Complex Systems **2** (1988) 91–141

Parity Filter Automata*

Charles H. Goldberg
Department of Computer Science, Trenton State College,
Trenton, NJ 08625, USA

Abstract. Parity filter automata are a class of two-state cellular automata on the integer grid points of the real line in which cells are updated serially from left to right in each time period rather than synchronously in parallel. Parity filter automata support large numbers of "particles," or persistent repeating configurations, and the collision of these particles is frequently a "soliton" collision in which the particles interact, but from which both emerge with their identities preserved. This paper presents a theory of such parity filter automata. Period and velocity theorems for particles, existence and uniqueness theorems, conservation and monotone nonconservation laws, duration and phase shifts in soliton collisions, and other results are proved.

1. Introduction

It is rare that a class of objects can be understood almost completely. This paper describes just such a happy circumstance.

Parity filter automata are a class of two-state cellular automata on the integer grid points of the real line in which cells are updated serially from left to right in each time period rather than synchronously in parallel as is the case for most cellular automata studied. Each parity filter automaton is characterized by the radius r of the neighborhood of cells whose state values influence the updating of the current cell. In a parity filter automaton, the neighborhood consists of the r cells to the left of the central cell in the current time period and of the central cell and the r cells to the right of it in the previous time period. Except if all of these state values are zero, the updated state value at a cell is the modulo 2 inverse of the sum of the state values in the neighborhood.

*The author wishes to thank Professor Ken Steiglitz of Princeton University for introducing him to this class of cellular automata, for showing him many examples, and for sharing his conjectures, many of which are proved in this paper. The author also wishes to thank the Princeton University Department of Computer Science for use of their computer systems in preparing this manuscript, and for the many courtesies extended to him while a Visiting Faculty Fellow at Princeton.

Parity filter automata are of interest because they support large numbers of "particles," or persistent repeating configurations, and the collisions of these particles is frequently a "soliton" collision in which two particles interact, but both emerge from the collision with their identities preserved. This paper presents a theory of such parity filter automata.

Parity filter automata were first defined by Steiglitz and studied by Park, Steiglitz, and Thurston in [1]. Steiglitz, Kamal, and Watson [2] use the phase conjugations in certain soliton collisions in a parity filter automaton to construct a carry-ripple adder.

This paper puts the empirical findings in [1] and [2] on a solid foundation. Stability of finite configurations is proved as are existence and uniqueness of classes of particles. Formulae are derived for velocity and period of particles. Sufficient conditions for soliton collisions are derived and formulae proved for phase and spatial displacements in soliton collisions.

Infinite families of complex particles described in [3] are reinterpreted as systems of *orbiting particles*, permitting finite bounds on the number of particles.

Conservation principles are defined and used to analyze *time reversable* phenomena. A monotone nonconservation principle analogous to the Second Law of Thermodynamics, is proved and used to analyze time irreversable phenomena including the ultimate resolution of any initial configuration into a collection of particles.

Prior to the work in this paper, nearly everything that was known about parity filter automata was derived from empirical studies. In contrast, all results in this paper are proved.

2. Overview

The paper is organized as follows. Section 3 starts with very general definitions of two major classes of cellular automata, *parallel synchronous* cellular automata in which all states are updated simultaneously, and *serial* cellular automata in which states are updated in a predetermined order and newly updated state values affect the calculations of new state values at neighboring nodes in the same time period. Attention is then specialized to serial automata on the integer points of the real line and to a family of updating rules, Parity Filter Automata, that are based on the parity of the sum of nearby state values except for one case necessary to preserve regions of zero values.

The Stability Theorem, stated in section 4 and proved in section 7, shows that for such cellular automata, finite initial configurations remain finite at all subsequent times. Pencil and paper studies, including the original proof of the Stability Theorem suggest a strong relationship between the state values s_j^t and s_{j-r}^{t+1} where r is the radius of the updating neighborhood N_j. *Time-shifted* state transition diagrams are defined in section 5 to take advantage of this fact by placing related states under one another.

We then show in section 7 that only one out of every $r+1$ consecutive vertices undergoes a state change in each time period, leading to a Rapid Updating Rule, and we define the *byte size* for a parity filter automaton to be $r+1$ adjacent state values. Small configurations (i.e., less than one byte in width) are then analyzed completely. They are all, with the exception of the zero and single nonzero bit configurations, particles for which the average velocity and period can be calculated from the initial configuration. Their periods and velocities turn out to depend only on the radius r of the updating window and the number of nonzero states in the initial configuration.

In order to derive similar descriptions for large particles, the concept of *energy* of a configuration is defined in section 9. The evolution of a large configuration has a simple description in terms of the evolution of its energy states, from which formulae for the average velocity and period of large particles are derived in section 10. These formulae explain the empirically determined frequencies of particle periods and velocities in [1].

The energy of a configuration provides a unifying principle that pervades the theory and simplifies the recognition, statement, and proof of results. The fundamental properties of energy are given in the suggestive, but probably misnamed, "Second Law Of Thermodynamics" that says that energy evolves monotonically, but not strictly monotonically, downhill, and in the Critical Transition Lemma that give the details of when energy is conserved and when it is lost.

Impossible configurations and energy distributions are discussed in section 14, leading to upper bounds on the size, period, and number of particles for a parity filter automaton. Existence and uniqueness theorems for particles of period 1 are proved in section 15, and uniqueness is extended to other particles in section 16.

Collisions of particles are analyzed in section 18. In the absence of null transition windows, they are *soliton* collisions in which the identities of the two particles are preserved. A precise description of the velocity of each particle during collision, and the spatial and phase displacement of each particle after the collision is given. Sufficient conditions for soliton collisions are given which explain some of the soliton frequencies in [1].

Based on an understanding of soliton particle collisions, some of the "complex particles" in [3], e.g., those in figures 2, 10, and 15, may be reinterpreted as collections of particles with the same average velocity, but different fine velocity structure or phase. If two such particles are sufficiently close together, at certain phases of their periods they will touch, collide, and cross one another. However, since their average velocities are the same, the particles will not move apart after the collision, and so they will recollide and recross each other. The result is an *orbiting system* of simpler particles defined and analyzed in section 19.

Section 20 studies *tangent* or *osculating particles*. In section 21, collisions of small particles are shown to be soliton collisions, and in section 22, the "soliton collisions" used in [2] to construct a carry-ripple adder are shown to be in reality the product of three successive soliton collisions of a *system of almost orbiting particles*. The paper closes with questions for further study.

3. Definitions

Although all the results of this paper are specific to the class of cellular automata called *parity filter automata*, the definitions in this section are given initially in far greater generality for two purposes. First, the more general definitions point the way toward possible generalizations of the results of this paper. If they have abstracted the right features of the more specialized examples, they may shed some insight into which properties are special to these examples and which properties flow from deeper properties of cellular automata in general. At the very least, they will stimulate discussion about what the proper generalizations should be.

The second reason is simpler. The more general definitions provide a framework for relating the parity filter automata studied in this paper to the more familiar *parallel synchronous* cellular automata. They are all instances of the more general cellular automata defined here, and their points of divergence are better seen in this context.

Definition 1. A simply transitive regular graph *is a regular graph* $G = (V, E)$ *and a group* Φ *of automorphisms (i.e., self-maps) of* G *such that for every* $v, w \in V$ *there is exactly one automorphism* $\phi_{vw} \in \Phi$ *that maps* v *to* w.

Looking ahead to the next definition, that of a *cellular automaton*, we see that the notion of each node having the "same" updating rule would not make sense unless every vertex had the same degree (i.e., the graph is *regular*), and further unless there is a unique way of defining corresponding isomorphic neighborhoods. A group of automorphisms of the graph, i.e., maps of the graph to itself, provide a canonical or standard way to equivalence neighborhoods. The *transitive* property of the group of automorphism means that there is at least one self-map in the group that takes any given vertex v to any other given vertex w. Strengthening this property to *simply transitive* means that there also aren't too many automorphisms, i.e., that there is one and only one automorphism of the graph carrying v to w. The group property of the set of self-maps of the graph also provides a "compatibility" condition on the canonical equivalences of neighborhoods: if the equivalence of a neighborhood of u with a neighborhood of v is composed with the equivalence of the neighborhood of v with a neighborhood of w, the resulting equivalence is the same as if the neighborhoods of u and w were compared directly.

Fortunately, the examples all satisfy these conditions in a straightforward way; otherwise, the definitions of their updating rules and behavior would not make much sense.

Definition 2. *A* cellular automaton *is a family of identical microprocessors, one for each node of a simply transitive regular graph* G. *There is a set* S *of possible states of an individual microprocessor, independent of vertex* $v \in V$, *so that at any time* t, *each microprocessor* P_v *is in a state* $s_v^t \in S$. *Surrounding* *each vertex v is a neighborhood N_v of vertices. These neighborhoods are compatible with the automorphism group in the sense that N_v is mapped to N_w by the automorphism ϕ_{vw}. The state s_v^{t+1} of the microprocessor P_v at time $t+1$ is determined from the states of neighboring microprocessors by a rule $s_v^{t+1} = F_v([s_w^{t(v,w)}]_{w \in N_v})$. The function F_v is independent of time and the functions F_u and F_v are identical when the wth argument of F_u is identified with the $\phi_{uv}(w)$th argument of F_v. The time dependence function $t(v,w)$ always has value t or $t+1$, but the value can depend on v and w. The rule F_v, or more precisely the collection of rules $\{ F_v \mid v \in V \}$, is called the updating rule of the cellular automaton.*

Definition 3. *An additional condition usually placed on the updating rule, the* null stability condition, *is that the distinguished state value 0 is self-perpetuating in the sense that $s_v^{t+1} = 0$ whenever $s_w^t = 0$ for all $w \in N_v$.*

All cellular automata in this paper satisfy the null stability condition. The function $t(v,w)$ in the update rule is the subject of the next two definitions.

Definition 4. *An updating rule for a cellular automaton is called* parallel synchronous *if $t(v,w) = t$ for all vertices $v, w \in V$.*

In a parallel synchronous cellular automaton, the state s_v^{t+1} of the microprocessor P_v at time $t+1$ is entirely determined by the states s_w^t of the microprocessors at the neighboring vertices $w \in N_v$ at time t. In effect, the states of all the microprocessors are simultaneously updated in parallel from time t to time $t+1$. Conway's "Game of Life" and the automata studied by Wolfram [5,6] are parallel synchronous cellular automata.

Definition 5. *An updating rule for a cellular automaton is called* serial *if there is a total order relation '<' on V preserved by the automorphisms of Φ and if*

$$t(v,w) = \begin{cases} t+1 & \text{if } w < v; \\ t & \text{if } w \geq v. \end{cases}$$

In a serial updating rule, microprocessors are updated from time t to time $t+1$ in sequence, starting at the smaller elements of V and proceeding toward the larger elements of V. As soon as a state value is updated, the new value is immediately available to neighboring nodes for use in their updating in the same time period. Steiglitz [1,2] calls serial updated cellular automata *filter automata* because their behavior resembles Infinite Impulse Response (IIR) digital filters. Serial updating arises naturally when one simulates a cellular automaton on a single processor von Neumann computer.

Example 1 (The Game of Life) *The graph G consists of the integer grid points in the plane and edges connecting horizontally and vertically adjacent grid points. Automorphisms Φ are integer translations in two dimensions.*

The state set S is the set $\{0,1\}$, with 1 meaning that the processor at a vertex is "alive" and 0 meaning that it is "dead." The neighborhood $N_{(x,y)}$ of a grid point (x,y) consists of the nine points $\{(x',y') \mid |x-x'| \le 1$ and $|y-y'| \le 1\}$. Updating is parallel synchronous. Using the traditional anthropomorphic terminology of this example, a new cell $s^{t+1}_{(x,y)} = 1$ is "born" at a previously unoccupied node $s^t_{(x,y)} = 0$ if and only if the sum of the state values at the nine points of its neighborhood is 3; a "living cell" $s^t_{(x,y)} = 1$ "dies of loneliness" at time $t+1$, i.e., $s^{t+1}_{(x,y)} = 0$, if the sum of the state values in its nine point neighborhood is less than 3, it "dies of overcrowding," i.e., $s^{t+1}_{(x,y)} = 0$, if the sum is greater than 4, and it "lives," i.e., $s^{t+1}_{(x,y)} = s^t_{(x,y)} = 1$, if the sum is 3 or 4.

Example 2 (Parity Filter Automata) *The graph G consists of the integer grid points on the real line with edges connecting adjacent integer points. Automorphisms are integer translations. The state set S is the set $\{0,1\}$, and the neighborhood N_x of a point x is the symmetric interval of $2r+1$ points centered at x, i.e., $N_x = \{x' \mid |x-x'| \le r\}$. The radius r of the neighborhoods is a parameter defining a family of parity filter automata. The updating rule is*

$$\begin{aligned} s_x^{t+1} &= F(s_{x-r}^{t+1}, \ldots, s_{x-1}^{t+1}, s_x^t, s_{x+1}^t, \ldots, s_{x+r}^t) \\ &= \begin{cases} 0 & \text{if } s_{x-r}^{t+1} + \cdots + s_{x-1}^{t+1} + s_x^t + s_{x+1}^t + \cdots + s_{x+r}^t \text{ is odd;} \\ 1 & \text{if } s_{x-r}^{t+1} + \cdots + s_{x-1}^{t+1} + s_x^t + s_{x+1}^t + \cdots + s_{x+r}^t \text{ is even} \\ & \text{and nonzero;} \\ 0 & \text{if } s_{x-r}^{t+1} + \cdots + s_{x-1}^{t+1} + s_x^t + s_{x+1}^t + \cdots + s_{x+r}^t \text{ is zero.} \end{cases} \end{aligned}$$

Updating is serial from left to right.

Example 3 (Pascal Triangle modulo p) *The graph G consists of the integer grid points on the real line with edges connecting adjacent integer points. The state set $S = \mathcal{Z}_p = \{0,1,\ldots,p-1\}$, the integers modulo p. The updating rule, $s_x^{t+1} = (s_{x-1}^t + s_{x+1}^t) \bmod p$, is parallel synchronous, but the neighborhood $N_x = \{x-1, x+1\}$ is a deleted neighborhood in the sense that it contains points near x, but not x itself. If the starting states of the vertices of the automaton are all zero except for a single state of 1 at the origin, the automaton evolves into a state where the nonzero vertex states are binomial coefficients modulo p separated by single zero states. The two-dimensional plot of the time evolution from this initial state is a Pascal Triangle modulo p with fractal-like appearance (see figure 1).*

Definition 6. *Following Wolfram [5], we call an updating rule* totalistic *if the new state s_v^{t+1} depends only on the sum of the state values at vertices $w \in N_v$.*

$$s_v^{t+1} = F\Big(\sum_{w \in N_v} s_w^{t(v,w)}\Big)$$

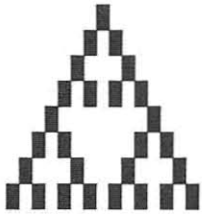

Figure 1: Pascal Triangle modulo 2.

Parity Filter Automata and Pascal Triangle automata are totalistic, although the neighborhoods in the Pascal Triangle examples must exclude the center point to satisfy the definition. (s_x^t does not appear in the sum.) Any totalistic rule can be made nontotalistic by enlarging the neighborhoods, but the converse is not true. The Game of Life fails to be totalistic because the updated state values depend on both the current state and the neighborhood state sum at a point.

4. Stability

In general, serial updating rules on a line automaton are *unstable* in the sense that finite configurations at time t can (and usually do) evolve to configurations at time $t+1$ that extend infinitely far to the right. The Stability Theorem proves that this does not happen in parity filter automata.

Starting with the Stability Theorem below, attention will be specialized to the class of Parity Filter Automata.

The set of states $s_w^{t'}$ that can be influenced by the state value s_v^t at vertex v and time t expands in a *cone of influence* from v. For parallel synchronous updating rules, if r is the radius of N_v, that is, the maximum distance (i.e., number of edges) from v to any $w \in N_v$, then $s_w^{t'}$ cannot be influenced by s_v^t if $\mathrm{dist}(v, w) > r(t'-t)$. The window radius r is a natural upper bound on the speed of propogation of state information in the automaton, which we call the *speed of light in the automaton.*

For general serial updating rules, the cone of influence satisfies the same relationship if $w < v$, but it is possible for s_v^t to affect s_w^{t+1} for all $w \geq v$.

Definition 7. *A cellular automaton is called* stable *if whenever only finitely many vertices are in nonzero states at time* t*, there will be only finitely many vertices in nonzero states at time* $t+1$.

All parallel synchronous cellular automata with neighborhoods of finite radius are stable by this definition. The principal theorem of this section asserts that parity filter automata are stable in this sense.

Theorem 1 (Stability Theorem) *If the state of a parity filter automaton at time* t *contains only finitely many nonzero values, then the state of the* *automaton at time $t+1$ will also contain only finitely many nonzero values. Moreover, if s_l^t is the leftmost nonzero value and s_m^t the rightmost nonzero value at time t, then $s_{l-r+1-k}^{t+1} = 0$ and $s_{m+k}^{t+1} = 0$ for all $k > 0$.*

The Stability Theorem was the first major result I proved about Parity Filter Automata. Its original proof was of necessity long because there was then no machinery with which to work. In this paper, we first develop the fundamental machinery of Parity Filter Automata implicit in the original proof, most particularly the concept of *time-shifted state diagrams.* The Stability Theorem is now proved in section 7 as an easy consequence of the Rapid Updating Rule.

5. Time-shifted state diagrams

The state value s_j^{t+1} is of necessity influenced by s_{j+r}^t, the state value that enters the computational window for the first time in the calculation of s_j^{t+1}. While it is always possible for a state value in a serially updated automaton to propogate to the left at the speed of light r, in a parity filter automaton, nearly all state value propogation is of this kind. For this reason, many properties of parity filter automata are easier to describe when the state values at time $t+1$ are shifted r places to the right relative to those at time t.

Definition 8. *The* time-shifted state diagram *of an initial configuration and cellular automaton defined on the integer points of the real line is the array of state values*

$$\{a_j^t\} = \{s_{j-tr}^t\}.$$

6. Evolution of configurations

For clarity of exposition, all subsequent space-time diagrams of the states of a parity filter automaton will be time-shifted diagrams, with state values at time $t+1$ shifted right r places relative to those at time t. Thus the computational window for state s_i^{t+1} previously drawn as

$$\begin{array}{ccccccc} & & & s_i^t & s_{i+1}^t & \cdots & s_{i+r}^t \\ s_{i-r}^{t+1} & \cdots & s_{i-1}^{t+1} & s_i^{t+1} & & & \end{array}$$

will now be drawn as the $(r+1) \times 2$ rectangle

$$\begin{array}{cccc} s_i^t & s_{i+1}^t & \cdots & s_{i+r}^t \\ s_{i-r}^{t+1} & \cdots & s_{i-1}^{t+1} & s_i^{t+1} \end{array}$$

or using the shifted notation $a_j^t = s_{j-rt}^t$,

$$\begin{array}{cccc} a_{j-r}^t & \cdots & a_{j-1}^t & a_j^t \\ a_{j-r}^{t+1} & \cdots & a_{j-1}^{t+1} & a_j^{t+1} \end{array}$$

Four special sets of transition values in the computational window are singled out for special names because they play special roles in the evolution of configurations.

Definition 9. *The computational window*

$$\begin{array}{cccc} 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \end{array}$$

is called the null transition window *and the associated state transition a* null transition.

It is the only exception to the usual case of calculating the new state value s_i^{t+1} by reversing the parity of the sum of the neighboring state values

$$s_i^{t+1} \equiv 1 + s_{i-r}^{t+1} + \dots + s_{i-1}^{t+1} + s_i^t + s_{i+1}^t + \dots + s_{i+r}^t \pmod 2.$$

which is equivalent to the more symmetric relationship

$$s_{i-r}^{t+1} + \dots + s_{i-1}^{t+1} + s_i^{t+1} + s_i^t + s_{i+1}^t + \dots + s_{i+r}^t \equiv 1 \pmod 2.$$

All other transitions are called *parity reversing* because they satisfy this equation for the calculation of s_i^{t+1}. The null transition is the only *parity preserving* transition that satisfies similar equations with the term 1 changed to 0.

A large part of the work in the proofs of theorems in this paper is devoted to proving that null transitions do not occur, so that all computational windows considered reverse parity.

Definition 10. *The computational window*

$$\begin{array}{cccc} 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \end{array}$$

is called the time irreversable transition window *and its associated transition* a time irreversable transition.

Although the time irreversable transition satisfies the parity reversing rule, it is the only valid computational window that would not represent a valid transition if the roles of times t and $t+1$ were reversed, that is, if time were "run backward" and updating done from right to left.

Definition 11. *The computational window*

$$\begin{array}{cccc} 0 & \dots & 0 & 1 \\ 0 & \dots & 0 & 0 \end{array}$$

is called the left particle boundary window *and its transition* a left particle boundary transition *because it occurs at the leftmost nonzero state value in a configuration. The computational window*

$$\begin{array}{cccc} 0 & 0 & \dots & 0 \\ 1 & 0 & \dots & 0 \end{array}$$

is called the right particle boundary window *and its associated transition* a right particle boundary transition.

Left particle boundary transitions occur at the leftmost nonzero state value in a configuration, as well as at the leftmost nonzero state value of any subconfiguration whose evolution at time t is not influenced by the nonzero states to the left of it. In particular, left particle boundary transitions occur at the leftmost nonzero states of subconfigurations called *particles*, to be studied next.

Similarly, right particle boundary transitions often occur at the rightmost nonzero state values in a configuration, as well as at the rightmost nonzero state value of any subconfiguration whose evolution at time t will not influence the evolution of nonzero states to the right of it. In particular, right particle boundary transitions will occur at the right end of particles.

7. Dynamics of small particles

Definition 12. *A* small configuration *is a configuration of the automaton in which all nonzero states are contained within a span of $r+1$ consecutive vertices.*

Definition 13. *The number $r+1$ is called the* byte size *for the parity filter automaton.*

We describe the evolution of a small particle with k nonzero states. If $k=1$, then the configuration "dies," i.e., decays to the zero configuration in one time period because no computational window has more than one nonzero value. We assume henceforth that $2 \le k \le r+1$. Using time-shifted diagrams, the situation is as follows.

$$\begin{array}{cccccccccccc} \dots & 0 & b_0 & b_1 & \dots & b_r & 0 & 0 & \dots & 0 & 0 & \dots \\ \dots & 0 & c_0 & c_1 & \dots & c_r & a_0 & a_1 & \dots & a_r & a_{r+1} & \dots \end{array}$$

where b_0 is the leftmost nonzero state value at time t. All computational windows to the left of the one for c_0 are null windows, and the window for c_0 is a left particle boundary window. Thus $c_0 = 0$. Thereafter, the computational window becomes "primed" in the sense of the following definition.

Definition 14. *A computational window with time-shifted diagram*

$$\begin{array}{cccc} a_{j-r}^t & \dots & a_{j-1}^t & a_j^t \\ a_{j-r}^{t+1} & \dots & a_{j-1}^{t+1} & a_j^{t+1} \end{array}$$

is called paired *if $a_{j-i}^t = a_{j-i}^{t+1}$ for all i in the range $1 \le i \le r$. It is called* primed *if there is exactly one i in the range $1 \le i \le r$ for which $a_{j-i}^t + a_{j-i}^{t+1} = 1$ and $a_{j-i}^t = a_{j-i}^{t+1}$ for the remaining i in this range. The etymology of the term derives from "priming a pump" and not from prime numbers.*

Since a primed window cannot be a null window, $a_j^{t+1} = a_j^t$ for primed windows. If a paired window is not null, then $a_j^{t+1} = 1 - a_j^t$ because parity must be reversed.

Lemma 1 (Primed Window Lemma) *Once a computational window becomes primed, it remains primed for r consecutive vertices and then it becomes paired.*

Proof. First we show that if the window for a_j^{t+1} is primed but the window for a_{j-1}^{t+1} is not primed, then the unmatched pair of states is $a_{j-1}^{t+1} \neq a_{j-1}^{t}$. Consider the time-shifted diagram

$$\begin{array}{ccccc} a_{j-r-1}^{t} & a_{j-r}^{t} & \cdots & a_{j-1}^{t} & a_{j}^{t} \\ a_{j-r-1}^{t+1} & a_{j-r}^{t+1} & \cdots & a_{j-1}^{t+1} & a_{j}^{t+1} \end{array}$$

Because $a_{j-i}^{t}+a_{j-i}^{t+1}=1$ for some i in the range $1 \leq i \leq r$, the window for a_{j-1}^{t+1} is not null. Thus parity is reversed and $a_{j-r-1}^{t} = a_{j-r-1}^{t+1}$. If i were greater than 1, the window for a_{j-1}^{t+1} would be primed. We conclude that $i = 1$ and $a_{j-1}^{t} + a_{j-1}^{t+1} = 1$.

As a result, the next $r-1$ computational windows are not null windows and therefore reverse parity. We easily show that $a_{j+i}^{t+1} = a_{j+i}^{t}$ for $0 \leq i \leq r-1$. Thus each of these computational windows is primed and the window for a_{j+r}^{t+1} is paired. ∎

The Primed Window Lemma provides an extremely rapid way of updating state values from time t to time $t+1$, which is expressed in the next theorem in the form of a three-state automaton. The updating mechanism is either in its initial state SLPB ("seeking a left particle boundary") where it copies strings of zero state values, or in the state PW ("primed window") where it copies possibly nonzero state values unchanged from time t to time $t+1$, or in the state CTBB ("critical transition on the byte boundary") where it inverts a state value and continues in the PW ("primed window") state. Updating is extremely rapid because at most one of each $r+1$ consecutive state values changes from time t to time $t+1$, and the sum of the state values in the computational window never needs to be calculated.

Theorem 2 (Rapid Updating Rule) *A parity filter automaton may be updated from time t to time $t+1$ by the following automaton with three states SLPB, PW, and CTBB. State SLPB is the initial state.*

State SLPB *If a zero state $a_j^t = 0$ is encountered, then a_j^{t+1} is set to zero and the automaton remains in state SLPB to read a_{j+1}^t. If $a_j^t = 1$ then a_j^{t+1} is set to zero, and the automaton enters state PW.*

State PW *r consecutive states a_j^t are copied to the corresponding a_j^{t+1}. If all copied states were zero, the automaton returns to state SLPB; otherwise it enters state CTBB to process the next node.*

State CTBB *The current state value a_j^t is inverted, i.e., $a_j^{t+1} = 1 - a_j^t$ and the automaton enters state PW regardless of the state value a_j^t.*

Proof. All cases follow immediately from the Primed Window Lemma and the definition of the state transition rule for parity filter automata. ∎

The Rapid Updating Rule provides a simple proof of the Stability Theorem, restated here in terms of time-shifted state diagrams. The original form of the Stability Theorem is immediately seen to be equivalent to the time-shifted version of the theorem, and therefore follows as soon as theorem 3 is proved.

Theorem 3 (Stability Theorem, Time-shifted Version) *If the state of a parity filter automaton at time t contains only finitely many nonzero values, then the state of the automaton at time $t+1$ will also contain only finitely many nonzero values. Moreover, if a_l^t is the leftmost nonzero value and a_m^t the rightmost nonzero value at time t, then $a_{l+1-k}^{t+1} = 0$ and $a_{m+r+k}^{t+1} = 0$ for all $k > 0$.*

Proof. The Rapid Updating Automaton is in state SLPB copying zero states until a_l^t where it switches to state PW after setting $a_l^{t+1} = 0$. This proves the assertion of the Stabiltiy Theorem at the left side of the configuration. At the right side, if $a_m^t = 1$ is encountered in the state CTBB, then $a_m^{t+1} = 0$, the next r zero states $a_{m+1}^t, a_{m+2}^t, \ldots, a_{m+r}^t$ are copied to $a_{m+1}^{t+1}, a_{m+2}^{t+1}, \ldots, a_{m+r}^{t+1}$, and the Rapid Updating Automaton enters the state SLPB forever, since there are no remaining nonzero states to the right. If on the other hand a_m^t is encountered as one of a group $a_{m+j-r}^t, a_{m+j-r+1}^t, \ldots, a_{m+j-1}^t$, $0 < j \leq r$, of r state values copied by the Rapid Updating Automaton in the state PW, then $a_{m+j}^t = 0$ is encountered in the state CTBB. In this case, a_{m+j}^{t+1} is set to 1, which turns out to be the rightmost nonzero state value at time $t+1$, the Rapid Updating Automaton switches to state PW to copy r consecutive zero states before returning to the state SLPB forever. In either case, the rightmost nonzero state at time $t+1$ is a_{m+j}^{t+1} for some $j, 0 \leq j \leq r$, and the conclusion follows. If a_m^t is encountered in the state SLPB, then $a_m^{t+1} = 0$, the next r zero states are copied with the Rapid Updating Automaton is in state PW, and the Rapid Updating Automaton enters state SLPB forever. In this case, the rightmost nonzero state value at time $t+1$, if any, is to the left of a_m^{t+1}, and the conclusion also follows. ∎

Evolution of small configurations

Returning to the analysis of small configurations and the singly subscripted notation introduced for small configurations, the Rapid Updating Theorem implies that all states at time $t+1$ to the left of c_0 are zero, that $c_0 = 0$ as the automaton enters the primed window state PW at a left particle boundary, and that $c_1 = b_1$, $c_2 = b_2$, $\ldots$, $c_r = b_r$. If b_0 is the only nonzero state, the automaton returns to state SLPB and stays there copying all the remaining zero states. If at least one additional b_i is nonzero, the automaton enters state CTBB, sets $a_0 = 1$, and returns to state PW to copy r zero states and then to state SLPB to copy the remaining zero states so that $a_j = 0$ for all $j \geq 1$. Thus the window for c_0 is a left particle boundary window terminating a sequence of null windows extending to minus infinity, and the window for a_r is a right particle boundary window followed by nothing but null windows extending to plus infinity.

The dynamics of small configurations can now be described precisely. If the number of nonzero state values $k = 1$, then the configuration dies in the next time period. If $k \geq 2$, the configuration transforms in each time period to another configuration that also has k nonzero state values contained within the span of a single byte of size $r + 1$. Each individual state value of the configuration remains fixed in the time-shifted diagram, except for the leftmost nonzero state which makes a quantum leap of exactly one byte width to the right. Thus the configuration transforms like a caterpillar tread of circumference $r + 1$ or a perfect undamped accoustical delay line.

In the second time period, the second nonzero state value (from the left) of the configuration is the leading edge of the configuration, and it jumps $r + 1$ positions to the right in the time-shifted diagram. Thus, after k time periods, each of the k nonzero state values in the initial configuration has jumped $r + 1$ positions to the right on exactly one occasion and remained fixed in the other $k - 1$ transitions. The result is that after k time periods, the original configuration reappears, displaced by $r + 1$ positions to the right in the time shifted diagram.

In the unshifted diagram, the displacement $d = kr - (r + 1)$ to the left. Although a small configuration appears to move with average velocity $v = d/k = r - \frac{r+1}{k}$, which is less than r, the speed of light, in fact this motion is composed of two parts:

1. Motion of the individual bits of the configuration to the left at the speed of light, and
2. Rotation of the nonzero states of the configuration by quantum leaps of $r + 1$ vertices to the right.

If n is a divisor of the byte size $r + 1$, and the initial configuration consists of n repetitions of a subpattern of width $(r + 1)/n$ vertices, then each subpattern will have k/n nonzero states, and the configuration will repeat after k/n time periods, a divisor of the full period k predicted for general small configurations with k nonzero states within one byte.

Definition 15. *A small configuration is called* a particle *if it repeats after p time periods at a displacement d_s to the right in the time-shifted diagram and $d = pr - d_s$ to the left in the unshifted diagram.*

Theorem 4 (Small Particle Period and Velocity Theorem) *All small configurations with $k \geq 2$ nonzero states within a span of one byte of size $r + 1$ consecutive vertices are particles. Their period is k, shifted displacement $d_s = r + 1$ to the right, unshifted displacement $d = kr - (r + 1)$ to the left, average shifted velocity $v_s = (r + 1)/k$ and unshifted velocity $v = (kr - (r + 1))/k$. If n divides $r + 1$ and the initial configuration consists of n repetitions of a subpattern of width $(r + 1)/n$, then there is a divisor period of k/n with displacements d_s/n and d/n.*

Unshifted Diagram Time-shifted Diagram

Figure 2: Electrons for $r = 4$.

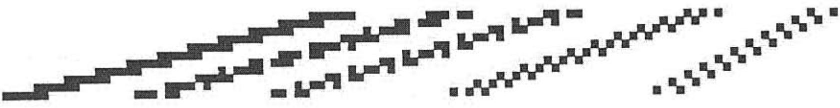

Figure 3: Unshifted configuration diagram for five small particles, $r = 5$.

8. Examples of small particles

The simplest examples of small particles consist of $r+1$ consecutive nonzero state values. In addition to its full period $k = r+1$, with displacements $d_s = r+1$ and $d = (r+1)(r-1)$, it has a divisor subperiod of $k/n = 1$ based on the divisor $n = r+1$. The displacements for this subperiod are $d_s = 1$ and $d = r-1$. The average velocity, $v = r-1$, for this particle is the fastest possible by the Stability Theorem, which is why this particle is called a "photon" in earlier literature [1,4].

The slowest possible small particle consists of exactly two nonzero states. (There are slower, even motionless, larger particles.) If the two nonzero states are adjacent, we have an "electron" or "inchworm" that moves with period 2 in cycles of one short step and one long step (see figure 2).

Figure 3 shows a collection of small particles of differing velocities for $r = 5$. The influence of density (i.e., number of nonzero states) on velocity is clearly illustrated.

Figure 4 shows their time-shifted diagrams in which the "rotation" of nonzero states is more evident because the component of the particle evolution consisting of uniform motion of states to the left at the speed of light has been factored out.

Figure 5 shows something called the *energy diagram* of the same five particles in the sense defined in section 9.

Figure 4: Time-shifted configuration diagram for five small particles, $r = 5$.

Figure 5: Energy diagram for five small particles with $r = 5$.

Spaced particles

Given a small particle in the parity filter automaton with window radius r, there are associated "spaced" particles in the automaton with window radius $r' = n(r+1)-1$ formed by inserting $n-1$ zero states after each state, zero or nonzero, of the original particle. Thus the "photon" 111 for $r = 2$ becomes the more tenuous particles 101010 for $r = 5$ and 100100100 for $r = 8$. The full period of a "spaced" small particle remains k, the same as the original particle, but the shifted displacement d_s and velocity v_s are multiplied by n. Any divisor periods present in the original particle are also present in the spaced particles.

9. Dynamics of large particles

We start by considering arbitrary, large configurations, i.e., those for which the nonzero states cannot be contained in a single byte of width $r+1$ consecutive vertices. Only configurations with a finite number of nonzero states at some time t are considered in this paper. The Stability Theorem then guarantees that such configurations will remain of finite extent for all subsequent time periods. We reserve the word *particle* for the following special class of configurations.

Definition 16. *A configuration or state of a cellular automaton is called a* particle *if*

1. *It reappears after p time periods at a right displacement d_s from its original position in the shifted diagram, and*
2. *In each time period, there is exactly one left particle boundary transition and one right boundary transitition.*

We remark that small particles, as defined in section 7, satisfy this definition. For such small particles, the period p is the number k of nonzero states, a quantity preserved in all phases of the particle's evolution. This result does not generalize in the most obvious way to large particles. The period of a large particle is not equal to the number of nonzero states in the particle, nor is the number of nonzero states even constant over time during the evolution of a large particle. The following quantity, called the *energy* of a configuration, takes over the role of the number of nonzero states of a small configuration in a way that permits generalizing the results about period and velocity to large configurations. It is then shown in retrospect that the calculations of period and velocity based on energy that apply to all particles, large or small, give the same results for small particles as those based on the number of nonzero state values.

Definition 17. *The* energy *of a configuration is the sum*

$$\mathcal{E}^t = \sum_{i=-\infty}^{\infty} |a_i^t - a_{i-r-1}^t|$$

For small particles, each nonzero state makes exactly two contributions to the sum, so the energy $\mathcal{E}^t = 2k$ at all times t.

The energy of a configuration may be decomposed into a double summation to show the independence of each relative position within a byte.

$$\mathcal{E}^t = \sum_{j=0}^{r} \sum_{n=-\infty}^{\infty} |a_{j+n(r+1)}^t - a_{j+(n-1)(r+1)}^t|$$

There is a contribution to the energy whenever a relative position in a byte is turned either on or off relative to the state of the same bit in the previous byte. Consequently, the energy of a (finite) configuration is always even. The most important fact about energy is contained in the following theorem, whose proof occupies the remainder of this section.

Theorem 5 (Second Law of Thermodynamics) *The energy of a configuration is never increased, i.e.,* $\mathcal{E}^{t+1} \leq \mathcal{E}^t$.

Corollary 1. *Every configuration evolves to a configuration of constant energy.*

To prove the Second Law, we must study the transitions of a configuration with particular regard to where the energy associated with a position at time t appears at time $t+1$. The first nonnull window encountered is the left particle boundary window associated with the leftmost nonzero bit b_0 of the configuration. (All diagrams are time-shifted.)

$$\begin{array}{cccccccccccc} \ldots & 0 & b_0 & b_1 & \ldots & b_r & b_{r+1} & \ldots & b_{2r+2} & b_{2r+3} & \ldots \\ \ldots & 0 & c_0 & c_1 & \ldots & c_r & c_{r+1} & \ldots & c_{2r+2} & c_{2r+3} & \ldots \end{array}$$

Since $c_0 = 0$, the energy associated with $b_0 = 1$ at time t is not carried over in the same position as energy associated with c_0 at time $t+1$. Instead, we may consider this energy as being transformed into energy associated with the priming of the computational window that occurs at a left particle boundary. The conservation or reduction of total configuration energy will depend on whether the energy associated with the primed window can be retransformed into positional energy associated with another vertex of the graph.

By the Primed Window Lemma, the windows for $c_1, c_2, \ldots, c_r$ are primed windows, so $c_1 = b_1$, $c_2 = b_2$, ..., $c_r = b_r$. Since the previous byte was all zero at both times t and $t+1$, whatever energy was carried by $b_1, b_2, \ldots, b_r$ is still carried by $c_1, c_2, \ldots, c_r$.

Definition 18. *In the updating of a parity filter automaton, the calculation of c_i is called a* critical transition *if the following three conditions are satisfied.*

1. *The calculational window for c_i is paired, i.e., $c_{i-1} = b_{i-1}$, $c_{i-2} = b_{i-2}$, $\ldots$, $c_{i-r} = b_{i-r}$.*
2. *$c_{i-r-1} = 1 - b_{i-r-1}$, i.e., there was a state change on the byte boundary in the previous byte.*
3. *Since the most recent left particle boundary, there is exactly one bit of positional energy at time t that has not yet been transformed into positional energy at time $t+1$.*

A transition satisfying these conditions is called a *critical transition* because what happens at c_i determines whether a particle ends there or whether it continues into the next byte. Critical transitions can only take place on a *byte boundary*, i.e., a multiple of $r+1$ positions to the right of the most recent left particle boundary transition.

Lemma 2 (Critical Transition Lemma) *If there is a critical transition at c_i then the following five cases, 1, 2a1, 2a2, 2b1, and 2b2, describe what can happen to the energy pending from the last left particle boundary.*

Case 1 *If $b_{i-k} \neq 0$ for some k with $1 \leq k \leq r$, then there will be no particle boundary at c_i, the window will be primed from c_{i+1} to c_{i+r} so the pending energy is still pending, and there will be another critical transition at c_{i+r+1}.*

Case 2a *If $b_{i-k} = 0$ for all k with $1 \leq k \leq r$ and $b_{i-r-1} = 0$, then there is a right particle boundary window at c_{i-1}, energy is conserved in the sense that all positional energy encountered at time t starting at the most recently encountered energy-bearing left particle boundary up to the current right particle boundary reappears at time $t+1$, and the computational window has r consecutive paired zeros seeking the next nonzero state at time t. Case 2a has two subcases.*

Case 2a1 *If $b_i = 0$, then there is a null transition at c_i.*

Case 2a2 *If $b_i = 1$, then there is another left particle boundary transition at c_i.*

Case 2b *If $b_{i-k} = 0$ for all k with $1 \leq k \leq r$ and $b_{i-r-1} = 1$, then there is a time irreversable transition at c_{i-1} and there are two subcases.*

Case 2b1 *If $b_i = 0$ then two bits of energy present at time t between the most recent left particle boundary and c_i are lost at time $t+1$.*

Case 2b2 *If $b_i = 1$ then there is an unusual left particle boundary window at c_i which did not contribute additional energy at time t, the windows are primed from c_{i+1} to c_{i+r} so energy from a previous left particle boundary is still pending, and there will be another critical transition at c_{i+r+1}.*

Proof. In Case 1, $b_{i-k} \neq 0$ for some k with $1 \leq k \leq r$. Thus parity is reversed in the calculaton of c_i and $c_i = 1 - b_i$. Calculating the energy at position i,

$$\begin{aligned}
|c_i - c_{i-r-1}| &= |1 - b_i - (1 - b_{i-r-1})| \\
&= |-b_i + b_{i-r-1}| \\
&= |b_i - b_{i-r-1}|
\end{aligned}$$

Thus there is energy at c_i at time $t+1$ if and only if there was energy at b_i at time t. Since the calculational window is paired when c_i is calculated from b_i, the Primed Window Lemma implies that the next r windows will be primed, the next r states will be copied unchanged, and that there will therefore be a critical transition at c_{i+r+1}.

In Case 2, $b_{i-k} = 0$ for all k with $1 \leq k \leq r$. We break Case 2 into two subcases depending on the value of b_{i-r-1}. In Case 2a, the state $b_{i-r-1} = 0$.

$$\begin{array}{cccc}
0 & 0 \; \ldots & 0 & b_i \\
1 & 0 \; \ldots & 0 & 0
\end{array}$$

Thus at the previous critical transition $c_{i-r-1} = 1$ as shown and the window for c_{i-1} is a right particle boundary window. There are two subcases depending on the value of b_i, and in both of them energy is conserved.

In Case 2a1, the state $b_i = 0$. Then $c_i = 0$ as the result of a null transition, and there is energy at c_i, but not at b_i. The part of the configuration between the most recent left particle boundary and the right particle boundary has transformed without loss of energy since the energy lost at the left particle boundary is regained at c_i.

In Case 2a2, the state $b_i = 1$. Again $c_i = 0$, but now there is positional energy at both b_i and c_i. However there is a way of interpreting the energy distribution that is consistent with Case 2a1. We consider the energy at c_i as completing the energy transformation of the particle whose right particle boundary window is at c_{i-1}. We regard b_i as the leftmost nonzero state of a new particle. With this interpretation, the energy at b_i is not expected to appear at c_i, but to go instead into energy associated with priming the computational window at the start of a new particle. This happens in Case 2a2, as it did for the initial left particle boundary transition.

To summarize what happens in Case 2a, a right particle boundary window is encountered. The energy of the primed calculational window is deposited at c_i, and energy is conserved. The difference between Cases 2a1 and 2a2 is that in Case 2a1, there are one or more null windows before the leftmost nonzero state of a new particle is encountered (if there is another one), while in Case 2a2, the left particle boundary of the next particle is encountered immediately at b_i.

Definition 19. *We call the situation of Case 2a2* osculating particles *or* tangent particles. *The defining characteristic is that the last bit of energy of one particle transforms to the positon that the first bit of energy of the next particle is leaving.*

In Case 2b, the state $b_{i-r-1} = 1$.

$$\begin{array}{cccc} 1 & 0 \;\ldots\; 0 & b_i \\ 0 & 0 \;\ldots\; 0 & 0 \end{array}$$

Thus $c_{i-r-1} = 0$ to have a primed window for c_{i-1}, and $c_i = 0$ as in Case 2a. The calculational window for c_{i-1} is a time irreversable window. There are two subcases, depending on the value of b_i.

In Case 2b1, the state $b_i = 0$. In this case, $c_i = 0$ and its window is the null window. There is energy at b_i but not at c_i, so a bit of energy of position is lost. Since the computational window also becomes unprimed at this time, the updating automaton reenters the (energy free) initial state SLPB to seek another energy-bearing left particle boundary, and a total of two bits of positional energy have been lost since the most recent left particle boundary window. The updating of the automaton then proceeds through null transitions seeking the next left particle boundary transition, if any.

In Case 2b2, the state $b_i = 1$. Again in this case, $c_i = 0$, but the calculational window for c_i is a left particle boundary window. However, unlike the initial left particle boundary window, there is no energy associated with b_i. Neither is energy gained at c_i because $c_i = c_{i-r-1} = 0$. Thus the computational window again becomes primed and there is still exactly one bit of positional energy from time t that is not yet transformed to positional energy at time $t+1$. The Primed Window Lemma completes the proof that there will be a critical transition at c_{i+r+1}. ∎

Case 2b2 is strange, unexpected, and poorly understood. It is why the definition of a particle specifies both exactly one left particle boundary transition and one right particle boundary transition. If Case 2b2 could occur in a particle, the number of left and right particle boundaries would not necessarily be equal. There is a reasonable chance that I may someday be able to prove that Case 2b2 cannot occur in a reasonable "particle," but in its current form, the proof is inelegant and probably also incomplete. The difficulty is avoided in this paper by defining particles in such a way that Case 2b2 cannot occur.

Completion of the proof of the Second Law: At every position except a byte boundary, state values and consequently energy contributions are not changed. On the byte boundaries, there are left particle boundary transitions and critical transitions of the five kinds. The Critical Transition Lemma says that combining the energy contributions of the most recent energy bearing left particle boundary transition with one of these five cases, energy is conserved in four of the cases and lost in Case 2b1. Applying this fact as many times as there are energy bearing left particle boundary transitions in the configuration, we prove that the energy of a configuration is never increased in the next time period. ∎

10. Dynamics of particles

We can now provide descriptions of the evolution of large particles similar to those given earlier for small particles. For simplicity, the particles are described in terms of their energy distribution pattern. Recall that the leftmost nonzero energy bit coincides with the leftmost nonzero configuration state, and that the rightmost nonzero energy bit lies exactly one byte (i.e., $r+1$ vertices) to the right of the rightmost nonzero configuration state.

Lemma 3 (Conservation of Energy in Particles) *The energy of a particle is the same in all phases of the particle's evolution, i.e.,* $\mathcal{E}^{t'} = \mathcal{E}^t$ *for all* $t' \geq t$.

Proof. The Second Law says that energy can never increase with time. Since the configuration of a particle reappears every p time periods with a displacement that does not affect the energy calculation, $\mathcal{E}^{t+np} = \mathcal{E}^t$. If for some $t' \geq t$ the energy of the particle decreased to $\mathcal{E}^{t'} < \mathcal{E}^t$, then for a suitable multiple of p such that $t + np \geq t'$, we have $\mathcal{E}^{t+np} = \mathcal{E}^t > \mathcal{E}^{t'}$, which contradicts the Second Law. Thus the energy of a particle does not decrease with time, so it remains constant. ∎

Theorem 6 (Period and Velocity Theorem) *If the positional energy of a particle is contained in precisely* w *contiguous bytes of size* $r+1$ *starting at the leftmost node which has nonzero state value or energy, then the particle has period* $p = \mathcal{E}^t$ *and shifted displacement* $d_s = w(r+1)$. *The unshifted displacement* $d = pr - d_s = r\mathcal{E}^t - w(r+1)$. *The energy of a particle is the same in all phases of the particle's evolution, i.e.,* $\mathcal{E}^{t'} = \mathcal{E}^t$ *for all* $t' \geq t$. *Moreover, the nonzero positions in the energy diagram of each phase of the particle's evolution are always contained in* w *contiguous bytes, but not in* $w-1$ *contiguous bytes.*

Corollary 2 (Divisor Period and Velocity Corollary) *If* n *is a divisor of* $w(r+1)$ *and if the energy diagram at time* t *consists of* n *repetitions of a subpattern, then the particle has this property at all subsequent times* $t' \geq t$, *and it has a divisor period* $\mathcal{E}^t/n$ *and a shifted displacement* $w(r+1)/n$ *for this divisor period.*

Proof. Since we are dealing with a particle and not an arbitrary configuration, critical transition Cases 2a2 and 2b2 cannot happen because they involve second left particle boundary transitions. Case 2b1 is impossible in a particle because it results in the loss of energy, which the Second Law says can never be regained, contradicting the Conservation of Energy Lemma. Thus only Cases 1 and 2a1 can happen. The following theorem completes the proof of the Period and Velocity Theorem and Corollary.

Theorem 7 (Particle Evolution Theorem) *At each time period, a particle's energy diagram transforms as follows. The leftmost energy bit (corresponding to the leftmost nonzero state value) disappears, i.e., is not present in this position at time* $t+1$. *All other positions containing positional energy at time* t *continue to contain positional energy at time* $t+1$, *and a new energy* *bit is created at time $t+1$ at a position $w(r+1)$ vertices to the right of the leftmost energy bit at time t.*

In terms of state values, the leftmost nonzero state disappears, all states not on subsequent byte boundaries remain unchanged, and all states on byte boundaries are changed, including creating one new nonzero state to the right of all previous nonzero state values in the time-shifted diagram.

Proof. At the leftmost nonzero state, energy is lost as the state flips from 1 to 0. The next r vertices have primed calculational windows and transform unchanged. Since the previous byte consists of $r+1$ zeros, all positional energy at these positions is preserved. At the second byte boundary, which is the next position, there is a critical transition. Only Cases 1 and 2a1 can happen, and Case 2a1, which carries forward a window with r paired zero states, can happen only once per time period because if there are any nonzero states to the right of a Case 2a1 critical transition, the first of these will cause another left particle boundary transition.

If the width of the energy diagram is w bytes, Case 1 critical transitions occur at the second through w^{th} byte boundaries, and a Case 2a1 critical transition occurs at the $(w+1)$st byte boundary. At positions not on the byte boundaries, all states remain unchanged, and therefore so does their energy. The Case 1 critical transitions flip state value, but leave positional energy unchanged on these byte boundaries because state values were also flipped at the previous byte boundary. Finally, at the Case 2a1 critical transition, the state value does not change, but energy is created (or redistributed) because the state value was 1 in the previous byte; it was the rightmost nonzero state value at time $t+1$. ∎

The following time-shifted diagram summarizes the state transitions of a particle.

$$\begin{array}{cccccccc}
\dots & 0 & 1 & b_1 & \dots & b_r & b_{r+1} & b_{r+2} \ \dots \\
\dots & 0 & 0 & b_1 & \dots & b_r & 1-b_{r+1} & b_{r+2} \ \dots
\end{array}$$

$$\begin{array}{cccccccc}
\dots & b_{2r+1} & b_{2(r+1)} & b_{2r+3} & \dots & b_{(w-1)(r+1)-1} & 0 & 0 \ \dots \\
\dots & b_{2r+1} & 1-b_{2(r+1)} & b_{2r+3} & \dots & b_{(w-1)(r+1)-1} & 1 & 0 \ \dots
\end{array}$$

The energy state transitions of a particle are even simpler: the leftmost energy bit at time t disappears from that position at time $t+1$ and reappears $w(r+1)$ positions to the right.

The proof of the Period and Velocity Theorem can now be completed. When the leftmost energy bit of time t moves $w(r+1)$ positions to the right, it exposes the second to leftmost nonzero energy bit as the new leftmost nonzero energy bit at time $t+1$. In the next transition, this nonzero energy bit moves $w(r+1)$ positions to the right, exposing the third to leftmost nonzero energy bit as the byte boundary for the following transition. After $\mathcal{E}^t$ time periods, each nonzero energy bit will have been the leftmost nonzero bit of a transition exactly once, and will have consequently moved right exactly $w(r+1)$ positions. The period of the particle is thus $\mathcal{E}^t$ and its shifted displacement $d_s = w(r+1)$. Since the unshifted displacement $d = pr - d_s$, its value is $r\mathcal{E}^t - w(r+1)$ and the theorem is proved. The Divisor Period and Velocity Corollary follows similarly. ∎

Theorem 8 (Consistency of Energy and Small Particle Descriptions) *The period and velocity theorems for small particles that calculate these quantities based on the number of nonzero states give the same answers as the energy based period and velocity theorems that apply to all particles.*

Proof. For small particles, $\mathcal{E}^t = 2k$ for all t, and the second energy byte of a small particle is always identical with the first energy byte. The energy diagram of a small particle therefore always has a divisor period of $\mathcal{E}^t/2 = k$, so that the energy based predictions based on this divisor period agree with the "full period" predictions based on the number of nonzero states. Divisor periods of the form k/n in the small particle model correspond exactly to divisor periods of the form $\mathcal{E}^t/2n$ in the energy model. It is easily checked that the calculated displacements and velocities agree. ∎

11. Examples of particles and their evolution

The Particle Evolution Theorem or the more general Rapid Updating Rule provide a very powerful and efficient way to calculate state transitions. For example, consider the following configuration with $r = 2$. It is a particle of energy width 4 bytes.

$$\begin{array}{ccccccccccccccc}
\bullet & \bullet & \bullet & & & \bullet & & & \bullet & \bullet & \bullet & & & \bullet & \\
1 & 1 & 1 & & 1 & 0 & 1 & & 0 & 1 & 0 & & 0 & 0 & 0
\end{array}$$

The states with dots over them carry positional energy. Thus $\mathcal{E}^t = 8$. The Period and Velocity Theorem predicts a fundamental period of 8 and displacement $d_s = w(r+1) = 12$. However, the energy diagram (but not the pattern of state values) consists of $n = 2$ repetitions of the energy subpattern 111 010, and so there is a divisor period of $8/2 = 4$ with displacement $12/2 = 6$. The transitions are calculated as follows, with $\boxed{0}$ indicating a state that changed to zero at a left particle boundary transition and underscored state values indicating that their states flipped due to critical transitions of type 1 on the byte boundaries.

$$\begin{array}{cccccccccccccccccc}
\bullet & \bullet & \bullet & & \bullet & & \bullet & \bullet & \bullet & & \bullet & & \bullet & \bullet & \bullet & & \bullet & \\
1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & & & & & & \\
\boxed{0} & 1 & 1 & \underline{0} & 0 & 1 & \underline{1} & 1 & 0 & \underline{1} & 0 & 0 & 0 & & & & & \\
 & \boxed{0} & 1 & 0 & \underline{1} & 1 & 1 & \underline{0} & 0 & 1 & \underline{1} & 0 & 0 & 0 & & & & \\
 & & \boxed{0} & 0 & 1 & \underline{0} & 1 & 0 & \underline{1} & 1 & 1 & \underline{1} & 0 & 0 & 0 & & & \\
 & & & & \boxed{0} & 0 & 1 & \underline{1} & 1 & 1 & \underline{0} & 1 & 0 & \underline{1} & 0 & 0 & 0 &
\end{array}$$

The transition diagram for energy is even simpler. In each time period, the leftmost nonzero energy bit jumps four bytes to the right and all other energy bits remain the same. It is obvious from the energy diagram that the period $p = 4$ shown is a divisor period and not the full predicted period because only half the nonzero energy bits of the configuration have rotated in four time periods.

1	1	1	0	1	0	1	1	1	0	1	0					
	1	1	0	1	0	1	1	1	0	1	0	1				
		1	0	1	0	1	1	1	0	1	0	1	1			
			0	1	0	1	1	1	0	1	0	1	1	1		
					0	1	1	1	0	1	0	1	1	1	0	1

12. Byte descriptions of particle evolution

In the examples of section 11, it was visually convenient to group together bytes of $r+1$ consecutive state or energy values and to put an extra separation between bytes for readability of the diagrams. In this section, we describe the evolution of energy and configuration in terms of operations on entire bytes.

Definition 20. *We say that $A_1, A_2, \ldots, A_n$ is a* byte decomposition *of a particle if each A_i is a byte of $r+1$ adjacent state values, if the position in the configuration of the rightmost bit of each A_i is adjacent to and to the left of the position of the leftmost bit of A_{i+1}, and if all the nonzero state values of the particle are contained in the bytes $A_1, A_2, \ldots, A_n$.*

Definition 21. *A byte decomposition is called* a canonical byte decomposition *if the leftmost bit of A_1 is nonzero and at least one bit of A_n is nonzero. These conditions are not in general required of a byte decomposition of a particle. The number of bytes n in a canonical byte decomposition is called the* configuration byte width.

Definition 22. *Similarly, we say that $E_1, E_2, \ldots, E_w$ is a* byte decomposition of the energy of a particle *if each E_i is a byte of $r+1$ adjacent energy values, if the position in the energy diagram of the rightmost bit of each E_i is adjacent to and to the left of the position of the leftmost bit of E_{i+1}, and if all the nonzero energy values of the particle are contained in the bytes $E_1, E_2, \ldots, E_w$.*

Definition 23. *A byte decomposition of the energy of particle is* canonical *if the leftmost bit of E_1 is nonzero and at least one bit of E_w is nonzero. The number of bytes w in a canonical byte decomposition of the energy of a particle is called the* energy byte width.

Definition 24. *We define the* bitwise sum modulo 2 *of two bytes $A \oplus B$ to be the byte that results from adding the corresponding bits of A and B modulo 2. If A^j denotes the j^{th} bit of A, then $(A \oplus B)^j = (A^j + B^j)$* mod 2.

Definition 25. *We denote the number of nonzero bits in a byte A by $N(A)$.*

Theorem 9 (Byte Description of Energy Evolution of a Particle I) *If w is the energy byte width of a particle and $E_1, E_2, \ldots, E_w$ is a byte decomposition of the particle, then the following diagram gives a byte decomposition description of the evolution of the particle.*

$$\begin{array}{cccccccccc}
E_1 & E_2 & E_3 & \ldots & E_w & & & & & \\
 & E_2 & E_3 & \ldots & E_w & E_1 & & & & \\
 & & E_3 & \ldots & E_w & E_1 & E_2 & & & \\
 & & & \ddots & \vdots & \vdots & \vdots & \ddots & & \\
 & & & & E_w & E_1 & E_2 & \ldots & E_{w-1} & \\
 & & & & & E_1 & E_2 & \ldots & E_{w-1} & E_w
\end{array}$$

The transition from the i^{th} to the $i+1$st lines in this diagram takes $N(E_i)$ time periods.

Proof. This theorem is really a restatement of the Particle Evolution Theorem in the new terminology. If one looks at the bitwise evolution of the energy of a particle after $N(E_1)$ time periods, after $N(E_1) + N(E_2)$ time periods, after $N(E_1) + N(E_2) + N(E_3)$ time periods, etc., one finds precisely the lines of the bytewise energy evolution diagram. ∎

Note that this theorem does not prove that the leftmost bits of E_2, E_3, $\ldots$, E_w are nonzero, nor could such a statement be proved because it is not necessarily true, even if the initial energy byte decomposition is canonical.

Theorem 10 (Byte Description of Energy Evolution of a Particle II) *If w is the energy byte width of a particle and $E_1, E_2, \ldots, E_w, E_{w+1}$ is a byte decomposition of the particle, then the following diagram gives a byte decomposition description of the evolution of the particle.*

$$\begin{array}{ccccccccccc}
E_1 & E_2 & E_3 & \ldots & E_w & E_{w+1} & & & & & \\
 & E_2 & E_3 & \ldots & E_w & E_{w+1}\oplus E_1 & & & & & \\
 & & E_3 & \ldots & E_w & E_{w+1}\oplus E_1 & E_2 & & & & \\
 & & & \ddots & \vdots & \vdots & \vdots & \ddots & & & \\
 & & & & E_w & E_{w+1}\oplus E_1 & E_2 & \ldots & E_{w-1} & & \\
 & & & & & E_{w+1}\oplus E_1 & E_2 & \ldots & E_{w-1} & E_w & \\
 & & & & & E_1 & E_2 & \ldots & E_{w-1} & E_w & E_{w+1}
\end{array}$$

The transition from the i^{th} to the $i+1$st lines in this diagram takes $N(E_i)$ time periods.

Proof. This is also a restatement of the Energy Evolution Theorem, but with byte boundaries falling so that the energy byte width appears to be one byte larger than it actually is. Note that in this case, E_{w+1} and E_1 have their nonzero bits in disjoint parts of the byte. ∎

Theorem 11 (Byte Description of Particle Evolution Theorem I) *If n is the configuration byte width of a particle and $A_1, A_2, \ldots, A_n$ is a byte* *decomposition of the particle, then the following diagram gives a byte decomposition description of the evolution of the particle configuration.*

$$\begin{array}{ccccccccccc}
A_1 & A_2 & A_3 & \dots & A_n & & & & & & \\
 & A_1 \oplus A_2 & A_1 \oplus A_3 & \dots & A_1 \oplus A_n & A_1 & & & & & \\
 & & A_2 \oplus A_3 & \dots & A_2 \oplus A_n & A_2 & A_1 \oplus A_2 & & & & \\
 & & & \ddots & \vdots & \vdots & \vdots & \ddots & & & \\
 & & & & A_{n-1} \oplus A_n & A_{n-1} & A_1 \oplus A_{n-1} & \dots & A_{n-2} \oplus A_{n-1} & & \\
 & & & & & A_n & A_1 \oplus A_n & \dots & A_{n-2} \oplus A_n & A_{n-1} \oplus A_n & \\
 & & & & & & A_1 & \dots & A_{n-2} & A_{n-1} & A_n
\end{array}$$

Proof. Referring to the byte description of energy evolution in theorem 9, we see that $w = n+1$ and $E_i = A_i \oplus A_{i-1}$ for $2 \le i \le n$. We can extend this equation to E_1 and E_w by defining $A_0 = A_{n+1} = 0$, the zero byte. We now reconstruct the sequence of configurations of the particle from the sequence of energy diagrams given by theorem 9. In general, the j^{th} configuration byte B_j of a byte decomposition is given in terms of the corresponding energy byte decomposition $\{ E_j(B) \}$ by the formula

$$B_j = \sum_{k=1}^{j} E_k(B).$$

All lines of configuration byte description in the current theorem are proved by applying this formula to the energy byte description and cancelling terms modulo 2. For example, the j^{th} byte on the first line is calculated as $E_1 \oplus E_2 \oplus \cdots \oplus E_j = A_1 \oplus (A_1 \oplus A_2) \oplus \cdots \oplus (A_{j-1} \oplus A_j) = A_j$. The configuration byte under this on the second line is calculated as $E_2 \oplus E_3 \oplus \cdots \oplus E_j = (A_1 \oplus A_2) \oplus (A_2 \oplus A_3) \oplus \cdots \oplus (A_{j-1} \oplus A_j) = A_1 \oplus A_j$. All others are calculated similarly. ∎

Theorem 12 (Byte Description of Particle Evolution Theorem II) *If n is the configuration byte width of a particle and $A_1, A_2, \dots, A_n, A_{n+1}$ is a byte decomposition of the particle, then the following diagram gives a byte decomposition description of the evolution of the particle configuration.*

$$\begin{array}{cccccccccccc}
A_1 & A_2 & A_3 & \dots & A_n & A_{n+1} & & & & & & \\
 & A_1 \oplus A_2 & A_1 \oplus A_3 & \dots & A_1 \oplus A_n & A_1 \oplus A_{n+1} & & & & & & \\
 & & A_2 \oplus A_3 & \dots & A_2 \oplus A_n & A_2 \oplus A_{n+1} & A_1 \oplus A_2 & & & & & \\
 & & & \ddots & \vdots & \vdots & \vdots & \ddots & & & & \\
 & & & & A_{n-1} \oplus A_n & A_{n-1} \oplus A_{n+1} & A_1 \oplus A_{n-1} & \dots & A_{n-2} \oplus A_{n-1} & & & \\
 & & & & & A_n \oplus A_{n+1} & A_1 \oplus A_n & \dots & A_{n-2} \oplus A_n & A_{n-1} \oplus A_n & & \\
 & & & & & & A_1 \oplus A_{n+1} & \dots & A_{n-2} \oplus A_{n+1} & A_{n-1} \oplus A_{n+1} & A_n \oplus A_{n+1} & \\
 & & & & & & A_1 & \dots & A_{n-2} & A_{n-1} & A_n & A_{n+1}
\end{array}$$

Proof. This byte description of configuration evolution follows from theorem 10 which applies to a byte decomposition with byte boundaries that increase the apparent byte width of the particle. The only extra observations necessary for the proof are that $A_1 = E_1$ and that $A_{n+1} = E_{n+2} = E_{w+1}$. ∎

13. Comparisons with empirical frequencies

In [1], Park, Steiglitz, and Thurston report on computer simulations to find all particles with bit width ≤ 16 for window radii $r = 2, 3, 4, 5$, and 6. Their tables of frequencies of periods and displacements show very large peaks at certain periods and displacements. Since the Period and Velocity Theorem shows that these quantities depend only on the byte width and energy of a particle, but not on other details of the particle's configuration, all these peaks may now be explained in terms of the large number of particles that share a common byte width and energy that produces these periods and displacements.

For example, all of the most frequently observed period-displacement pairs in [1] for $r = 4$, $r = 5$, and $r = 6$ are characteristic of particles of configuration byte width 3 and "average" energy for their size. Since the size of a byte is 5, 6, or 7 in these cases and the search was conducted using configurations of bit width at most 16, we find very few particles of configuration byte width 4 in their tables for $r = 4$, and none whatsoever for $r = 5$ and $r = 6$.

However, there is ample room within 16 bits to have sampled all of the particles of byte width 3 for $r = 4$ and $r = 5$, and nearly all of them for $r = 6$. Their periods are given by their energy and are thus are even integers, as predicted. For all three radii, the maximum observed frequency occurs for period $p = \mathcal{E} = 2(r+1)$, an energy most readily obtained from particles with two energy bits in each relative position in the byte. Since for such particles, the leftmost energy bit of the particle can be paired with a corresponding energy bit in any of the three remaining energy bytes, and the remaining r relative positions in a byte can have their nonzero energy bits chosen in $\binom{4}{2}$ ways, and each such energy configuration will appear in $\mathcal{E}$ different phases during a period, a rough estimate of the number of such particles is $3\binom{4}{2}^r/\mathcal{E} = 3 \cdot 6^r/2(r+1)$, which approximates the observed frequencies. (For $r = 6$, a substantial fraction of these three-byte particles have bit width more than 16 and would not be included in the tables of [1].)

Accompanying each collection of particles of configuration byte width 3 are particles of divisor period 2 having half the period and displacement. These are necessarily far fewer in number because their energy pattern must consist of two identical halves.

For $r = 3$, the maximum observed frequencies correspond to particles of byte width 4 which fall within the 16 bit configurations sampled. Most frequent periods and therefore energies are 10 and 12, approximately half the positions in the energy width of 20 possible positions carrying nonzero energy. Secondary peaks in frequency correspond to particles of byte width 3. Particles of byte width 5 are not represented, as expected.

14. Impossible particles

Not all state configurations can evolve as particles. The purpose of this section is to show that there are a sufficient number of classes of forbidden configurations for particles to prove that the total number of distinct particles supported by the parity filter automaton with window radius r is finite.

It is shown in [2] that the number of particles with fixed period p is bounded, but this is a much stronger result, that there are upper bounds (although necessarily large ones) on the total number of particles and on the maximum period of a particle.

Lemma 4. *The energy of a particle cannot have $r+1$ consecutive positions of energy zero between its leftmost and rightmost nonzero energy positions.*

Proof. Consider that stage of the energy rotation of the particle when the nonzero energy bit to the left of the $r+1$ zeros is the leftmost energy bit of the particle. When this bit rotates to the right, the energy byte width of the particle decreases by one because the new leftmost nonzero energy position is more than $r+1$ positions to the right of the previous one. This reduction in width cannot be recovered during later evolution of the configuration, so the starting energy pattern of the particle never reappears, a contradiction. Thus there cannot be $r+1$ consecutive zero states in the middle of the energy pattern of a particle. ∎

Lemma 5. *Two consecutive nonzero bytes A_i and A_{i+1} of the byte decomposition of a particle cannot be equal.*

Proof. If $A_i = A_{i+1}$ then $E_{i+1} = 0$, which is forbidden by lemma 4. ∎

Lemma 6. *No two bytes A_i and A_j of a byte decomposition of a particle can be equal, nor can any interior byte be zero.*

Proof. Assume for purposes of contradiction that the configuration is a particle and $A_i = A_j$. If $j = i+1$ or $i = j+1$, lemma 5 gives the conclusion. Assume $j \neq i+1$ and $i \neq j+1$. Define $A_0 = 0$, the zero byte, so that a typical byte A_j in the top row in the byte description of the particle's evolution can be written as $A_0 \oplus A_j$, and a typical byte of the middle column can be written as $A_j \oplus A_0$. Now every possible bitsum of bytes, $0 \leq i, j \leq n$, appears twice in the diagram, once as $A_i \oplus A_j$ and once as $A_j \oplus A_i$.

If the leading byte $A_i \oplus A_{i+1}$ of the row with $A_i \oplus A_j$ has its leftmost bit nonzero, this row represents a canonical byte decomposition of the particle at this time, and there will be a right particle boundary when $A_i \oplus A_j$ is reached by the rapid updating automaton. Since $j \neq i+1$ and $i \neq j+1$, this particle boundary is in the middle of the particle, a contradiction. Similarly, $A_j \oplus A_{j+1}$ cannot have its leftmost bit nonzero. Thus the leftmost bits of A_i, A_{i+1}, A_j, and A_{j+1} are equal.

We now do an induction proof, assuming that for all bits to the left of relative position k, the corresponding bits of A_i, A_{i+1}, A_j, and A_{j+1} are equal.

If relative bit k of $A_i \oplus A_{i+1}$ is nonzero, the rapid updating automaton will encounter $r+1$ consecutive zero states after bit k of $A_i \oplus A_j$, some of them at the end of the byte $A_i \oplus A_j$ and the rest of them at the start of the byte $A_i \oplus A_{j+1}$. (Recall that the leftmost bits of A_{j+1} are equal to the corresponding bits of A_j by the induction hypothesis.) These consecutive zero states starting at relative position k which is being used as the byte boundary by the rapid updating automaton would again cause a right particle boundary in the middle of the particle, a contradiction, so relative bit k of $A_i \oplus A_{i+1}$ is zero. Similarly, the k^{th} bit of $A_j \oplus A_{j+1}$ is zero. Thus the k^{th} bits of A_i, A_{i+1}, A_j, and A_{j+1} are equal, and the induction is complete.

As a result, the complete bytes $A_i = A_{i+1} = A_j = A_{j+1}$, and lemma 5 shows that the configuration cannot be a particle. ∎

Theorem 13 (Finite Number of Particles Theorem) *For fixed parity filter automaton with window radius r, the number of distinct particles is finite.*

Proof. Since no interior byte of a particle can be zero and no two nonzero bytes can be equal, the byte width n of a particle is at most $2^{r+1}-1$ bytes. Since all bytes must be distinct, there are at most $n!$ particles. ∎

15. Particles of period 1

Particles with period 1 are a special class of particles that can be described completely. Since the period is 1, the velocity v and shifted velocity v_s are equal respectively to the displacement d and shifted displacement d_s. These particles range in speed from the fastest possible particle, the so-called "photon," with velocity $r-1$ to the slowest possible particle, the *stationary particle*, with velocity zero. The most important result about particles of period 1 is that for each velocity in this range, there exists one and only particle with that velocity. In other words, particles of period 1 are uniquely determined by their velocity.

Theorem 14 (Existence and Uniqueness of Particles of Period 1.) *For any velocity v with $0 \le v \le r-1$ there exists one and only one particle with period 1 and velocity v.*

Proof. The proof proceeds by constructing the energy pattern of a particle of period 1 and velocity v. At each step, the choice is forced, so there is at most one particle with velocity v. Then it is shown that the energy pattern so constructed corresponds to a particle, which establishes the existence part of the result.

If there is a particle with period 1 and velocity v, its shifted displacement $d_s = r - d = r - v$. Thus d_s lies in the range $1 \le d_s \le r$. Since the energy configuration of a particles evolves by moving the leftmost nonzero bit w bytes to the right while all other energy bits remain fixed, the second nonzero energy bit (from the left) must be d_s nodes to the right of the first

10010 01001 00100 10010 01001 00100
00010 01001 00100 10010 01001 00100 10000

Figure 6: Energy pattern for a particle of period 1 with $r = 4$, $v = 2$, and $d_s = 3$.

one. Since the period is 1, the energy pattern at time $t+1$ must be the same as that at time t displaced to the right by d_s nodes. Thus the third nonzero energy bit, if any, must be d_s nodes to the right of the second nonzero energy bit, and so forth. Figure 6 shows such an energy pattern for $r = 4$, $v = 2$, and $d_s = 3$.

Since each nonzero energy bit represents a difference of configuration state value between one node and the node $r + 1$ positions to the left of it, and since all nodes sufficiently far to the left and to the right are in state zero, the energy pattern of a configuration must have an even number of ones in each relative position within a byte. The repetitive energy pattern of a particle of period 1 starts with a 1 in the leftmost position of a byte and does not again have a 1 in the same relative position in a byte until $\mathrm{lcm}(d_s, r+1)$ nodes to the right. Thereafter, the pattern repeats in the same relative positions within bytes, so that after $2\,\mathrm{lcm}(d_s, r + 1)$ nodes, the contributions to the energy are even in each relative position within a byte for the first time. In fact, each relative position has either 0 or 2 nonzero energy bits. All that remains to be shown is that this stopping place in the energy pattern corresponds to a particle and that continuing the pattern further results in a configuration that has at least two left particle boundaries and thus cannot be a single particle.

The configuration corresponding to this energy pattern adds new nonzero states each byte throughout the first half of the energy pattern because each new nonzero energy bit is in a different relative position within the byte. In byte $\mathrm{lcm}(d_s, r + 1)/(r + 1)$, every possible relative position of the form $kd_s \bmod (r+1)$ is nonzero. Thereafter, configuration bits become zero in the same order they became 1, so that byte $2\,\mathrm{lcm}(d_s, r+1)/(r+1)$ is all zero, and the previous byte has at least one nonzero state not in the leftmost position of the byte (See figure 7). This means that the PW (primed window) state in the Rapid Updating Automaton does not copy r consecutive zeros until the rightmost energy byte we have considered, which corresponds to the all zero configuration byte. Thereafter, the automaton enters state SLPB (seeking left particle boundary). If there were another nonzero energy bit to the right of the $2\,\mathrm{lcm}(d_s, r + 1)$ bits we considered, there would be another nonzero configuration state and consequently another left particle boundary transition. Thus stopping the energy pattern at $2\,\mathrm{lcm}(d_s, r+1)$ positions gives a particle and continuing further cannot produce a single particle because of the extra left particle boundary. The repetition of the energy pattern every d_s positions guarantees that this is the particle desired. ∎

10010 11011 11111 01101 00100 00000

10010 11011 11111 01101 00100 00000
00010 $\underline{0}$1011 $\underline{0}$1111 $\underline{1}$1101 $\underline{1}$0100 $\underline{1}$0000 00000

Figure 7: Configuration pattern corresponding to the energy pattern in figure 6 and its evolution.

As figure 7 shows, the configuration pattern of a particle of period 1 displays almost none of the regularity of its energy pattern. To calculate the exact width w of the configuration from its leftmost nonzero state to its rightmost nonzero state, we start with the $2\,\mathrm{lcm}(d_s, r+1)$ nodes it takes before the energy pattern lands on the byte boundary for the third time, subtract the last full byte of zero configuration states and the $d_s - 1$ final zero states in the previous byte. The resulting formula is

$$\begin{aligned} w &= 2\,\mathrm{lcm}(d_s, r+1) - (r+1) - (d_s - 1) \\ &= 2\,\mathrm{lcm}(d_s, r+1) - r - d_s \end{aligned}$$

The constant displacement between successive nonzero bits in the energy pattern means that if the pattern is reversed, i.e., written with left and right interchanged and leftmost nonzero bits aligned, it remains the same. We say that such a pattern is *symmetric*.

The next theorem says that symmetric energy patterns correspond to symmetric particle configurations.

Theorem 15 (Symmetric Configuration Theorem) *A configuration is symmetric if and only if its energy pattern is symmetric.*

Proof. Start by pairing the leftmost nonzero energy bit (which coincides with the leftmost nonzero configuration bit) with the energy bit $r+1$ positions to the right of the rightmost nonzero configuration bit. These energy bits are both nonzero. Move the pairing one position at a time toward the center of the configuration. At each stage, the two configuration bits involved in the calculation of the energy bit are symmetrically placed with respect to the center of the configuration. Thus the paired energy bits are all equal if and only if all the symmetrically placed configuration bits are equal.

Formally, if $b_i^t = a_{-i}^t$ is the reverse of configuration a, and $e_i^t(a) = \left|a_i^t - a_{i-r-1}^t\right|$ is its energy pattern, then the energy pattern of b is

$$e_{-i+r+1}^t(b) = \left|b_{-i+r+1}^t - b_{-i}^t\right| = \left|a_{i-r-1}^t - a_i^t\right| = e_i^t(a)$$

Thus, except for a shift, the energy pattern of b is the reverse of the energy pattern for a. ∎

Corollary 3 (Symmetry of Particles of Period 1) *Any particle of period 1 has a symmetric configuration.*

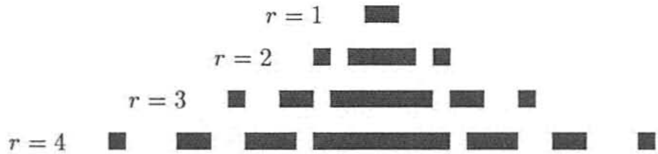

Figure 8: Stationary particles for $r = 1, 2, 3$, and 4.

Stationary particles

At one extreme of the particles of period 1 are the "photons," the particles of highest velocity, and at the other extreme are the *stationary particles*, those of velocity zero. Steiglitz discovered these particles and derived their configuration pattern for any window radius r. He also observed that they are symmetric configurations, a property now known to be true for all particles of period 1.

Figure 8 shows the configurations of stationary particles for window radii 1 to 4. For such particles, $d = 0$, $d_s = r$, and the configuration width

$$\begin{aligned} w &= 2\,\mathrm{lcm}(d_s, r+1) - r - d_s \\ &= 2r(r+1) - r - r \\ &= 2r^2 \end{aligned}$$

16. Uniqueness of particles

The method used in the previous section to find the energy pattern of a particle of period 1 can be used to find the energy pattern of any particle for which the sequence of (shifted) displacements $d_{s1}, d_{s2}, \ldots, d_{sp}$ are given. Starting at the leftmost nonzero energy bit, the second nonzero energy bit must be d_{s1} positions to the right, the third nonzero energy bit d_{s2} positions to the right, and so forth until the $p+1$st nonzero energy bit is placed d_{sp} positions to the right of the previous one. Since a particle of period p recurs in its original configuration after p time periods, subsequent displacements repeat the sequence $d_{s1}, d_{s2}, \ldots, d_{sp}$. If there is a particle with this sequence of shifted displacements, its energy pattern must be some initial subsequence of the pattern we generated, terminating when the associated configuration pattern has its first full byte of zeros to produce a right particle boundary transition. Thus for each sequence of displacements, there is one and only one candidate configuration for a particle with the specified period and displacements, which proves the following theorem.

Theorem 16 (Uniqueness of Particles) *For any period p and sequence of shifted displacements $d_{s1}, d_{s2}, \ldots, d_{sp}$, there is at most one particle of this period with these displacements.*

Although there is only one candidate for a particle, there is no guarantee that the candidate energy pattern will indeed correspond to a particle.

The questions of which configurations cannot be particles, which energy sequences cannot represent particles, and which sequences of displacements cannot form the fine structure of the velocity pattern of a particle are thus seen to be equivalent. Some partial results are given below, but a complete characterization of these impossible configurations remains an open question.

17. Mirror image particles and reversability of time

If you take any particle evolution diagram and turn it upside down by rotating it 180°, the diagram still looks like a particle evolution diagram. In fact it is the evolution diagram of a particle, and the particle is the *mirror image* of the original particle.

Definition 26. *If a_i is a state configuration of the cellular automaton, then its* mirror image *is the configuration $b_i = a_{-i}$.*

Theorem 17 (Mirror Image Theorem) *The mirror image of a particle is a particle with the same period, energy, shifted and unshifted displacements, and its particle evolution diagram is the 180° rotation of the evolution diagram of the original particle.*

Proof. Let a_i^t be the shifted evolution diagram for the particle a_i^0, which is defined for $t \geq 0$. Let $p = \mathcal{E}(e_i^0)$ and d_s be its time-shifted displacement. Then $e_i^t = e_{i+d_s n}^{t+pn}$ for all positive integers n. If $t < 0$ then there is some n big enough that $t + pn > 0$. Define $e_i^t = e_{i+d_s n}^{t+pn}$ for negative t. With this definition, the diagram represents the evolution of the particle a_i^0 from time $t = -\infty$ to $t = \infty$.

Let $b_i^t = a_{-i}^{-t}$. We claim that b_i^t is the evolution diagram for the particle $b_i^0 = a_{-i}^0$, the mirror image particle. The conclusions on equality of period, energy, and displacements follow easily from this stronger claim.

We examine what the rapid updating automaton does in the transition of b from time t to time $t+1$. First, it copies zeros in the SLPB state until it finds the leftmost nonzero b_i^t, from which it sets $b_i^{t+1} = 0$. This is the left particle boundary of b and the right particle boundary of a where the zero state a_{-i}^{-t-1} changes to $a_{-i}^t = 1$. The updating automaton then copies r consecutive states $b_i^{t+1} = b_i^t$, at least one of which must be nonzero since at least one of the corresponding $a_{-i}^{-t} = a_{-i}^{-t-1}$ was nonzero to sustain the chain of critical transitions of Type 1 to the right particle boundary of a. The next state of b is inverted, i.e., $b_i^{t+1} = 1 - b_i^t$, which matches the critical transition in a of $a_{-i}^{-t} = 1 - a_{-i}^{-t-1}$. If n_a is the configuration byte width, then this process of copying r states and inverting one state continues for n_a times in both b and a until the left particle boundary $a_{-i}^{-t} = 0$ and $a_{-i}^{-t-1} = 1$ is reached. This corresponds to a critical transition $b_i^{t+1} = 1$ and $b_i^t = 0$. Thereafter, the rapid updating automaton operating on b encounters nothing but zero states because there were nothing but zero states to the right of the left particle boundary a_{-i}^{-t-1} of a. Thus the time-reversed mirror image of the updating of a particle is the correct transition of the mirror image configuration, which is therefore a particle. ∎

18. Collisions of particles

Many large configurations split naturally into smaller subconfigurations that evolve, at least for a time, independently of each other. If each of the subconfigurations is a particle, well separated from the other subconfigurations, we may think of the large configuration as a collection of particles, each moving with its own characteristic velocity and rotation until it comes sufficiently close to another particle to interact with it.

One of the most interesting phenomena that Steiglitz [1] found in parity filter automata is that a large fraction of such particle collisions are "soliton" collisions from which both particles emerge with their original identities intact. There is, however, for each particle a displacement from its original path due to the collision, and a *phase shift* consisting of a modification of the times that different configurations within the period of the particles appear after the collision as compared to when they would have appeared if the two particles had never interacted. This phase shift in soliton collisions is exploited in [2] to build a carry ripple adder.

This section makes precise definitions of what it means for a configuration to split into subconfigurations called *connected components*, what it means for these subconfigurations to be particles, and what it means for two particles to collide. The most important theorem, The Soliton Collision Theorem, calculates the number of time periods duration of a collision of two particles, and the phase shift and displacement of each particle after the collision.

Definition 27. *A configuration is decomposed into its* connected components *in the following way. Starting at the leftmost nonzero state, the configuration is partitioned into* bytes *of $r+1$ adjacent nodes. A byte is called a* zero byte *if all its states are zero, and a* splitting byte *if its rightmost r states are zero irrespective of the state of the leftmost node, which is called the* byte boundary. *The leftmost* connected component *of a configuration consists of all nodes from the leftmost nonzero node of the configuration to the right node of the first (i.e., leftmost) splitting byte.*

By this definition, a connected component always ends with at least r zero states and is an integral number of bytes wide. In fact, a connected component can have as much as $2r$ consecutive zero states at its right end.

The second connected component is obtained by realigning byte boundaries on the first nonzero state to the right of the splitting byte of the first (leftmost) connected component. It continues from this nonzero node until the rightmost node of the next splitting byte. The byte boundaries for the second connected component need not agree with the division points one would get by continuing the byte boundaries of the first connected component beyond its splitting byte.

All subsequent connected components are obtained by the same method as the second. Byte boundaries are realigned at the leftmost nonzero state to the right of the complete splitting byte of the previous connected component, and the connected component extends from there until the right end of the next splitting byte using the new byte boundaries.

Definition 28. *The smallest possible connected component of a configuration consists a single splitting byte (i.e., a single nonzero state followed by r zero states). It is called* a moribund component *because it "dies" or disappears completely in the next time period.*

Definition 29. *A configuration is called* connected *if it has exactly one connected component.*

Definition 30. *A component of a configuration is called* a constituent particle, *or simply* a particle *if it would evolve as a particle when all states of all other components are set to zero.*

Comparing the definition of connected components of a configuration with the Critical Transition Lemma, we see that critical transitions take place on the byte boundaries used to determine connected components, and that there is a Case 2 critical transition if and only if the previous byte is a splitting byte.

Theorem 18. *A particle is a connected configuration.*

Proof. It was shown earlier that particles can have only Case 1 and Case 2a1 critical transitions, with the only Case 2a1 critical transition occurring at the right particle boundary. Thus the only splitting byte of a particle is at its right end, and a particle configuration has a single connected component and is therefore connected. ∎

Connected components are sufficiently separated from each other that they evolve for at least one time period independently of each other. A splitting byte is precisely where the Rapid Updating Rule automaton leaves the PW (primed window) state and reenters the SLPB state to seek a new left particle boundary transition.

We can trace the evolution of a component of a configuration in two different senses; first as a component of the evolving configuration, and second as the component would evolve independently if all states of all other components were set to zero.

Definition 31. *We say that a connected component α of a configuration at time t evolves into a connected component α' of the configuration at time $t+1$ if α' has nonzero states at precisely those nodes where α would have generated nonzero states if allowed to evolve with all states of all other components at time t set to zero. We may then use the same name to refer to the connected component in both time periods, calling $\alpha = \alpha(t)$ and $\alpha' = \alpha(t+1)$.*

Definition 32. *We say that two nonmoribund connected components α and β* collide *at time t if $\alpha(t-1)$ and $\beta(t-1)$ are (necessarily adjacent) connected components of the configuration at time $t-1$, but their independent evolution configurations $\alpha(t)$ and $\beta(t)$ are not separate connected components of (and in fact are not entirely present in) the configuration at time t.*

For the remainder of this section, α and β will be particles colliding at time t. Since α and β were not in collision at time $t-1$, there is a full null splitting byte of $r+1$ zero states at the right of $\alpha(t-1)$ plus a possible gap of $g \geq 0$ additional zero states before the leftmost nonzero state of $\beta(t-1)$. Using the Rapid Updating Rule to calculate states at time t, we see that the leading 1 of $\alpha(t-1)$ moves to the leftmost bit of the splitting byte for $\alpha(t-1)$, but the remaining r states of this byte remain zero. Since the byte boundary for β moves $d_s(\beta, t-1) > 0$ positions to the right, there are exactly $r + g + d_s(\beta, t-1) \geq r + 1$ zero states between α and β at time t.

Let $A_1, A_2, \ldots, A_{n_\alpha}$ be the canonical byte decomposition of $\alpha(t)$ and $B_1, B_2, \ldots, B_{n_\beta}$ be the canonical byte decomposition of $\beta(t)$. Label the individual states of $\alpha(t)$ from a_0 at the left of A_1 to $a_{n_\alpha(r+1)-1}$ at the right of A_{n_α}. Similarly label the states of $\beta(t)$ from b_0 to $b_{n_\beta(r+1)-1}$.

The byte boundaries for $\alpha(t)$ are $d_s(\alpha, t-1)$ further to the right than those of $\alpha(t-1)$, leaving only $r + g + d_s(\beta, t-1) - d_s(\alpha, t-1)$ zero states after the byte boundary in what would have been the splitting byte of $\alpha(t)$. Since the particles are in collision at time t, this number is less than the r zero states required for a splitting byte, so that $d_s(\alpha, t-1) > d_s(\beta, t-1) + g$.

Definition 33. *The quantity $k = d_s(\alpha, t-1) - d_s(\beta, t-1) - g > 0$, is called the* collision offset *of two colliding particles.*

The quantity k is called the *collision offset* for the following reason. At time $t-1$, the leftmost nonzero bits of $\alpha(t-1)$ and $\beta(t-1)$ are $(n_\alpha+1)(r+1)+g$ positions apart. At time t, they are only $(n_\alpha + 1)(r + 1) + g - d_s(\alpha, t-1) + d_s(\beta, t-1) = (n_\alpha+1)(r+1) - k$ positions apart so that when $\alpha(t)$ is updated, there are critical transitions on the byte boundaries of $A_2, A_3, \ldots, A_{n_\alpha}$, and the next critical transition is at b_k, the position which is offset k to the right of the byte boundary of B_1. Thus byte boundaries of $\alpha(t)$ are offset by k positions to the right with respect to those of $\beta(t)$.

We need one more technical condition on the intersection of two particles α and β before we can state the Soliton Collision Theorem. The leftmost nonzero state of $\alpha(t)$ is a_0 and the rightmost nonzero state is a_{m_α}, where $m_\alpha = (n_\alpha - 1)(r+1) - d_s(\alpha, t-1)$. Similarly, the leftmost and rightmost nonzero states of $\beta(t)$ are b_0 and b_{m_β} where $m_\beta = (n_\beta - 1)(r+1) - d_s(\beta, t-1)$. We define $a_i = 0$ if $i < 0$ or $i > m_\alpha$ and define $b_j = 0$ if $j < 0$ or $j > m_\beta$.

Condition ICC (Intersection Compatibility Condition) Whenever a_i carries energy in $\alpha(t)$, that is, whenever $a_i \neq a_{i-r-1}$, and whenever for that i the integer n makes the intersection of subscript intervals $[\,k+i+n(r+1)-r, \ldots, k+i+n(r+1)-1\,] \cap [\,0, \ldots, m_\beta\,]$ nonempty, there is at least one j with $1 \leq j \leq r$ for which $a_{i-j} \neq b_{k+i+n(r+1)-j}$. In other words, the r configuration states of $\alpha(t)$ to the left of a_i do not exactly match the r configuration states of $\beta(t)$ to the left of $b_{k+i+n(r+1)}$.

We now have the terminology to state the main theorem of the section, the Soliton Collision Theorem, describing precisely what happens when two particles collide without loss of energy.

Theorem 19 (Soliton Collision Theorem) *If a configuration consists of two constituent particles $\alpha(t-1)$ and $\beta(t-1)$ at time $t-1$ with α to the left of β, and if these constituent particles collide at time t, and if the intersection compatibility condition, Condition ICC, holds, then*

1. *α and β will remain in collision for exactly $\mathcal{E}(\alpha)$ time periods.*
2. *During the collision, the nonzero energy bits of α will continue to rotate normally, except that when they reappear at the right they appear exactly $(n_\beta + 1)(r+1)$ nodes to the right of where they would have appeared if α had evolved independently. (Recall that n_β is the number of bytes in the canonical byte decomposition of β, and $n_\beta + 1$ is the number of bytes in the canonical byte decomposition of the energy diagram of β.)*
3. *During the collision, the energy bits of β will remain fixed in the shifted diagram and move left at the speed of light in the unshifted energy diagram.*
4. *After the $\mathcal{E}(\alpha)$ time periods of the collision, the particles α and β will evolve as separate constitutent components of the configuration for at least one time period. The particle β will be to the left of α.*

Figures 9, 10, and 11 show respectively the unshifted and time-shifted configuration evolutions, and the time-shifted energy evolution of a collision of the same two constituent particles with $r = 4$. All three figures show both particles emerging intact from the collision with their relative positions reversed, i.e., it is a *soliton* collision. However both shifted and unshifted configuration diagrams are confusing concerning the locations of the two particles *during* the collision. On the other hand, the energy evolution diagram clearly shows the right particle β stopping its rotation and remaining fixed in one location throughout the 8 time periods of the collision, and it clearly shows the left particle α leaping two extra byte widths over β, while rotating one energy bit at a time, consistent with the prediction of the Soliton Collision Theorem that the collision will take $\mathcal{E}(\alpha) = 8$ time periods and the particle α will jump to the right $(n_\beta + 1)(r + 1) = (1+1)(4+1) = 10$ positions.

The individual identities of the two particles during the eight time periods of the collision, especially the identity of β are much less clear in the configuration evolution figures. The unshifted configuration diagram even shows some apparent diagonal patterns of slope 1 that turn out to be artifacts of this particular collision and have no general meaning. Knowing that the energy configuration of β remains fixed in time-shifted diagrams for the 8 time periods of the collision, one can locate the corresponding 8×5 rectangle where the configuration states of β might be expected to remain fixed, and there find a sum modulo 2 of the fixed, unrotating configuration states of β and offset copies of the rotating configuration states of α during those 8 time periods as they more accurately "step on" β rather than nimbly

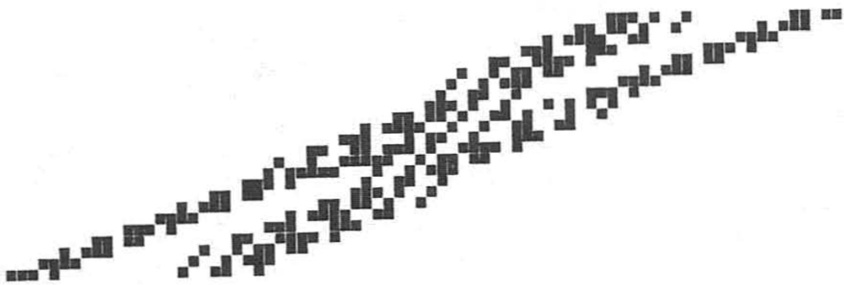

Figure 9: Unshifted configuration diagram for collision of two particles.

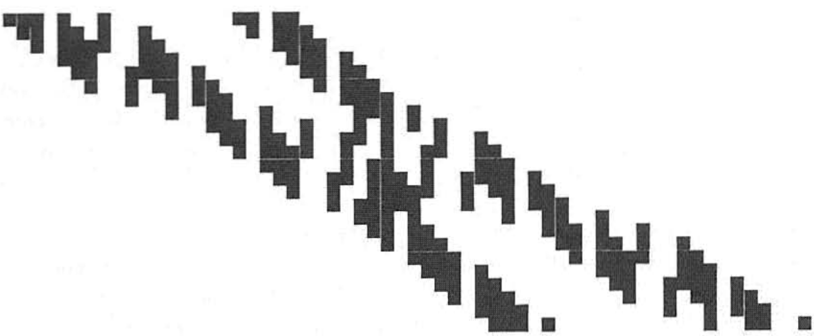

Figure 10: Time-shifted diagram for collision of two particles.

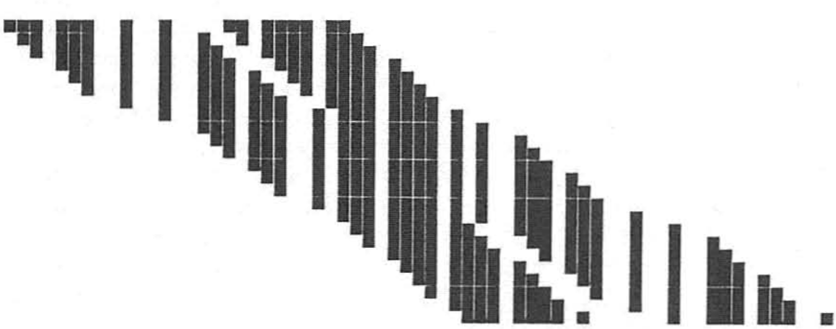

Figure 11: Energy diagram for collision of two particles.

"leap over" β on their way to the right. Miraculously, β reappears intact and ready to resume its rotation in the last time period of the collision. All these observations will now be proved as consequences of the evenness of the energy of α in every relative position of a byte, and of the fact that the energy configuration of β does not change throughout the time periods of collison.

Proof of the Soliton Collision Theorem. We will show that during the collision, the combined configuration has exactly one left particle boundary and one right particle boundary, and that between these, the critical transitions that take place are of two kinds: those that would have taken place in the same time period (although not necessarily in the same position) if α were allowed to evolve independently with β set to zero, and those that involve interactions of α and β. The former are "well-behaved" critical transitions of Type 1 because α is a particle and can have only Type 1 critical transitions between its particle boundaries. That the latter are all Type 1 critical transitions turns out to be equivalent to Condition ICC, the Intersection Compatibility Condition.

The proof proceeds in three stages: (1) The evolution of the configuration from time t to time $t+1$ shows that a_0, the leftmost nonzero state of $\alpha(t)$, reappears with an extra spatial displacement of $n_\beta + 1$ bytes to the right in addition to its normal displacement of $n_\alpha + 1$ bytes in the independent rotation of α. The transitions in this time period establish the width of the combined configuration during the collision, begin to show that α will consist of two pieces, one part to the left of β, and the rest of α to the right of β. Stage 1 forms the basis for the induction. (2) In each of the next $\mathcal{E}(\alpha) - 1$ time periods, one energy bit of α rotates to the right $n_\alpha + n_\beta + 2$ bytes without any change of the energy of the combined configuration between these two positions, and without a splitting byte opening up. (3) At time $t + \mathcal{E}(\alpha)$, the combined configuration splits into constituent particles β and α, which once more begin to evolve independently.

(1) The rotation of a_0: In the transition from t to $t+1$, the configuration bit $a_0 = 1$ disappears at the left particle boundary of $\alpha(t)$. Then there are n_α subsequent critical transitions of Type 1, just as there would have been if $\alpha(t)$ evolved as a separate particle. The last of these places the rightmost nonzero state of $\alpha(t+1)$ on the byte boundary to the right of A_{n_α}. However, here the evolution of the colliding particles configuration diverges from that of the independent evolution of α. This nonzero state does not form a right particle boundary because the next byte boundary falls at b_k, which is $k > 0$ positions into B_1, the first byte of $\beta(t)$. Since b_0, the leftmost bit of $\beta(t)$ is nonzero, the updating automaton inverts the state b_k, and continues in the "primed window" state PW. In the independent evolution of α, only zero states would have been encountered in this byte, and the updating automaton would have returned to the state SLPB.

We now show that Condition ICC implies that the updating automaton will also invert the k^{th} state of $B_2, B_3, \ldots, B_{n_\beta}$, copying all other states of β.

Applying Condition ICC to the case $i = 0$, and noticing that by definition $a_{i-j} = 0$ for all $j \geq 1$, Condition ICC says that for fixed n, not all of the states $b_{k+n(r+1)-1}, b_{k+n(r+1)-2}, \ldots, b_{k+n(r+1)-r}$ are zero if at least one of their subscripts lies in the range $0, \ldots, m_\beta$. For these n, this is precisely the condition that the transition at $b_{k+n(r+1)}$ is not preceded by a splitting byte, and is thus a critical transition of Type 1.

To show that this argument based on Condition ICC extends to establishing a critical transition in the bytes B_{n_β} and $B_{n_\beta+1}$, we need to know that the k^{th} position of B_{n_β} is to the left of b_{m_β}, the rightmost nonzero state of $\beta(t)$. The number of trailing zero states in B_{n_β} is $d_s(\beta, t-1) - 1$, so the rightmost nonzero state is in position $r - d_s(\beta, t-1) + 1$ of the byte. However $k = d_s(\alpha, t-1) - d_s(\beta, t-1) - g \leq r - d_s(\beta, t-1)$, proving the assertion. Thus there is a critical transition of Type 1 guaranteed by Condition ICC in bytes B_{n_β} and $B_{n_\beta+1}$.

In fact, the nonzero state b_{m_β} guarantees that the next critical transition to the right of B_{n_β} is of Type 1, without reference to Condition ICC, and this transition inverts a zero state to nonzero. Thereafter, all states to the right are zero, so this is a right particle boundary, exactly $n_\beta + 1$ bytes to the right of where it would have been if $\alpha(t)$ had evolved independently.

Notice that a_0, the leftmost nonzero configuration bit at time t reappears in the combined configuration at time $t+1$ at two different places, respectively n_α and $n_\alpha + n_\beta + 1$ bytes to the right of where it started. Between these two positions, the state of the k^{th} position of each of the configuration bytes $B_1, B_2, \ldots, B_{n_\beta}$ of $\beta(t)$ is inverted, so the configuration pattern of $\beta(t)$ begins to disappear. However, the energy in these positions is unchanged, which is ultimately the reason why $\beta(t)$ will reappear at time $t + \mathcal{E}(\alpha)$, after all the energy of α has rotated to the right of these positions.

(2) Rotation of the remaining energy of $\alpha(t)$: We use as induction hypothesis that at time t', all configuration states to the left of b_0 are as they would be in $\alpha(t')$, the independent evolution of α to time t', that all configuration states $a_j^{t'}$ to the right of b_{m_β} are related to states of $\alpha(t')$ by the formula $a_j^{t'} = \alpha(t')_{j-(n_\beta+1)(r+1)}$, and that for each t'' with $t \leq t'' < t'$ the chain of critical transitions that starts at the leftmost nonzero state at time t'' has $n_\alpha + n_\beta + 1$ critical transitions of Type 1, the last of which places a right particle boundary state, followed by a splitting byte and a critical transition of Type 2.

We have shown that the induction hypothesis holds for $t' = t + 1$. Now assume that the induction hypothesis holds for some t' with $t \leq t' < t + \mathcal{E}(\alpha)$. We show that it holds for $t' + 1$. Since $t' - t < \mathcal{E}(\alpha)$, there is at least one energy bit of $\alpha(t)$ that as not yet moved in the independent evolution of $\alpha(t)$ to $\alpha(t')$. The leftmost of these nonzero energy bits $a_i^{t'}$ corresponds to a (possibly zero) configuration state a_i that carries energy in $\alpha(t)$. Since all the energy of $\alpha(t)$ was to the left of b_0, the induction hypothesis says that $a_i^{t'}$ is also the leftmost nonzero state of the combined configuration at time t'. The Rapid Updating Automaton begins by setting $a_i^{t'+1} = 0$ and making critical transitions on subsequent byte boundaries. As long as the positions of these critical transitions are at b_0 or to the left of b_0, all states in the preceding byte are as they would be in $\alpha(t')$; thus the transitions are of Type 1.

We pause now for a lemma.

Lemma 7 (Interior Updating Lemma) *Let $a_i^{t'}$ be the leftmost nonzero state of a configuration at time t', and suppose that for all t'' with $t \leq t'' < t'$ the chain of critical transitions starting at the leftmost nonzero state of the configuration at time t'' has only critical transitions of Type 1 or Type 2a2 to the left of or at position j, then if j' is the largest integer of the form $j - n(r+1)$ that is less than i, the configuration state in position j at time t' is given by the formula $a_j^{t'} = a_j^t \oplus a_{j'}^t$.*

Proof. We rely on the expression of configuration states in terms of energy states, $a_j^t = \sum_{n \geq 0} e_{j-n(r+1)}^t$ and $a_j^{t'} = \sum_{n \geq 0} e_{j-n(r+1)}^{t'}$. Subtracting and noting that under the conditions of the lemma, $e_{i'}^{t'} = e_{i'}^t$ if $i \leq i' \leq j$ and $e_{i'}^{t'} = 0$ if $i' < i$, we get $a_j^t - a_j^{t'} = \sum_{j-r(n+1) \leq i} e_{j-n(r+1)}^t = a_{j'}^t$. The last equality results from noticing that j' is the largest possible subscript in the sum, which therefore represents the configuration state $a_{j'}^t$. Transposing terms by addition modulo 2, we obtain $a_j^{t'} = a_j^t \oplus a_{j'}^t$. ∎

We resume the proof of the Soliton Collision Theorem. When any of the r positions preceding a critical transition fall within the range b_0 to b_{m_β}, we apply Condition ICC. If the critical transition is at $b_{k+i+n(r+1)}$, Condition ICC says that there is some j with $1 \leq j \leq r$ such that $a_{i-j} \neq b_{k+i+n(r+1)-j}$. However, $b_{k+i+n(r+1)-j}$ lies within the range of positions to which the Interior Updating Lemma applies, so its value at time t' is given by the formula $b_{k+i+n(r+1)-j}^{t'} = b_{k+i+n(r+1)-j} \oplus a_{i-j}$. Combining these two results, we get $b_{k+i+n(r+1)-j}^{t'} \neq 0$, so the critical transition at $b_{k+i+n(r+1)}^{t'}$ is of Type 1.

For the above argument to be completely correct, we should note that if $j' = k + i + n(r+1) - j$ is outside the range $[0, \ldots, m_\beta]$, Condition ICC uses the defined value $b_{j'} = 0$, but the Interior Updating Lemmma uses the actual state value in relative position $b_{j'}$. Fortunately, the actual state value $b_{j'}$ is also zero because at time t all states to the right of $\beta(t)$ and at least $r+1$ states to the left of $\beta(t)$ are zero.

When all the r positions preceding a critical transition are to the right of b_{m_β}, the induction hypothesis says that exactly the same critical transition would have taken place in $\alpha(t'')$ exactly $n_\beta + 1$ bytes to the left of where it occurs in the combined configuration. Again, this implies that the critical transition is of Type 1, except if it corresponds to the rightmost critical transition of $\alpha(t'')$. The resulting configuration state at this transition is thus the same as the one in $\alpha(t''+1)$, except shifted $n_\beta + 1$ bytes to the right, so it makes that part of the induction hypothesis true for $t''+1$. Since the chain of critical transitions begins at an unshifted left particle boundary of $\alpha(t'')$ and ends at a shifted copy of the right particle boundary of $\alpha(t'')$, there are $n_\alpha + n_\beta + 1$ critical transitions of Type 1 before a Type 2 transition occurs. Thus the complete induction hypothesis is true for $t''+1$.

(3) Splitting of the combined configuration at time $t + \mathcal{E}(\alpha)$: We wish to apply the Interior Updating Lemma to the range of times $t \leq t'' < t + \mathcal{E}(\alpha)$ to show that the configuration $\beta(t)$ reappears in the same position at time $t + \mathcal{E}(\alpha)$. If $0 \leq j \leq (n_\beta + 1)(r + 1)$, then all left particle boundaries for t'' in this range of times are to the left of b_j because they are to the left of b_0. The position of the first Case 2 critical transition at time t'' moves right with increasing t'', and for time $t'' = t$, the smallest in the range, it occurs at the position of $b_{k+(n_\beta+1)(r+1)}$. Thus the Interior Updating Lemma applies to b_j.

The leftmost nonzero configuration state at time $t + \mathcal{E}(\alpha)$ is b_0 because α has period $\mathcal{E}(\alpha)$, and the independent evolution of $\alpha(t)$ for $\mathcal{E}(\alpha)$ time periods moves each configuration bit of $\alpha(t)$ exactly $n_\alpha + 1$ bytes to the right. Since these new positions are all at or to the right of b_k, and thus not to the left of b_0, the induction hypothesis at $t'' = t + \mathcal{E}(\alpha)$ implies that the combined configuration has no nonzero states to the left of b_0, and that the only nonzero states to the right of b_{m_β} derive from nonzero states of $\alpha(t + \mathcal{E}(\alpha))$ shifted right $n_\beta + 1$ bytes, or, what is equivalent, states of $\alpha(t)$ shifted right by $n_\alpha + n_\beta + 2$ bytes. The leftmost of these is at the position of $b_{k+(n_\beta+1)(r+1)}$, so that if $\beta(t)$ reappears in its original position at time $t + \mathcal{E}(\alpha)$, it will have a full splitting byte of zero states $B_{n_\beta+1}$.

Now that we have shown that the Interior Updating Lemma applies over the time interval from t to $t + \mathcal{E}(\alpha)$ to each b_j with $0 \leq j \leq (n_\beta + 1)(r + 1)$, and that the $r + 1$ states to the left of b_0 at time t are zero, the lemma proves that $b_j^{t+\mathcal{E}(\alpha)} = b_j \oplus 0 = b_j$.

Thus $\beta(t)$ reappears at time $t + \mathcal{E}(\alpha)$ in its original position in the time-shifted diagram, complete with a full splitting byte of zero states, and $\alpha(t)$ reappears at time $t + \mathcal{E}(\alpha)$, shifted right $n_\alpha + n_\beta + 2$ bytes, which places it to the right of $\beta(t + \mathcal{E}(\alpha))$. Because there is a splitting byte at the right of $\beta(t + \mathcal{E}(\alpha))$, the two component partices evolve independently for at least one time period. This completes the proof of the Soliton Collision Theorem. ∎

Figure 1 is the time-shifted configuration of the same collision as figure 9, this time with nonzero configuration states identified with respect to whether they derive from α or from β. The modified rotation (and jump) of α is now seen clearly, as well as the gradual disappearance and reappearance of the nonzero states of β during the collision. If you look carefully, you can even see the "footprints" of α as it steps across the fixed particle β in this time-shifted diagram.

Theorem 20 (Converse of the Soliton Collision Theorem) *If two particles α and β collide with α initially to the left of β, and they stay in collision for at least $\mathcal{E}(\alpha)$ time periods, then Condition ICC holds, they stay in collision for exactly $\mathcal{E}(\alpha)$ time periods, and the collision is a soliton collision.*

Proof. The term *staying in collision*, means that there is only one component of the combined configuration in these time periods. As a result, there can never be a splitting byte or a Type 2 critical transition to the left of a known nonzero state, or even at the first byte boundary after a known nonzero state.

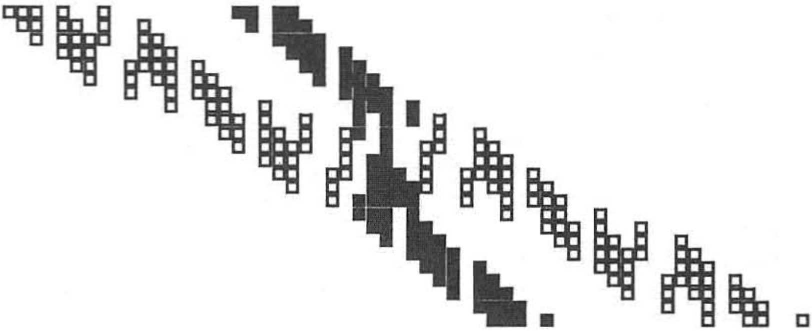

Figure 12: Time-shifted diagram of the collision of two particles with $r = 4$. Nonzero states of α are open boxes and nonzero states of β are filled boxes.

We use the same notation as in the proof of the Soliton Collision Theorem. At time t, the leftmost nonzero state is a_0 and the rightmost nonzero state is b_{m_β}. As before, the critical transitions of interest occur at $b_{k+n(r+1)}$, that is, at position k of every byte B_j. As before, position k of B_{n_β} is to the left of b_{m_β}, so there will be a critical transition of Type 1 in position k of $B_{n_\beta+1}$ depositing the rightmost nonzero state for time $t+1$, and a critical transition of Type 2 at position k of $B_{n_\beta+2}$ depositing the energy carried by a_0 at time t.

Since the r states of $\alpha(t)$ to the left of a_0 are zero states, Condition ICC applied to the energy bearing state a_0 is equivalent to the condition that not all of the r states to the left of $b_{k+n(r+1)}$ are zero states for $0 \leq n \leq n_\beta + 1$. However, these $b_{k+n(r+1)}$ are precisely where we have shown there are Type 1 critical transitions, so Condition ICC holds for time t.

As in the proof of the Soliton Collision Theorem, the left particle boundary for each of the $\mathcal{E}(\alpha)$ time periods starting at t corresponds to an energy bearing state a_i of $\alpha(t)$. Condition ICC applies only to these states. Since there is only one component during these time periods, and since the position of the rightmost nonzero state of the single component moves right with increasing time because energy bearing a_i are never more than r positions apart, there cannot be a splitting byte until well to the right of b_{m_β} for any time t' with $t \leq t' < t + \mathcal{E}(\alpha)$. Let t' be the time when $a_i^{t'}$ is the left particle boundary. The Interior Updating Lemma applies to the range of times from t to t' and range of positions from b_{-r} to $b_{m_\beta+r}$ so for all t' and $k+i+n(r+1)-j$ in these ranges, $b^{t'}_{k+i+n(r+1)-j} = b_{k+i+n(r+1)-j} + a_{i-j}$. The absence of splitting bytes in this range means that for each applicable n, there is at least one j with $1 \leq j \leq r$ for which $b_{k+i+n(r+1)-j} \neq 0$. For this j, we have $a_{i-j} \neq b_{k+i+n(r+1)-j}$ and Condition ICC holds for time t'.

Since every energy bearing state a_i of $\alpha(t)$ appears as a left particle boundary during the $\mathcal{E}(\alpha)$ time periods the two particles are assumed to be in collision, Condition ICC is fully verified. As a result, the Soliton Collision

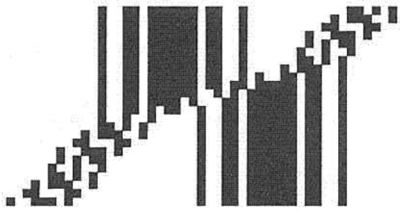

Figure 13: Unshifted configuration diagram of the soliton collision of a stationary particle with a particle of average velocity 1. Note the gradual jump of the stationary particle 12 positions to the right taking place over 8 time periods, and the speeding up of the other particle to the speed of light, $r = 3$, during the collision.

Theorem may be applied to show that this collision is a soliton collision and lasts for precisely $\mathcal{E}(\alpha)$ time periods. ∎

Constructing a clear unshifted collision diagram

The principal facts about the unshifted configuration state transition diagram of a soliton collision are (1) that the left particle α "jumps" to the right by $n_\beta+1$ bytes; however this jump does not take place all at once, but gradually, one energy bit at a time over the $\mathcal{E}(\alpha)$ time periods of the collision, (2) the right particle β speeds up to the speed of light during the collision, and (3) the left particle α continues rotating, while the right particle β does not. The next example we construct will illustrate properties (1) and (2).

To illustrate the "jump" of α, we use a stationary particle for α because the persistence of the configuration states of a stationary particle in fixed positions in the unshifted diagram makes it easy to see when any part of α moves. To show the speed-up of β to the speed of light, we choose a particle for β whose average velocity in the unshifted diagram is recognizably different from the speed of light, and a value for r, the speed of light, that is not so high that the trace of β during the collision does not appear nearly horizontal. Choosing $v_\beta = 1$ and $r = 3$ satisfies these objectives. The width of the stationary particle with $r = 3$ is exactly $3r^2 = 18$ positions which is wide enough for a satisfactory diagram. We choose a width of β of 2 bytes to provide a jump of $(n_\beta + 1)(r + 1) = 12$ positions, $2/3$ of the width of α. The energy $\mathcal{E}(\alpha) = 8$, so there is a moderate time to observe the speed of β during the collision. Figure 13 is the result.

After a soliton collision

The Soliton Collision Theorem guarantees only that the particles evolve independently for one time period after the collision. If the average unshifted velocity of β exceeds that of α, or what is equivalent, the average shifted velocity of α exceeds that of β, (which in most instances is why they collided in the first place), once they have participated in a soliton collision and

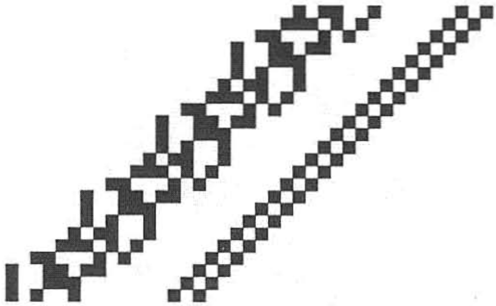

Figure 14: Unshifted configuration diagram for two particles not orbiting, $r = 3$.

switched places so that β is on the left, their differences in velocity will tend to take them even further apart in each succeeding time period.

However, neither of these two conditions is guaranteed. As we shall see in section 19, particles of equal average velocity can collide, identical particles can collide if they are at different phases of their rotational cycle, and even particles of slower average velocity can overtake and collide with particles of faster average velocity if they start close enough together and if their velocities during the early part of the cycle differ markedly from their average velocities.

19. Systems of orbiting particles

When two particles have the same average velocity and sufficient initial separation, they do not collide. For example, the two particles in figure 14 both have shifted velocity 2 and unshifted velocity 1. The window radius $r = 3$. The right particle β has period 1 (really a divisor period) and thus always has the same displacement $d_s(t, \beta) = 2$ in all time periods. The particle β is actually a spaced version of the "photon" for $r = 1$. The left particle α, however, has a more variable "instantaneous velocity" which sometimes exceeds the average velocity and sometimes is less than it. Thus the independent motion of α consists of motion at the average velocity and a pertubation that is sometimes to the left of the average path and sometimes to the right.

When these two particles start somewhat closer together, as shown in figure 15, they evolve independently for a few time periods until α moves to the right of its average position. Then they collide. The collision is a soliton collision, and after $\mathcal{E}(\alpha) = 6$ time periods, α is to the right of β. Since β does not rotate during the collision, and α goes through a complete revolution, they are in the same relative phase when independent evolution resumes after the collision. However, since their average velocities are equal, they do not tend to move apart, and when α reaches that part of its period, say Δt time intervals later, when its pertubation is to the left of its average path, the particles are again close enough to collide.

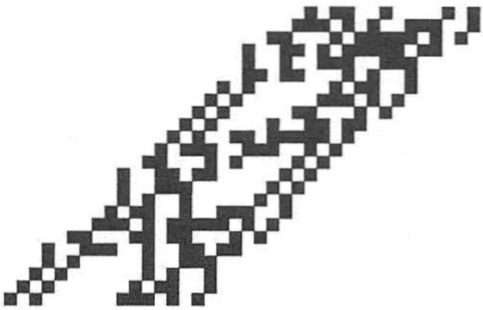

Figure 15: Unshifted configuration diagram for two orbiting particles, $r = 3$.

This time the collision takes $\mathcal{E}(\beta) = 4$ time periods. The particle α does not rotate during this interval, and the particle β goes through one complete revolution. During the first collision, the energy bits of α jumped to the right by $n_\alpha + n_\beta + 2$ bytes, and during the second collision, the energy bits of β jumped to the right by $n_\beta + n_\alpha + 2$ bytes, the same amount. Thus when the particles resume independent evolution after the second collison, they are in precisely the same relative position as they would have been evolving independently for Δt time periods from the time of the initial contact. As a result, after $\mathcal{E}(\alpha) - \Delta t$ additional time periods, α has completed two full rotations. In general, we cannot expect β to be in its initial phase after $\mathcal{E}(\alpha)$ time periods, but the particular β of this example has the divisor period 1 which divides the period $\mathcal{E}(\alpha)$ of α. Thus, when α completes its second revolution, they are in exactly the same relative position they were in at the initial contact, and the sequence of pairs of soliton collisions repeats for these two particles every $2\mathcal{E}(\alpha) + \mathcal{E}(\beta) = 16$ time periods.

Differences between single particles and systems of orbiting particles

It is important to make a distinction between single particles in the sense we have made precise earlier and such *systems of orbiting particles*. For one thing, there are infinite families of such systems of orbiting particles, while the number of single particles for each r is large, but finite. For a second thing, the period and velocity of a single particle are completely determined by the particle's energy and size in bytes, while these formulae do not apply to systems of orbiting particles.

If the period of β does not divide $\mathcal{E}(\alpha)$ or if the pertubation of motion of one or both particles is more complex, there may be additional collisions before the particles repeat their original relative position and rotational phase. However, since their average relative position remains constant except for the jumps of $n_\alpha + n_\beta + 2$ positions when the particles collide, as long as there are times in their joint independent rotaton when α is relatively further to the right of its average position than β and other times when β is relatively farther to the right of its average position than α, the particles will repeatedly cross and recross as a system of orbiting particles.

20. Tangent or osculating particles

Tangent or osculating particles occur when the gap g between the splitting byte of the first particle and the left particle boundary of the second particle is zero. As a result, the byte boundaries of the second particle are aligned with and continue the pattern of byte boundaries of the first particle. There is a critical transition of Type 2a2 at the left particle boundary of the second particle, the energy interpretation of which is ambiguous (see the Critical Transition Lemma). It may be considered as a transition where the energy lost at the left particle boundary of the first particle is restored to the configuration at the same position where energy is being lost to the primed window at the left particle boundary of the second particle, or it may be considered just another energy preserving critical transition that leaves the computational window primed for the next critical transition $r+1$ positions to the right.

Another way to express the ambiguity of tangent particles is to notice that you get exactly the same configuration in the next time period if you consider the configuration as two particles evolving independently, or as one large particle that evolves by rotating its leftmost nonzero energy state to the first byte boundary to the right of the complete configuration.

Tangency and collisions

Suppose two particles α and β are tangent, and remain tangent for several time periods. Since tangent particles have aligned byte boundaries, their displacements must be the same during these time periods, and therefore the initial (i.e., leftmost) states of their energy diagrams and configurations must be equal. It makes equal sense to say that the initial nonzero energy states of α are rotating to the right of the combined configuration while all the energy of β remains fixed, i.e., that they are in collision, as it is to say that the nonzero energy states that are appearing on the right of the configuration are the initial nonzero energy states of β which are disappearing from the left particle boundary of β somewhere in the middle of the combined configuration, only to be replaced by the equal initial nonzero energy states of α.

Suppose now α slows down relative to β in the next time period. A positive gap $g > 0$ will open up between the particles, the byte boundaries will no longer be aligned, and the particles will evolve independently. Since the tangency of the two particles does not result in a full collision lasting $\mathcal{E}(\alpha)$ time periods, it was considered preferable to define tangent particles as not in collision.

Suppose however that after Δt time periods of tangency, α speeds up relative to β in the time-shifted diagram. The gap is then negative and the particles are in collision at time t. Assuming all goes well and the collision is a soliton collision, at time $t + \mathcal{E}(\alpha)$, their relative phases are as they were at time t, and α has moved $n_\alpha + n_\beta + 2$ bytes to the right relative to β. We now back up to time $t + \mathcal{E}(\alpha) - \Delta t$ and claim that at this time we can already find the complete particle α at the right of the configuration. This is Δt time periods earlier than predicted.

By backing up Δt time periods to time $t + \mathcal{E}(\alpha) - \Delta t$, we do not allow the last Δt nonzero energy states of α to rotate to the right. These were the energy states that got to the right end of α during the Δt periods of tangency, and are thus matched by a set of equal energy states at the right of β at the time t of the collision. These sets of states are an exact number of bytes apart, namely $n_\beta + 1$ bytes apart. Thus we may interpret the combined configuration at time $t + \mathcal{E}(\alpha) - \Delta t$ in two ways, first as a configuration in the midst of a collision during which Δt nonzero energy states of α have yet to rotate, and second as two particles, the particle $\beta(t - \Delta t)$ to the left of and tangent to a second particle which is the transposition of the particle $\alpha(t - \Delta t)$ to the right by $n_\alpha + n_\beta + 2$ bytes.

Strictly speaking, the conclusions of the Soliton Collision Theorem are not true in this instance because the configuration splits into separate particles Δt time periods earlier than predicted by the Theorem. (This is not a counterexample to the theorem because Condition I is violated by the equal initial energy patterns of α and β.) However, both in the time interval from t to $t + \mathcal{E}(\alpha)$ and from $t - \Delta t$ to $t + \mathcal{E}(\alpha) - \Delta t$, (as well as in all the time intervals of length $\mathcal{E}(\alpha)$ between these two), the other conclusions of the Soliton Collision Theorem are fulfilled. It therefore makes sense to weaken the hypotheses of the Soliton Collision Theorem and the definition of "particles staying in collison" in its converse so that tangency is permitted within a collision.

21. Collisions of small particles

Condition I, the intersection compatibility condition in the Soliton Collision Theorem guarantees that as the left particle α "steps" over the right particle β, it does not accidentally create a string of r consecutive zero states that cause the colliding particles configuration to split prematurely. Some of the transitions to which Condition I may be applied are also proved to be Type 1 critical transitions because of known nonzero states such as the leftmost and rightmost nonzero states of β and the first nonzero state of α to reappear to the right of β. If both colliding particles are small particles, i.e., contained in one byte, we now show that premature splitting can never take place.

Theorem 21 (Small Particle Collision Theorem) *The collision of two small particles is always a soliton collision.*

Proof. Suppose the particles α and β collide at time t. For notational convenience, we label nodes in the cellular automaton by an ordered pair, (byte number, relative position), with $(0,0)$ at the leftmost nonzero state of $\alpha(t-1)$. Then all of α is contained in byte 0, and all the states of byte 1 are zero. If $g \geq 0$ is the gap between particles at time $t-1$, the leftmost nonzero state of β is at $(2,g)$, and all states to the right of $(3,g)$, inclusive, are zero. Let $(0,i)$ be the position of the second nonzero state of $\alpha(t-1)$ from the left, and $(2,g+k)$ be the position of the second nonzero state of $\beta(t-1)$.

At time t, the nonzero states of $\alpha(t)$ extend from $(0,i)$ to $(1,0)$, and those of $\beta(t)$ extend from $(2,g+k)$ to $(3,g)$. Since the particles are in collision at time t, we have $i > g+k$. Consider the evolution of the configuration for those time periods when the leftmost nonzero state is in byte 0. The critical transition in byte 1 is of Type 1 because of the nonzero state at $(1,0)$, the transition in byte 2 is of Type 1 because of the nonzero state at $(2,g+k)$, and the transition in byte 3 is of Type 1 because of the nonzero state at $(3,g)$. In the next time period, the leftmost nonzero state of the configuration is at $(1,0)$. The transition at $(2,0)$ is Type 1 because of the nonzero state at $(1,i)$, the transition at $(3,0)$ is Type 1 because of the nonzero state at $(2,g+k)$, and the transition at $(4,0)$ is Type 1 because of the nonzero state at $(3,i)$.

Next we consider the remaining time periods when the leftmost nonzero state is in byte 1. Since $i > g+k$, the transition in byte 2 is of Type 1 because of the nonzero state at $(2,g+k)$, the transition in byte 3 is Type 1 because of the nonzero state at $(3,g)$ if $g>0$, and the transition in byte 4 is Type 1 because of the nonzero state at $(4,0)$. The remainder of the proof in the case when $g=0$ will be handled separately below after completing the proof in case $g>0$. For $g>0$, the next (and it will turn out to be the last) time period of the collision has the leftmost nonzero state of the configuration at $(2,0)$. The transition at $(3,0)$ is Type 1 because of the nonzero state at $(2,g+k)$, the transition at $(4,0)$ is Type 1 because of the nonzero state at $(3,g)$, and the transition at $(5,0)$ is Type 1 because of the nonzero state at $(4,i)$. In the next time period, every state from $(3,g+1)$ to $(4,i-1)$ is zero because it started at zero at time t and was inverted either two or no times. The leftmost nonzero state of the configuration is at $(2,g+k)$, it moves to $(3,g+k)$, but there is a splitting byte before $(4,g+k)$ since $i > g+k$. Particles now evolve independently with the left particle boundary of α at $(4,i)$.

In case $g=0$, the two particles are tangent at time $t-1$. Assume the they were first tangent at time $t-\Delta t$ for $\Delta t \geq 1$. When two particles are tangent, their leftmost nonzero states have the same relative position within a byte. During the Δt time periods of tangency, in each particle these nonzero states moved exactly one byte to the right, so at time t, the rightmost Δt nonzero states of α and β have the same relative position in a byte, and the $\Delta t+1$st nonzero state is farther to the left in α. Thus the transition to byte 3 of all but the last Δt nonzero state of α is guarantee of Type 1 because of the $\Delta t+1$st nonzero state of $\beta(t)$ from the right. Since there were Δt time periods of tangency before the two particles collided, the remaining Δt time periods of a soliton collision (in the extended sense) consist of tangent evolution of the two particles, as was shown in section 20. ∎

Comparison with empirical results

The Small Particle Collision Theorem explains the large percentages of soliton collisions reported in [1] when the window radius r approaches the sampled particle size. In this case, nearly all the sampled collisions are collisions of small particles, which by the theorem must be soliton collisions.

The empirical statistics in [1] for the case of window radius $r = 9$ and particles of size at most 10 were initially of some concern. Whereas Table II in [1] reports that 99.42% of a random sample of 2000 such collisions were soliton collisions, the Small Particle Collision Theorem predicts that all collisions of such particles should be solition collisions. Replication of the original empirical study and examination of the 23 particle interactions reported as non-soliton collisions show that in each case the interaction is indeed a soliton collision of two particles of nearly equal velocity, but the simulations on which Table II are based were run an insufficient number of time periods after the soliton collision to allow these particles to separate sufficiently so that the correct nature of these collisions could be detected by the criteria used in [1]. (They use a gap of at least $2r$ consecutive zero states to partition a configuration into components.) Thus even the most problematic of the empirical statistics in [1], when examined closely, not only do not provide counterexamples to the Small Particle Collision Theorem; in fact they provide 2000 examples confirming the theorem.

Collisions of particles of configuration byte width 2 can also be analyzed by the same techniques. They are not all soliton collisions because some of the compatibility relations that are part of Condition I are not automatically satisfied. The probability of randomly chosen particles violating any one of the compatibility conditions is in most cases the same as the probability of two random sequences of r binary digits agreeing, which decreases exponentially with r. The number of such conditions in Condition I increases quadratically with r (for fixed byte size of the two particles), so for large r, almost all the requirements of Condition I are satisfied with high probability. However, compatibility conditions near the end of the collision rely on agreement of very small numbers of states and therefore bound the probability of soliton collisions of such particles away from 1.

22. Almost orbiting particles

Soliton collisions in filter automata are used in [2] to embed computation in the evolution of particles in the filter automaton by observing the phase shift of selected particles during soliton collisions. For the particles and window radius they choose, they observe phase shifts in both particles. The Soliton Collision Theorem, however, says that only one of the colliding particles should have a phase shift. The right particle β does not rotate during the $\mathcal{E}(\alpha)$ time periods of the collision, so its phase would be expected to be delayed by $\mathcal{E}(\alpha)$, but the left particle α does rotate throughout the time of the collision, and so should experience no phase shift. The resolution of this seeming inconsistency is that the "soliton collision" they use in their

Figure 16: Almost orbiting particles used in a carry-ripple adder. In each collision, two particles of nearly equal velocity orbit three times before finally separating.

carry adder is really the product of three successive soliton collisions of two particles of nearly equal average velocities. As can be seen in figure 16, the particles collide at a low relative velocity, nearly orbit each other for several oscillations, but since their average velocities differ, the faster particle eventually escapes before the slower particle can overtake it again for yet another soliton collision and orbital period.

23. Open questions

Progressing from the general to the specific, there are a number of interesting questions as yet unanswered. First, what is the proper context in which to give a general definition of a cellular automaton? Should a group of symmetries of the underlying graph be taken into account and the updating function required to be invariant under these symmetries? Most parallel synchronous cellular automata studied have this property, but serial automata may have a fundamental assymetry forced by the order relation. What happens when the time dependence function in the updating rule also looks at time $t-1$ so that three (or more) time periods are involved in the relation?

Is there an interesting serial automaton on the integer grid points in the plane that exhibits all or some of the interesting properties of parity filter automata on the line? What is the evolution of infinite configurations in a parity filter automaton? (The updating rule will have to be modified at minus infinity, perhaps to start in state PW.) How many of the results carry over to a finite circular integer graph? Are there interesting serial automata on the line with k possible state values?

For parity filter automata, give a complete characterization of "impossible" configurations, that is, configurations that cannot appear after one or

n time periods, or which cannot appear in a particle. Show that the restriction to exactly one left and one right particle boundary in a particle is unnecessary.

Finally, is there any connection between the "solitons" found in parity filter automata and the "solitons" found as solutions of differential equations?

References

[1] James K. Park, Kenneth Steiglitz, and William Thurston, "Soliton-Like Behavior in Automata", *Physica* **19D** (1986) 423–432 (North-Holland, Amsterdam). Appeared earlier as Technical Report No. 340, Department of Computer Science, Princeton University, April 1985.

[2] Kenneth Steiglitz, Irfan Kamal, and Arthur Watson, "Embedding Computation in One-Dimensional Automata by Phase Coding Solitons", *IEEE Transactions on Computers*, **37:2** (1988) 138–145. Appeared earlier as Technical Report No. 15, Department of Computer Science, Princeton University, November 1985.

[3] Kenneth Steiglitz, diagrams from an unpublished article on Parity Filter Automata.

[4] Kenneth Steiglitz, private communication.

[5] Stephen Wolfram, "Some Recent Results and Questions About Cellular Automata", *Proceedings of the Workshop on Dynamical Behaviour of Automata: Theory and Applications*, (Marseilles, September 1983) ed. J. Demongeot, E. Goles, and M. Tchuente, Academic Press.

[6] Stephen Wolfram, *Theory and Applications of Cellular Automata*, (World Scientific Publishing Co., Singapore, 1986).