A NONLINEAR DYNAMICS PERSPECTIVE OF WOLFRAM’S NEW KIND OF SCIENCE.
PART XII: PERIOD-3, PERIOD-6, AND PERMUTIVE RULES

Leon O. Chua and Giovanni E. Pazienza
Department of Electrical Engineering and Computer Sciences,
University of California at Berkeley,
Berkeley, CA 94720, USA
* Cellular Sensory and Wave Computing Laboratory
Computer and Automation Research Institute
MTA-SZTAKI
Budapest, Hungary

Abstract

This 12th part of our Nonlinear Dynamics Perspective of Cellular Automata concludes a series of three articles devoted to CA local rules having robust ω-limit orbits. Here we consider only the two rules, [131] and [133], constituting the third of the six groups in which we classified the 1D binary Cellular Automata. Among the numerous theoretical results contained in this article, we emphasize the complete characterization of the ω-limit orbits, both robust and non-robust, of these two rules and the proof that period-3 and period-6 ω-limit orbits are dense for [131] and [133], respectively. Furthermore, we
will also introduce the fundamental concepts of *perfect period-T orbit sets* and *riddled basins*, and see how they emerge in rule \[131\].

As stated in the title, we also focus on *permutive* rules, which have been introduced in a previous installment of our series but never thoroughly studied. Indeed, we will review some of the well-known properties of such rules, like the *surjectivity*, examining their implications for finite and bi-infinite Cellular Automata.

Finally, we propose a *new* list of the 88 globally-independent local rules, which is slightly different from the one we have used so far but has the great advantage of being via a rigorous methodology and not an arbitrary choice. For the sake of completeness, we display in the appendix the basin tree diagrams and the portraits of the $\omega$-limit orbits of the rules from this refined table which have not yet been reported in our previous articles.

Keywords: cellular automata; nonlinear dynamics; period-3 rules; period-6 rules; group 3 rules; permutive rules; surjective rules; surjectivity; dense orbits; perfect orbit sets; $\omega$-limit orbits; attractors; Isles of Eden; gardens of Eden; orbit concatenation; basin tree diagrams; isomorphic basin trees; riddled basins of attraction; Wolfram.
1. **List of the 88 minimal equivalence rules**

The universe of one-dimensional cellular automata is small yet so rich in nonlinear dynamics phenomena, as we have delineated over the past eleven episodes of our chronicle: Part I [Chua et al., 2002], Part II [Chua et al., 2003], Part III [Chua et al., 2004], Part IV [Chua et al., 2005a], Part V [Chua et al., 2005b], Part VI [Chua et al., 2006], Part VII [Chua et al., 2007a], Part VIII [Chua et al., 2007b], Part IX [Chua et al., 2008], Part X [Chua et al., 2009a], and Part XI [Chua et al., 2009b]. The final aim of our research is to formalize the enormous amount of empirical results collected in Wolfram’s monumental tome [Wolfram, 2002] via the rigorous theory of Nonlinear Dynamics.

This paper contains many relevant results, especially on the dynamics of period-3, period-6, and permutive rules. However, we devote this first section to a fundamental issue that will affect our future work. As proved in [Chua et al., 2004], the 256 one-dimensional binary Cellular Automata can be partitioned into 88 global equivalence classes, each containing 1, 2 or 4 elements. For the sake of simplicity, we have so far chosen the rule with the lowest number as the representative specimen of the whole class. For instance, in our discourse we often mentioned rule 60, but never rule 102, because it is globally equivalent to 60; namely, \( 102 = T^1(60) \). It is important to emphasize that this choice is arbitrary.

However, in [Chua et al., 2009b] we discussed the importance of the number 137 in the history of physics, and we hence suggested to consider rule 137, and not rule 110, for representing the class of universal rules. This episode prompted us to reconsider our original, rather arbitrary, procedure in selecting a representative rule for each equivalence class and reflect on the possible alternatives to it. Indeed there is a better choice: we
select as representative of each of the 88 global equivalence classes the rule with the *smallest number of firing patterns* [Chua et al., 2003] or, in other words, with fewer red vertices in the Boolean cube. If two rules have the same number of firing patterns, pick the smaller rule. From the perspective of training our monkey mascots, as depicted in our cartoons [Chua, 2006], to raise a red flag whenever he sees a firing pattern, this implies a minimal learning time: This procedure\(^1\) allows us to include \(137\) among the 88 ‘minimal’ rules, reported in Table 1a. Remarkably, only six rules change number with respect to the previous list.

In [Chua et al., 2009b] we introduced the concept of *quasi-global equivalence* which relates the space-time pattern of non-globally equivalent rules, proving that there are six pairs of quasi-globally equivalent CA local rules. The list of the 82 minimal quasi-equivalence rules is displayed in Table 1b. Here, we made two exceptions with respect to the above selection procedure; namely, we included rule \(170\) instead of rule \(15\), and rule \(204\) instead of rule \(51\). In both cases the reason has been the particular importance that \(170\) and \(204\) have had throughout our work, specially since \(170\) is exactly the Bernoulli shift when \(L \to \infty\). The Boolean cubes of the minimal and quasi-minimal equivalence rules are displayed in Tables 2a and 2b, respectively. As already mentioned, only six rules have been renumbered: two of them – rules \(131\) and \(133\) – belong to Group 3 and are the subject of this article; two others – rules \(129\) and \(161\) – belong to Group 5 (Hyper Bernoulli rules) and they are discussed in Appendix A; the last two – rules \(137\) and \(166\), belong to Group 6 (Complex Bernoulli rules) and they are discussed in Appendix B.

To conclude this section, we present two tables summarizing the main properties – such as the complexity index, the group number, the time-reversibility etc. – of the 256 local rules, in Table 3, and the 88 minimal rules, in Table 4.

---

\(^1\) Observe that a similar approach has been used in other works (see [Li & Packard, 1990]).
2. Basin Tree Diagrams, Omega-Limit Orbits and time-τ characteristic function of rules from Group 3

In [Chua et al., 2009a] and [Chua et al., 2009b] we presented all basin tree diagrams and portraits of ω-limit orbits of the local rules belonging to Groups 1 and 2, respectively. Here, we do the same for the two rules of Group 3; namely, 131 and 133. First of all, we recall some fundamental concepts of our Nonlinear Dynamics Perspective of Cellular Automata.

Each binary bit string is coded by an integer number equal to the decimal equivalent of an L-bit binary bit string as follows:

\[ [x_0, x_1, x_2, ..., x_I] \rightarrow n = \sum_{i=0}^{I} \beta_i \cdot 2^{(I-i)} x_i \]  

(1)

where \( I \triangleq L-1 \) and \( \beta_i \in \{0, 1\} \).

Consequently, for each string length \( L \), there are \( 2^L \) distinct binary strings labeled from \( n = 0 \) to \( n = 2^L - 1 \). For computational purposes, such as in deriving the time-1 return maps, it is necessary to convert a binary bit string to its associated real number \( \phi \in [0, 1) \) via the formula [Chua et al., 2005a]:

\[ \phi = \sum_{i=0}^{I} 2^{-(i+1)} x_i \]  

(2)

where \( I \triangleq L-1 \).

The \( 2^L \) binary strings for \( 3 \leq L \leq 8 \) along with their decimal number code \( n \) and the real number \( \phi \) are displayed in Table 7 of [Chua et al., 2009a].

For each rule \( [N] \) and length \( L \), the basin tree diagrams give a complete catalog of the evolution from all possible \( 2^L \) initial binary bit strings of length \( L \). In general, they are organized into several isolated directed graphs. Each of these connected graphs is called a basin tree in [Chua et al., 2006] for conciseness, even though some of these graphs are not trees in the graph theoretic sense. Any connected component graph in the basin tree...
diagrams that has an empty set as its basin of attraction\(^2\) is called an Isle of Eden; conversely, any connected component graph in the basin tree diagrams that has a non-empty set as its basin of attraction is called an attractor [Chua et al., 2007a]. We usually refer to both types of asymptotic orbits as \(\omega\)-limit orbits [Chua et al., 2009a]. Note that according to the classical definition [Birkhoff, 1927], an \(\omega\)-limit orbit is always periodic for finite \(L\) whereas the classical usage of the word orbit allows it to be the entire trajectory (both transient and steady state). Hence, the \(\omega\)-limit orbit excludes the transient part of the orbit and, for finite \(L\), it coincides with the periodic orbit.

2.1 Basin Tree Diagrams of rules from Group 3

In Tables 5a and 6a we present a gallery of the basin tree diagrams\(^3\) for rules \([131]\) and \([133]\), respectively. They are organized into two Galleries, each identified by a two digit number “\(N-m\)”, where the left digit \(N\) is identified with the local rule \([N]\), where \(N \in \{131,133\}\), and the right digit \(m\) pertains to page \(m\) of Gallery \(N\). The first page (i.e., Gallery \(N-1\)) of each Gallery \(N\) displays the following relevant information:

1. Rule number \([N]\)
2. Boolean cube of rule \([N]\)
3. Explicit formula for generating the truth table of rule \([N]\), where \(x_{i-1}, x_i, x_{i+1} \in \{0,1\}\)
4. Truth table of rule \([N]\)
5. Characteristic function \(\chi^1_{[N]}\) of rule \([N]\)
6. Sample set of time-1 return maps corresponding to different random bit strings \(\phi_0 \in [0,1)\), where the transient components have been deleted to avoid clutter. Each color codes for an \(\omega\)-limit orbit.

\(^2\) We define here the basin of attraction of a basin tree diagram as the collection of all “transient” (non-recurring) bit strings which converge to an attractor.

\(^3\) The basin tree diagrams for all 256 CA local rules are also available on the webpage http://sztaki.hu/~gpazienza/cellular_automata
Each Gallery \( N−m \) of rule \([N]\) is composed of attractors and their basins of attraction, coded in magenta, and Isles of Eden, coded in blue (a mnemonic for islands surrounded by the blue sea). To estimate the robustness of each distinct attractor and each distinct Isle of Eden, we simply calculate the ratio of the total number of bit strings, for each bit length \( L \), over the total number \( 2^L \) of bit strings:

\[
\rho_m \triangleq \frac{\text{No. of bit strings in attractor } "m" \text{ or Isle of Eden } "m"}{2^L}
\]  

We will henceforth call the number \( \rho_m \) the robustness coefficient of the attractor “\( m \)”, or the Isle of Eden “\( m \)” of the local rule \([N]\).

### 2.2 Portraits of the \( \omega \)-Limit Orbits from rules of Group 3

Given any local rule \([N]\), and any initial bit string

\[
[x_0 \ x_1 \ x_2 \ldots x_{L-1}]_0 \rightarrow \phi_0 \in [0,1)
\]  

a fundamental problem is to determine the time-asymptotic behavior

\[
[x_0 \ x_1 \ x_2 \ldots x_{L-1}]_n \rightarrow \phi_n \in [0,1)
\]  

as \( n \rightarrow \infty \). For finite \( L \), the evolution must converge to a period-\( T \) orbit, where \( T < 2^L \).

For ease of future reference, all relevant qualitative properties of the \( \omega \)-limit orbits of rules \([131]\) and \([133]\) have been extracted, organized, and exhibited in two portraits in Tables 5b and 6b, respectively. For each bit string length \( L \), identified in column 1, only distinct isomorphic\(^4\) attractors and Isles of Eden are displayed: they are identified in column 2 by a red integer \( i \). The information extracted from Tables 5a and 6a as described above are collected in the columns 3 to 9. In addition, column 7 (Bernoulli Parameters) shows the three relevant Bernoulli parameters (\( \beta, \sigma, \tau \)) [Chua et al., 2005a] for each rule \([N]\). Column 8 specifies the integer \( \delta_{\text{max}} \) defined as the distance from the farthest Garden of Eden to the attractor of each basin tree diagram. Clearly, \( \delta_{\text{max}} = 0 \) for

\(^4\) A thorough discussion about isomorphic \( \omega \)-limit orbits can be found in [Chua et al., 2009a].
all *Isles of Eden*. The last column 9 is devoted to the *robustness* coefficient for each isomorphic attractor, or Isle of Eden. The robustness coefficient $\rho_m$ for each orbit is calculated and displayed in the form $\rho_m = k \times \left( \frac{\text{number of bit strings}}{2^L} \right)$ so that the number $k$ of isomorphic copies of attractors, or *Isles of Eden*, can be identified directly. Note that the sum of all $\rho_m$ for each table is equal to 1.

Observe that in Tables 5a and 6a, we display all basin tree rules grouped by *genotype* whereas in Table 5b and 6b, for the sake of briefness, we often grouped basin trees according to their *phenotype*. For this reason, not always the values of seeds and robustness correspond in these two tables.

### 2.3 Time-$\tau$ characteristic diagram of rules from Group 3

The first page of Table 5a displays the time-1 characteristic function of rule $131$ extracted from Table 2 of [Chua et al., 2005a]. However, it is much more relevant analyzing the time-3 characteristic function of this rule, since $131$ has robust *period-3 $\omega$-limit* orbits. Figure 1 is an empirical evidence of this phenomenon, because most of the points of the time-3 characteristic function lie on the diagonal, and hence they correspond to *period-3 $\omega$-limit* orbits, whereas only a few do not lie on the diagonal, and hence they correspond to transients and non-robust $\omega$-limit orbits. From Fig. 2, we observe that a great percentage of the $2^L$ bit strings belongs to period-3 $\omega$-limit orbits for rule $131$ even for low values of $L$, such as $L \geq 15$.

Similarly, the first page of Table 6a displays the time-1 characteristic function of rule $133$, also extracted from Table 2 of [Chua et al., 2005a]. For this rule, it is much more relevant analyzing its *time-6 characteristic* function, since $133$ has robust *period-6 $\omega$-limit* orbits. Figure 3 is an empirical evidence of this phenomenon, since most of the points of the time-6 characteristic function lie on the diagonal, and hence they correspond to *period-6 $\omega$-limit* orbits, whereas only a few do not lie on the diagonal, and hence they correspond to transients and non-robust $\omega$-limit orbits. From Fig. 2, we observe that a great percentage of the $2^L$ bit strings belongs to period-6 $\omega$-limit orbits for rule $133$ even for low values of $L$, such as $L \geq 15$.

---

5 A *genotype* characterizes a set of isomorphic basin trees; if non-isomorphic basin trees have the same digraph representation, then they are said to have the same *phenotype* [Chua et al., 2009a].
correspond to transients and non-robust $\omega$-limit orbits. The existence of robust period-6 $\omega$-limit orbits is even more evident is Fig. 4, in which we focused on a smaller portion of the axis $\phi \in [0.5,1]$ and with an higher value of $L$. From Fig. 5, we observe that a great percentage of the $2^b$ bit strings belongs to period-6 $\omega$-limit orbits for rule $133$ even for low values of $L$, such as $L \geq 27$.

3. Robust $\omega$-limit orbits of rules from Group 3

Rule $131$ and rule $133$ have been included in Group 3 because they have robust period-3 and period-6 orbits, respectively, as found numerically in [Chua et al., 2007b]. We proposed this classification after verifying that several dozens of randomly-chosen long (400 bits) bit strings indeed belonged to period-$T$ $\omega$-limit orbits, where $T$ is equal to 3 for rule $131$ and 6 for rule $133$. This empirical approach has in fact a mathematical justification: if we let $q$ denote the fraction of non-robust period-$T$ orbits for a rule from Group 3 (either rule $131$ or rule $133$) then the probability $p$ that $m$ randomly-chosen bit strings belong to a non-robust orbit is $p=q^m$; for instance, for $q=0.01$ (which means that only one bit string out of 100 belongs to a non-robust orbit) and $m=50$ (we choose randomly 50 bit strings), the probability that all of these iterates converge to a non-robust orbit $p=q^m=10^{-100}$. Therefore, the probability that $131$ or $133$ are misclassified is extremely low.

Even though this methodology is effective, our ‘Nonlinear dynamics perspective of Cellular Automata’ is based on rigorous proofs and firm results, and hence we cannot rely exclusively on computer simulations. For this reason, we will present in Sec. 3.2 the formal proofs concerning the robust $\omega$-limit orbits for rules $131$ and $133$, after reviewing in Sec. 3.1 some basic definitions. We recall that a similar rigorous approach was used in [Chua et al., 2009a] to prove that Group 1 rules have robust period-1 $\omega$-limit orbits, and in [Chua et al., 2009b] to prove that Group 2 rules have robust period-2 $\omega$-limit orbits.
3.1. Definitions and notation

We start our analysis by formalizing the concept of period-T robustness as follows:

Definition 3.1.1. Robust period-T $\omega$-limit orbits

A local rule has robust period-T $\omega$-limit orbits if the probability for a random bit string of finite length $L$ to converge to a period-T $\omega$-limit orbit$^6$ tends to 1 when $L \to \infty$.

This definition does not exclude a period-T rule from having other $\omega$-limit orbits (attractors and Isles of Eden) with period different from $T$; however, if such $\omega$-limit orbits exist, they are non-robust, in the sense that they appear only for some values of $L$ and their basins of attraction are composed of a limited number of bit strings satisfying very specific conditions. Note that this definition extends those given in [Chua et al., 2009a] and [Chua et al., 2009b] for period-1 and period-2 rules, respectively. Definition 3.1.1 implies that a period-T rule is characterized by the fact that for a given $L$ most of the $2^L$ bit strings belong to the basin of attraction of period-T $\omega$-limit orbits (attractors and Isles of Eden).

Also, we would like to recall a definition and a fundamental theorem about the periodicity of the single bits of a generic bit string $x^n$ [Chua et al., 2009b]:

Definition 3.1.2. Period-T bit

Given a bit string $x^n$ belonging to a periodic orbit and denoting by $x^n_i$ the bit of $x^n$ at position $i$, the bit $x^n_i$ has period $T$ if $x^{n+T}_i = x^n_i$.

In other words, a bit has period $T$ if it repeats in the same position after every $T$ iterations. Recursively, we find that if a bit has period $T$, then $x^{n+kt}_i = x^n_i$, $\forall k \in \mathbb{N}$.

$^6$ For finite $L$, the $\omega$-limit orbit coincides with the periodic orbit associated with either an attractor, or an isle of Eden. However, for $L = \infty$, a $\omega$-limit orbit need not be periodic.
Theorem 3.1.1.
The period of a bit string \( x^n = (x_0^n x_1^n ... x_t^n) \) is equal to the least common multiple of the periods of its bits.

As a corollary, if all bits of a bit string have the same minimal periodicity \( T \), then also the bit string has period \( T \); however, the converse is not true, because the fact that a bit string has period \( T \) does not imply that all its bits have period \( T \). Indeed, some bits may have shorter periods in view of some local symmetry within the bit string.

We would like to conclude this section by giving a few details about the notation used here for consistency with [Chua et al., 2009a] and [Chua et al., 2009b], and hence already familiar to the readers of our previous works.

In the following, we use the expression ‘isolated 0’ (resp., ‘isolated 1’) to indicate a bit 0 (resp., 1) whose left and right neighbors are 1 (resp., 0), and the expression ‘runs of \( k \) 0’ (resp., ‘runs of \( k \) 1’) to indicate \( k \) consecutive bits 0 (resp., 1). Moreover, the next section contains numerous figures showing the predecessors of a given pattern under rule \([N]\).

Such pattern is always on the last row – labeled Pattern – whereas its possible predecessors are on the other rows, labeled Case I, Case II, etc. These bit strings are created bit by bit starting from the left and trying all possible valid possibilities; we employ the symbol \( X \) when no bit in that position is possible under rule \([N]\). These figures allow us to prove that some sequences cannot appear in the periodic orbits of \([131]\) and \([133]\). As an example, we give a detailed and thorough explanation of Fig. 7, in order to help the readers to follow our proofs.

Example 3.1.3 – Interpretation of Figure 7
Our goal is to prove that the bit strings belonging to the \( \omega \)-limit orbits of rule \([131]\) cannot contain the pattern 101101. For example, it would be sufficient to show that 101101 can appear only in Gardens of Eden, or, in other words, it has no predecessors. We can denote the generic predecessor of 101101 by the bit string \( x_0 x_1 x_2 x_3 x_4 x_5 x_6 x_7 \);
hence, \( x_0x_1x_2x_3x_4x_5x_6x_7 \xrightarrow{[131]} 101101 \), where the symbol \( \xrightarrow{[131]} \) means ‘generated in one iteration under rule \([131]\)’. Consequently, we would like to find \( \{x_0, x_1, \ldots, x_7\} \) such that

\[
\begin{align*}
x_0x_1x_2 & \rightarrow 1 \quad (6.1) \\
x_1x_2x_3 & \rightarrow 0 \quad (6.2) \\
x_2x_3x_4 & \rightarrow 1 \quad (6.3) \\
x_3x_4x_5 & \rightarrow 1 \quad (6.4) \\
x_4x_5x_6 & \rightarrow 0 \quad (6.5) \\
x_5x_6x_7 & \rightarrow 1 \quad (6.6)
\end{align*}
\]

Our task is proving that no combination of \( \{x_0, x_1, \ldots, x_7\} \) can give 101101.

From Table 7, we can also observe that the only firing patterns of \([131]\) are 000, 001, and 111. Since \( x_0x_1x_2 \) must be a firing pattern (see Eq. 6.1), it is sufficient to perform a case study of the three cases: a) \( (x_0x_1x_2) = (111) \); b) \( (x_0x_1x_2) = (001) \); c) \( (x_0x_1x_2) = (000) \).

In Case I, depicted in the first row of Fig. 7, we suppose that in the first three columns we have \( (x_0x_1x_2) = (111) \); therefore, \( 111x_3x_4x_5x_6x_7 \xrightarrow{[131]} 101101 \). We can now focus on \( x_3 \): since \( x_1 = 1 \) and \( x_2 = 1 \), \( x_3 \) must be 0 to meet the condition of Eq. (6.2), because 111 is a firing pattern. Therefore, \( 1110x_4x_5x_6x_7 \xrightarrow{[131]} 101101 \). It is easy to observe that we have obtained a contradiction, because neither \( x_4 = 0 \) nor \( x_4 = 1 \) can meet the condition of Eq. (6.3) since both 100 and 101 are quenching patterns; hence, \( (x_0x_1x_2) \neq (111) \). We will henceforth denote a disallowed case with a red X.

Let us suppose next that \( (x_0x_1x_2) = (001) \), i.e., \( 001x_3x_4x_5x_6x_7 \xrightarrow{[131]} 101101 \). We focus on \( x_3 \): since \( x_1 = 0 \) and \( x_2 = 1 \), the condition of Eq. (6.2) is always met, because
both 010 and 011 are quenching patterns. We analyze the case \( x_3 = 1 \) in Case II, and the case \( x_3 = 0 \) in Case III.

*Case II* is depicted in the second row of Fig. 7, and, according to the assumptions above, \( x_3 = 1 \) and \( 0011x_4x_5x_6x_7 \xrightarrow{[3]} 101101 \). Observe that, because of Eqs. (6.3) and (6.4), both \( x_4 \) and \( x_5 \) must be 1; hence, \( 001111x_6x_7 \xrightarrow{[3]} 101101 \). We can find the value of \( x_6 \) from Eq. (6.5): since \( x_4x_5x_6 \) must be a quenching pattern and \( x_4 = x_5 = 1 \), then \( x_6 = 0 \), because 111 is a firing pattern; hence, \( 0011110x_7 \xrightarrow{[3]} 101101 \). Again, we have obtained a contradiction, because neither \( x_7 = 0 \) nor \( x_7 = 1 \) can meet the condition of Eq. (6.6), because both 100 and 101 are quenching patterns.

*Case III* is depicted in the third row of Fig. 7, and, according to the assumptions above, \( x_3 = 0 \) and \( 0010x_4x_5x_6x_7 \xrightarrow{[3]} 101101 \). Once more, we have obtained a contradiction, because neither \( x_4 = 0 \) nor \( x_4 = 1 \) can meet the condition of Eq. (6.3) since both 100 and 101 are quenching patterns. Therefore, \( (x_4, x_5) \neq (001) \).

The last case to analyze, *Case IV* depicted in the fourth row of Fig. 7, is the one in which \( (x_0, x_1) = (000) \), i.e., \( 000x_3x_4x_5x_6x_7 \xrightarrow{[3]} 101101 \). Also in this case, we obtain a contradiction, because neither \( x_3 = 0 \) nor \( x_3 = 1 \) can meet the condition of Eq. (6.2) since both 000 and 001 are firing patterns. Therefore, \( (x_0, x_1) \neq (000) \).

Thanks to this exhaustive analysis, we have proved that no combination of \( \{x_0, x_1, \ldots, x_7\} \) can give rise to 101101, which can hence be contained only in some Gardens of Eden.

A similar approach is repeated for all figures of this section. Clearly, a thorough description of all of them would be impractical, but the reader can try to repeat the proofs in order to get familiar with this *reductio ad absurdum* methodology.

3.2. Rigorous results about the robust \( \omega \)-limit orbits of rules [131] and [133]
In this section, we prove that rule 131 and rule 133 have robust period-3 and period-6 ω-limit orbits, respectively. The behavior of these two local rules, both belonging to Group 3, is more complex than those of the first two groups; for example, they have very long transients and many non-robust ω-limit orbits. Consequently, the proofs are also more complicated, even though they require only the knowledge of the truth tables of rules 131 and 133, which are shown in Table 7.

We start with the theorem concerning rule 131 and its globally-equivalent local rules.

**Theorem 3.2.1. Period-3 rules**

Rule 131 and its globally-equivalent local rules have robust period-3 ω-limit orbits.

**Proof**

This proof is particularly complex and hence, for the reader’s convenience, we divided it into three parts: in the first part, we prove that the bit strings belonging to the ω-limit orbits of rule 131 cannot contain an arbitrary number of consecutive 0s or 1s; in the second part, we demonstrate that all ω-limit orbits are either period-3 or στ-Bernoulli shift with σ=-1 and τ=2; in the third part, we show that only period-3 ω-limit orbits are robust.

**Part 1 – Form of the bit strings belonging to the ω-limit orbits of rule 131**

The final aim of this part is finding the maximum number of consecutive 0s or 1s that can be contained in the bit strings belonging to ω-limit orbits. For the moment, we do not consider the bit strings 00...0 and 11..1 which will be analyzed in the second part of the proof.

As for the runs of 0s, we observe that a bit string belonging to a periodic orbit cannot include a run of seven or more 0s because all its valid predecessors have to contain either the bit string 101101 or the bit string 10101, as proved in Fig. 6. However, these two bit strings can appear only in some Gardens of Eden, as proved in Figs. 7 and 8, respectively. Similarly, a run of six 0s cannot be part of a bit string of a periodic orbit.
since it is transformed, after four iterations, into the bit string 001100 (see Figs. 9) which belongs either to a bit string of the transient regime or to a $\sigma_e$-Bernoulli shift $\omega$-limit orbit with $\sigma=-1$ and $\tau=2$, as proved later in this section (refer also to Fig. 14a). Therefore, the bit strings of the $\omega$-limit orbits of rule $[131]$ cannot contain more than five consecutive 0s.

As for the runs of 1s, Fig. 5 shows that a run of five or more 1s has two possible valid predecessors: either a run of $k+1$ 0s or a run of $k+2$ 1s. Therefore, the longest run of 1s that can belong to a bit string of an $\omega$-limit orbits is 1111, because of the restriction of the consecutive number of 0s. Consequently, the bit strings belonging to $\omega$-limit orbits of rule $[131]$ cannot contain more than four consecutive 1s.

These considerations limit the number of consecutive 0s and 1s that can appear in periodic orbits (no more than five consecutive 0s or four consecutive 1s), and hence we can perform a thorough case-by-case study. Moreover, since runs of $k$ 0s generate runs of $k-1$ 1s in one iteration, we can restrict our analysis to only four cases; namely, the patterns 011110, 01110, 0110, and 010. The only possibility left out is the pattern 101, which will be analyzed separately in Part 2 below.

**Part 2 – Finding the periodicity of the $\omega$-limit orbits of rule $[131]$**

The final aim of this part is proving that all possible cases that can arise in $\omega$-limit orbits, which have been identified in the previous part, are either period-3 or $\sigma_e$-Bernoulli shift with $\sigma=-1$ and $\tau=2$.

First, we analyze a run of four 1s (i.e., pattern 011110). Here, we consider eight different situations, corresponding to all possible combinations of the three bits on the left of the pattern 011110. We can find that there are three possible outcomes: a) if these three bits are 000, 010 or 110 then the only $\omega$-limit orbit to which the resulting pattern can belong to must be period-3, as shown in Figs. 11a and 11b; b) if these three bits are 100 then we are in a transient regime because, as shown in Fig. 11c, after three iterations we find the pattern 0111110 which cannot be included in any bit string of a periodic orbit (refer to Fig. 9); c) if these three bits are 001, 101, 011, or 111 we obtain a bit string including the pattern 10111 which, as proved in Fig. 12, can be included only in a transient orbit.
Second, we analyze a run of three 1s (i.e., pattern 01110). Also in this case we have to analyze eight different situations, corresponding to all possible combinations of the three bits on the left of the pattern 01110. It is possible to observe that there are five possible outcomes: a) if the three bits are 000 then the only $\omega$-limit orbit to which the resulting pattern can belong to must be period-3, as shown in Fig. 13a; b) when the three bits are 010, 110 or 100, after three iterations we obtain either the bit string 01110 again (see Fig. 13b) or the bit string 011110 (see Fig. 13c) which, if belonging to an $\omega$-limit orbit, is period-3, as proved before; c) if the three bits are 001 then we obtain after three iterations the pattern 001100 (see Figs. 13d) which belongs either to a transient regime or to a $\sigma_\tau$-Bernoulli shift $\omega$-limit orbit with $\sigma=-1$ and $\tau=2$, as proved later in this section (refer also to Fig. 14a); d) if the three bits are either 011 or 111 then the resulting bit string would contain the pattern 10111, which cannot be included in any bit string of a periodic orbit (refer to Fig. 12); e) when the three bits are 101 the resulting bit string would contain the pattern 10101, which cannot be included in any bit string of a periodic orbit (refer to Fig. 8).

Third, we analyze a run of two 1s (i.e., pattern 0110) which are tackled by considering the nearest neighbors (one on the left and one on the right) of the pattern 0110 and examining the three possible outcomes: a) if both the left and the right neighbor are 0, then we obtain a $\sigma_\tau$-Bernoulli shift $\omega$-limit orbit with $\sigma=-1$ and $\tau=2$, as shown in Fig. 14a; b) when only one of the two neighbors is 1, as in Figs. 14b and 14c, after two iterations we obtain a run of four 1s (observe that, as shown in Fig. 14d, it is not possible to obtain only three consecutive 1s) which belongs either to a transient or to a period-3 $\omega$-limit orbit, as proved earlier in this section; c) the case in which both neighbors are 1, corresponding to the pattern 101101, can occur only in a Garden of Eden (refer to Fig. 7).

Fourth, we analyze what happens to isolated 1s – pattern 010 – through the usual method, i.e., by considering all possible combinations of the three bits on the left of the pattern 010 and examining the four possible outcomes: a) when the three bits are 000, 010, 110, 001 or 111 (see Figs. 15a, 15b, 15c and 15d), the pattern can only belong to a period-3 $\omega$-limit orbit (or, of course, to a transient); b) if the three bits are 100 (see Fig. 15e) then we obtain after one iteration the pattern 001100 which belongs either to a transient regime or to a period-3 $\omega$-limit orbit, as proved earlier in this section; c) when
the three bits are 011 (see Fig. 15f) we obtain a run of four 1s which belongs either to a transient regime or to a period-3 $\omega$-limit orbit, as proved earlier in this section; d) when the three bits are 101 the resulting bit string would contain the pattern 10101 which cannot be included in any bit string of a periodic orbit (refer to Fig. 8).

Fifth, in Fig. 16 we show that the pattern 101 is transformed in two iterations into one of the patterns already studied.

We conclude this part with an important observation. At the beginning of Part 1 above, we specified that we were not considering the bit strings 00...0 and 11...1: the reason is that they converge to a period-1 $\omega$-limit orbit (11...1 → 11...1) which has a reduced basin of attraction since it includes only bit strings having some very special spatial symmetry. Moreover, such orbit is at the same time period-1, period-3 and $\sigma_\tau$-Bernoulli shift $\omega$-limit orbit with $\sigma=-1$ and $\tau=2$, and hence we do not consider it as a separate case.

This proves all bit strings for any finite L belong to two kinds of orbits: either period-3 or $\sigma_\tau$-Bernoulli shift with $\sigma=-1$ and $\tau=2$.

Part 3 – Robustness of the $\omega$-limit orbits of rule 131

In the previous part, we proved that all $\omega$-limit orbits of rule 131 are either period-3 or $\sigma_\tau$-Bernoulli shift. Now, we will show that only the former are robust. Looking carefully at the case study performed in Part 2, we find that there are two possible period-3 patterns: the first one, which we indicate with the symbol $P_1$, consists of a run of three 1s (see Fig. 13a) and its space-time pattern is depicted in Fig. 17; the second one, which we indicate with the symbol $P_2$, consists of a run of four 1s (see Figs. 11a and 11b) and its space-time pattern is depicted in Fig. 18. Also, there is only one $\sigma_\tau$-Bernoulli shift pattern, which we indicate with the symbol $B$, consisting of a run of two 1s and whose space-time pattern is depicted in Fig. 19. Observe that patterns containing isolated 1s can be period-3 (see Figs. 15c and 15d), $\sigma_\tau$-Bernoulli with $\sigma=-1$ and $\tau=2$ (see Fig. 15e), or even both period-3 and $\sigma_\tau$-Bernoulli shift at the same time (see Figs. 15a and 15b).
Now, we need to see how the $\sigma$-$Bernoulli$ shift pattern $B$ interacts with the period-3 patterns $P_1$ and $P_2$. Through an exhaustive case study, not reported here to avoid numerous and tedious examples, we found that there are only two possible scenarios, illustrated in Figs. 20 and 21. In the first case, the $Bernoulli$ shift pattern $B$ ‘hits’ the period-3 pattern $P_1$, generating a period-3 pattern $P_2$. In the second case, the $Bernoulli$ shift pattern $B$ ‘hits’ the period-3 pattern $P_2$, generating another $Bernoulli$ shift pattern $B$. Thanks to this observation, we can draw a fundamental conclusion: if the number of patterns $B$ generated in the transient regime is greater that the one of patterns $P_1$, then the resulting $\omega$-limit orbit is a $\sigma$-$Bernoulli$ shift with $\sigma=-1$ and $\tau=2$; otherwise, the $\omega$-limit orbit is period-3. Therefore, we need to focus now on what happens in the transient regimes of rule 131.

Unfortunately, this rule has particularly long transients, which are impractical to analyze case by case (especially because their lengths depend on $L$); this makes it difficult to find rigorous conclusions. However, we can here give an intuitive argument that explains the results found experimentally, and hence helps us to prove the theorem. Let us consider a bit string of length $L$, where $L$ is possibly very long, belonging to a period-3 $\omega$-limit orbit; consequently, the number of patterns $P_1$ generated in the transient regime is greater than or equal to that of patterns $B$. Now, we add three arbitrary bits on the right, as shown in Fig. 22, and we analyze what the possible outcomes are. If the bit string is long enough, the number of $P_1$ and $B$ patterns generated in the ‘central part’ of the bit string, shown in dark grey in Fig. 22, will not change (or change slightly), whereas new $B$ and $P_1$ patterns will be generate by the three added bits and their two neighbors (in light grey in Fig. 22). If we make an exhaustive analysis of the behavior of the $2^5=32$ possible 5-bit patterns, shown in Fig. 23, we find that in 22 cases out of 32 the new pattern is also period-3 while only in 14 cases out of 22 the patterns generated are $\sigma$-$Bernoulli$ shifts. Therefore, if the initial bit string belongs to a period-3 orbit, then in the majority of cases we still obtain a period-3 orbit when we increase such bit strings by three bits; however, if the original bit string belongs to a $\sigma$-$Bernoulli$ shift orbits, a new equilibrium between $B$ and $P_1$ patterns can be created, and a new period-3 bit string may still be generated as a result.
Summary and Conclusion

In the first part of the proof, we focused on the form of the bit strings belonging to the \( \omega \)-limit orbits of \([131]\), finding that they cannot contain more than five consecutive 0s or four consecutive 1s. Thanks to this result, which allowed us to perform an exhaustive case analysis on the different situations that can arise, we proved, in the second part, that all \( \omega \)-limit orbits of \([131]\) are either period-3 or \( \sigma \)-Bernoulli shifts. Moreover, as shown in the third part, the percentage of period-3 \( \omega \)-limit orbits of rule \([131]\) is always increasing over the total as \( L \) tends to infinity, which is exactly the definition of robust \( \omega \)-limit orbits.

In conclusion, rule \([131]\) has robust period-3 \( \omega \)-limit orbits.

The robustness of the period-3 \( \omega \)-limit orbits can be observed even better in Fig. 24, in which we analyze separately the three cases for \( L = m + 3k, \ k \in \mathbb{N} \): i) \( m \equiv 0 \mod 1 \); ii) \( m \equiv 0 \mod 2 \); iii) \( m \equiv 0 \mod 3 \).

We can now focus on rule \([133]\) and its globally-equivalent local rules.

Theorem 3.2.2. Period-6 rules

Rule \([133]\) and its globally-equivalent local rules have robust period-6 \( \omega \)-limit orbits.

Proof

Also this proof is very complex and we divided it into three parts: in the first part, we prove that the bit strings belonging to \( \omega \)-limit orbits for rule \([133]\) cannot contain an arbitrary number of consecutive 0s or 1s; in the second part, we observe that all robust \( \omega \)-limit orbits have a common feature (i.e., they have what we will dub ‘walls’); in the third part, we demonstrate that all bits of the bit strings belonging to robust \( \omega \)-limit orbits can
be exclusively period-1, period-2, or period-3. The conclusion that all robust $\omega$-limit orbits of rule 133 have period-6 follows from Theorem 3.1.1.

Part 1 – Form of the bit strings belonging to the $\omega$-limit orbits of rule 133

The final aim of this part is finding the maximum number of consecutive 0s or 1s that can be contained in the bit strings belonging to $\omega$-limit orbits.

As for the runs of 0s, we observe that a run of three 0s is transformed, in one iteration, into a stable pattern (i.e., not disrupted during the evolution of the Cellular Automaton) containing an isolated 1 (see Fig. 25); hence, it cannot be part of a bit string belonging to an $\omega$-limit orbit. A similar argument can lead to the conclusion that all patterns of $k$ consecutive 0s, $k$ odd, cannot be part of a bit string belonging to an $\omega$-limit orbit, as shown in Fig. 26 for $k=5$.

As for the runs of 1s, if $k$ is odd, $k$ greater than or equal to 3, then the only valid predecessor is a run of $k+2$ 1s (see Fig. 27), but this implies that we are in a transient regime and not in an $\omega$-limit orbit.

We conclude this first part of the proof by emphasizing the two main results obtained so far: first, the bit strings belonging to $\omega$-limit orbits cannot contain runs of $k$ consecutive 0s or 1s, where $k$ is odd and greater than 1; second, isolated 1s (i.e., the pattern 010) are stable (i.e., period-1), which means that the ‘walls’ formed by an isolated 1 cannot be disrupted during the evolution of the Cellular Automaton.

Part 2 – Robust $\omega$-limit orbits of rule 133

In the previous part we saw that the sequence 010 forms a ‘wall’ in the space-time pattern, because $010 \rightarrow 010$. However, since the probability that a randomly-chosen bit string of length $L$ includes the pattern 010 tends to 1 as $L$ tends to infinity, we can state that any bit string belonging to a robust $\omega$-limit orbit contains at least one such a ‘walls’. This implies that no $\sigma$-Bernoulli shift $\omega$-limit orbit can be robust.

Moreover, for finite $L$, the ‘wall’ limits the dynamic of the Cellular Automaton on both sides, due to the periodic boundary conditions. Our strategy will then be to analyze
thoroughly what dynamics arise between two ‘walls’ and combine them through Theorem 3.1.1.

An important observation is that rule \([133]\) is bilateral; i.e., \([133] = T^1(133)\).

Consequently, we can analyze only what happens at the right side of a ‘wall’, because the same behavior occurs on the left side too.

**Part 3 – Periods of the \(\omega\)-limit orbits of rule \([133]\)**

The final part of this proof is devoted to explore the periodicity of robust \(\omega\)-limit orbit of rule \([133]\). We will analyze all possible cases described in Part 1 in the framework proposed in Part 2 (i.e., right side of a ‘wall’).

First, Fig. 28 shows that a run of six or more consecutive 1s on the right of a ‘wall’ results in a run of three 0s after two iterations, which form a new ‘wall’ (refer to Fig. 25). Therefore, a run of \(k\) 0s, where \(k\) is greater than 5, cannot be in an \(\omega\)-limit orbit.

Second, Fig. 29a shows that a run of four 1s gives, after three iterations, a run of three 1s: since five consecutive 1s cannot be generated in a periodic orbit, as proved before, we need to test whether either a run of three 1s or a run of four 1s can be produced. The first case is excluded by the considerations made in the first part of this proof, whereas the second case is confirmed by Fig. 29c: to sum up, a run of four 1s can be part of a bit string belonging to a period-3 \(\omega\)-limit orbit.

Third, Fig. 30a shows that a run of two 1s is transformed, after two iterations, into a run of at least two 1s; therefore, two alternatives are possible: the first possibility is that the pattern produced is 0110, which means that the run of two 1s is a period-2 \(\omega\)-limit orbit, as illustrated in Fig. 30b; the second possibility is that the pattern produced is 011110, which means that the run of two 1s was either in a transient regime or in the basin of attraction of a period-3 \(\omega\)-limit orbit.

To sum up, any run of \(k\) consecutive 0s, \(k\) even, must be in a transient, a period-2 \(\omega\)-limit orbit, or a period-3 \(\omega\)-limit orbit.

Now, let us analyze how runs of \(k\) consecutive 0s, \(k\) even, evolve. In Fig. 31 we observe that a run of four consecutive 0s is transformed, in one iteration, into a pattern containing at least two consecutive 1s which, as seen previously, can be in a transient
regime, a period-2 \( \omega \)-limit orbit, or a period-3 \( \omega \)-limit orbit. Figure 32a analyzes a run of two consecutive 0s which evolves, in one iteration, in a pattern containing at least two consecutive 0s. This implies that such pattern is either period-2, as depicted in Fig. 32b, or is transformed in a run of four 0s.

In conclusion, all patterns on the right (and consequently on the left, due to the bilateral property of rule 133) of a ‘wall’ can be only period-1 (Fig. 32b), period-2 (Fig. 30b) or period-3 (Fig. 29b).

Finally, we have to check what happens on the right of these patterns. If we find that only the three patterns mentioned before can arise, then our proof is concluded. Through an exhaustive study of all possible cases, we found that the only way in which the pattern in Fig. 29b can be in an \( \omega \)-limit orbit is through the scheme of Fig. 33 (and the resulting pattern is still period-3), whereas for the pattern in Fig. 30b there are two possible cases, in Figs. 34 and 35 (and the resulting pattern is still period-2 in both cases). This proves that all bit strings belonging to \( \omega \)-limit orbit and having at least one isolated 1s, corresponding to the robust case, are period-6.

**Summary and Conclusion**

In the first part of this proof, we focused on the form of the bit strings belonging to the \( \omega \)-limit orbits of rule 133, finding that they cannot contain runs of \( k \) consecutive 0s or 1s, where \( k \) is odd and greater than 1. Then, in the second part we presented some fundamental properties of the robust \( \omega \)-limit orbits of rule 133. These considerations have led to the proof, presented in the third part, that all bits of the bit strings belonging to robust \( \omega \)-limit orbits can be exclusively period-1, period-2, or period-3.

In conclusion, from Theorem 3.1.1 it follows that all robust \( \omega \)-limit orbits (attractors or Isles of Eden) of rule 133 have period 6.

As mentioned in the proof, this statement is certainly true for strings containing the pattern 010 (which we called the ‘wall’) and hence for the robust case; however, non-robust orbits may, or may not, be period-6. As an example, we show in Fig. 36 the case
of a period-6 $\omega$-limit orbit for $L=10$, and in Fig. 37 the case of a period-14 $\omega$-limit orbit for $L=14$.

Theorems 3.2.1 and 3.2.2 follow a series of rigorous results concerning the classification of CA local rules: *Group 1* rules were analyzed in [Chua et al., 2009a], *Group 2* rules in [Chua et al., 2009b], *Group 5* rules in [Chua et al., 2007a], and *Group 6* rules in [Chua et al., 2007b]. We will conclude this discourse in our next paper, in which we will focus on the rules belonging to *Group 4* (Bernoulli $\sigma$-$\tau$-shift rules).

### 3.3. Dense and perfect $\omega$-limit orbits, Riddled basins of attraction

Thanks to our incisive research on the dynamics of rules 131 and 133, we are able to characterize the distribution of their $\omega$-limit orbits. We start by presenting the following theorem concerning the concatenation of an $\omega$-limit orbit of a generic rule $\bar{N}$:

**Theorem 3.3.1 Concatenation of $\omega$-limit orbits**

Given a bit string $x = (x_0 x_1 \ldots x_{L-1})$ of length $L$ representing an $\omega$-limit orbit $\Gamma(L)$ of rule $\bar{N}$, the bit string $x' = (xx\ldots x) = (x_0 x_1 \ldots x_{L-1} \ldots x_0 x_1 \ldots x_{L-1})$, obtained by concatenating $m$ times $x$, belongs to the $\omega$-limit orbit $\Gamma'(L)$ obtained by concatenating $m$ times $\Gamma(L)$.

**Proof**

It is a direct consequence of Theorem 3.3.1 (*$\omega$-Limit orbits generation algorithm*) proved in [Chua et al., 2009a], when $L' = L''$ and $\Gamma'(L') \equiv \Gamma''(L'')$.

**Corollary 3.3.1 Concatenation of $\omega$-limit orbits**

---

7 Note that our definition of orbit does not require that it is periodic. This is consistent with standard usage in dynamical systems [Hasselblatt and Katek, 2003]. An orbit is just the trajectory from any initial condition, and is not periodic unless the initial state falls on a *periodic* orbit.
Given a bit string $x = (x_0x_1\ldots x_{L-1})$ of length $L$ belonging to a period-$T$ $\omega$-limit orbit of rule $N$, the bit string $x' = (xx\ldots xx) = (x_0x_1\ldots x_{L-1}x_0x_1\ldots x_{L-1})$ of length $mL$, obtained by concatenating the bit string $x$ $m$ times is also a period-$T$ $\omega$-limit orbit of rule $N$.

The application of this corollary is evident in the examples of Figs. 38 and 39 for rules 131 and 133, respectively.

In order to present some fundamental definitions and theorems, we need to introduce the following notation: $\Sigma(L)$ denotes the bit string space of all $2^L$ bit strings of length $L$; $\tilde{x}_r(L)$ denotes a period-$T$ bit string of rule $N$ from bit string space $\Sigma(L)$; $\tilde{x}_r(pL)$ denotes the period-$T$ bit string of rule $N$ derived by concatenation of $\tilde{x}_r(L)$ by $p$ multiples, where $p$ is an integer. Now, we can give the definition of dense period-$T$ orbits:

**Definition 3.3.1 Dense period-$T$ orbits**

The period-$T$ orbits of rule $N$ is said to be dense iff for any $\varepsilon > 0$, and any period-$T$ orbit $\tilde{x}_r(L) \in \Sigma(L)$, there exists an integer $m$ such that for all $p \geq m$, there is another period-$T$ orbit $\tilde{y}_r(pL) \in \Sigma(pL)$ such that $|\tilde{x}_r(pL) - \tilde{y}_r(pL)| < \varepsilon$.

Figure 40 gives a graphical representation of this definition. If we concatenate $m$ times a bit string $\tilde{x}_r(L)$ of length $L$, we obtain a bit string $\tilde{x}_r(mL)$ of length $mL$. Saying that another bit string $\tilde{y}_r(mL)$ must be closer than $\varepsilon$ to $\tilde{x}_r(mL)$ is equivalent to saying that the first $t$ bits of $\tilde{y}_r(mL)$ and $\tilde{x}_r(mL)$ are the same, where $t = \lceil \log_2 \varepsilon \rceil$. Those bits beyond $x_i$ and $y_i$ can be considered as a ‘noise tail’. This concept is expressed in the following Proposition:

---

8 We follow the notation in [Chua et al., 2009a] where $\Gamma(L)$ denotes any $\omega$-limit orbit of rule $N$, of length $L$. 

24
Proposition 3.3.1  
**Noise-tail bounding estimate**

The distance between any two bit strings whose first $n$ bits are identical is less than $\frac{1}{2^{n+1}}$.

Example 3.3.1

Let $x$ and $y$ be two strings with $L = 20$ such that $x = (1101101011010010101)$ and $y = (1101101011101001011)$. Since $x_i = y_i, i \in [0, 10]$ and $x_{11} \neq y_{11}$, then the distance between this two bits is less than $2^{-12} = 0.000244141$. Indeed, by using formula (2), we find that $|\phi_x - \phi_y| = 0.000173569 < 2^{-12}$.

Therefore, Definition 3.3.1 implies that for any $\epsilon > 0$ (and hence for any position $t$), given any period-$T$ orbit $\tilde{x}_T(L)$ and concatenating it $m$ times to generate the bit string $\tilde{x}_T(mL)$, where $mL > t$, we can create another period-$T$ bit string $\tilde{y}_T(mL)$ where the first $t$ bits of $\tilde{x}_T(mL)$ and $\tilde{y}_T(mL)$ coincide.

If all period-$T$ $\omega$-limit orbits of rule $N$ are dense for all possible periods, then such set of $\omega$-limit orbits is said to be perfect, according to the following definition:

**Definition 3.3.2  Perfect period-$T$ orbit set**

The set $\tilde{X}_T = \{x : x = \tilde{x}_T\}$ of all period-$T$ orbits, for all possible periods, is said to be perfect iff each $\tilde{x}_T \in \tilde{X}_T$ is dense.

Observe that for the limit when $L \to \infty$, this definition is similar to the notion of a perfect metric space [Hasselblatt and Katok, 2003].

Now, we can analyze whether the robust orbits for rules $131$ and $133$ are dense.

**Theorem 3.3.2  Dense period-3 orbits of rule $131$**
Rule [131] has a dense set of period-3 orbits.

Proof

In Theorem 3.2.1 it was proved that a bit string $\tilde{x}_r(L)$ is period-3 under rule [131] if, and only if, the number $n_{p_3}$ of patterns $P_i$ generated in the transient regime is greater than or equal to the number $n_{a_3}$ of patterns $B$. Therefore, we need to analyze how these numbers change when we concatenate strings.

Let us suppose that $\tilde{x}_r(L)$ is a period-3 bit string; Theorem 3.2.1 implies that $n_{p_3} \geq n_{a_3}$. As we explained above and shown in Fig. 40, the value of $\varepsilon$ determines the position of the bit $x_{\tau+1}$, which is the starting point of the noise tail. According to Definition 3.3.1, we want to find a period-$T$ bit string $y(mL)$ of length $mL$ in which the first $t+1$ bits coincide with those of $\tilde{x}_r(mL)$, obtained by concatenating $m$ times $\tilde{x}_r(L)$; however, this implies that $y(mL)$ has a noise tail of length $mL - t - 1$, where $m$ is arbitrary. Moreover, since all bits of the noise tail of $y(mL)$ are also arbitrary, we can always set them in such a way that the number $n_{p_3}$ of patterns $P_i$ is greater than or equal to the number $n_{a_3}$ of patterns $B$, and hence the bit string $y(mL)$ has also period-3, $y(mL) \equiv \tilde{y}_r(mL)$.

In conclusion, rule [131] has a dense set of period-3 orbits. ■

Theorem 3.3.3 Dense period-$T$ orbits of rule [131]

Rule [131] is perfect.

Proof

Since rule [131] has only two types of periodic orbits, period-3 and $\sigma_\tau$-Bernoulli shift orbits with $\sigma=-1$ and $\tau=2$ (see Theorem 3.2.1), and we have already proved in Theorem 3.3.2 that period-3 orbits are dense, we only need to prove that also the $\sigma_\tau$-Bernoulli shift orbits, which are periodic with period $T \leq \sigma L$, are dense. But this can be
done by using the same procedure as for Theorem 3.3.2, in which we swap the role of $n_\sigma$ and $n_{\tau}$. In practice, we can easily prove that given a $\sigma$-Bernoulli shift bit string $\tilde{x}_T(L)$ for rule [131], there exists an integer $m$ such that $n_\sigma > n_{\tau}$ in the noise tail of $y(mL)$.

Consequently, all periodic orbits of rule [131] are dense, and hence rule [131] is perfect.

There is an important observation to make about the $\omega$-limit orbits of rule [131], which was already mentioned at the end of Part 2 of Theorem 3.2.1. For $\forall L, L \in \mathbb{N}$, the bit string 11...1 is a period-1 attractor for rule [131]. However, according to the definition of period-T and $\sigma$-Bernoulli shift orbits we gave in [Chua et al., 2005a], such attractor can also be considered as a period-T orbit, $\forall T$, and as a $\sigma$-Bernoulli shift orbit, $\forall \sigma, \tau$.

Here, our choice is considering it either as period-3 or as $\sigma$-Bernoulli shift with $\sigma=1$ and $\tau=1$, according to our convenience, so that we do not have to analyze further cases in Theorem 3.3.2 and 3.3.3. Observe that in Tables 5b and 6b we indicate such orbit as period-1.

Now, we can focus on the density of the robust $\omega$-limit orbits of rule [133].

Theorem 3.3.4  Dense period-6 orbits of rule [133]

Rule [133] has a dense set of period-6 orbits.

Proof

According to what we proved in Theorem 3.2.2, any finite length periodic bit string containing at least one ‘wall’ can be only period-1, period-2, period-3, or period-6. Moreover, we proved that the two ‘walls’ isolate the dynamic behavior of the bit sequence within such ‘walls’.

---

9. Table 14(B) of [Chua et al., 2008] shows a list of all rules endowed with Type B period-1 orbits, which includes rules [131] and [133].
With a procedure similar to the one discussed in the proof of Theorem 3.3.3, we can affirm that the noise tail of the bit string \( y(mL) \), which we use to ‘test’ the density of rule \([131]\), can be arbitrarily long; hence, we can always create a noise tail delimited by ‘walls’ and containing a period-6 bit string, so that the whole bit string is period-6, \( y(mL) \equiv \tilde{y}_6(mL) \) as a consequence of Theorem 3.1.1.

In conclusion, rule \([133]\) has a dense set of period-6 orbits. ■

We cannot give any definitive result on whether rule \([133]\) is perfect, because our empirical analysis shows that this rule has an unbounded number of \( \omega \)-limit orbits of distinct periods.

Before concluding this section, we would like to introduce yet another fundamental concept:

**Definition 3.3.3**  
*Riddled basins of attraction for rule \([N]\)*

Rule \([N]\) is said to have *riddled basins of attraction* iff for any \( \varepsilon > 0 \) and any period-\( T \) orbit \( \tilde{x}_r(L) \in \Sigma(L) \), for all possible periods \( T \), \( \exists \) an integer \( m \) such that for all \( p \geq m \) there is a period-\( T' \) orbit \( \tilde{y}_r(pL) \in \Sigma(pL) \), \( \tilde{y}_r(pL) \neq \tilde{x}_r(pL) \), for all possible periods \( T' \), such that \( |\tilde{x}_r(pL) - \tilde{y}_r(pL)| < \varepsilon \).

This definition implies that for any \( \varepsilon > 0 \) (and hence for any position \( t \), as defined in Fig. 40), given any period-\( T \) orbit \( \tilde{x}_r(L) \) and concatenating it \( m \) times to generate the bit string \( \tilde{x}_r(mL) \), where \( mL > t \), we can create another period-\( T \) bit string \( \tilde{y}_r(mL) \) where the first \( t \) bits of \( \tilde{x}_r(mL) \) and \( \tilde{y}_r(mL) \) coincide. Our definition of riddled basins is inspired by the one given in [Alexander et al., 1992] for continuous systems.

A simple example can illustrate the meaning of the notion of riddled basins.

**Example 3.3.2**
Let us consider the bit string $\tilde{x}_r(L) = \tilde{x}_r(12) = (001001111001)$, which is period-3 for rule $[131]$, as shown in Fig. 41, and let us fix $\varepsilon = 2^{-25} \approx 3 \cdot 10^{-8}$. Since $[131]$ has either $\sigma_r$-Bernoulli shift orbits or period-3 orbits (see Theorem 3.2.1), if we suppose that $[131]$ has riddled basins of attraction (as it will indeed be proved in Theorem 3.3.5) then we should be able to find a bit string $\tilde{y}_r(mL)$ belonging to a $\sigma_r$-Bernoulli shift orbit closer than $\varepsilon$ to the bit string $\tilde{x}_r(mL)$ obtained by concatenating $m$ times $\tilde{x}_r(L)$. From the expression of $\varepsilon$ reported above, we find easily that the position of the bit $t$ (see Fig. 40) is $x_{24}$; consequently, $m \geq 3$ because $L = 12$ for $\tilde{x}_r(L)$. In Table 8, we displayed the binary and the decimal representations of the bit strings $\tilde{x}_r(L)$ and $\tilde{x}_r(3L)$, whereas their space-time patterns are in Figs. 41 and 42, respectively. Now, we need to find a bit string $\tilde{y}_r(3L)$ belonging to a $\sigma_r$-Bernoulli shift orbit which has the first 25 bits of $\tilde{x}_r(L)$: we found experimentally that the bit string $\tilde{y}_r(3L)$ reported in Table 8, and whose space-time pattern is in Fig. 43, meets these conditions. A visual representation of this example is given in Fig. 44.

It is not by chance that two bit strings belonging to different kind of $\omega$-limit orbits were arbitrarily close, because for rule $[131]$ the following theorem holds:

**Theorem 3.3.5 Riddled basins of attraction of rule $[131]$**

The basins of attraction of rule $[131]$ are riddled.

**Proof**
It follows directly from the proofs of Theorems 3.3.2 and 3.3.3. From the arbitrariness of the noise tail in $\tilde{y}_r(mL)$, we conclude that we can change the proportion of patterns $n_{\sigma_r}$ and $n_{\sigma}$, and hence the kind of orbit, according to our convenience.■
We will see in our forthcoming articles that \(131\) is not the only rule having riddled basins of attraction.

4. Permutive rules

In [Chua et al., 2008], we described a particular class of CA local rules called ‘permutive’, whose Boolean cubes exhibit on anti-symmetry with respect to some vertical plane through the center of the cube, as depicted in Fig. 8 of [Chua et al., 2008]. In particular, we say that a local rule \(N\) is: 1) \textit{Left-Permutive iff} the vertical symmetry plane is parallel to the paper; 2) \textit{Right-Permutive iff} the vertical symmetry plane is perpendicular to the paper; 3) \textit{Bi-Permutive, iff} it is both \textit{Left-} and \textit{Right-Permutive}.

Finally, a local rule \(N\) is said to be \textit{Permutive iff} \(N\) is \textit{Left} and / or \textit{Right-Permutive}.

An examination of the 256 Boolean cubes in Table 1 of [Chua et al., 2008] shows that there are only 16 \textit{Left-Permutive} rules, 16 \textit{Right-Permutive} rules, and 4 Bi-Permutive rules, as displayed in Tables 9, 10, and 11, respectively.

The symmetry in the Boolean Cubes of \textit{Permutive} rules is a consequence of the symmetry of their truth tables (see Appendix C of [Chua et al., 2008]). We recall that if \(N\) is \textit{left-Permutive} then

\[
T_N (x_{i-1}^i x_i x_{i+1}^i) = x_i^{t+1} \Rightarrow T_N (\overline{x}_{i-1}^i x_i x_{i+1}^i) = \overline{x}_i^{t+1}
\]

and if \(N\) is \textit{right-Permutive} then

\[
T_N (x_{i-1}^i x_i x_{i+1}^i) = x_i^{t+1} \Rightarrow T_N (x_{i-1}^i \overline{x}_i x_{i+1}^i) = \overline{x}_i^{t+1}
\]

for \(x_{i-1}, x_i, x_{i+1} \in \{0,1\}\). If we use the compact representation [Chua et al., 2002] of a local rule \(N = (\beta_7, \beta_6, \beta_5, \beta_4, \beta_3, \beta_2, \beta_1)\), where \(N = \sum_{i=0}^{7} \beta_i \cdot 2^i\), we can easily find that a local rule is \textit{left-Permutive} iff (see also Fig. 8a of [Chua et al., 2008])

\[
\beta_7 = \overline{\beta}_3, \ \beta_6 = \overline{\beta}_2, \ \beta_5 = \overline{\beta}_1, \ \text{and} \ \beta_4 = \overline{\beta}_0, \ \text{where} \ \beta_i \in \{0,1\} \quad (7a)
\]

or, alternatively
and it is right-permutive iff (see also Fig. 8b of [Chua et al., 2008])
\[ \beta_i = \overline{\beta}_0, \quad \beta_5 = \overline{\beta}_4, \quad \beta_3 = \overline{\beta}_2, \quad \text{and} \quad \beta_1 = \overline{\beta}_0, \quad \text{where} \quad \beta_i \in \{0,1\} \]  
\[ \text{(8a)} \]

or, alternatively
\[ \mathbf{N} = (\overline{\beta}_0 \beta_0 \overline{\beta}_4 \beta_1 \beta_2 \beta_1 \beta_0) \]  
\[ \text{(8b)} \]

In fact, the original definition of Permutive rules [Hedlund, 1929] was given through the property of the truth tables of the local rules.

Not all the permutive rules are globally-independent, as already observed in [Chua et al., 2008]. The following theorem allows us to find the relation among left- and right-permutive rules:

Theorem 4.1 Right-permutivity and Global transformations

Let \( \mathbf{N} \) be a right-permutive rule, then: 1) \( \mathbf{N}^\dagger = T^\dagger (\mathbf{N}) \) is left-permutive; 2) \( \mathbf{N} = T (\mathbf{N}) \) is right-permutive; 3) \( \mathbf{N}^* = T^* (\mathbf{N}) \) is left-permutive.

Proof

First, we recall from [Chua et al., 2004] that \( \mathbf{N}^\dagger = (\beta_i \beta_3 \beta_5 \beta_0 \beta_2 \beta_4 \beta_6) \). If \( \mathbf{N} \) is right-permutive, from Eq. (8a) we obtain \( \mathbf{N}^\dagger = (\overline{\beta}_0 \beta_2 \beta_4 \beta_0 \beta_2 \beta_4 \beta_0) \), which is left-permutive, as follows from Eq. (7b).

Second, we recall from [Chua et al., 2004] that \( \mathbf{N} = (\overline{\beta}_0 \beta_0 \beta_5 \beta_4 \beta_1 \beta_5 \beta_1) \). If \( \mathbf{N} \) is right-permutive, from Eq. (8a) we obtain \( \mathbf{N} = (\overline{\beta}_0 \beta_0 \beta_5 \beta_4 \beta_1 \beta_5 \beta_1) \) which is right-permutive, as follows from Eq. (8b).

Third, we recall from [Chua et al., 2004] that \( \mathbf{N}^* = (\beta_0 \beta_4 \beta_2 \beta_5 \beta_6) \). If \( \mathbf{N} \) is right-permutive, from Eq. (8a) we obtain \( \mathbf{N}^* = (\beta_0 \beta_4 \beta_2 \beta_5 \beta_6) \), which is left-permutive, as follows from Eq. (7b).
Remark
Theorem 4.1 can be proved by inspection of the respective Boolean cubes of Figs. 8(a), 8(b), and 8(c), respectively.

A similar theorem holds for left-permutive rules, mutatis mutandis:

Theorem 4.2  Left-Permutivity and Global transformations
Let us be a left-permutive rule, then: 1) is right-permutive; 2) is left-permutive; 3) is right-permutive.

Table 12 shows the 10 globally-independent permutive minimal rules, as defined in Sec. 1. Permutive rules have interesting properties, especially for bi-infinite bit strings. Observe that there is a dramatic change between L finite and L infinite. As an example, we can see how the notion of surjectivity varies in these two cases.

Definition 4.1 Surjective local rule
A local rule is said to be surjective if given an arbitrary bit string there always exists at least another bit string generating under rule ,

We proved in [Chua et al., 2008] that all rules, except for the six lossless rules , , , , , and , have Gardens of Eden for finite L. Therefore, for L finite, only the lossless rules are surjective. This is not true when L is infinite, since the following theorem holds:

Theorem 4.2
For , all permutive rules are surjective.

Proof
It is sufficient to prove that all bit strings have at least one predecessor, or, in other words, there are no Gardens of Eden.

Here, we give the proof for a right-permutive rule $N$; the proof for left-permutive rules follows easily, mutatis mutandis. We consider an arbitrary bi-infinite bit string $x^n$, and we try to find its predecessor starting with two generic bits $x_{i-1}^n$ and $x_i^n$: our task is to show that it is always possible to find $x_{i-2}^{n-1}, x_{i-1}^{n-1}, x_i^{n-1}, x_{i+1}^{n-1}$ which generate the two generic bits $x_{i-1}^n$ and $x_i^n$. From Eq. (8b), it follows that $\forall x_{i-1}^{n-1}, x_i^{n-1} \in \{0,1\}, \exists! x_{i+1}^{n-1} \in \{0,1\}$ such that $T_N(x_{i-1}^{n-1}, x_i^{n-1}, x_{i+1}^{n-1}) = x_i^n$; consequently, $x_{i+1}^{n-1}$ depends only on $x_i^n$ and it always exists.

As for $x_{i-2}^{n-1}$, from Eq. (8b) we find that if $\exists x_{i-2}^{n-1}$ such that $T_N(x_{i-2}^{n-1} 00) = x_{i-1}^n$ then either $T_N(x_{i-2}^{n-1} 01) = x_i^n$ or $T_N(x_{i-2}^{n-1} 00) = x_{i+1}^n$, then both $T_N(001) = x_{i-1}^n$ and $T_N(101) = x_i^n$. This means that for any $x_{i-1}^n$, $\exists x_{i-2}^{n-1} \in \{0,1\}$ such that $T_N(x_{i-2}^{n-1} x_i^{n-1}, x_{i+1}^{n-1}) = x_{i-1}^n$ if $(x_{i-1}^{n-1}, x_i^{n-1}) = (00)$, or $(x_{i-1}^{n-1}, x_i^{n-1}) = (01)$. Similar considerations lead to the conclusion that for any $x_{i-1}^n$, $\exists x_{i-2}^{n-1} \in \{0,1\}$ such that $T_N(x_{i-2}^{n-1} x_i^{n-1}, x_{i+1}^{n-1}) = x_{i-1}^n$ if $(x_{i-1}^{n-1}, x_i^{n-1}) = (11)$, or $(x_{i-1}^{n-1}, x_i^{n-1}) = (10)$. Therefore, $\forall x_{i-1}^n, x_i^n \in \{0,1\}, \exists x_{i-2}^{n-1} \in \{0,1\}$ such that $T_N(x_{i-2}^{n-1} x_{i-1}^{n-1}, x_i^{n-1}) = x_{i-1}^n$. This proves that there is always a predecessor of $x^n$ $\forall x_{i-1}^n, x_i^n$. Since $x_{i-1}^n$ and $x_i^n$ are arbitrary, the proposition holds in general.

Theorem 4.3
For $L = \infty$, all bit strings of bi-permutive rules have exactly four predecessors.

Proof
We consider an arbitrary bi-infinite bit string $x^n$, and we try to find its predecessor starting with two generic bits $x_{i-1}^n$ and $x_i^n$: our task is to show that it is always possible to find $x_{i-2}^{n-1}, x_{i-1}^{n-1}, x_i^{n-1}, x_{i+1}^{n-1}$ which generate the two generic bits $x_{i-1}^n$ and $x_i^n$. From Eq. (8b), it

\[\text{We recall that the notation } \exists! \text{ means “there exists only one”}\]
follows that \( \forall x_{i-1}^{n-1}, x_i^{n-1} \in \{0,1\}, \exists! x_{i-1}^{n-1} \in \{0,1\} \) such that \( T(x_{i-1}^{n-1}, x_i^{n-1}, x_{i+1}^{n-1}) = x_i^n \); consequently, \( x_{i-1}^{n-1} \) depends only on \( x_i^n \) and it always exists. From Eq. (8b), it follows that \( \forall x_{i-1}^{n-1}, x_i^{n-1} \in \{0,1\}, \exists! x_{i-2}^{n-1} \in \{0,1\} \) such that \( T(x_{i-2}^{n-1}, x_{i-1}^{n-1}, x_i^{n-1}) = x_i^n \); consequently, \( x_{i-1}^{n-1} \) depends only on \( x_i^n \) and it always exists. From the uniqueness of \( x_{i-1}^{n-1} \) and \( x_{i-1}^{n-1} \), and the arbitrariness of \( x_{i-1}^{n-1} \) and \( x_{i-1}^{n-1} \), the proposition follows.

**Remark**

The results of Theorems 4.2 and 4.3 are not new, since they can be found also in the classical works of the CA literature, such as [Hedlund, 1929] and [Wolfram, 2002]. However, here we presented compact and straightforward proofs for them.

5. **Concluding remarks**

In this paper, we have introduced several fundamental notions of our ‘Nonlinear Dynamics Perspective of Cellular Automata’. For example, we have presented a new criterion for selecting 88 globally-independent local rules (and the 82 quasi globally-independent local rules) as our prototype, which has the great advantage of being the result of a rigorous choice. Our choice also allows us to train our monkey mascot to learn only the smallest number of firing patterns for each prototype rule. In this new listing, the “universal Turing machine” rule \([137]\), which was extensively cited in our last two papers, is listed as one of the 88 globally-independent local rules.

The main part of this paper is concerned with the two rules of Group 3; namely, \([131]\) and \([133]\). Their dynamical behaviors are unique among the 88 globally-independent local rules: rule \([131]\) has robust period-3 \( \omega \)-limit orbits and non-robust \( \sigma_\tau \)-Bernoulli shift \( \omega \)-limit orbits, whereas rule \([133]\) has robust period-6 \( \omega \)-limit orbits and non-robust period-T \( \omega \)-limit orbits with \( T \neq 6 \). Here, for the first time, we found these results through a rigorous procedure and not by means of computer experiments. We also defined the notion of dense set of period-T orbits, which is the starting point to introduce more complex concepts such as the perfect period-T orbit sets and the riddled basins of
attractions. For instance, we proved that all ω-limit orbits of \[131\] are dense and hence \[131\] is perfect, in this precise sense, and that the basins of attraction of \[131\] are riddled. We conjecture that similar results can be found for many of the rules of Groups 1 and 2 as well. We emphasize the importance of the Noise-tail Bounding Estimate (Proposition 3.3.1) in our discourse, since it allows us to introduce a metric for binary bit strings.

Finally, we have given a few results on permutive rules, which were defined in our previous papers but never studied thoroughly. There are only 10 globally-independent permutive rules: 7 of them are right- / left- permutive, and 3 of them are bi-permutive. Observe that in Theorem 4.1 we have proved that the distinction between right- and left-permutive rules is fictitious, since every right-permutive has a globally-equivalent left-permutative rule, and vice versa. Also, we have shown a straightforward proof of two well-known results regarding the predecessors of permutive rules.

This paper can be considered as a sort of ‘bridge’ between our previous works and our future ones: the table of the 88 globally-independent rules is now cast in stone and properly justified, both from our scientific and neural perspective; for the first time, we were able to characterize completely the ω-limit orbits, robust and non-robust, of one local rule, \[131\]; our results on the surjectivity of permutive rules show how different the same rule can behave for finite or bi-infinite bit strings. All these elements are the cornerstones of our future discourse.

Acknowledgement

This work is supported in part by the Hungarian Academy of Science and the USA Office of Naval Research under grant no. N000 14-09-1-0411. The authors thank Dr. J. Shin for drawing the basin tree diagrams and the phase portraits.

Appendix A

Group 5 (Hyper Bernoulli rules) contains ten globally-independent local rules. Two of them have been renumbered according to the criterion illustrated in Sec. 1: namely, \[129\] substitutes \[126\] and \[161\] substitutes \[122\]. The basin tree diagrams and
the portraits of the $\omega$-limit orbits of these two rules are displayed in Tables A1, A2, A3, and A4, respectively.

Appendix B

Group 6 (Complex Bernoulli rules) contains eight globally-independent local rules. Two of them have been renumbered according to the criterion illustrated in Sec. 1: namely, 137 substitutes 110 and 166 substitutes 154. The basin tree diagrams and the portraits of the $\omega$-limit orbits of these two rules are displayed in Tables B1, B2, B3, and B4, respectively.

Appendix C

The basin tree diagrams of the rules belonging to Group 5 (Hyper Bernoulli rules) were displayed in [Chua et al., 2007a]. In Tables C1-C8, we display the portraits of the $\omega$-limit orbits of such rules, except for 129 and 161, which can be found in Appendix A.

Appendix D

The basin tree diagrams of the rules belonging to Group 6 (Complex Bernoulli rules) were displayed in [Chua et al., 2007a]. In Tables D1-D6, we display the portraits of the $\omega$-limit orbits of such rules, except for 137 and 166, which can be found in Appendix B.