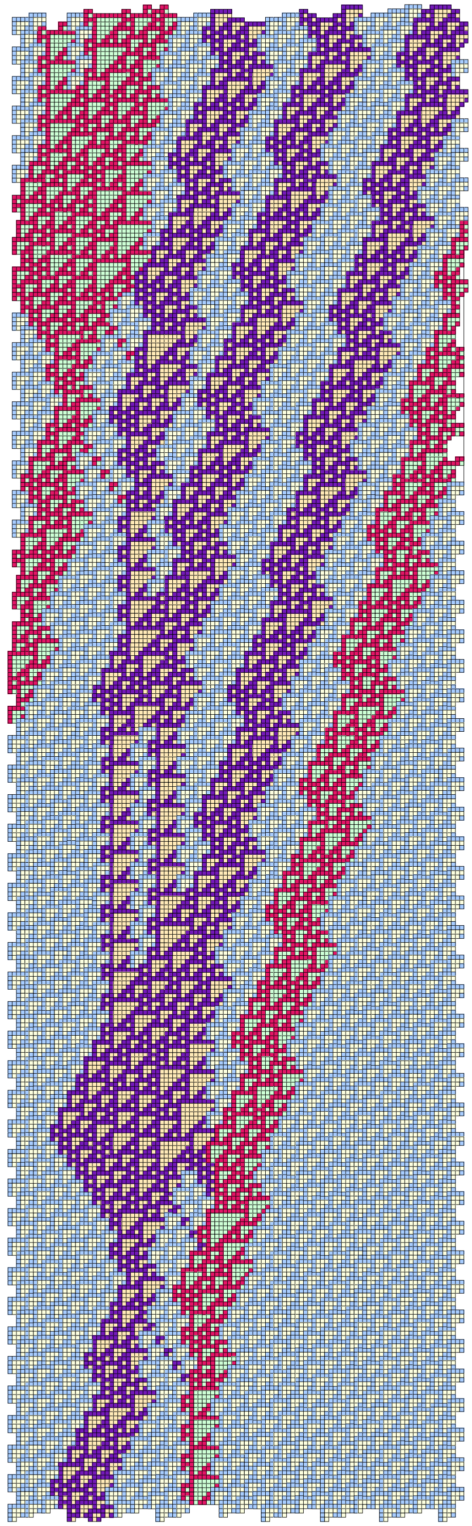

Figure 20: One EBar in three gets by.

7 Shimming Succeeding EBar Packs

Since we wouldn't want to erase everything, the leapfrog has to be halted. That is done by inserting a shim between one series of EBar sextets and the next, consisting of an E5 and an E2 running alongside of each other. Actually the shim is thicker, but these two gliders convert into the EBar and an E1 which lead the pack initiating a new predicate cycle.

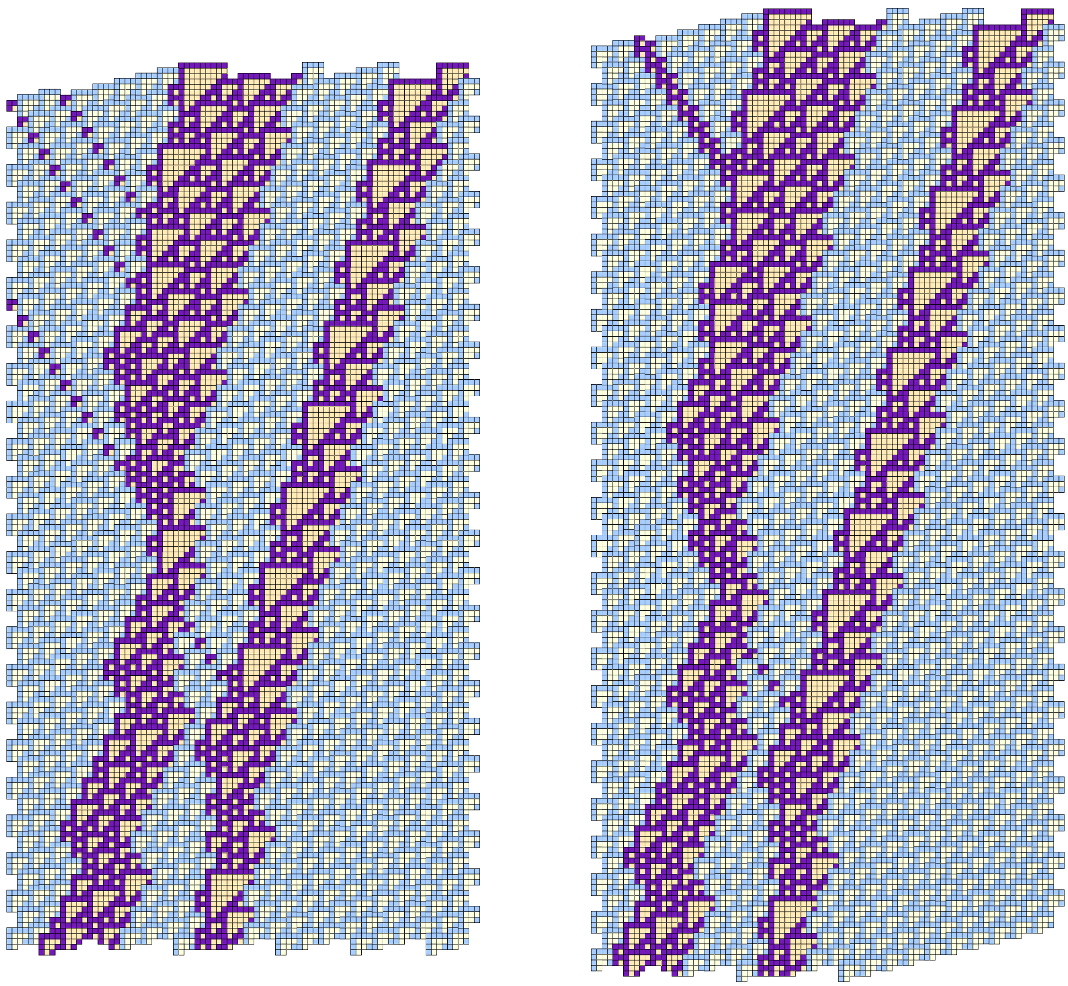

Figure 21: The leapfrog can be stopped by an (E5, E2) pair, which becomes an (EBar, E1) pair. But the same configuration must also stop the C3-C1 pair sequence which filters EBars, as shown in the left drawing. There it is triggered by one particular A triplet, rather than an A trimer.

8 The Western Badlands

After working out possible data and program representations, there remains the job of verifying that transmission across a program is feasible, and that the information which has been sent can be stopped. With transmission and the decision to use C2 and EBar gliders, there is only one possibility, consisting in the solitonic relation between the two. For convenience it is repeated in the right drawing of Figure 22, where it is seen that the spacing between EBars and C2's must lead to an encounter at Bresenham index 3.

Additional increments of integral EBar lengths are possible. Having a unique arrangement is fortunate, because all C2's are displaced equally by the passage of solitons, leaving their relative spacings intact. The only problems arise in guaranteeing that they are eventually stopped at the correct places.

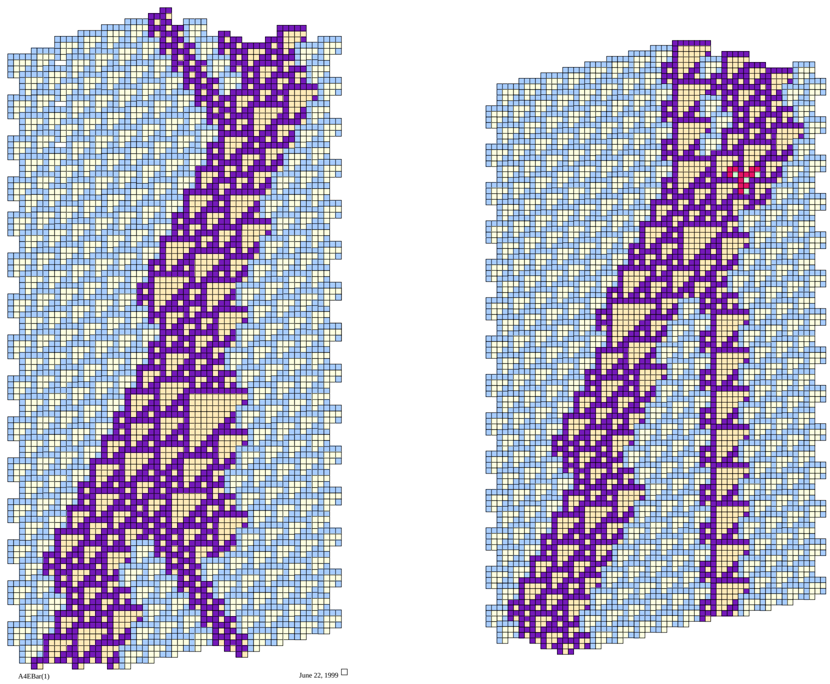

Figure 22: EBar interactions for which there are no alternative forms.

Interpreting the predicate creates spurious EBar gliders, leaving the two possibilities of ignoring them or destroying them. The latter is problematic; parity implies leaving residues. For A tetramers the fortunate alternatives exist, of letting EBars pass which arrive with Bresenham index 1, or of stopping those with indices 2, 3, or 6. Passage is shown in the left drawing of Figure 22.

Deciding which combinations are to be used is part of designing the Cyclic Tag System, and are not further discussed here. Just as accomodating the EBar packets arriving from the far right so that they can either be erased or transmitted (albeit with filtering) requires a synchronizing independent of the inner structure of the packets, so the stopping choices at the left require an invariance. The collision at Bresenham indes 2 is especially prompt.

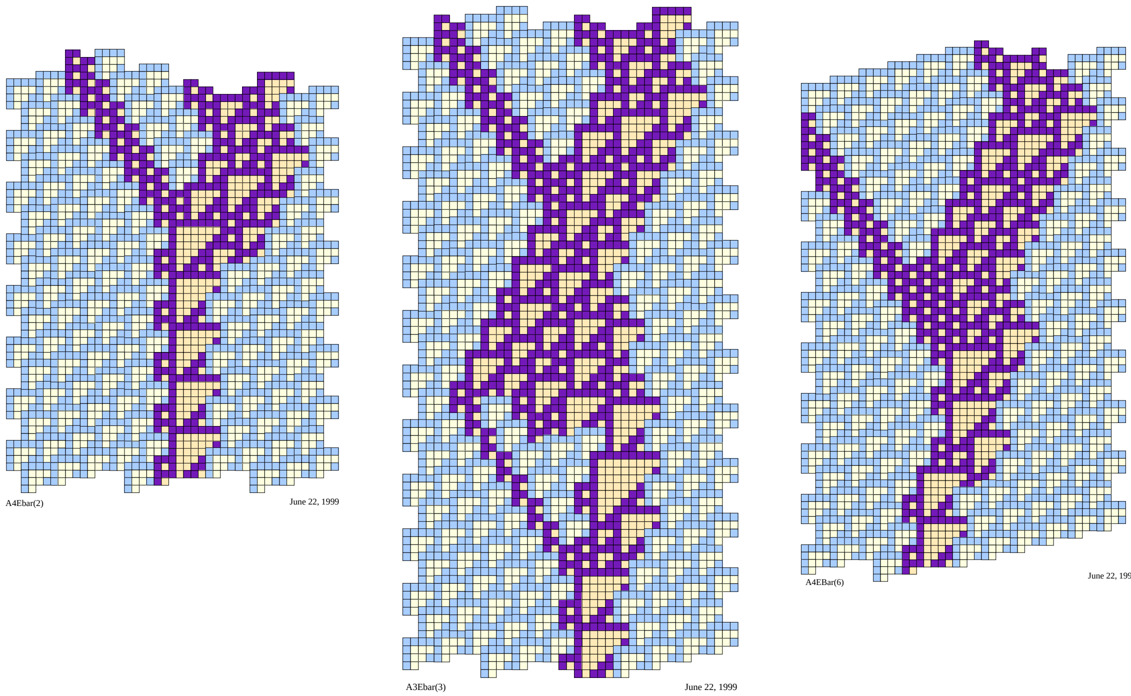

Figure 23: Three alternatives by which an A tetramer can stop an EBar and convert it into a C2.

Beyond checking the synchronization of the activities at the left and far right of the Cyclic Tag System, the verification that it is actually a computer is important. Presumably this verification exists in the literature, and in any event could be carried out symbolically without any drawings of the evolution of Rule 110.

Once that was done, it would still be entertaining to watch some simple calculations, such as the operation of a binary counter, or even of simple monary arithmetic with addition, subtraction, multiplication and division.

References

[1] Matthew Cook, "Introduction to the activity of rule 110" (copyright 1994-1998 Matthew Cook) `http://w3.datanet.hu/~cook/Workshop/CellAut/ Elementary/Rule110/110pics.html`

[2] Jim Giles, "What kind of science is this?" *Nature* **417** 216-218 (16 May 2002).

[3] Branko Grünbaum and G. C. Shepard, *Tilings and Patterns*, W. H. Freeman and Company, New York, 1987 (ISBN 0-7167-1193-1).

[4] Mariusz H. Jakubowski, Ken Steiglitz, and Richard K. Squier, "When can solitons compute?" *Complex Systems* **10** 1-21 (1996).

[5] Wentian Li and Mats G. Nordahl, "Transient behavior of cellular automaton rule 110," *Physics Letters A* **166** 335-339 (1992).

[6] Kristian Lindgren and Mats G. Nordahl, "Universal Computation in Simple One-Dimensional Cellular Automata," *Complex Systems* **4** 299-318 (1990).

[7] Harold V. McIntosh, "Wolfram's Class IV automata and a good Life," *Physica D* **45** 105-121 (1990).

[8] Harold V. McIntosh, "Rule 110 as it relates to the presence of gliders," Internet, CIEA, 1999 `http://delta.cs.cinvestav.mx/~mcintosh/comun/summer2001/bookcollisionsHtml/` .

[9] Harold V. McIntosh, "A concordance for Rule 110," Internet, CIEA, 1999 `http://delta.cs.cinvestav.mx/~mcintosh/comun/summer2001/bookcollisionsHtml/` .

[10] Genaro Juárez Martínez, "ATLAS," Internet, CIEA, 1999 `http://delta.cs.cinvestav.mx/~mcintosh/comun/summer2001/newweb/marco_home.html/` .

[11] A. M. Turing, "On computable numbers, with an application to the Entscheidungsproblem," *Proceedings of the London Mathematical Society Ser. 2* **42** 230- (1936).

[12] John von Neumann, *Theory of Self-reproducing Automata* (edited and completed by A. W. Burks), University of Illinois Press, 1966.

[13] Edward F. Moore, "Machine models of self reproduction," American Mathematical Society *Proceedings of Symposia in Applied Mathematics* **14** 17-33 (1962).

[14] Arthur W. Burks (editor), *Essays on Cellular Automata*, University of Illinois Press, Urbana, 1970.

[15] Sven Skyum, "Confusion in the Garden of Eden," *Proceedings of the American Mathematical Society* **50** 332-336 (1975).

[16] Alfred North Whitehead and Bertrand Russell, *Principia Mathematica* Cambridge University Press, 1910.

[17] Marvin Minsky, *Computation: Finite and Infinite Machines*, Prentice Hall, 1967.

[18] Stephen Wolfram (Ed.), *Theory and Applications of Cellular Automata*, World Scientific Press, Singapore, 1986 (ISBN 9971-50-124-4 pbk).

[19] Stephen Wolfram, *A New Kind of Science*, Wolfram Media, Inc., Champaign, Illinois, 2002 (ISBN 1-57955-008-8).

[20]

July 10, 1991