A Weakly Universal Cellular Automaton in the Heptagrid of the Hyperbolic Plane

Maurice Margenstern
LGIPM, Department of Computer Science and Applications
Université de Lorraine
3 rue Augustin Fresnel, BP 45112, 57073 Metz, Cédex 03, France

In this paper, we construct a weakly universal cellular automaton (CA) in the heptagrid, the tessellation $\{7, 3\}$ that takes place in the hyperbolic plane. The CA is not rotation invariant but is truly planar. This result, under these conditions, cannot be improved for the tessellations $\{p, 3\}$ of the hyperbolic plane.

Keywords: cellular automata; tessellations; hyperbolic plane; weak universality; railway model

1. Introduction

This paper presents a result of weak universality in a tiling of the hyperbolic plane. Weak universality is a terminology encouraged by the author and followed by some others, like [1, 2]. Classically, the Turing machine starts its computation from an initial finite configuration; the term weak universality stresses that the condition on the initial configuration is weakened in some way. In many cases of such results, the relaxed condition is the finiteness of the initial condition. This is the case here. As in many other studies where that relaxation occurs, the initial configuration cannot be arbitrary: outside a finite frame, it must be constructed by some automatic process, possibly with the help of single-stack machinery. A famous example of a result established in a context where the initial configuration is at large constructed with the help of a finite automaton is the weak universality of rule 110 of elementary cellular automata (ECAs) according to Wolfram's classification. The result was conjectured in 1983 by Wolfram and the proof was published in [3, 4] 20 years later.

In this paper, the space where the considered cellular automaton lives is the hyperbolic plane. The interest of this frame lies in the infinite number of regular tessellations on which cellular automata (CAs) can be defined as usual, considering such a tessellation as the space of the cells. Outside the theoretical interest of this new frame, possible applications can be viewed in the fact that hyperbolic geometry is suited as a frame for restricted relativity. Another possible application

https://doi.org/10.25088/ComplexSystems.27.4.315

is the possibility in this richer geometrical space to represent graphs and especially trees in a more convenient frame. We refer the reader to [5, 6] for an introduction to tessellations in the hyperbolic plane suited to such a paper.

This paper is basically an improvement of [7] and [8], where the author proved the same result in the tessellations $\{9, 3\}$ and $\{8, 3\}$, respectively. The reason for this improvement lies in the relatively small number of rules for [7] and the fact, noticed in that paper, that several rules were uselessly duplicated. Also, as is usual in this process of reducing the possibilities of the automaton, here its neighborhood, it is necessary to change something in the previous scenario of the simulation. Here, the scheme explained in [9] is repeated. The key idea of that paper was to combine two existing structures in order to eliminate one of the structures used so far in the simulation scheme explained in [6]. Note that the present result cannot be improved for this class of CAs, which explicitly make use of nonrotation-invariant rules: indeed, the heptagrid is the tessellation $\{p, 3\}$ of the hyperbolic plane with the smallest possible value for p, which is 7. This paper makes use of the new system of coordinates introduced in [10] for the tilings $\{p, 3\}$ and $\{p-2, 4\}$. In this paper, the same model as in [11] and [12] and the other quoted papers is used.

For the reader's convenience, Section 2 introduces the model simulated by the automaton. In Section 3, the model is implemented in the heptagrid. In Section 4, the rules of the automaton are given, stressing the way the rules are defined in a context where rotation invariance is no longer required, which allows us to prove the following result:

Theorem 1. There is a weakly universal cellular automaton (CA) on the heptagrid of the hyperbolic plane, the tessellation $\{7, 3\}$, that is truly planar and that has two states.

Presently, we turn to the proof of the result, repeating that the rules are not rotation invariant: the statement of the theorem does not mention that condition.

2. The Railway Model

This model was introduced by Stewart in [13]. It defines a circuit that takes place in the Euclidean plane on which a single locomotive is running. The circuit consists of segments of straight lines, quarters of circles, crossings and switches. The crossing is a structure allowing two tracks to cross each other. A switch is the meeting of three straight lines a, b and c. The locomotive may arrive at the switch through a and then leave it either through b or c: it is called an *active* passage or crossing. The track through which the locomotive leaves the switch is called the *selected* track. The locomotive may also arrive through *b* or *c* and then leave the switch through *a*: this is called a *passive* crossing.

There are three kinds of switches: fixed switch, flip-flop and memory switch. The fixed switch accepts both active and passive crossings. But in an active passage, the selected track is always the same: either always *b* or always *c*. The flip-flop accepts active passages only, but the selected track is changed after each passage of the locomotive. The memory switch also accepts both active and passive crossings. However, the rule for changing the selected track is different: the selected track is the track of the last passive crossing by the locomotive.

Assembling all those elements allows us to construct a circuit in which the traversal by the locomotive mimics any Turing machine (see [13]) or any register machine; see [7, 10–12, 14] as well as [6] for the results obtained before the previous ones. At each time, the configuration of the circuit is defined by the state of all its switches, where the state indicates which track is the selected one.

Figure 1 and its caption illustrate the working of the *basic element* of the circuit. The element contains one bit of information. The figure explains how the information is read. It also explains how it is written. Note that the writing is performed only for changing the current value to the opposite one. The circuit must be managed in such a way that the locomotive enters *W* if and only if a change of the bit must be performed at that moment.

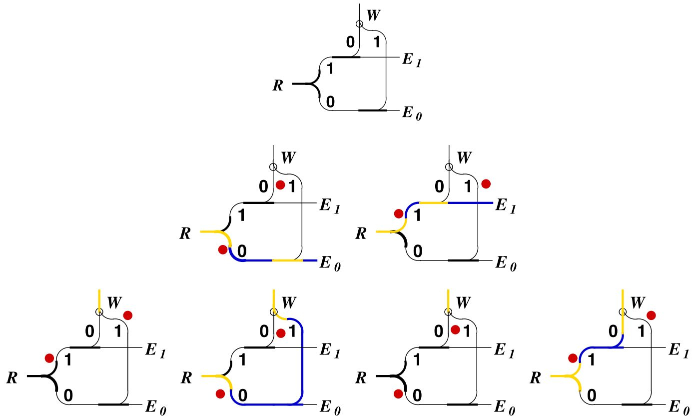


Figure 1. The basic element: first row, the element with *R*, reading entry and *W*, writing entry. Second row: the action of the reading entry: left, reading 0; right, reading 1. Third row, action of the writing entry. Left, marking the change from 1 to 0 at *W*, then at *R*. Right, change from 0 to 1 at the same places.

3. The Scenario of the Simulation

In [7], we could reduce the number of faces of the cells from 11 in [11] down to nine by introducing the decomposition of the crossing and of the previously mentioned switches into simpler structures. In [7], these structures are the passive fixed switch, the fork, the doubler, the selector between a simple and a double locomotive, the controller and the sensor.

That decomposition reinforces the importance of the tracks: their role for conveying key information is more and more decisive. Here too, tracks are blank cells marked by appropriate black cells we call *milestones*. We carefully study this point in Section 3.1. Later, in Section 3.2, we adapt the configurations described in [7] to the heptagrid, the tessellation $\{7, 3\}$.

3.1 The Tracks

In this implementation, the tracks are represented in a way that is a bit similar to that of [7, 8]. Figure 2 indicates the basic feature of the implementation.

First, the sides of a cell are numbered. That numbering of the sides will systematically be used throughout the paper. As far as rotation invariance is no longer required, we may decide which side of the cell is side 1. That choice is at our disposal and, for each cell, it is fixed once and for all. It also fixes the numbers of the other sides of the cell: they are numbered starting from side 1, moving counterclockwise around the cell. That drives us to consider that the tracks are one way. Consequently, the following convention is introduced: in each cell of the track, side 1 is the side shared by the next cell of the track. Note that the same side, which is shared by two cells, most often receives two different numbers in the cells that share it. An example of that situation is given in Figure 2: in the central cell, denoted by 0(0), side 1 is side 6 in the neighbor of the central cell sharing that side. In Section 4.1, the tracks consist of assembling the elements indicated in Figure 2.

Note that Figure 2 shows two rays starting from M, the midpoint of side 2 of the central cell. Those rays allow us to introduce the numbering of the tiles based on [10]. It will be used in the figures illustrating this paper.

Those rays, u and v, are defined as follows. They both start from the midpoint M of side 2 of the largest cell of the figure. The ray u passes through the midpoint of its side 1. It also passes through the midpoint of side 7 of the neighbor of the central cell, which is seen through its side 1 and which we denote 1(1). The ray v cuts sides 5 and 4 of 1(1) at their midpoints. Its support also passes through the midpoint of side 3 of the central cell.

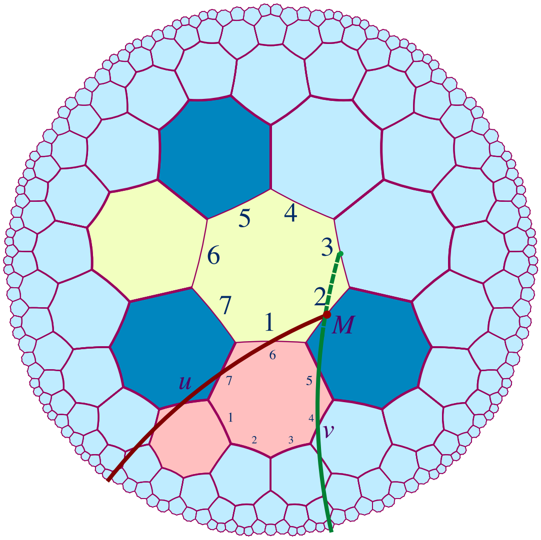


Figure 2. Element of the tracks: in pink, the exit cells; in yellow: the entrance to the cell and the cell itself.

In the figures in this paper, the central cell is the tile whose center is the center of the circle in which the figure is inscribed. The central cell is numbered 0, denoted by 0(0). We number the sides of the tile as indicated in Figure 2. For $i \in \{1..7\}$, the cell that shares side i with the central cell is called *neighbor* i and it is denoted by $1(i)$. The rays u and v denote *sector* 1. The rotation around 0(0) allows us to attach a sector to each tile $1(i)$. Number 1 in this notation is the number given to the root of the tree attached to the sector defined from this tile; see [5, 10]. Here, that definition is adapted to the case of the tessellation $\{7, 3\}$, which we call the heptagrid from now on, as in many previous papers. The reader is invited to follow the present explanation of Figure 1. Consider the sector defined by the rays u and v. The neighbors of cell 1(1) sharing its sides j, $j \in \{1..3\}$ are numbered $j+1$ and are denoted by $j+1(1)$. We say that cell 1(1) is a *W*-cell and its daughters are defined by the rule W→BWW, which means that 2(1) is a *B*-cell. This means that the daughters of 2(1) are defined by the rule B→BW, where the *B*-daughter has two consecutive sides crossed by u in their midpoints. Those daughters of 1(1) constitute level 1 of the tree. The daughters of 2(1) starting from its *B*-daughter are numbered 5 and 6, denoted by 5(1) and 6(1), respectively. By induction, level $n+1$ of the tree is the daughter of the cells that lie on level n, applying the preceding rules. The cells are numbered from level 0, the root, level by level, and on each level, from left to right, that is, from ray u to ray v. What we have seen on the numbering of the daughters of 2(1) is enough to see how the process operates on the cells. From now on, we use this numbering of the cells in the figures of this paper.

Further, Figure 3 illustrates how to assemble elements of the track on which the locomotive passes. As mentioned in the caption, the trajectory of the locomotive is illustrated in yellow. From the point of view of the CA, this is not a new state: yellow cells are blank cells. This representation is used to facilitate the reader's understanding.

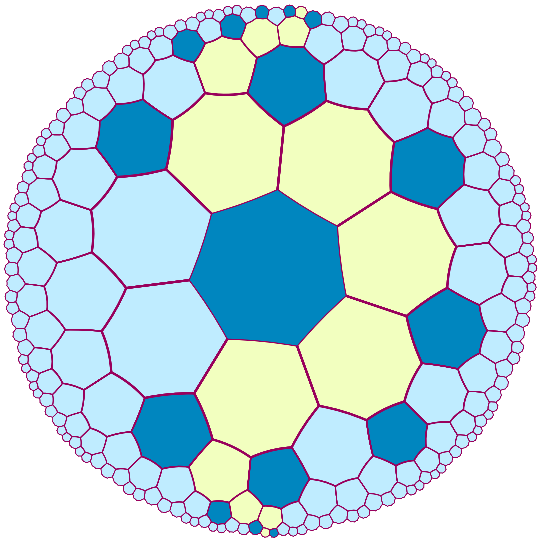


Figure 3. Element of the tracks: in yellow, the elements of the track where the locomotive passes.

As can be seen in the figures of Section 4, the locomotive is implemented as a single black cell: it has the same color as the milestones of the tracks. Only the position of the locomotive with respect to the milestones allows us to distinguish it from the milestones. As is clear from the next subsection, we know that besides this *simple locomotive*, the locomotive also occurs as a *double locomotive* in some portions of the circuit: two consecutive black cells. In a double locomotive, the cell that is a neighbor of the next cell occupied by the locomotive is the *front* of the locomotive, while its other cell, neighboring the front, is called the *rear*. The cell of a simple locomotive is also called the front of the locomotive.

In Figure 3, assuming that the locomotive goes from top to bottom, the milestones are in neighbors 2, 5 and 7 for cell 1(6) or in neighbors 2, 4 and 6 for cell 3(1). It is important to notice that such tracks allow us to join any pair of points. In Section 4, we check that the rules will satisfy that constraint.

The circuit also makes use of signals that are implemented in the form of a simple locomotive. Accordingly, at some point, it may happen that we have three simple locomotives traveling on the circuit: the locomotive and two auxiliary signals involved in the working of some switch. For aesthetic reasons, the black color that is in contrast to the blank is dark blue in the figures.

3.2 The Structures of the Simulation

The crossings of [13] are present in many of the author's papers. In [15] and later papers, they are replaced by roundabouts, a road traffic structure, in the author's simulations in the hyperbolic plane. At a roundabout where two roads are crossing, if you want to continue in the direction you arrived at the roundabout, you need to leave the roundabout at the second road. The structure is illustrated by Figure 4. Its caption illustrates the position of the components of the structure: a fixed switch, a doubler and a selector.

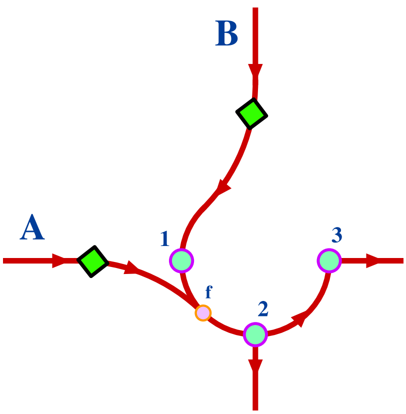


Figure 4. Implementation scheme for the roundabout: the green diamond is a doubler; circle f is a fixed switch. Circles 1, 2 and 3 are selectors.

The arriving simple locomotive first reaches a doubler that transforms it into a double locomotive. Then, the new locomotive meets a first selector that, detecting that it is double, sends it to the second selector. The second selector, detecting a single locomotive, sends it on a track that leaves the roundabout. In this section, we present the implementation of these structures, which are those of [8] adapted to the present tessellation.

3.2.1 The Fixed Switch, the Doubler and the Fork

We look at the fixed switch first, and then at the doubler and the fork, as the doubler is a combination of the fork and the fixed switch.

3.2.2 The Fixed Switch

As the tracks are one-way and as an active fixed switch always sends the locomotive in the same direction, no track is needed for the other direction: there is no active fixed switch. Passive fixed switches are still needed, as mentioned previously.

Figure 5 illustrates the passive fixed switch when there is no locomotive around: we say that such a configuration is *idle*. We shall again use this term in a similar situation for the other structures and for individual cells too.

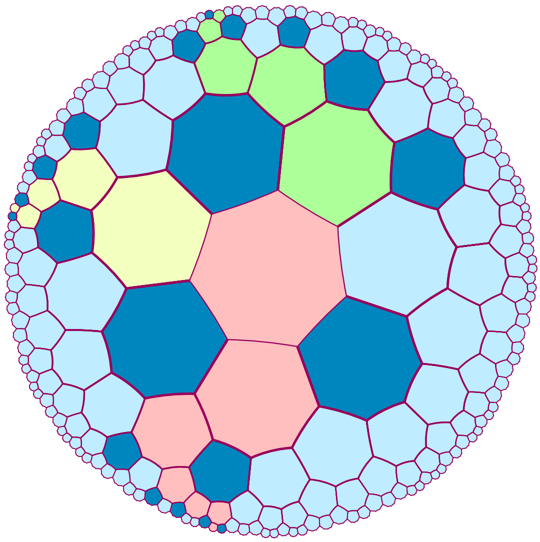


Figure 5. Idle configuration of the passive fixed switch. In yellow, the arriving path from the left; in green, the arriving path from the right; in pink, the path leaving the switch.

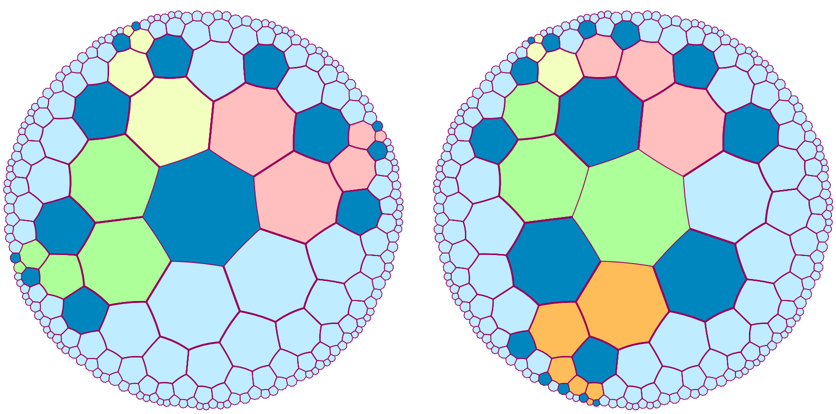
We can see that it consists of elements of the tracks that are simply assembled in the appropriate way in order to drive the locomotive to the bottom direction in the graphic, no matter from which upper side the locomotive arrived at the switch. The path followed by the locomotive to the switch is in yellow or in green until the central cell, which is pink. The path from the left-hand side, yellow in the figure, consists in this order of the cells: 29(2), 11(2), 10(2), 3(2) and 1(2). From the right-hand side, green in the figure, it consists of the cells 23(1), 9(1), 3(1), 2(1) and 1(7). Of course, 1(2) and 1(7) are neighbors of 0(0). The path followed by the locomotive from 0(0) is in pink in the figure. It consists of the following cells in this order: 0(0), 1(4), 2(4), 7(4), 8(4), 9(4) and 24(4). Note that cell 0(0) in Figure 5 is a standard element of the track with three milestones in 1(5), 1(1) and

1(3), its neighbors 2, 5 and 7, respectively. Note that 1(3) and 1(1) are milestones for cell 1(2), that 1(3) and 1(5) are milestones for 1(4) and that 2(7) and 1(1) are milestones for 1(7). Note that the milestones of 1(7) are its neighbors 3, 5 and 7.

From our description of the way the roundabout works, a passive fixed switch must be crossed by a double locomotive as well as a simple locomotive. Later, in Section 4.2.1, we shall check that the structure illustrated by Figure 5 allows those crossings.

3.2.3 The Fork and the Doubler

The fork is the structure illustrated in Figure 6(a). Note that its structure is very different from that of the tracks or of the fixed switch. The central cell 0(0) is black and two paths start from 1(1), each on one side of the central cell with respect to its axis, which crosses its side 1 and passes through the vertex that is opposite side 1; it is shared by sides 4 and 5. The paths each take two cells around 0(0) and then leave the neighborhood of cell 0(0). The cells leading to 1(1) are yellow in the figure. The left-hand track is green, consisting of the following cells, in this order: 1(2), 1(3), 3(3), 7(3) and 20(3). The right-hand track is pink. It consists of the cells 1(7), 1(6), 4(6), 5(7) and 13(7). The locomotive, a simple one, arrives through the yellow path: 28(1), 10(1), 4(1) and 1(1). From 1(1), two simple locomotives appear: one in 1(2), going along the green path; the other in 1(7), traveling along the pink path.



(a) (b)

Figure 6. Idle configurations. (a) The fork. (b) The doubler. In both of them: arrival of the locomotive through the yellow track. Then, one locomotive on the green and on the pink tracks. In the doubler: the double locomotive leaves the switch through the orange track.

The doubler is a structure that receives a simple locomotive and yields a double locomotive. The idea is to use a fork to produce two simple locomotives and then to gather them at a fixed switch in order to produce the double locomotive. The process is illustrated by Figure 6(b). The structure is inspired by that of [8], but it turns out that here it is much simpler than there. The reason is that in [8], the even number of sides compelled the author to devise a detour in order that two locomotives arrive at the same time, one after the other, at one entrance of the fixed switch. In the heptagrid, the odd number of sides allowed us to perform a simpler implementation. The odd number allows us to have two equal paths around the common milestones of the concerned elements of the tracks. It is enough to place the central cell of the fixed switch at the end of one of the paths, the green path in Figure 6(b). The graphic uses the same colors as the graphic of the fork with the same meaning. Consider the green path. Its cells are, in this order: 2(2), 1(2) and 0(0), which makes three cells. The pink path consists of the following cells, in this order: 3(1), 2(1) and 1(7). It can be seen that the cells around 0(0) are exactly the neighbors of the central cell of a fixed switch; see Figure 5. According to this description, the two simple locomotives created at the same time in 2(2) and 3(1), respectively, do not arrive at the same time at cell 0(0). When the locomotive created in 2(2) arrives at cell 0(0), the locomotive created in 3(1) is at 1(7), so that the two black cells in 1(7) and 0(0) constitute a double locomotive arriving from the right whose front is in the central cell of the fixed switch. Then the double locomotive leaves the switch through the orange path, in this order: 1(4), 2(4), 7(4), 8(4), 9(4) and 24(4). Accordingly, the structure works as expected for a doubler. Note that only elements of the track are involved.

3.2.4 The Selector

The selector is illustrated by Figure 7. This structure is less symmetric than the corresponding structure of [8], which makes another difference with that paper.

We have a yellow track through which the locomotive arrives, simple or double; both cases are possible. The track consists of the cells 25(6), 9(6), 10(6), 4(6), 1(6) and 0(0). When a simple locomotive arrives, it leaves the cell through 1(1) via the pink path, which consists of the cells 1(1), 2(1), 7(1) and 18(1). When a double locomotive arrives, a simple locomotive leaves the structure through the green path, consisting of the cells 1(4), 2(5), 5(5), 12(4) and 33(4). Both cells 1(5) and 1(7) can detect whether the locomotive is simple or double. They can do that when the front of the locomotive is in 0(0). Then, if the locomotive is double, its rear is in 1(6). Both cells 0(0) and 1(6) are neighbors of 1(5) and of 1(7) too.

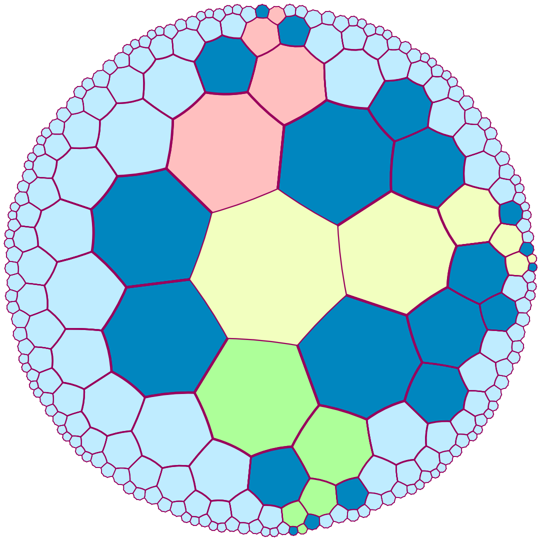


Figure 7. Idle configuration of the selector. Cells 1(7) and 1(5) detect whether the locomotive is simple or double. Arriving through the yellow track, a simple locomotive leaves through the pink track; a double locomotive leaves through the green track as a simple locomotive.

In Section 4, the rules will show the implementation of this process.

3.3 The Controller and the Sensor

In this subsection, we look at the additional structures used for the flip-flop and for the memory switch; see [6, 13] for the definitions and for the implementation in the hyperbolic plane. As explained in [6], the flip-flop and the active memory switch are implemented by using the fixed switch, the fork and a new structure we shall study in Section 3.3.1: the *controller*. The structure is illustrated by Figure 8. For the passive memory switch, we need the fork, the fixed switch and another new structure we shall study in Section 3.3.2: the *sensor*, illustrated by Figure 9.

Suitably assembled, these structures allow us to implement the required switches. Figure 10 indicates how a specific assembly using the fixed switches, the forks and the controllers only allows us to implement the flip-flop and the active memory switch: we remind the reader that due to the one-way property of the tracks, the active memory switch is split into an active one corresponding to the tracks of the active passage and a passive one corresponding to the passive crossing.

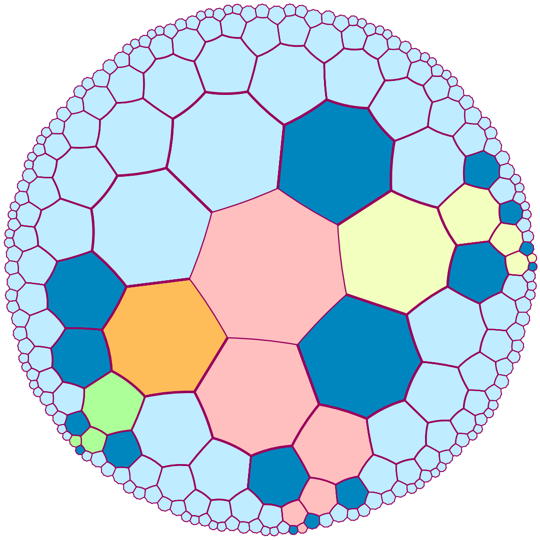


Figure 8. Idle configuration of the controller of the flip-flop and of the active memory switch. In orange, cell 1(3): the sensor that controls the working of the device. In pink, the portion of the track that is allowed when cell 1(3) is black only.

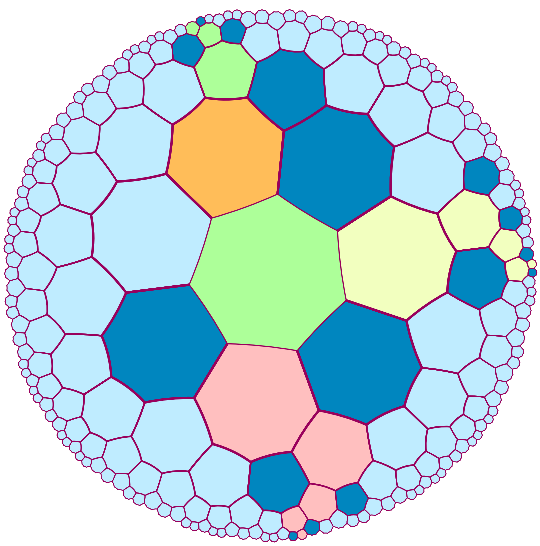


Figure 9. Idle configuration of the sensor of the passive memory switch. In yellow, cell 1(1), the sensor cell.

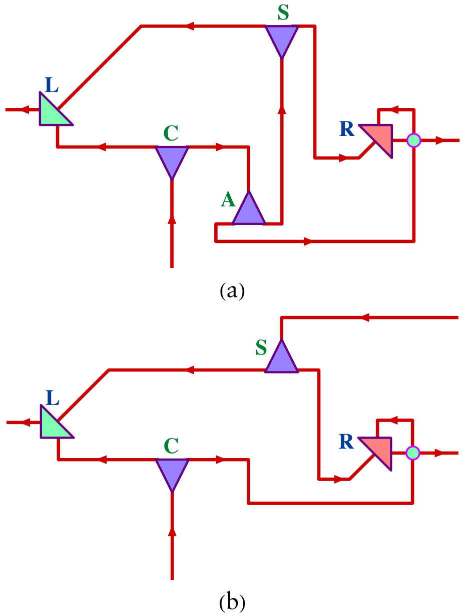


Figure 10. Assembling fixed switches, forks and controllers. (a) It yields a flip-flop. (b) An active memory switch.

In both cases, the controller is crossed by a simple locomotive and it acts in the same way: when it is black, it lets the locomotive cross the switch. When it is blank, it cancels the locomotive. This behavior is explained by the assembly itself: the arriving locomotive first crosses a fork C that yields two simple locomotives. One of them is sent to the controller L, while the other is sent to the other controller R. The controller L is black; the controller R is white: in Figure 10, it corresponds to a left-hand-side selected track. Both switches are implemented in the same way for what is the management of the selected track.

A difference occurs in the change of the selected track. In the flip-flop, the change is triggered by the passage of the locomotive. This is why the track from C to R first crosses another fork, A. One of the two simple locomotives produced by A is sent to the controller R, while the second locomotive is sent to a fork S. In both cases, the two simple locomotives sent by S go to both controllers L and R. They arrive there through another entry than that used by the locomotives sent by C. Arriving at the controller, the locomotive sent by S changes its color to the opposite one. In the active memory switch, the change is triggered by a locomotive sent from the passive memory switch: the locomotive arrives at the fork S. It has the same effect as in the case of the flip-flop. We shall see how to deal with the passive memory switch in Section 3.3.2.

3.3.1 The Controller

As shown by Figure 8, the controller sits on an ordinary cell of the track. The locomotive that runs on that track is always simple. The track consists of the yellow path that passes through 25(6), 9(6), 10(6), 4(6) and 1(6), a neighbor of 0(0), and the pink path that starts from 0(0) and that is crossed by the locomotive when the controller is black. The *color* of the controller is defined by cell 1(3), in orange in Figure 8. The pink path consists of the following cells, in this order: 0(0), 1(4), 2(5), 5(5), 12(4) and 33(4), a path already seen in previous figures. When cell 1(3) is black, then cell 0(0) is an ordinary element of the track, so that the locomotive goes on its way along the pink path, leaving the controller. If cell 1(3) is white, then cell 0(0) can no longer work as an element of the track. It remains white, which means that the locomotive is stopped at 1(6): after that, it vanishes. This implements the action of a selection in an active passage of the switch: the locomotive cannot run along a nonselected track. Here it can do it for a while, but at some point, it is stopped by the controller. Note that the occurrence of a locomotive in the structure does not change the color in 1(3). The change of color in that cell is performed by a signal that takes the view of a simple locomotive arriving through another track: 31(3), 12(3) and 4(3), the last cell being a neighbor of 1(3). When the locomotive signal arrives at 4(3), it makes cell 1(3) change its color: from white to black and from black to white.

3.3.2 The Sensor

Let us now turn to the passive memory switch whose structure is illustrated by Figure 11, taken from [7]. It assembles two fixed switches, four forks and two new structures: the sensors, the green and red small squares of the figure. As in [7], we put a fork on the tracks that arrive at the passive memory switch. One of the simple locomotives created by the forks S_1 and S_2 goes to the fixed switch F_1, from which it continues to another switch. The indication of the selected track is indicated by the sensors: as controllers, they have a *color*. When the sensor is white or black, it corresponds to the unselected or selected track, respectively, as in the case of the active switches. The other locomotive created by S_1 and S_2 goes to a sensor. Consider the case of S_1, which in the figure corresponds to the selected track because the associated sensor L is green. The sensor is white, but the locomotive coming from S_1 is stopped: there is no change to perform on the color of the sensors. Consider the case of S_2. This time, sensor R indicates that the track leading to S_2 is not selected. The locomotive sent by S_2 to R finds a black sensor. The sensor turns to white and lets the locomotive go on its way, but now, that locomotive is used like a signal sent to the other sensor L and to the controllers of the active memory switch in order to change their colors. This is why the locomotive is sent to another fork, S_3, which sends one locomotive to L and the other to the fixed switch F_2 in order to go to the active memory switch associated to the passive one. There is a similar sensor S_4 for the case when the sensor L is black. This is why the fixed switch F_2 is placed in order to collect the signal that arrives either from one side or from the other. Note the occurrence of two roundabouts in the structure, as tracks for managing the signal have to cross each other there, at some points.

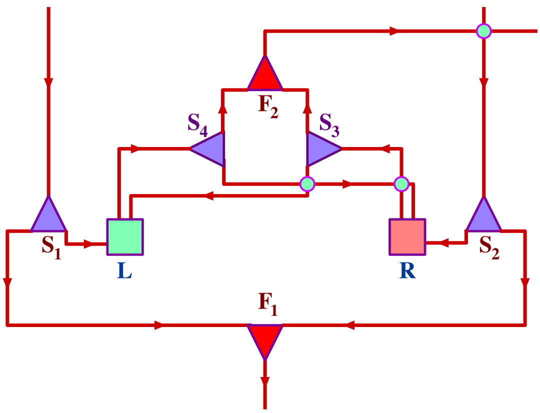


Figure 11. Organization of the passive memory switch with fixed switches, forks and sensors. Note that the sensors are not represented with the same symbol as the controllers in Figure 10.

Let us now turn to the implementation of the sensor illustrated by Figure 9. Note that the working of the sensor is different from that of the controller: in the sensor, the reaction of the locomotive according to the color is opposite to its reaction in the controller. The sensor is again installed on a track, the same as in Figure 8. The cell that plays the role of a sensor is this time cell 1(1), whose state we call the *color* of the sensor. Note that the neighborhood of that cell in Figure 9 is the same, up to rotation, as the neighborhood of cell 1(3) in Figure 8: the green path here consists of cells 26(1), 9(1) and 3(1), the last one being a neighbor of 1(1).

Figure 9 shows a very different structure for cell 0(0) compared with that of Figure 8. When the sensor is white, its neighborhood is exactly that of cell 0(0) when the controller is black: it is an ordinary element of the track, so that the locomotive goes on its way on the track. The difference in both structures allows us to implement the logic of each switch. In the case of the controller, when the locomotive goes on its way, it is the locomotive of the circuit going to another switch or to a roundabout. In the case of the sensor, the locomotive that goes on its way on the track becomes a signal sent to the other sensor and to the active switch associated to the passive switch.

We can just note that the change of color is different in the sensor: when the sensor is white, if a locomotive passes, it must become black: the signal is the locomotive itself, as will be seen in Section 4.5. This is why cell 0(0) is green in Figure 9: cell 1(1) can see the locomotive only when it is in 0(0). When the sensor is black, it has to be changed if a locomotive passed through the other sensor, which then changed from white to black. The locomotive that arrived at the formerly white sensor is sent to the still-black one in order to make it change to white. The locomotive arrives through the green path of Figure 9. As the configuration is the same around cell 1(1) of that figure as that around 1(3) in Figure 8, the change from black to white is performed.

4. Rules

The figures of Section 3 help us to establish the rules. Their application is illustrated by the figures of this section, which are taken from figures drawn by a computer program. The program wrote the PostScript files of the latter graphics (see [9]) from the computation of the application of the rules to the configurations of the various types of parts of the circuit. The computer program also established the traces of execution that contribute to checking the application of the rules.

Let us explain the format of the rules and what is allowed by the relaxation from rotation invariance. We remind the reader that a rule has the form $\underline{X}_o X_1 .. X_7 \underline{X}_n$, where X_o is the state of the cell c, X_i is the *current* state of the neighbor i of c and X_n is the *new* state of c applied by the rule. As the rules no longer observe rotation invariance, we may freely choose which is side 1 for each cell. We already indicated in Section 3.1 how we define which is neighbor 1 for the cells of the tracks. There are exceptions, namely the cells that belong to several tracks, which is typically the case for the central cell of a switch or for some of its neighbors. In particular, when a cell belongs to two tracks, side 1 is arbitrarily chosen among the two possible cases. In some places, side 1 may be chosen in order to allow the CA to apply the expected rule. Most often, the milestones have their side 1 shared with an element of the track, which also contributes to reducing the number of rules. Note that in that case there may be several possibilities, in particular when a milestone belongs to two tracks or when it belongs to several cells of the same track.

There are two types of rules. Those that keep the structure invariant when it is idle, we call *conservative*, and those that control the motion of the locomotive, we call *motion rules*. Those latter rules are applied to the cells of the tracks as well as their milestones and sometimes to the cells of the structures that may be affected by the passage of the locomotive. Next, in each subsection, we give the rules for the motion of the locomotive in the tracks, then for the fixed switch, then for the doubler and for the fork, then for the selector, then for the controller and, eventually, for the sensor. In each subsection, we also illustrate the motion of the locomotive in the structure, as well as provide a table giving traces of execution for the cells of the track involved in the crossing. The rules are numbered, and in the tables and when they are referred to in the text, we write in red the number of a rule where the new state of the cell is different from its current state.

4.1 The Rules for the Tracks

Figure 2 shows us a single element of the track. Figure 3 shows us how to assemble elements as illustrated in Figure 2 in order to constitute tracks. In Figure 3, the track is represented by the yellow cells, in this order when going from top to bottom: the cells 16(1), 6(1), 7(1), 3(1), 1(1), 1(7), 1(6), 1(5), 1(4), 3(4), 10(4), 11(4) and 29(4).

A close look at the tracks shows us at least two kinds of cells, despite all of them being three-milestoned cells, another difference with [8], where four-milestoned elements of the tracks are often present. In Figure 3, there are three-milestoned cells with milestones in their neighbors 2, 4 and 7 as, for instance, 1(6) and 3(1), and three-milestoned cells with their milestones in neighbors 3, 5 and 7, as 1(1) and 7(1), for instance. Table 4 shows us that for cells 1(6) and 3(1), rules 4, 36, 17, 25 are applied, while rules 3, 38, 40, 43 are applied for cells 1(1) and 7(1). The rules are taken from Table 2, which gives all rules used by the locomotives on the tracks.

Another assortment of the milestones is in neighbors 2, 5 and 7. Table 1 gives the motion rules corresponding to these cells and a few others as indicated by Tables 4 and 5. In that table and in the following ones, the black state corresponding to a simple locomotive is denoted by B.

Note that Table 1 gives all possible neighborhoods for three isolated milestones, requiring that neighbor 1 be blank in the idle configuration. For instance, take the positions 2, 4, 6 for the milestones.

Rule 14 is the conservative rule and rule 31 corresponds to the case of a simple locomotive being in the cell. As the locomotive always leaves the cell through neighbor 1, rule 34 applies after rule 31. Now, there are *a priori* three possible entrances for the locomotive: one is neighbor 5 as in rule 28, displayed in Table 1. Table 2 shows that neighbors 3 and 7 are also used; look at rules 63, WWBBBBWBWB and 41, WWBWBWBBBB. The same observation can be made for the other dispatches of the milestones in Table 1.

Accordingly, for the display 2, 4, 7, besides rule 36 corresponding to an entrance through neighbor 3, rule 41 again corresponds to another one through neighbor 6; it is now read as WWBWBWBBBB, and rule 45 WWBWBBWBB corresponds to yet another one through neighbor 5. This can be repeated for the other neighbors; see Table 3, where the number in brackets indicates the neighbor through which the locomotive enters. Rule 71 will be used later, in the fixed switch.

2, 4, 6		2, 4, 7		2, 5, 7		3, 5, 7	
14	WWBWBWBWW	4	WWBWBWWBW	7	WWBWWBWBW	3	WWWBWBWBW
28	WWBWBBBWB	36	WWBBBWWBB	16	WWBWWBBBB	38	WWWBWBBBB
31	BWBWBWBWW	17	BWBWBWWBW	24	BWBWWBWBW	40	BWWBWBWBW
34	WBBWBWBWW	25	WBBWBWWBW	29	WBBWWBWBW	43	WBWBWBWBW

Table 1. The motion rules for a simple locomotive.

from bottom to top: simple locomotive							
1	WWWWWWWWW	13	WBWWWWWWW	25	WBBWBWWBW	37	BWWWBWWWB
2	BWWWWWWWB	14	WWBWBWBWW	26	WBWWWWWBW	38	WWWBWBBBB
3	WWWBWBWBW	15	BWWBWWWWB	27	WWWBWWWWW	39	BWWWWBWWB
4	WWBWBWWBW	16	WWBWWBBBB	28	WWBWBBBWB	40	BWWBWBWBW
5	WWWBWWWBW	17	BWBWBWWBW	29	WBBWWBWBW	41	WWBWBWBBB
6	WBWWWWBWW	18	WBBWWWWWW	30	WBWWWWBBW	42	WBBBWWWWW
7	WWBWWBWBW	19	BBWWWWWWB	31	BWBWBWBWW	43	WBWBWBWBW
8	WWBWWWWBW	20	WBBWWWWBW	32	WWBBWBWBB	44	BWWWWWBWB
9	WWBWWWWWW	21	WWBBWWWWW	33	WWBBWWWBW	45	WWBWBBWBB
10	WWWWWWWBW	22	BWBWWWWWB	34	WBBWBWBWW		
11	WWWWBWBWW	23	BWWWWWWBB	35	WBWBWWWBW		
12	WBWBWWWWW	24	BWBWWBWBW	36	WWBBBWWBB		
from bottom to top: double locomotive							
46	BWBBWWWWB	51	BBBWWBWBW	56	BWBBBWWBB	60	BBWBWBWBW
47	BWBWWBBBB	52	BWBWBBBWB	57	BWWBBWWWB	61	BWBWBWBBB
48	BBBWBWWBW	53	BBBWBWBWW	58	BWWWBBWWB	62	BWWWWWBBB
49	BBBWWWWWB	54	BWBBWBWBB	59	BWWBWBBBB		
50	BBWWWWWBB	55	WBBBWWWBW				
from top to bottom, when the locomotive is:							
simple				double			
63	WWBBBWBWB	64	BWWWWBBWB	65	BWBBBWBWB	66	BWBWBBWBB

Table 2. Rules managing the motion of a locomotive on the tracks.

3, 5, 7 :	38	WWWBWBBBB	[6]	71	WWWBBBWBB	[4]	32	WWBBWBWBB	[2]
2, 5, 7 :	16	WWBWWBBBB	[6]	45	WWBWBBWBB	[4]	32	WWBBWBWBB	[3]
2, 4, 7 :	36	WWBBBWWBB	[3]	41	WWBWBWBBB	[6]	45	WWBWBBWBB	[5]
2, 4, 6 :	28	WWBWBBBWB	[5]	63	WWBBBWBWB	[3]	41	WWBWBWBBB	[7]

Table 3. The other rules involved for the motion of a simple locomotive.

	11_4	10_4	3_4	1_4	1_5	1_6	1_7	1_1	3_1	7_1	6_1
1	25	24	16	14	7	4	4	3	4	3	7
2	4	29	24	28	7	4	4	3	4	3	7
3	4	7	29	31	32	4	4	3	4	3	7
4	4	7	7	34	24	36	4	3	4	3	7
5	4	7	7	14	29	17	36	3	4	3	7
6	4	7	7	14	7	25	17	38	4	3	7
7	4	7	7	14	7	4	25	40	41	3	7
8	4	7	7	14	7	4	4	43	17	38	7
9	4	7	7	14	7	4	4	3	25	40	45

Table 4. Execution of rules 1 to 45: motion along the tracks from bottom to top for a simple locomotive.

	6_1	7_1	3_1	1_1	1_7	1_6	1_5	1_4	3_4	10_4	11_4
1	34	24	63	7	7	7	3	7	4	4	7
2	14	29	31	32	7	7	3	7	4	4	7
3	14	7	34	24	16	7	3	7	4	4	7
4	14	7	14	29	24	16	3	7	4	4	7
5	14	7	14	7	29	24	38	7	4	4	7
6	14	7	14	7	7	29	40	45	4	4	7
7	14	7	14	7	7	7	43	24	36	4	7
8	14	7	14	7	7	7	3	29	17	36	7

Table 5. Execution of rules 1 to 45: motion along the tracks from top to bottom for a simple locomotive.

Call the rules of Table 1 for a given neighborhood, the conservative rule, the front rule, the cell rule and the witness rule, the names being self-explanatory.

Table 1 also shows us an interesting feature: the neighborhood of rule 36 is WBBBWWB and that of rule 29 is BBWWBWB. It is not difficult to see that we can pass from one neighborhood to the other by a circular permutation. Accordingly, rule 36 and rule 29 are not rotationally compatible: rule 36 is a front rule; rule 29 is a witness rule. The other conclusion we can draw from the comments regarding other variants of the rule allowing the locomotive to enter the cell is that this number of variants gives us an important flexibility for devising tracks that go from one tile to another. In the example of Figure 3, we can see that the track going to 1(6) can also go to 2(6), 3(6) or 4(6). Figure 12 shows us how to proceed to continue a path to the daughters of an already reached element.

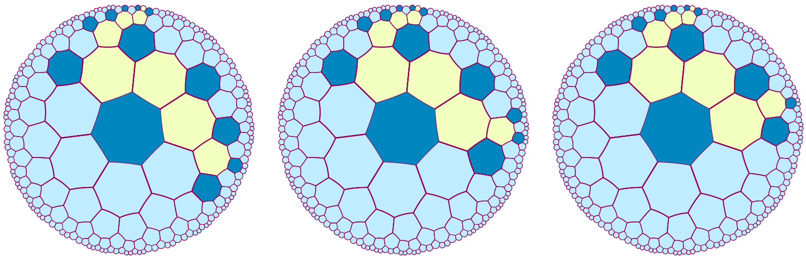


Figure 12. Element of the tracks: in yellow, the elements of the track where the locomotive passes.

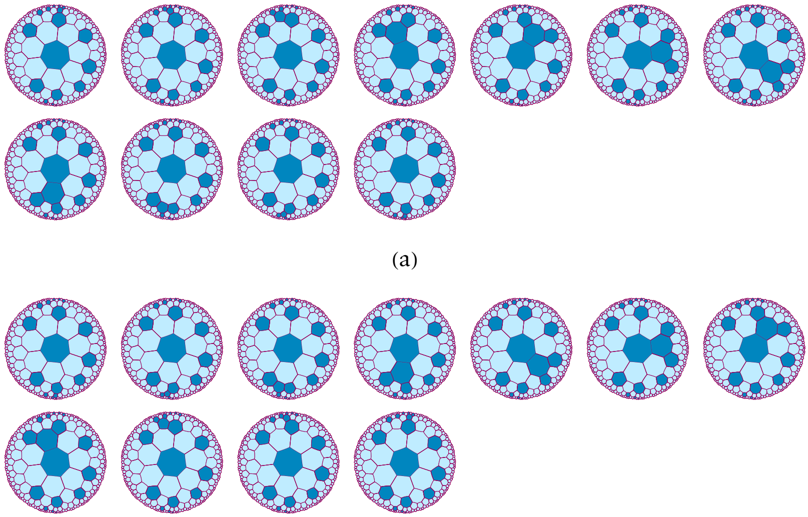
In Figure 3, the neighborhood of 1(6) is of the type 2, 5, 7. In the graphics of Figure 12, we have the following neighborhoods:

1(6):	2, 4, 6	3, 5, 7	2, 4, 7
sons:	2(6)	3(6)	4(6)
	2, 4, 7	2, 4, 7	2, 5, 7
	2, 5, 7	2, 5, 7	3, 5, 7
			2, 4, 6

It is worth noticing that many rules appearing in Table 4, showing the rules used when a simple locomotive goes up along the tracks illustrated by Figure 3, also appear in Table 5, which displays the rules used when the same locomotive goes down, assuming that the sides 1 have been changed in order to allow the motion from top to bottom on the same cells. Of course, in the circuit, for any cell, side 1 is fixed once and for all.

The application of the rules so far considered for the motion of a simple locomotive is illustrated by Figure 13, where the simple locomotive runs over the tracks illustrated by Figure 3 in one direction, and then in the other.

Before turning to the rules for a double locomotive, note that in the conservative rules for the elements of the tracks, the neighborhoods are rotated forms of each other.



(a)

(b)

Figure 13. A simple locomotive along a track: (a) in one direction; (b) in the opposite one.

The rules for the double locomotive are displayed in Table 6. They can be derived from the previous rules as follows. The conservative and the front rules are the same: at those times, the cell does not know whether the locomotive is simple or double. Once the front is in the cell, the cell rule cannot be applied, as the rear is seen at the place where the front was one step previously. Accordingly, the cell rule is replaced by two new rules: the rear rule, which makes the rear of the locomotive enter the cell, and the clearing rule, which makes it leave the cell. The witness rule is the same as for a simple locomotive: when the rear is in the next cell on the track, the cell cannot remember that the locomotive was double. Note that the rear and clearing rules are obtained from the front and witness rules, respectively, by changing the current state from W to B. Table 6 gives the rules applied in the respective neighborhoods considered in Table 1. That table has the same property as Table 1. Other entries are possible for the double locomotive. Table 3 indicates to us for each neighborhood the sequence of rules constituted, in this order, by the front, the rear and the clearing rules. Also, brackets indicate, after each triple, which is the neighbor through which the locomotive enters the cell. In that table, the locomotive is marked as B. In Table 6, the front of the locomotive is B and its rear is B. Table 7 indicates all possible neighborhoods with the entry of the front of the locomotive in brackets. Note the same phenomenon as in Table 1: a few pairs of rear and clearing rules are the same for different neighbors. In fact, that feature appears when two neighbors differ by one place, for example 2, 4, 7 and 2, 5, 7. Clearly, the positions of the black neighbors are the same for the front and rear rules when the entrance is neighbor 5 and 4, respectively. Note that in Table 6, rule 34 occurs with B, while in Table 1, the same rule occurs with B at the same place. It corresponds to the fact that the single cell of a simple locomotive is both the rear and the front.

	2, 4, 6		2, 4, 7		2, 5, 7		3, 5, 7
14	WWBWBWBWW	4	WWBWBWWBW	7	WWBWWBWBW	3	WWWBWBWBW
28	WWBWBBBWB	36	WWBBBWWBB	16	WWBWWBBBB	38	WWWBWBBBB
52	BWBWBBBWB	56	BWBBBWWBB	47	BWBWWBBBB	59	BWWBWBBBB
53	BBBWBWBWW	48	BBBWBWWBW	51	BBBWWBWBW	60	BBWBWBWBW
34	WBBWBWBWW	25	WBBWBWWBW	29	WBBWWBWBW	43	WBWBWBWBW

Table 6. The motion rules for a double locomotive.

3, 5, 7 :	38, 59, 60 [6]	71, 72, 60 [4]	32, 54, 60 [2]
2, 5, 7 :	16, 47, 51 [6]	45, 66, 51 [4]	32, 53, 51 [3]
2, 4, 7 :	36, 56, 48 [3]	41, 61, 48 [6]	45, 67, 48 [5]
2, 4, 6 :	28, 52, 53 [5]	63, 65, 53 [3]	41, 61, 53 [7]

Table 7. The other rules involved for the motion of a double locomotive.

Table 6 contains 20 rules. The cell rule is different in Table 1 so that, together, those tables contain 24 rules. Table 7 brings in 16 new rules. Accordingly, we have 40 rules for the elements of the track only. There are other rules concerning the milestones: $\underline{B}W^{\alpha}BW^{\beta}\underline{B}$ with $\alpha, \beta \geq 0$ and $\alpha + \beta = 6$ together with $\alpha < 5$ when a simple locomotive moves; see rules 19, 22, 15, 37 and 39. For a double locomotive, we have all possible rules of the form $\underline{B}W^{\alpha}BBW^{\beta}\underline{B}$ with $\alpha, \beta \geq 0$ and $\alpha + \beta = 5$ together with $\underline{B}BW^{5}B\underline{B}$; see rules 49, 46, 57, 58, 64, 62 and 50. We also have to append rule 1, $\underline{W}W^{7}\underline{W}$, the conservative rule of the blank cells that have no black cell among their neighbors, as well as rule 2, $\underline{B}W^{7}\underline{B}$, which is the conservative rule of the milestones of the elements of the track. Table 8 illustrates the use of many of those rules, together with some others for four cells: 4(1), a white cell, and three milestones: 2(1), 0(0) and 4(4). Cell 4(1) illustrates a situation when a white cell can see two consecutive elements of the tracks. We use the same color conventions as in Tables 1 and 6 for the front and for the rear of a locomotive.

It can be noted that the rules of Table 8 do not change the current state of the cell: it is conformal with the role of witness devoted to these cells. We also can remark that the change of direction in the motion boils down to a change in the order of application of the rules. We also can see the change in the display of the colors in the rules attached to the motion of a double locomotive.

4(1):	simple ↑		double ↑		simple ↓		double ↓	
	5	WWWBWWWBW	5	WWWBWWWBW	5	WWWBWWWBW	5	WWWBWWWBW
	35	WBWBWWWBW	35	WBWBWWWBW	33	WWBBWWWBW	33	WWBBWWWBW
	33	WWBBWWWBW	55	WBBBWWWBW	35	WBWBWWWBW	55	WBBBWWWBW
	5	WWWBWWWBW	33	WWBBWWWBW	5	WWWBWWWBW	35	WBWBWWWBW
			5	WWWBWWWBW			5	WWWBWWWBW
2(1):	simple ↑		double ↑		simple ↓		double ↓	
	2	BWWWWWWWB	2	BWWWWWWWB	2	BWWWWWWWB	2	BWWWWWWWB
	22	BWBWWWWWB	22	BWBWWWWWB	39	BWWWWBWWB	39	BWWWWBWWB
	19	BBWWWWWWB	49	BBBWWWWWB	44	BWWWWWBWB	64	BWWWWBBWB
	23	BWWWWWWBB	50	BBWWWWWBB	23	BWWWWWWBB	62	BWWWWWBBB
	44	BWWWWWBWB	62	BWWWWWBBB	19	BBWWWWWWB	50	BBWWWWWBB
	39	BWWWWBWWB	64	BWWWWBBWB	22	BWBWWWWWB	49	BBBWWWWWB
	2	BWWWWWWWB	39	BWWWWBWWB	2	BWWWWWWWB	22	BWBWWWWWB
			2	BWWWWWWWB			2	BWWWWWWWB
0(0):	simple ↑		double ↑		simple ↓		double ↓	
	2	BWWWWWWWB	2	BWWWWWWWB	2	BWWWWWWWB	2	BWWWWWWWB
	19	BBWWWWWWB	19	BBWWWWWWB	39	BWWWWBWWB	39	BWWWWBWWB
	22	BWBWWWWWB	49	BBBWWWWWB	37	BWWWBWWWB	58	BWWWBBWWB
	15	BWWBWWWWB	46	BWBBWWWWB	15	BWWBWWWWB	57	BWWBBWWWB
	37	BWWWBWWWB	57	BWWBBWWWB	22	BWBWWWWWB	46	BWBBWWWWB
	39	BWWWWBWWB	58	BWWWBBWWB	19	BBWWWWWWB	49	BBBWWWWWB
	2	BWWWWWWWB	39	BWWWWBWWB	2	BWWWWWWWB	19	BBWWWWWWB
			2	BWWWWWWWB			2	BWWWWWWWB
4(4):	simple ↑		double ↑		simple ↓		double ↓	
	2	BWWWWWWWB	2	BWWWWWWWB	2	BWWWWWWWB	2	BWWWWWWWB
	15	BWWBWWWWB	15	BWWBWWWWB	23	BWWWWWWBB	23	BWWWWWWBB
	22	BWBWWWWWB	46	BWBBWWWWB	19	BBWWWWWWB	50	BBWWWWWBB
	19	BBWWWWWWB	49	BBBWWWWWB	22	BWBWWWWWB	49	BBBWWWWWB
	23	BWWWWWWBB	50	BBWWWWWBB	15	BWWBWWWWB	46	BWBBWWWWB
	2	BWWWWWWWB	23	BWWWWWWBB	2	BWWWWWWWB	15	BWWBWWWWB
			2	BWWWWWWWB			2	BWWWWWWWB

Table 8. Rules for cells witnessing the motion on the tracks.

Tables 9 and 10 show which instructions are applied to the cells of the track when a double locomotive passes: from bottom to top in Table 9, from top to bottom in Table 10. Figure 14 illustrates the application of the rules when a double locomotive runs over the tracks illustrated by Figure 3.

We conclude this section by a remark: in [7] and in [8], the tracks were implemented by using both three- and four-milestoned cells as elements of the tracks. Here we succeeded in using three-milestoned cells only. The large number of motion rules allowed us to assemble such elements in very efficient structures. Also note the importance of the choice of side 1. As an example, for cell 4(4), its side 1 is shared with 3(4).

	11_4	10_4	3_4	1_4	1_5	1_6	1_7	1_1	3_1	7_1	6_1
1	25	51	47	28	7	4	4	3	4	3	7
2	4	29	51	52	32	4	4	3	4	3	7
3	4	7	29	53	54	36	4	3	4	3	7
4	4	7	7	34	51	56	36	3	4	3	7
5	4	7	7	14	29	48	56	38	4	3	7
6	4	7	7	14	7	25	48	59	41	3	7
7	4	7	7	14	7	4	25	60	61	38	7
8	4	7	7	14	7	4	4	43	48	59	45

Table 9. Execution of rules 1 to 66: the double locomotive on the tracks from bottom to top.

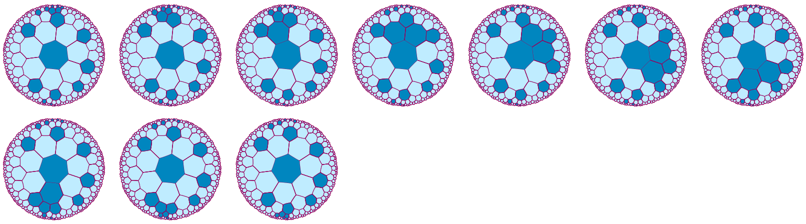
	6_1	7_1	3_1	1_1	1_7	1_6	1_5	1_4	3_4	10_4	11_4
1	34	51	65	32	7	7	3	7	4	4	7
2	14	29	53	54	16	7	3	7	4	4	7
3	14	7	34	51	47	16	3	7	4	4	7
4	14	7	14	29	51	47	38	7	4	4	7
5	14	7	14	7	29	51	59	45	4	4	7
6	14	7	14	7	7	29	60	66	36	4	7
7	14	7	14	7	7	7	43	51	56	36	7
8	14	7	14	7	7	7	3	29	48	56	16

Table 10. Execution of rules 1 to 66: the double locomotive on the tracks from top to bottom.

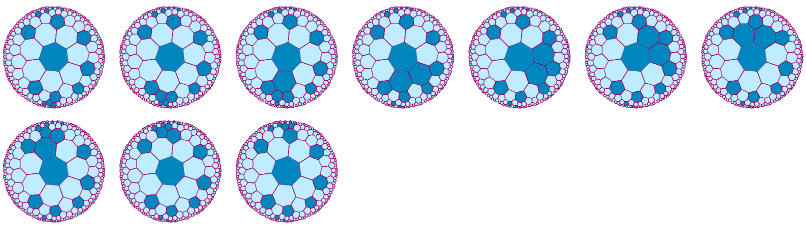
Figures 13 and 14 illustrate the motions described by Tables 4, 5, 9 and 10. They are produced by PostScript commands computed by a computer program. The program applies the rules given in Table 2 and the other tables of rules given in the following subsections to the cellular automaton. From those calculations, it computes the position of the cell(s) representing the locomotive on the tracks. The same program did the same for the various configurations we shall further investigate.

4.2 The Rules for the Fixed Switch, the Fork and the Doubler

We now turn to the study of the fixed switch, the fork and the doubler. Table 11 gives new rules that are used for the crossing of those structures, together with already used rules. We start our study with the fixed switch, which is a passive structure as noted in Section 3.2.



(a)



(b)

Figure 14. A double locomotive along a track: (a) in one direction; (b) in the opposite one.

fixed:	from the left						from the right
simple		double		simple		double	
67	WWBWWWBWW	70	WBBWWWBBW	71	WWWBBBWBB	72	BWWBBBWBB
68	WWBWWWBBW						
69	WBBWWWBWW						
doubler:				fork:			
73	BWWBWBWWB			76	WWWWWBWBW		
74	WBBBWBWBW			77	WWBWBWWWW		
75	BWBWWWBWB						

Table 11. Rules for the crossing of a fixed switch, a doubler and a fork.

4.2.1 The Fixed Switch

As can already be seen in Figure 5, the structure is mainly constituted by elements of the tracks assembled in an appropriate way. In particular, the central cell is a three-milestoned cell, as in the elements of the tracks. Its neighborhood is a rotated image of any neighborhood of a cell of the tracks, as studied in Section 4.1. The rules for the fixed switch are displayed by Table 11.

Table 13 shows the instructions applied during the crossing of a simple locomotive through the fixed switch from the left and from the right, for the cells of the tracks only. We can notice that the rules involved in the table are those of the motion of the locomotive on the tracks. The information of those tables is completed by that of

Table 12, showing the rules applied at cell 1(1), which has a view on each track arriving at the central cell. For that latter table, we again used the colors distinguishing the front from the rear in a double locomotive. Note that in rule 23, the color of the locomotive depends on whether it represents the front or the rear of a simple or double locomotive, respectively.

simple, left		double, left		simple, right		double, right	
2	BWWWWWWWB	2	BWWWWWWWB	2	BWWWWWWWB	2	BWWWWWWWB
44	BWWWWWBWB	44	BWWWWWBWB	15	BWWBWWWWB	15	BWWBWWWWB
23	BWWWWWWBB	62	BWWWWWBBB	22	BWBWWWWWB	46	BWBBWWWWB
2	BWWWWWWWB	23	BWWWWWWBB	19	BBWWWWWWB	49	BBBWWWWWB
		2	BWWWWWWWB	23	BWWWWWWBB	50	BBWWWWWBB
				2	BWWWWWWWB	23	BWWWWWWBB
						2	BWWWWWWWB

Table 12. Rules for cell 1(1), which witnesses the motion on the tracks.

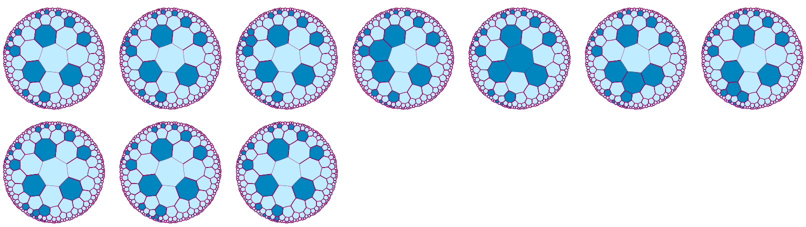
	simple locomotive												
	from the left				from the right				leaving the cell				
	10_2	3_2	1_2	0_0	3_1	2_1	1_7	0_0	1_4	2_4	7_4	8_4	9_4
1	24	38	7	7	40	41	3	7	7	14	7	4	7
2	29	40	45	7	43	17	38	7	7	14	7	4	7
3	7	43	24	16	3	25	40	45	7	14	7	4	7
4	7	3	29	24	3	4	43	24	16	14	7	4	7
5	7	3	7	29	3	4	3	29	24	63	7	4	7
6	7	3	7	7	3	4	3	7	29	31	32	4	7
7	7	3	7	7	3	4	3	7	7	34	24	36	7
8	7	3	7	7	3	4	3	7	7	14	29	17	16
	double locomotive												
	from the left				from the right				leaving the cell				
	10_2	3_2	1_2	0_0	3_1	2_1	1_7	0_0	1_4	2_4	7_4	8_4	9_4
1	51	59	45	7	60	61	38	7	7	14	7	4	7
2	29	60	66	16	43	48	59	45	7	14	7	4	7
3	7	43	51	47	3	25	60	66	16	14	7	4	7
4	7	3	29	51	3	4	43	51	47	63	7	4	7
5	7	3	7	29	3	4	3	29	51	65	32	4	7
6	7	3	7	7	3	4	3	7	29	53	54	36	7
7	7	3	7	7	3	4	3	7	7	34	51	56	16
8	7	3	7	7	3	4	3	7	7	14	29	48	47

Table 13. Execution of the rules for the fixed switch when a locomotive crosses the switch: upper half, simple locomotive; lower half, double locomotive.

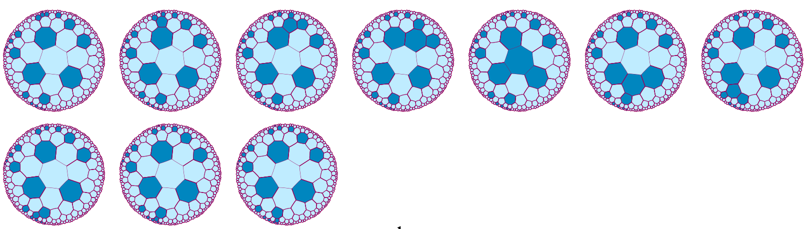
Table 13 shows the rules applied to the elements of the tracks traversed by the locomotive when it crosses the switch. Table 13 deals with the locomotive, whether it is simple or double. For each side of the switch, the table indicates the group of cells involved by the arrival of the locomotive at the central cell.

We can see that the rules applied to the central cell have already been seen in Section 4.1. The rules for a simple locomotive coming from the left are those of cell 10(4) in Table 4. For a double locomotive from the left, the rules appear in Tables 6 and 7 for cells whose neighborhood is 2, 5, 7. For a simple locomotive coming from the right, the front rule used here is in Table 3 with the neighborhood 2, 5, 7. When a double locomotive comes from the right, Tables 6 and 7 indicate the corresponding rules.

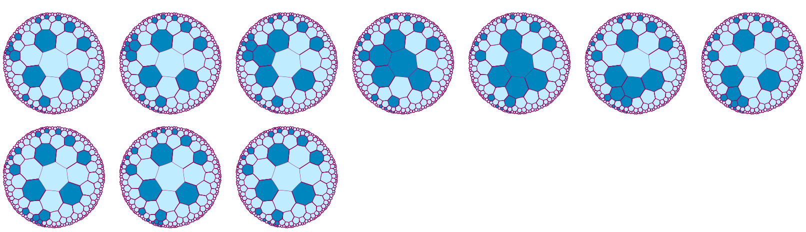
Figure 15 illustrates the motion of the locomotive for the four motions we have to consider for the fixed switch. The figure illustrates what was indicated in Table 13 for such motions.



(a)

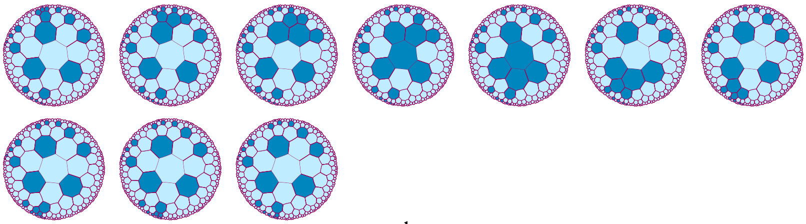


(b)



(c)

Figure 15. (*continues*)



(d)

Figure 15. Illustration of the motion of a locomotive through a fixed switch.

4.2.2 The Rules for the Doubler and the Fork

As is clear from Table 11, only a few rules are needed by the doubler and the fork. As the doubler contains both the fork and the fixed switch, Table 11 displays the three additional rules required by the doubler before the two rules required by the fork, as tested in the configuration of Figure 6(a). Here, we distinguish between the two locomotives created by cell 4(1) by giving them colors: green for the locomotive that will follow the green path; dark pink for the one that will go along the pink path.

In Table 14, we give the rules used by the doubler when the locomotives cross cells 4(1), 1(1) and 0(0). Note that when the locomotive enters cell 4(1), at the next time, two locomotives leave the cell, which is witnessed by the cell; see rule 74 in Table 14. Cell 1(1) also witnesses the duplication, as shown by rules 37, 73, 75, 50 and 23. That last rule already appeared for cell 1(1) in the passive fixed switch, witnessing that the rear of the new double locomotive leaves the neighborhood of the cell. Note that rule 50 witnesses the junction of the two simple locomotives into a double locomotive. Note that cell 4(1) has applied a sequence of rules that differs from a sequence indicated in Section 4.1 by the witness rule: instead of rule 29 as in the motion rules, we have here rule 74, as the cell can see two locomotives created in its neighbors 1 and 3. In cell 0(0), rule 66 witnesses the junction of the two simple locomotives into a double locomotive. This is why in the sequence of rules 7, 16, 47, 51 and 29 in the crossing of a cell by a double locomotive (see the rules for cell 1(6) in Table 10) rule 47 is replaced by rule 66 as the occurrence of the second cell, the rear, appears from the side that is opposite to the expected one. Figure 16 illustrates the motion of the locomotive for the doubler and the fork.

Table 15 shows us the rules of Table 11 that are applied by the cells that constitute the track in a doubler and then in a fork. These executions can also be followed on Figure 16(a) for the doubler and on Figure 16(b) for the fork.

4(1):		0(0):		1(1):	
7	WWBWWBWBW	7	WWBWWBWBW	2	BWWWWWWWB
16	WWBWWBBBB	16	WWBWWBBBB	37	BWWWBWWWB
24	BWBWWBWBW	66	BWBWBBWBB	73	BWWBWBWWB
74	WBBBWBWBW	51	BBBWWBWBW	75	BWBWWWBWB
7	WWBWWBWBW	29	WBBWWBWBW	50	BBWWWWWBB
		7	WWBWWBWBW	23	BWWWWWWBB
				2	BWWWWWWWB

Table 14. Rules for cells 4(1), 0(0) and 1(1), which witness the motion of the locomotives in the doubler.

doubler

	12_4	4_1	2_2	1_2	0_0	1_7	2_1	3_1	1_4	2_4	7_4	8_4
1	25	24	36	4	7	3	7	41	7	14	7	4
2	4	74	17	36	7	3	16	17	7	14	7	4
3	4	7	25	17	16	38	24	25	7	14	7	4
4	4	7	4	25	66	60	29	4	16	14	7	4
5	4	7	4	25	51	43	7	4	47	63	7	4
6	4	7	4	4	29	3	7	4	51	65	32	4
7	4	7	4	4	7	3	7	4	29	53	54	36
8	4	7	4	4	7	3	7	4	7	34	51	56

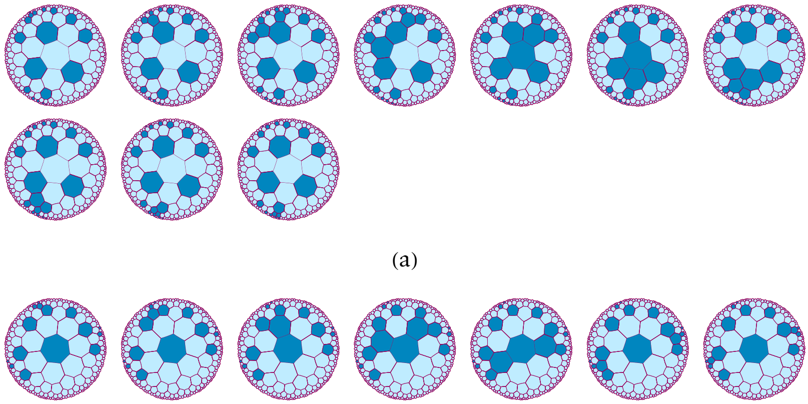
fork

	10_1	4_1	1_1	1_2	1_3	3_3	7_3	1_7	1_6	4_6	5_7
1	29	17	16	4	7	7	4	4	4	4	7
2	7	25	24	36	7	7	4	41	4	4	7
3	7	4	74	17	16	7	4	17	36	4	7
4	7	4	7	25	24	16	4	25	17	36	7
5	7	4	7	4	29	24	36	4	25	17	16

Table 15. Upper, lower part: execution of the rules for the doubler and the fork, respectively, corresponding to the illustrations of Figure 3.

4.3 The Rules for the Selector

The selector is the last structure we need to implement roundabouts. The new rules needed by the structure are given in Table 16, while the execution of the rules used by the crossing of a locomotive is given in Table 18: the left- and right-hand side subtables give the rules used by a simple and double locomotive, respectively. Figure 17 illustrates both situations.



(a)

(b)

Figure 16. (a) Illustration of the crossing of: (a) the doubler and (b) the fork, by the locomotive.

simple locomotive							
78	WBWBBWBWW	85	BBWBWWWBB	92	BBWBBWBWW	99	WBBWWBBWW
79	WBBWBBBWW	86	WBWBBWBBB	93	BBWWWBBWW	100	BWWBWWBBB
80	BWBBWWBBB	87	BWWWWBBBB	94	WBBBBBBWW	101	BWWBBWWBB
81	BBBWBWWWB	88	BBBWBBBWW	95	BBWBBWWWB	102	BWBWWWWBB
82	BBBWWBWWB	89	BBBBWWBBB	96	WBBBBBWWW	103	BWWBBWBWB
83	WBBWBBBBB	90	BWBBBWWWB	97	BWWWBWWBW	104	WWBBBWWWW
84	BBBWWWBWB	91	BBBWWWWBB	98	WWBWWBBWB		
double locomotive							
105	BBBWBBWWB	109	BBBBWWWBB	113	BBBBBWWWW	117	WWWBBWWBB
106	BBBWWBBWB	110	BBWBBWBBW	114	WWBBBBBWW	118	WBWWBBBWW
107	BBBWBBBBB	111	BBWWWBBBB	115	BWBWWBWWW	119	BBWWWBWWB
108	BBBWWWBBB	112	BBBBBBBWW	116	BWBWWBBWB	120	BWWBWBBWB

Table 16. Rules for the locomotive through the selector.

In both subtables of Table 18, we can see that the track leading the locomotive to the selector makes use of motion rules examined in Section 4.1, cell 1(6) excepted. That cell, which constitutes the entrance to the selector, has a specific neighborhood involving five milestones.

Among them, cells 1(7) and 1(5), which constitute the sensors of the selector: they detect whether a simple or a double locomotive arrived at cell 0(0). Table 17 shows the rules applied at cells 1(7), 1(5), 1(6) and 0(0). For each cell, the table gives the rules when a simple locomotive arrives and then when a double locomotive arrives. Note that W indicates that the cell 1(7) or 1(5) became white for one time in order to cancel the locomotive prepared for the corresponding path.

1(7) simple		1(7) double		1(5) simple		1(5) double	
57	BWWBBWWWB	57	BWWBBWWWB	64	BWWWWBBWB	64	BWWWWBBWB
90	BWBBBWWWB	90	BWBBBWWWB	87	BWWWWBBBB	87	BWWWWBBBB
95	BBWBBWWWB	113	BBBBBWWWW	93	BBWWWBBWW	111	BBWWWBBBB
101	BWWBBWWBB	117	WWWBBWWBB	98	WWBWWBBWB	116	BWBWWBBWB
103	BWWBBWBWB	57	BWWBBWWWB	64	BWWWWBBWB	120	BWWBWBBWB
57	BWWBBWWWB					64	BWWWWBBWB

1(6) simple		1(6) double		0(0) simple		0(0) double	
79	WBBWBBBWW	79	WBBWBBBWW	78	WBWBBWBWW	78	WBWBBWBWW
83	WBBWBBBBB	83	WBBWBBBBB	86	WBWBBWBBB	86	WBWBBWBBB
88	BBBWBBBWW	107	BBBWBBBBB	92	BBWBBWBWW	110	BBWBBWBBW
94	WBBBBBBWW	112	BBBBBBBWW	96	WBBBBBWWW	114	WWBBBBBWW
99	WBBWWBBWW	118	WBWWBBBWW	78	WBWBBWBWW	78	WBWBBWBWW
79	WBBWBBBWW	79	WBBWBBBWW				

Table 17. Rules for cells 1(7) and 1(5), 1(6) and 0(0), which witness the motion of the locomotives in the selector.

simple locomotive

	9_6	10_6	4_6	1_6	0_0	1_1	2_1	7_1	1_7	1_5
1	29	17	36	79	78	7	4	7	57	64
2	7	25	17	83	78	7	4	7	57	64
3	7	4	25	88	86	7	4	7	90	87
4	7	4	4	94	92	16	4	7	95	93
5	7	4	4	99	96	24	36	7	101	98
6	7	4	4	79	78	29	17	16	103	64
7	7	4	4	79	78	7	25	24	57	64

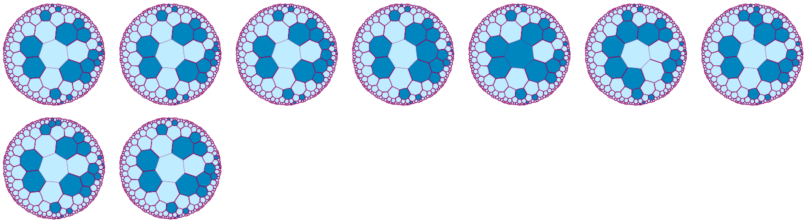
double locomotive

	9_6	10_6	4_6	1_6	0_0	1_4	2_5	5_5	12_4	1_7	1_5
1	29	48	56	83	78	4	7	7	4	57	64
2	7	25	48	107	86	4	7	7	4	90	87
3	7	4	25	112	110	36	7	7	4	113	111
4	7	4	4	118	114	17	16	7	4	117	116
5	7	4	4	79	78	25	24	16	4	57	120
6	7	4	4	79	78	4	29	24	36	57	64
7	7	4	4	79	78	4	7	29	17	57	64

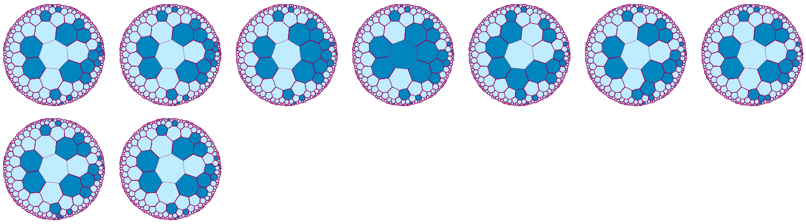
Table 18. Execution of the rules for a locomotive passing through the selector.

Figure 17 illustrates the motion of the locomotive in the selector, whether it is simple, Figure 17(a); or double, Figure 17(b).

At this point, it is important to point out the fact that for cell 1(6), side 1 is not shared with 0(0) but with a milestone of 1(6), namely 2(7). In the same way, for 0(0), side 1 is not shared with a cell of the tracks, it is also shared with a milestone: cell 1(7). The reason for these choices lies in the fact that the idle neighborhood of 1(6) coincides with the rotated form of a neighborhood of 0(0) in the selector: this can be seen with rules 79 and 86, whose neighborhoods are rotated forms of each other and which, consequently, are rotationally incompatible.



(a)



(b)

Figure 17. Illustration of the crossing of the selector: (a) by a simple locomotive; (b) by a double locomotive.

4.4 The Rules for the Controller

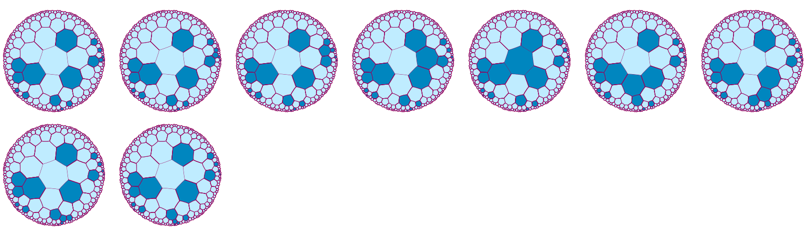
Let us now consider the rules for the controller of the active switches. The rules are displayed by Table 19. As mentioned in the table itself, the two columns on the left-hand side deal with the passage of the locomotive, while the last column deals with the change of color of the controller. We remind the reader that the color of the controller is the color of cell 1(3) in Figure 8. Table 20 gives the execution trace of the crossing of a black controller by the locomotive. The first row of Figure 18 together with the first two figures of the second row illustrates the crossing of a black controller by the locomotive. Table 22 indicates the rules that are applied when the locomotive arrives at a white controller and those that are applied when the signal for changing its color arrives at the controller. Starting from the third graphic, the second row of Figure 18 illustrates the motion of the locomotive through a white controller. Figure 19 illustrates the change of color of the controller when it is reached by the appropriate signal.

passage of the locomotive				signal	
black		white		B → W	
121	WWWWWWBBW	125	WWWWWWBWW	129	BWWBBBWWW
122	WBWWBWWWW	126	WWWBBWWWW	W → B	
123	WBWBBWWWW	127	WBWWBWBWW	130	WBWWBWBBB
124	WBBWBWWWW	128	WWWWBWWWW	131	WWWBBBWWB
				132	BBWWBWBWW

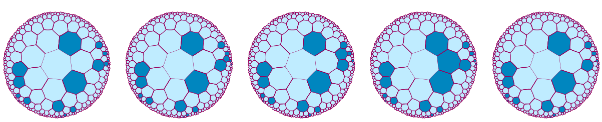
Table 19. Rules for the control: passage of the locomotive and signal for changing the selected track.

	9_6	10_6	4_6	1_6	0_0	1_4	2_5	5_5	12_4	1_3
1	29	17	63	4	4	4	7	7	4	57
2	7	25	31	45	4	4	7	7	4	57
3	7	4	34	17	36	4	7	7	4	57
4	7	4	14	25	17	36	7	7	4	95
5	7	4	14	4	25	17	16	7	4	101
6	7	4	14	4	4	25	24	16	4	57
7	7	4	14	4	4	4	29	24	36	57

Table 20. Execution of the rules used during the traversal of a black controller by the locomotive.

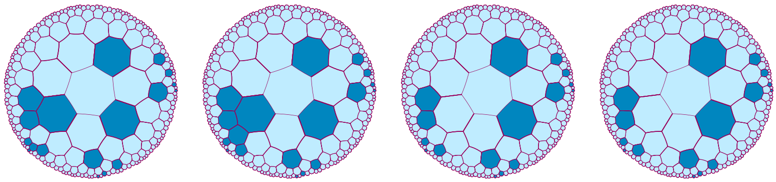


(a)

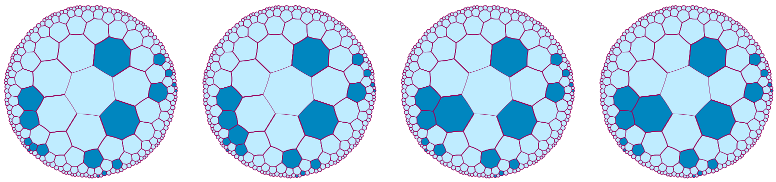


(b)

Figure 18. Illustration of the crossing of the controller by the locomotive: (a) when it is black; (b) when it is white.



(a)



(b)

Figure 19. Illustration of the arrival of the signal at the controller. (a) From black to white; (b) from white to black.

All cells of the track obey the rules we have considered for the tracks for three-milestoned rules. As an example, the same rules are applied to cell 1(4) as cell 1(6) in Tables 1 and 4. We shall look closely at cell 1(3), the control cell of the structure, and at central cell 0(0).

In Table 21, as in previous tables, black and white cells have different meanings with respect to the simulation. For the convenience of the reader, we indicate that B marks the locomotive, B marks the signal, B shows us that the controller is black, which allows the passage of the locomotive, while W shows us that it is white, which forbids the passage of the locomotive. The way the rules are working should now be clear without further comments.

	passage				signal			
0(0):	black		white		black		white	
	4	WWBWBWWBW	77	WWBWBWWWW	4	WWBWBWWBW	77	WWBWBWWWW
	36	WWBBBWWBB	104	WWBBBWWWW	77	WWBWBWWWW	4	WWBWBWWBW
	17	BWBWBWWBW	77	WWBWBWWWW				
	25	WBBWBWWBW						
	4	WWBWBWWBW						
1(3):	black		white		black		white	
	57	BWWBBWWWB	126	WWWBBWWWW	57	BWWBBWWWB	126	WWWBBWWWW
	95	BBWBBWWWB			129	BWWBBBWWW	131	WWWBBBWWB
	101	BWWBBWWBB			126	WWWBBWWWW	57	BWWBBWWWB

Table 21. Rules for cells 0(0) and 1(3) of the controller. Rules for the passage of the locomotive and for the signal, whatever the color of the controller.

Accordingly, the neighborhood for central cell 0(0) when the controller is black is WBWBWWB as the third milestone, which is in 1(3), is the cell that performs the control. In that case, the motion rules for cell 0(0) are rule 4, rule 36, rule 17 and rule 25. This corresponds to a cell of the tracks whose milestones are at the neighbors 2, 4 and 7, as indicated in Table 1. When the controller is white, the neighborhood of cell 0(0) is now WBWBWWW. The motion rules are then rule 77 and 104. Rule 77 appeared for the fork and rule 104 appeared in the selector when it is crossed by a simple locomotive.

The neighborhood of cell 1(3) is WWBBWWW, where the milestones occur at neighbors 3 and 4. That neighborhood works for an idle configuration, whatever the state of cell 1(3). The conservative rule for a black controller is rule 57, which already appeared in the motion of a double locomotive along the tracks. The conservative rule for a white controller is rule 126, a new rule appearing in Table 19.

The motion rules are rule 57, rule 95 and rule 101 already appearing in Tables 1 and 16, for the motion rules on the track and for the selector, respectively, in both cases, when it is crossed by a simple locomotive.

The change of color for the controller is triggered by a signal that arrives at its neighbor 5. This is indicated by rule 129, which follows rule 57 and is then followed by rule 126: rule 129 performs the transition from an idle black configuration to an idle white one. The opposite transition is performed by rule 131; see Tables 21 and 22. Figure 19 illustrates that behavior.

locomotive, 1(3) white

	9_6	10_6	4_6	1_6	0_0
1	29	17	63	4	77
2	7	25	31	45	77
3	7	4	34	17	104

signal for changing the color

W → B	12_3	4_3	1_3	1_2	B → W	12_3	4_3	1_3	1_2
1	25	132	131	125	1	25	92	129	121
2	4	78	57	121	2	4	127	126	125

Table 22. Execution of the rules when the locomotive arrives at a white controller and when the signal for changing the color arrives.

4.5 The Rules for the Sensor

In this last subsection of Section 4, we examine the rules that manage the working of the sensor, the specific control structure of the passive memory switch. Section 3.3 explained the working of the structure, pointing at the differences between the controller and the sensor illustrated by Figures 8 and 9. Table 23 illustrates the few rules that have to be appended to the already examined 132 rules in order to make the structure work as expected.

passage				signal	
white				B → W	
133	WBWBWBWWW	139	WBWBWBBWW	143	BWWWBBBWW
134	WWWWBBWWW	140	WBBWBBWBW	144	BBBBWWWWB
135	WBWBBBWWB	black			
136	BBWBWBWWW	141	WBWBBBBWW		
137	BBWWWWBWB	142	BWBWWWBBB		
138	WBBBWBBWW				

Table 23. Rules for the sensor of the passive memory switch.

As can be seen in the comparison of Figures 8 and 9, many rules used for the controller are also used for the sensor. As an example, as long as the sensor is white, the rules executed in the cells of the tracks when the locomotive passes are the same as those used in the same action when the controller is black; see Tables 20 and 24.

	9_6	10_6	4_6	1_6	0_0	1_4	2_5	5_5	12_4	1_1
1	29	17	63	4	133	4	7	7	4	134
2	7	25	31	45	133	4	7	7	4	134
3	7	4	34	17	135	4	7	7	4	134
4	7	4	14	25	136	36	7	7	4	131
5	7	4	14	4	138	17	16	7	4	58
6	7	4	14	4	139	25	24	16	4	58
7	7	4	14	4	139	4	29	24	36	58

Table 24. Execution of the rules when the sensor is white and then a locomotive passes.

Table 25 indicates the rules used for cells 0(0) and 1(1) when the locomotive crosses a white sensor. Note the colors given to some letters. We use the same conventions of colors as for the controller: the global meaning of B and of W are the same, and we keep the marks B and W for cell 1(1), which is the sensor cell of the structure. We remind the reader that B means that there is nothing to change, so that here B blocks the locomotive so that no change is performed for the considered sensor or for the other. Similarly, W means that the change must be performed. It is performed at the considered sensor where cell 1(1) turns from W to B. As the locomotive passes, it is after that point a signal that triggers the change in the other sensor and in the active memory switch, according to the scheme given by the second graphic of Figure 10. The rules for that latter change are also given in Table 25. Another look at those changes is given in Tables 24 and 26, which give the rules used along with the cells crossed by the locomotive and the signal during the corresponding times.

			passage		signal	
0(0):	white		black		black	
	133	WBWBWBWWW	139	WBWBWBBWW	139	WBWBWBBWW
	135	WBWBBBWWB	141	WBWBBBBWW	133	WBWBWBWWW
	136	BBWBWBWWW	139	WBWBWBBWW		
	138	WBBBWBBWW				
	139	WBWBWBBWW				
1(1):	white		black		black	
	134	WWWWBBWWW	58	BWWWBBWWB	58	BWWWBBWWB
	131	WWWBBBWWB			143	BWWWBBBWW
	58	BWWWBBWWB			134	WWWWBBWWW

Table 25. Rules for cells 0(0) and 1(1) of the sensor. Rules for the passage of the locomotive and for the signal, whatever the color of the sensor.

	locomotive						signal			
	9_6	10_6	4_6	1_6	0_0		9_1	3_1	1_1	2_1
1	29	17	63	4	139	1	29	92	143	144
2	7	25	31	45	139	2	7	127	134	15
3	7	4	34	17	141	3	7	127	134	15
4	7	4	14	4	139					

Table 26. Execution of the rules for the black sensor, for the locomotive and for the signal.

We just mention that here neighbor 1 of cells 0(0) and cell 1(1) is cell 1(3) and cell 4(1), respectively, for an examination of Figure 9. This allows us to better understand the rules of Table 25. The motion rules for cell 0(0) when cell 1(1) is white correspond to a neighborhood of 0(0), which is 1, 3, 5, a new configuration; see rule 133 in Tables 23 and 25. The conservative rule for 0(0) when the sensor is white is rule 133, while it is rule 139 for a black sensor. That is clear in Table 25.

Note that the neighborhood of 1(1) has its milestones at 4,5. The rule is conservative when the sensor is black: see rule 58; it is not when it is white: see rule 131. Cells with only two milestones and contiguous ones were already met in the motion on the tracks as rules 18 and 21 or rules 121 and 126 for the controller. In all those latter rules, which are conservative, the current state of the cell is white.

That again points at the importance of the relaxation of the rotation invariance hypothesis.

At last, note that the change of color in 1(1) is not triggered in the same way. From white to black, the change is triggered by the locomotive; see rule 131, where the black cell of the locomotive is marked with B. From black to white, it is triggered by the signal; see rule 143, where the locomotive signal is marked B. Those features can be read in Table 25 in the part devoted to the rules applied to cell 1(1).

Figure 20 shows the three different motions we studied with Tables 24 to 26. The first two lines show us the locomotive when it crosses a white sensor. The third line shows us that when cell 1(1) is black, it stops the advance of the locomotive, preventing the changes. The last line shows us how the signal arriving at cell 3(1) makes cell 1(1) turn from black to white.

We completed the examination of the rules for the sensor. Accordingly, Theorem 1 is proved. □

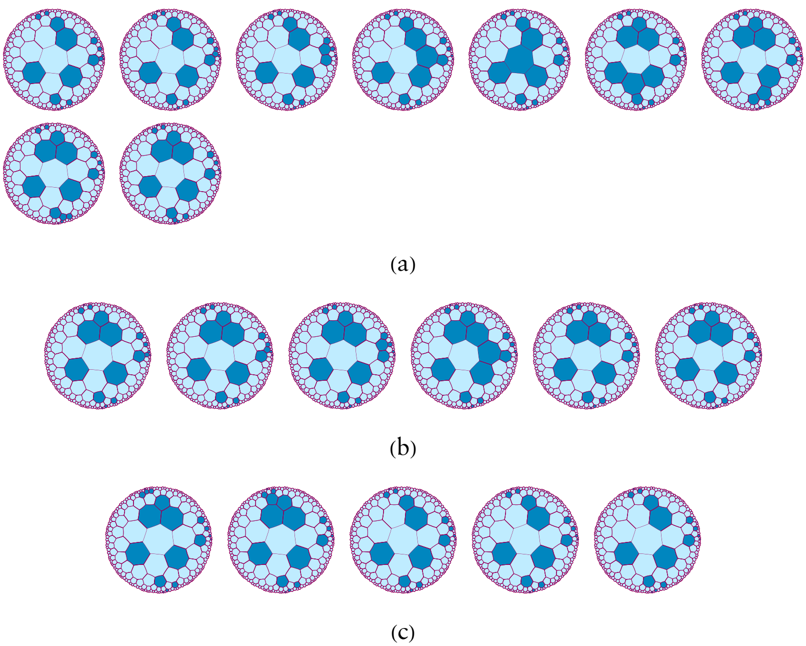


Figure 20. Illustration of the working of the sensor. (a) Passage of the locomotive; (b) the locomotive is stopped; (c) the signal changes a black sensor to a white one.

5. Conclusion

As mentioned in the introduction, Theorem 1 is the best result for tessellations involving a regular convex polygon with the angle $2\pi / 3$: the tessellation $\{7, 3\}$ is the tessellation $\{p, 3\}$, where p has the smallest value possible for the hyperbolic plane. With this model, the implementation in the heptagrid seems to be very difficult if not impossible under the rotation invariance assumption.

References

[1] T. Neary and D. Woods, "Small Weakly Universal Turing Machines," *Proceedings of the 17th International Conference on Fundamentals of Computation Theory (FCT'09)*, Wrocław, Poland, Berlin, Heidelberg: Springer-Verlag, 2009 pp. 262–273. dl.acm.org/citation.cfm?id=1789494.1789520.

[2] T. Neary and D. Woods, "The Complexity of Small Universal Turing Machines: A Survey," *Theoretical Computer Science*, **410**(4–5), 2009 pp. 443–450. doi:10.1016/j.tcs.2008.09.051.

[3] M. Cook, "Universality in Elementary Cellular Automata," *Complex Systems*, **15**(1), 2004 pp. 1–40. www.complex-systems.com/pdf/15-1-1.pdf.

[4] S. Wolfram, *A New Kind of Science*, Champaign, IL: Wolfram Media, Inc., 2002.

[5] M. Margenstern, *Cellular Automata in Hyperbolic Spaces, Vol. 2, Implementation and Computations*, Philadelphia: Old City Publishing, 2008.

[6] M. Margenstern, *Small Universal Cellular Automata in Hyperbolic Spaces: A Collection of Jewels*, New York: Springer, 2013.

[7] M. Margenstern, "A Weakly Universal Cellular Automaton on the Tessellation $\{9, 3\}$." arxiv.org/abs/1605.09518.

[8] M. Margenstern, "A Weakly Universal Cellular Automaton on the Tessellation $\{8, 3\}$." arxiv.org/abs/1606.03938.

[9] M. Margenstern, "A Weakly Universal Cellular Automaton in the Heptagrid." arxiv.org/abs/1606.09488.

[10] M. Margenstern, "A New System of Coordinates for the Tilings $\{p, 3\}$ and $\{p-2, 4\}$." arxiv.org/abs/1605.03753.

[11] M. Margenstern, "A Weakly Universal Cellular Automaton with 2 States in the Tiling $\{11, 3\}$," *Journal of Cellular Automata*, **11**(2–3), 2016 pp. 113–144.

[12] M. Margenstern, "A Weakly Universal Cellular Automaton on the Pentagrid with Two States." arxiv.org/abs/1512.07988v1.

[13] I. Stewart, "A Subway Named Turing," Mathematical Recreations, *Scientific American*, **271**(3), 1994 pp. 104–107.

[14] M. Margenstern, "A Weakly Universal Cellular Automaton on the Pentagrid with Three States." arxiv.org/abs/1510.09129.

[15] M. Margenstern, "A Family of Weakly Universal Cellular Automata in the Hyperbolic Plane with Two States." arxiv.org/abs/1202.1709.