Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Quantum Logic Cellular Automata as Reservoirs: Unveiling the Information-Processing Role of Class IV

Yuki Tokuyama
Waseda University
Yukio-Pegio Gunji
pegioyukio@gmail.com

Waseda University

Research Article

Keywords: Quantum Logic, Cellular Automata, Reservoir Computing, Criticality, Complexity

Posted Date: November 4th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11335448/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Quantum Logic Cellular Automata as Reservoirs: Unveiling the Information-Processing Role of Class IV

Yuki Tokuyama[1] and Yukio-Pegio Gunji[1*]

[1]Department of Intermedia Art and Science, WASEDA University, 3-4-1 Okubo, Shinjuku-ku, 169-8555, Tokyo, Japan.

*Corresponding author(s). E-mail(s): pegioyukio@gmail.com;
Contributing authors: yukiyuuki205@gmail.com;

Abstract

Reservoir computing (RC) achieves high computational performance near critical regimes, often termed the edge of chaos. However, inducing and maintaining such dynamics typically requires careful parameter tuning, limiting robustness and applicability. We propose a tuning-free critical reservoir architecture based on a Quantum-Logic Cellular Automaton (QLCA), whose transition relations induce a non-distributive orthomodular lattice via rough-set fixed points. Here, "quantum-logical" denotes a non-distributive information structure, not a quantum physical system.

Using the five-bit memory task with distractors, QLCA reservoirs consistently achieve high success rates across diverse rules and outperform elementary cellular automata (ECA) reservoirs, particularly for intermediate and long distractors. Rules exhibiting Class-IV-like patterns maintain robust performance even at distractor lengths up to $\boldsymbol{T_d = 1000}$, where conventional CA reservoirs rapidly degrade.

By decomposing these dynamics into periodic backgrounds and propagating foreground structures, we show that long-horizon robustness depends on their interaction rather than foreground criticality alone. This interaction sustains information retention while mitigating distractor-induced degradation.

These findings suggest that non-distributive quantum-logical structure promotes critical dynamics and enhances reservoir robustness without parameter tuning, offering a new design principle for RC with potential applications to physical and bio-inspired computational substrates.

Keywords: Quantum Logic, Cellular Automata, Reservoir Computing, Criticality, Complexity

1 Introduction

Reservoir computing (RC) has emerged as an effective framework for temporal information processing, exploiting the transient dynamics of a fixed high-dimensional system while training only a linear readout[1, 2]. A central challenge in RC design is to balance memory retention and nonlinear separability, two properties that are often in trade-off[3]. Previous studies have shown that this balance is optimized when reservoir dynamics operate near critical regimes, commonly described as the "edge of chaos," where information storage, transmission, and dynamic range are maximized [4–6].

Criticality has also been reported as a generic organizing principle in biological and neural systems. Experimental and theoretical studies suggest that neuronal networks operate near critical states, enabling flexible yet robust information processing [7, 8]. More recent reviews have emphasized the relevance of criticality for cognition and computation, highlighting its role in optimizing information-theoretic quantities and computational capability [9, 10]. These findings have motivated the use of critical dynamics as a guiding principle for reservoir design.

Despite its importance, criticality in RC is typically achieved through careful parameter tuning, such as adjusting coupling strengths, spectral radii, or updateing rules[11]. This tuning requirement limits robustness across tasks and undermines portability across implementations. Cellular automata (CA) provide a minimal and interpretable substrate for RC, and previous studies have demonstrated that CA-based reservoirs can achieve high performance when rules exhibit Class-IV-like dynamics [12, 13]. However, such rules form only a small subset of the rule space, again implying an implicit tuning or search process.

This situation raises a fundamental design question: *Can critical dynamics be induced structurally, rather than parametrically, in reservoir systems?*

In parallel, quantum logic, formalized as a non-distributive orthomodular lattice, has been studied as a generalized information structure that captures contextuality and incompatibility beyond classical Boolean logic[14–16]. Although quantum-logical structures originate in quantum physics, they have also been explored in broader informational and cognitive contexts[17–24]. However, their potential role as a design principle for dynamical computation, particularly in relation to criticality, has not been systematically investigated.

In our previous work, we introduced the Quantum-Logic Cellular Automaton (QLCA), a cellular automaton whose transition relations induce a non-distributive orthomodular lattice via rough-set fixed points [25, 26]. We showed that QLCA generically exhibits Class-IV-like spatiotemporal patterns without parameter tuning, suggesting a structural origin of critical dynamics. In the present study, we extend this line of work by investigating whether such structurally induced criticality enhances reservoir computing performance. Although inspired by quantum logic, the proposed framework does not rely on quantum hardware or quantum states; rather, it exploits non-distributive logical structure as an information-theoretic design principle.

Specifically, we implement QLCA as a reservoir and evaluate its computational capability on the five-bit memory task with distractors. By systematically comparing QLCA-based reservoirs with elementary cellular automata (ECA) reservoirs, we

demonstrate that the quantum-logical structure promotes robust critical dynamics and significantly improves long-horizon memory performance. Furthermore, by decomposing spatiotemporal patterns into periodic background components and propagating foreground structures, we clarify a concrete mechanism through which the quantum-logical structure sustains computation near criticality.

The main contributions of this paper are as follows:

- We propose a tuning-free reservoir architecture based on quantum-logical (non-distributive) transition relations.
- We demonstrate superior robustness of QLCA reservoirs in distractor-heavy memory tasks compared to ECA-based reservoirs.
- We identify a mechanistic role of the background–foreground interaction in sustaining critical computation.

2 Methods

2.1 Composition of QLCA

Before looking at the implementation of QLCA as a reservoir, we briefly recall the composition of QLCA [25, 26]. QLCA uses quantum logic as the rule of calculation. The calculation is derived as follows. The bit sequence in any time step t is represented as

$$b^t = (a_0^t, a_1^t, ..., a_{N-1}^t) \tag{1}$$

where $a_i^t \in \mathbf{B} = \{0, 1\}$ for $i = 0, 1, ..., N-1$, i.e. $b^t \in \mathbf{B}^N$. When computing, b^t will be separated into triplets $(a_i^t, a_{i+1}^t, a_{i+2}^t)$ to regard them as decimal numbers $c_i^t \in \{0, 1, ..., 7\}$:

$$c_i^t = 4a_i^t + 2a_{i+1}^t + a_{i+2}^t. \tag{2}$$

For example, when $(a_i^t, a_{i+1}^t, a_{i+2}^t) = (1, 1, 0)$, the decimal number of this triplet is $c_i^t = 6$. Thus, equation (1) can be rewritten as follows.

$$b^t = (c_0^t, c_3^t, ..., c_{N-}^t). \tag{3}$$

Now we see the application of the computation rule to bit sequences in

$$b^{t+1} = G(r)b^t \tag{4}$$

where each r and $G(r)$ represents a transition rule and applies the rule r. As a whole process, we take $G(r)$ for $T-1$ times. Then a set of all bit sequences $(b^0, ..., b^{T-1})$ forms a monochrome pattern consisting of $N \times T$ cells by re-coloring cells 0 and 1 white and black, respectively.

The transition rule r is defined as

$$r = (u(0), u(1), ..., u(7)) \tag{5}$$

where $i, u(i) = \{0, 1, ..., 7\}$, that is, the value of $u(0), ..., u(7)$ is selected from $0, 1, ..., 7$ without repetition. The output value of c_i^{t+1} is determined by an input value c_i^t, neighboring cells $(a_{i-2}^t, a_{i-1}^t, a_{i+3}^t, a_{i+4}^t)$ and this rule $r = (u(0), ..., u(7))$.

The connection between an input table and a transition rule r depends on an input-output table. In QLCA, this input-output table is induced from an orthomodular lattice corresponding to quantum logic. Figure 1 shows an input-output table of the symmetric QLCA, shown in our previous research[26], and also used in all the experiments in this paper. The square on the left of the figure represents a map of the relations between the input and output value. Each blue cell expresses a relation there and white cells mean no relation is there. We can see 3 diagonal relations (3×3, 2×2, 3×3, that is symmetric) and the surrounding relations of them. It represents quantum logic, or orthomodular lattice, consisting of three boolean sublattices and shared elements (maximum and minimum) between them, which correspond to quantum entanglement. The diagram on the right shows a transition rule that is calculated using neighbor cells of the fixed cells. Cellular automata usually refer to neighbor cells of the fixed cells for calculating its state on next time step. QLCA also uses neighbor cells for calculation. The gray numbers are the temporal states of fixed cells like $(a_i^t, a_{i+1}^t, a_{i+2}^t)$, that is, the input states. The black numbers represent states one or two neighboring cells. The next states of the fixed cells are decided by following this table. For example, if now the state of the fixed cells $(a_i^t, a_{i+1}^t, a_{i+2}^t) = (1, 0, 0)$, that is, $c_i^t = 4$, and their neighboring cells are $(a_{i-2}^t, a_{i-1}^t, a_{i+3}^t, a_{i+4}^t) = (1, 0, 0, 1)$, then the next state will decide as $c_i^{t+1} = u(2)$ based on the input-output map shown on the right of Figure 1. The value of $u(2)$ is decided by r in equation (5). If $r = [0, 5, 6, 2, 1, 4, 7, 3]$, then $u(2) = 6$ so we obtain $c_i^{t+1} = 6$ and $(a_i^{t+1}, a_{i+1}^{t+1}, a_{i+2}^{t+1}) = (1, 1, 0)$ as the next state.

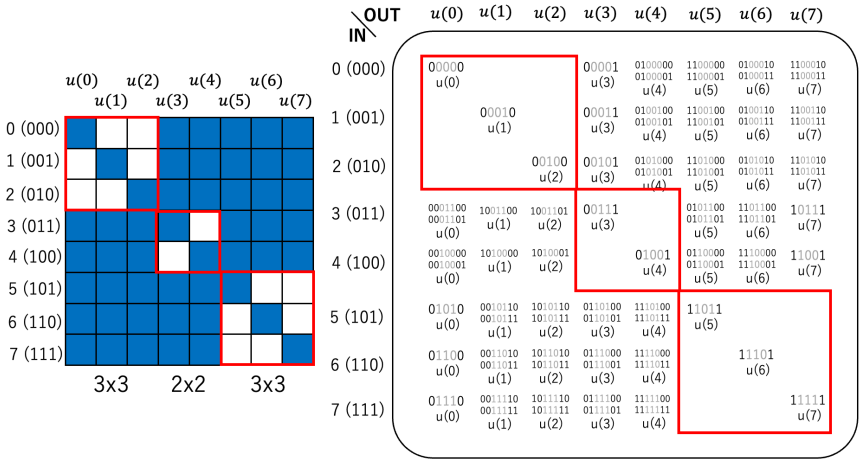


Fig. 1 An input-output table of the symmetric QLCA. On the left side of the figure, the blue area means there is a relation between input and output, while the white area means there is not. On the right, the input-output table is interpreted as deterministic, using neighbor cells of fixed cells.

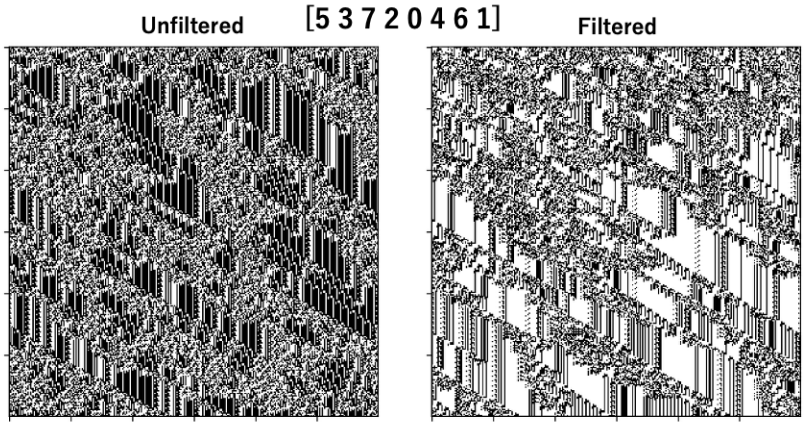


Fig. 2 A typical classIV-like pattern generated from QLCA. The left figure is a plane pattern and the right figure is filtered one removing background periodic (oscillation) patterns.

QLCA generates spatiotemporal patterns. Figure 2 displays typical class-IV like patterns generated from the symmetric QLCA. The figure on the left is the plane image. We can see a periodic pattern in the background and dendritic foreground patterns that traverse the background pattern. This is a typical behavior of classIV patterns, often referred to as complex behavior. ClassIV, which is from a classification by Wolfram[27], means that it is not able to be categorized into classI (fixed), classII (periodic), or classIII (random), and is also explained as "complex" behavior. In addition, the power spectrum of the filtered classIV patterns often displays 1/f noise[25, 26]. 1/f belongs to critical behavior, which is often seen in natural or social systems. QLCA shows a high ratio of criticality without tuning any parameter, i.e. its inherent criticality.

2.2 Quantum Logic in Transition Rules

In this section, we discuss the implications of quantum logic for transition rules. The algebraic structure of transition rules can be seen through binary relations representing transition rules, such as those shown in Figure 1.

From this binary relation, $R \subseteq D \times H$, we define the upper and lower approximations of rough sets[28, 29]. For any $X \subseteq D$ and any $Y \subseteq H$,

$$K^*(X) = \{h \in H | \exists d \in X((h,d) \in R)\} \tag{6}$$

$$S_*(Y) = \{\, d \in D \mid \forall h \in H \, \big((h,d) \in R \Rightarrow h \in Y\big) \,\} \tag{7}$$

Collecting the fixed points of the composition of these approximations produces a complete lattice, that is, a partially ordered set closed under join and meet [28, 30], that is,

$$L = \{X \subseteq D | S_*(K^*(X)) = X\}. \tag{8}$$

Here, we call this a rough set induced lattice.

When a binary relation is represented by $n \times n$ cells, and $m \times m$, where $m < n$, are related only in the diagonal cells and no relation exists in the other cells, it is called a sub-diagonal relation. When multiple sub-diagonal relations are covered by relation cells, the rough set induced lattice always becomes a non-distributive orthomodular lattice, i.e., quantum logic. The binary relation shown in Figure 1 is a binary relation in which the 3×3, 2×2 and 3×3 sub-diagonal relations are covered by relation cells. Here, we will call this a 323 relation. A non-distributive orthomodular lattice has a structure in which multiple Boolean algebras (Boolean lattices) are glued together while sharing multiple elements. Figure 3 shows the type of lattice structure that the 323 relation of the quantum automaton we use results in. Each $n \times n$ diagonal relation becomes a Boolean lattice with 2^n elements. Here, the lattice is represented by a Hasse diagram, which connects the ordering relations with lines. When the diagonal relations fill the binary relations without overlapping and the relation cells cover the periphery, the common elements of multiple Boolean lattices are the smallest and largest elements. In Figure 3, the common elements of two 2^3-Boolean lattices and one 2^2-Boolean lattice are shown as black circles, and the Hasse diagram of the final glued non-distributive orthomodular lattice is shown on the far right.

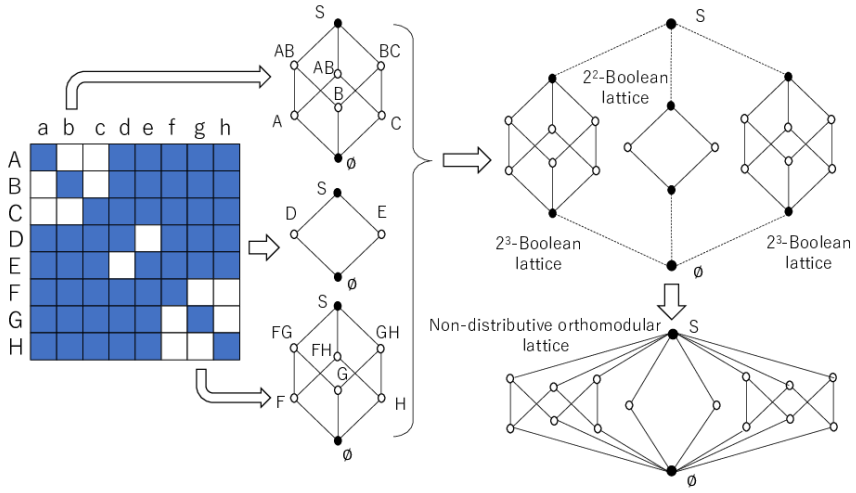


Fig. 3 Non-distributive orthotmodular lattice derived from binary relation representing the transition rule.

Figure 4 shows how subrelations affect the structure of quantum logic in an 8×8 transition rule. Figure 4A shows the 333 relation, but the first and second 3×3 subrelations overlap by one cell. This results in a non-distributive orthomodular lattice formed by the first and second subrelations. Each results in a 2^3-Boolean lattice, but because the diagonal cells overlap, the two Boolean algebras have overlapping elements (shown in red) in addition to the minimum and maximum elements. This itself therefore becomes a non-distributive orthomodular lattice consisting of 12 elements. When the 2^3-Boolean lattice obtained from the third 3×3 subrelation is glued to this, the

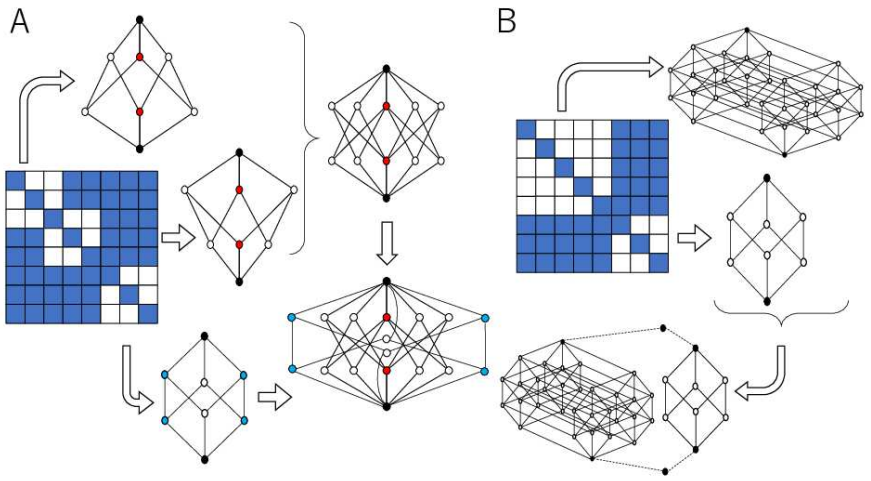


Fig. 4 Variation of binary realtion. A. The 333 relation leading to a non-distributive orhotmodular lattice consisiting of a non-distributive orthomodular lattice and 2^3-Boolean lattice. B. The 333 relation leading to a non-distributive orhotmodular lattice consisiting of 2^5- and 2^3-Boolean lattice.

8×8 binary relation itself becomes a non-distributive orthomodular lattice. It becomes a non-distributive orthomodular lattice. Figure 4B shows the 53 relation, which is a non-distributive orthomodular lattice formed by gluing a 2^5-Boolean lattice and a 2^3-Boolean lattice together.

The glued Boolean algebras become Boolean sub-lattices in a non-distributive orthomodular lattice. The distributive law holds as long as operations are performed only on elements within each sub-lattice. However, when operations are performed on elements from multiple sub-lattices, the distributive law breaks down. This is significant when we consider that this binary relation is a transition rule. In our quantum automaton, when the entire 8×8 binary relation is diagonal, its algebraic structure becomes a single Boolean lattice, and over time, it generates only order, i.e., oscillations. This holds true for non-distributive orthomodular lattices, where the sub-lattice exhibits periodic oscillations. However, when multiple Boolean algebras have overlapping elements and the distributive law breaks down, periodic oscillations can interact with each other. That is, periodic oscillations move in space, collide, and modulate the oscillations. This is how so-called Class IV behavior emerges. In particular, in our quantum automaton, implementing interactions between adjacent cells introduces effects beyond the transition between triplet cells, making the interactions of the oscillatory traveling waves more complex.

However, if the overlap of the sub-diagonal relations becomes excessively large, the overall lattice regains the distributive law, and the algebraic structure of the entire system becomes a Boolean lattice[31]. Figure 5 illustrates this. When the 8×8 binary relations representing the transition rules are composed of sub-diagonal relations such as 333, 343, and 353, they are all non-distributive orthomodular lattices. However, when the binary relations overlap with two or more diagonal components, such as 363 and 73, the distributive law is restored, and the overall algebraic structure becomes

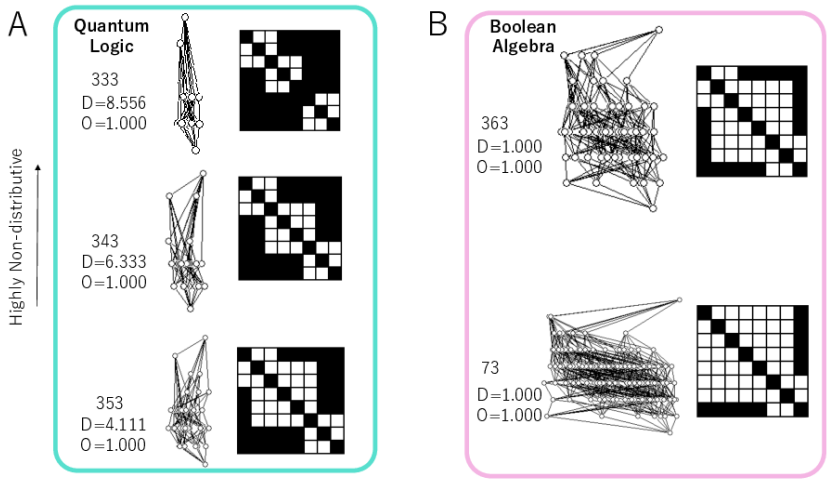


Fig. 5 The effect of overlapping of sub-relations in 8×8 relation. A. The 333, 343 and 353 relations leading to non-distributive orthomodular lattice. B. The 363 and 73 relstions leading to Boolean algebra.

a Boolean lattice. Here, to evaluate the existence of the distributive law and the orthomodular law, we calculate the following distribution rate (D) and orthomodular rate (O) such that

$$D = \frac{\#\left\{a \in L \mid \#\left\{a_c \in L \mid a \wedge a_c = 0,\ a \vee a_c = 1\right\}\right\}}{\#\left\{a \in L \mid \exists\, a_c \in L\right\}}, \tag{9}$$

$$O = \frac{\#\left\{(a,b) \in L \times L \mid a \leq b \ \Rightarrow\ a \vee (a_c \wedge b) = b\right\}}{\#\left\{(a,b) \in L \times L \mid a \leq b\right\}}. \tag{10}$$

The distribution rate is defined as a real number $D \geq 1.0$, and when the value is 1.0, the distributive law holds. Similarly, the orthomodular rate is a real number in the range $1.0 \geq O \geq 0.0$, and when the value is 1.0, the orthomodular law holds. When both the distributive law and the orthomodular law hold, the lattice is a Boolean lattice. However, even in Boolean algebra, the binary relation of Boolean algebra shown in Figure 5 is one-to-many in the input-output relationship in the transition rule, so in our automaton it is no longer simply a periodic oscillation.

2.3 Implementation for reservoir

Uragami implemented reservoir computing (RC) using asynchronous tuned elementary cellular automata (AT-ECA), a customized version of ECA proposed by Nichele and Gundersen for ECA-based RC, following their methodology [12, 32]. In this study, we adapt this methodology to QLCA. A conceptual diagram of the architecture is shown in Figure 6. The configuration shown corresponds to the benchmark task used to evaluate QLCA-based RC.

In this study, we use the five-bit memory task to evaluate the information-processing capability of QLCA-based RC. This benchmark task evaluates short-term

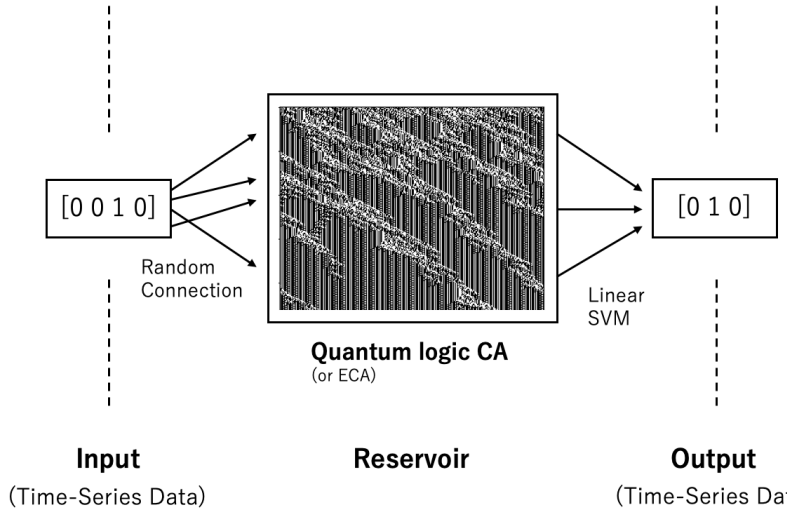


Fig. 6 A schematic diagram of whole structure of RC with QLCA.

memory by requiring the system to reproduce a specific portion of an input sequence after a fixed delay period containing distractor inputs. The duration of this delay period is denoted by T_d. When $T_d = 0$, no distractor inputs are presented. To evaluate the robustness of RC, we increase T_d to values of 20, 100, 200, 400, and 1000. The success rate is defined as the proportion of bits correctly reproduced relative to the target output sequence. Each rule is evaluated over 100 independent runs, and the mean success rate is used for total analysis.

For execution of this task on QLCA-RC, the training data consist of two time series: an input sequence composed of four-bit segments and a target output sequence composed of three-bit segments. Both sequences have a length of T. Rather than being fed into the reservoir one bit at a time, the input is presented in blocks of four bits. The input time series is randomly mapped to binary sequences. Each reservoir configuration consists of R subsequences, each containing L cells, giving a total sequence length of $L \times R$. Each subsequence receives the input and evolves for I time steps. After these I time steps, the input for the next time step is randomly mapped onto a copy of the final row, following the procedure described above. This process of input presentation and temporal evolution is repeated for all T time steps, generating T spatiotemporal patterns. These patterns constitute the reservoir's computational processing of the input sequence.

Once the T spatiotemporal patterns have been generated, learning takes place in the readout layer. Supervised learning is used to learn the mapping between the reservoir patterns and the corresponding three-bit target outputs. In light of the task's non-linearity and spatial constraints, we chose a linear support vector machine (SVM) as our learning algorithm instead of the conventional ridge regression.

The five-bit memory task is performed with $L = 20$, $R = 8$, and $I = 4$. Other parameter settings, such as $I = 8$, are beyond the scope of this study.

3 The performance of the QLCA reservoir

3.1 Overall Performance Trends of QLCA-RC

As mentioned above, the performance of cellular automaton-based RC depends on the transition rule. Elementary cellular automata (ECA), among the simplest one-dimensional cellular automata, have 256 possible transition rules. Before examining QLCA-based RC in detail, we conducted a comprehensive investigation of the learning success rate for the five-bit memory task across all ECA rules.

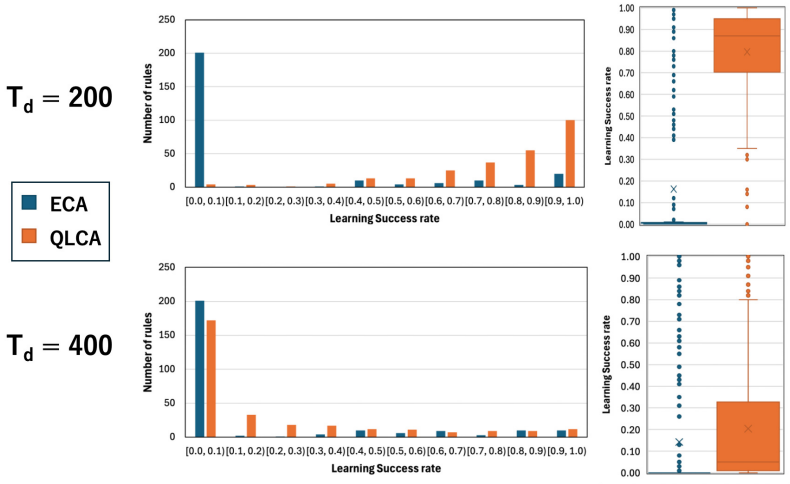


Fig. 7 An overall tendency of learning success rates on five bit memory task in QLCA and ECA reservoir (for each 256 rules). The distractor length $T_d = 200$(upside) and 400(downside). The blue graph represents ECA while the orange one represents QLCA.

Figure 7 shows the overall distributions of RC performance across 256 rules for QLCA (orange) at $T_d = 200$ and $T_d = 400$. The results for all rules of elementary cellular automata (ECA) are also plotted in blue. For each rule, we evaluated the reservoir's performance on the five-bit memory task and recorded its learning success rate. At $T_d = 200$, the most frequent performance ranges differ substantially between the two models. The number of ECA rules is highest in the range $[0.0, 0.1)$, reaching approximately 200, whereas QLCA exhibits a peak in the range $[0.9, 1.0]$, with approximately 100 rules. There is a significant difference($P(T \leq t) = 3.8364 \times 10^{-92} < 0.05$) at Welch's t-test. Furthermore, most QLCA rules achieve success rates of 0.5 or higher. At $T_d = 400$, the frequency of QLCA rules with success rates in the range $[0.9, 1.0]$ decreases, while the range $[0.0, 0.1)$, which contains the peak for ECA, becomes more prominent. However, the plots still indicate a difference in performance between the two models, suggesting that QLCA retains a significant performance

advantage(actually, $P(T \leq t) = 0.014308 < 0.05$). These results indicate that, at least up to $T_d = 400$, QLCA-based RC outperforms ECA-based RC, which serves as a elementary computational model.

3.2 Comparison Class-IV with Class-III

As mentioned in the previous section, the transition rules of CA are classified into Wolfram classes by their behavior. Previous research has shown that RC implementations using special CAs capable of generically exhibiting Class IV behavior demonstrate stable, high performance regardless of the length of the distractor[32]. Class IV behavior refers to complexity that is neither periodic nor chaotic. Positioned between the order and chaos phases, it can also be interpreted as "critical behavior." Indeed, it can be interpreted as a critical phenomenon, exhibiting characteristics such as 1/f fluctuations. Class IV rules exhibiting such behavior are quite rare in general CAs such as ECA. However, in our CA here(the symmetric QLCA), approximately one-third of the transition rules are classIV, generating critical behavior[25, 26]. Here, we examined the impact of Class IV rules in QLCA on their implementation in RC, comparing them with Class III rules.

The Wolfram classification of CA lacks established quantitative methods, allowing only rough categorization. This paper classifies QLCA patterns using three approaches: visual inspection(appearance), entropy variance, and maximum mutual information. Since the spatiotemporal pattern is determined for each rule, its entropy H(t) is also calculated for the number of rules. The entropy of the one-time step binary sequence $H(t)$ is calculated by

$$H(t) = -\sum_{i=0}^{7} P_i(b^t) \log P_i(b^t) \tag{11}$$

where $P_i(b^t)$ denotes the probability of appearance of $i = \{0, 1, 2, ..., 7\} i.e. \{000, 001, ..., 111\}$ in a fixed sequence b^t. Then the mean value of the entropy of a spatiotemporal pattern $Mean(H)$ is calculated by

$$Mean(H) = \frac{1}{T} \sum_{t=0}^{T-1} H(t) \tag{12}$$

and the entropy standard deviation $Std(H)$ is, using equation 12,

$$Std(H) = \sqrt{\frac{\sum_{t=0}^{T-1} (H(t) - Mean(H))^2}{T}}. \tag{13}$$

The value of mutual information is defined by

$$I(b^t, b^{t+1}) = \sum_{b^t} \sum_{b^{t+1}} P(b^t, b^{t+1}) \log \frac{P(b^t, b^{t+1})}{P(b^t) P(b^{t+1})}. \tag{14}$$

Although visual classification is the most primitive and reliable method, it relies on subjective metrics. Therefore, we introduce the entropy variance and maximum mutual information as objective measures. The objective is to discern trends in Class IV information processing via QLCA-RC through these three metrics. Figure 8 presents the Wolfram classification of QLCA patterns using the 260 rules used in this study, categorized by the three methods mentioned above. Although visual classification yields approximately 40 percent in Class IV, classification based on entropy and mutual information remains below 20 percent. This is because patterns in Class IV, where background and foreground waves are similar but barely distinguishable visually, cannot be effectively identified through entropy or mutual information calculations. All other classes were classified as Class III by all methods, while only a very small number of Class I and Class II instances were visually observed.

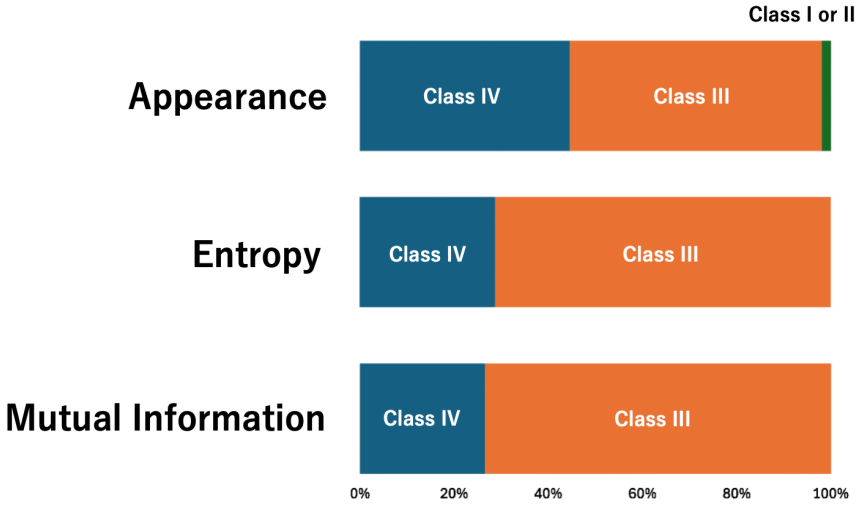


Fig. 8 Bar diagram showing the classification ratio of QLCA patterns in 260 rules. From the top, classification based on appearance, standard deviation of entropy, and maximum mutual information.

Figure 9 shows a brief comparison of RC with class-III and IV patterns, clasified by its appearance. 5 transition rules for each class were plotted. The above is class IV, and the following one is classIII. In class III, the learning success rate shows a significant fall along the length of the distractor(T_d) growing. In contrast, the class IV ratio remains high while T_d changes. This result displays that classIV patterns, having complex behavior, are desirable as a reservoir to resist long distractor.

Figure 10 shows the trends in the learning success rate of the 5-bit memory task when comparing the cases of classIV and classIII rules, classified by their appearance. We can easily see that class III success rate plots of $T_d = 400$ are seen mainly around 0.0 and rarely appear in more than 0.2 in Figure 10A. The class III reservoir did not reach high success rate (more than 0.4) with long distractor while many of class IV

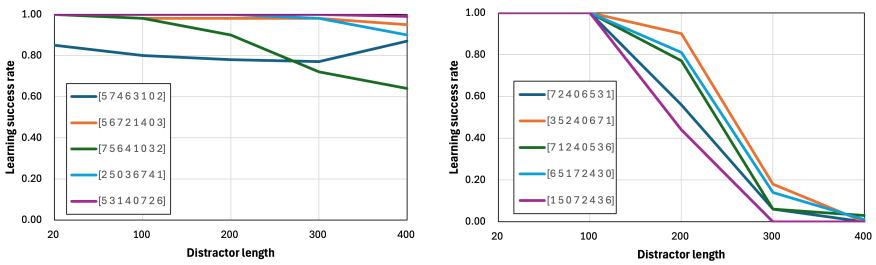


Fig. 9 A comparison class-3 and class-4 patterns on learning success rate on 5-bit memory tasks with QLCA reservoir in 5 transiton rules. The left: classIV rules. The right: classIII rules.

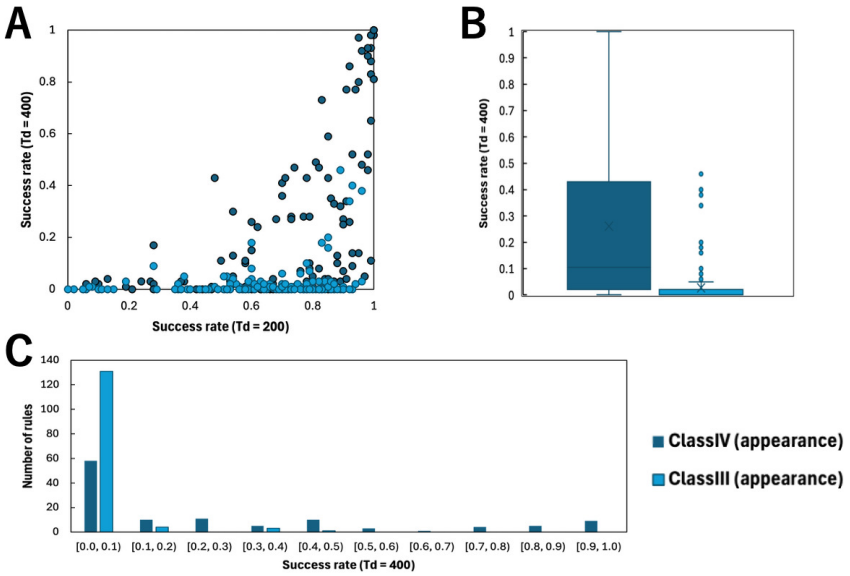


Fig. 10 Statistics of the success rate of QLCA-RC in Class IV vs Class III for total 260 rules, classified by those appearance. A(left top): Plot of the success ratio where T_d=400 against where T_d=200. B(right top): Boxplot of the success ratio where T_d=400. C(bottom): Histogram of the success rate where T_d=400.

rules did. Note that for both Class IV and Class III, the success rate at $T_d = 400$ tends to be lower than the success rate at T_d=200. This indicates that as distractors become longer, it becomes more difficult to retain the correct information—which is, in a sense, to be expected. Figure 10B shows the essential difference between classIV and classIII as reservoir. The average of success rate with the classIV rules reservoir is about 0.25 while about 0.03 in classIII rules. From Figure 10C, we also see that the success ratio of the classIV rules is reached to around 1.0, while the reservoir of the classIII rules almost appears around 0.0.

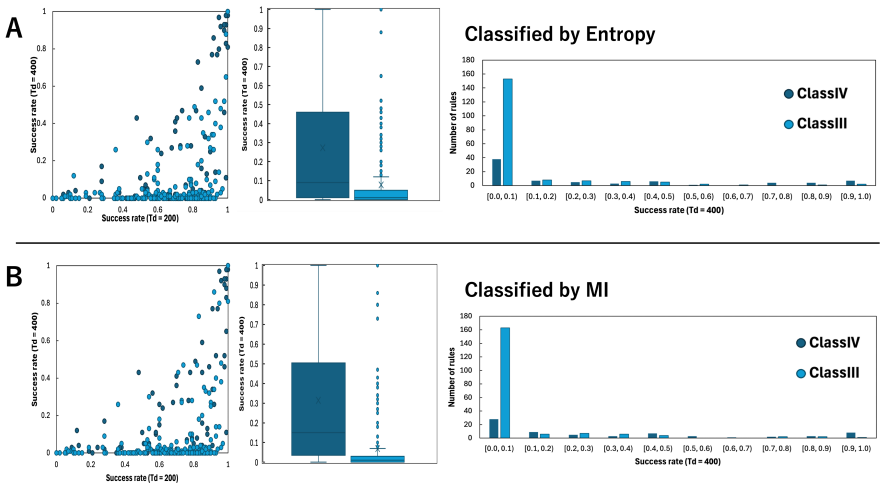


Fig. 11 Statistics of the success rate of QLCA-RC in Class IV vs Class III for total 260 rules, classified by the standard deviation of those entropy(A), or maximum value of those mutual information(B). The details of each diagram are the same as for appearance classification.

In addition, Figure 11 is comparisons of the learning success rates for 5-bit memory tasks in class III and class IV under classification based on the standard deviation of the entropy and the maximum value of mutual information, respectively. Both approaches are the same as in Figure 10, and the resulting trends are almost the same. We further tested the difference in accuracy rates between Class IV and Class III using Welch's t-test. For appearance classification, $P = 1.8291 \times 10^{-12}$, for entropy-based classification $P = 7.3327 \times 10^{-6}$, and for mutual information-based classification $P = 1.1332 \times 10^{-07}$. All values were less than 0.05, indicating a statistically significant difference.

Table 1 Examples for the changes of learning success rate of QLCA reservoir in 5 classIV rules where T_d=200, T_d=400 and T_d=1000.

Rule	T_d **= 200**	T_d **= 400**	T_d **= 1000**
[5 7 4 6 3 1 0 2]	0.78	0.81	0.83
[5 6 7 2 1 4 0 3]	0.96	0.96	0.95
[2 5 0 3 6 7 4 1]	1.00	0.95	0.86
[0 4 3 2 5 7 1 6]	0.85	0.59	0.52
[2 5 0 1 3 6 4 7]	0.96	0.92	0.75

Within some of classIV rules, reservoir computing with QLCA recorded high performance especially in T_d=400, the longer distractor. These results implies that classIV rules bring robustness for irrelevant inputs. Moreover, RC with classIV will continue to preserve its high rate when T_d=1000 as seen in Table 1. Here are 5 classIV rules

whose learning success rate is nearly 0.9 in $T_d = 400$. These reservoir performances did not change with longer distractor($T_d = 1000$) and preserved their high success ratio. In addition, Figure 12 shows the plot of the success rate of QLCA-RC in 22 classIV rules where $T_d = 1000$ versus $T_d = 400$. These rules were chosen under the condition that their success rate should have been greater than 0.5 in $T_d = 400$. In this figure, each data point generally lies on a straight line with a slope of 1. That is, the learning success rate at T_d=400 and the success rate at T_d=1000 show almost no change. That is, rules with high success rates at T_d=400 also exhibit high values at T_d=1000, showing no performance degradation. This indicates that, at least for the Class IV rules presented here, the performance of the reservoir is not affected by changes in the length of the distractors. Taking into account also the high success rate in $T_d = 200$, it is evident that the Class IV rules exhibit a high robustness against distractors.

QLCA exhibited higher success rates on a wider range of transition rules compared to ECA, suggesting that its inherent criticality supports stable computational performance without parameter tuning. Class IV rules maintained high success rates even at large distractor lengths (T_d), while Class III rules rapidly degraded. This indicates that critical dynamics enhances the ability to handle long-term dependencies.

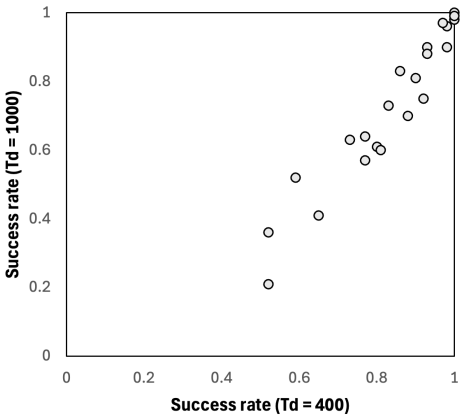


Fig. 12 The plot of the learning success rate where T_d=1000(horizontal) against T_d=400(vertical). The number of rules is 22 chosen by the condition that the success rate were more than 0.5 where T_d = 400. All data were obtained from classIV rules.

3.3 Difference by background filtering

From the previous subsection, we found that classIV-like patterns usually lead to robustness on RC for inserted distractor. Then what essential characteristic in the classIV-like pattern brings to robustness? As mentioned above, as in Figure 2, the Class IV patterns generated by QLCA can be divided into foreground waves from the background by applying a filter, and only with such a foreground wave do the patterns have a tendency to show a 1/f fluctuation by conversion to a power spectrum.

Here, we investigate the effect of foreground or background waves in the patterns from QLCA. The influence of the fore/background wave for RC was estimated by the success rate with reservoir filtering. In this experiment, filtering was performed using a two-dimensional median filter. The filter window size was 5 in the vertical (time) direction and 11 in the horizontal (spatial) direction. Background wave filtering was performed as follows. First, the median value of the cells within the filter window was taken and set as the fixed cell value (standard 2D median filter). Background waves were removed by comparing the newly obtained pattern with the original spatiotemporal pattern cell by cell and inverting the values of cells showing no change. Specifically, if the i-th cell at time t after filter application is denoted as l_i^t, the filtering process here is as follows.

$$l_i^t = \begin{cases} 1 - a_i^t & a_i^t = f(a_i^t) \\ a_i^t & elsewhere \end{cases} \tag{15}$$

Here, a_i^t represents a cell in the original spatiotemporal pattern, and f denotes the operation of applying a median filter. Figure 13 shows the results of equation (15). We can see that most of the periodic background patterns have disappeared with the adoption of this filter.

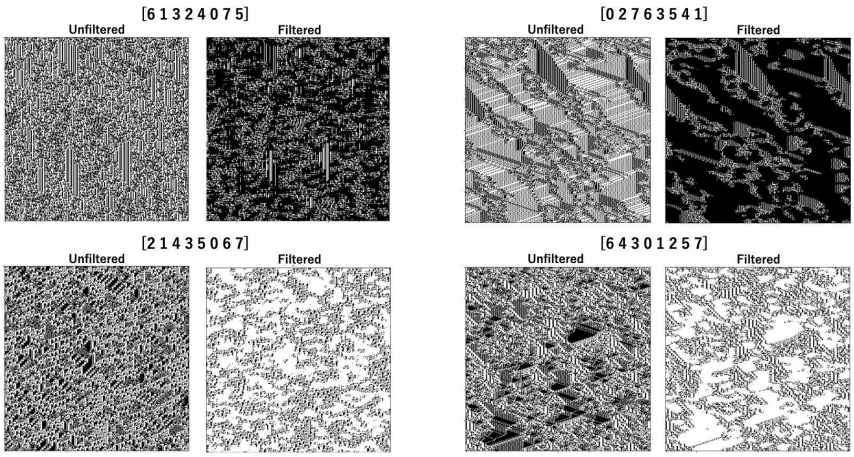


Fig. 13 Examples of unfiltered and filtered pattens derived from equation (15) on 4 ClassIV rules. The original unfiltered patterns are shown on the left side and filtered patterns are on the right side.

Figure 14 shows the example plots of the success rate versus the length of distractor before and after filtering for the 4 rules in classIV, each representing a comparison between unfiltered and filtered patterns: the blue line represents the case without filtering, and the orange line represents the case with filtering. In these 4 rules, the success rate dropped significantly when periodic background waves were removed. The success rate with filtering tends to be lower than that without filtering, within a significant difference at T_d=400.

To see the effect of background filter for more classIV rules reservoir, we investigate the change of learning ability of QLCA-RC between unfiltered and filtered in 120

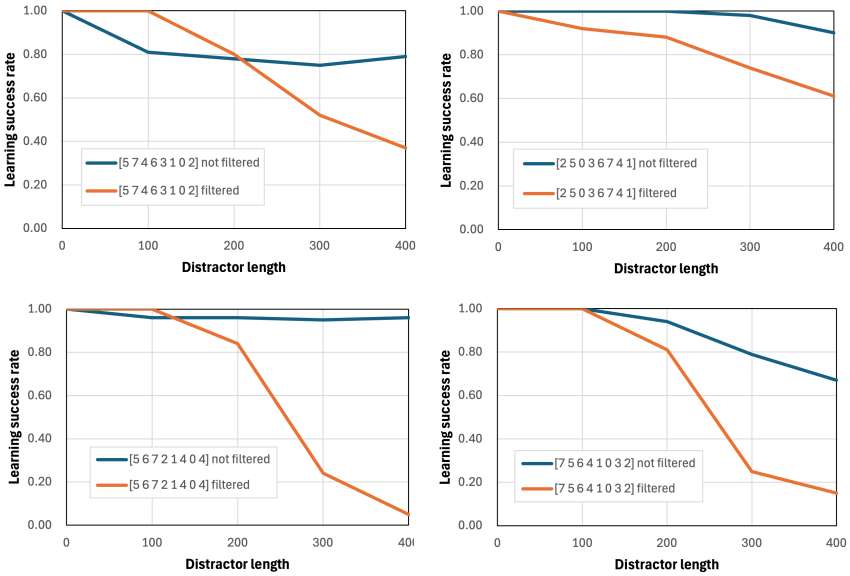


Fig. 14 Examples of the transition of success rate comparing the cases unfiltered patterns(blue) and filtered patterns(orange). All results generated by QLCA-RC with classIV rules for each.

classIV rules, as seen in Figure 15. These results are some comparison for the success rate before and after filtering, revealing the disadvantage of the filter. The data shown in blue represent the pre-filtered results, while the data shown in orange represent the post-filtered results. Figure 15A plots the ratio at T_d=400 against them at T_d=200. This plot shows that the success rate at T_d=200 and T_d=400 both decreased due to the application of the filter. The histogram of Figure 15B and the boxplot of Figure 15C compare the success rate at $T_d = 200$ before and after applying the filter. These figures show that the success rate before applying the filtered, i.e., when the periodic background wave was not removed (blue), achieved a higher value compared to when the filter was applied and the background wave was removed (orange). In a histogram in Figure 15D and a boxplot in Figure 15E, the filtered data also show a lower success rate compared to the unfiltered at $T_d = 400$. Welch's t-test was performed on these data, yielding significant differences ($p(T \leq t) = 1.2349 \times 10^{-11} < 0.05$ at $T_d = 200$ and $p(T \leq t) = 5.6276 \times 10^{-4} < 0.05$ at $T_d = 400$). These results mean that filtering prevented correct learning with long range distractor and completely decreased their success rate, especially in $T_d = 400$. In other words, the background wave is required to maintain the high reservoir computational power of classIV QLCA-RC.

The operation "filtering" was done here to remove the background periodic wave. We now see the filtering method along the calculation progress, and the success rate was reduced. Of course, the foreground waves, showing critical behavior, is brought out not only from its own but all of the time development involving the background wave. These results reveal that the computational power of the reservoir is achieved as a result of the overall behavior, including the background waves.

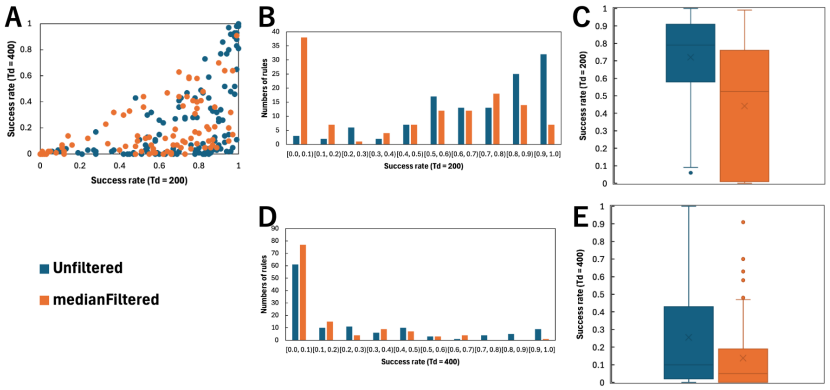


Fig. 15 The graphs of learning success rate on QLCA-RC whose reservoir unfiltered (blue) or filtered (orange) in 120 classIV rules, classified by appearance. A: plots of the ratio where T_d=400 against where T_d=200. B, D: a distribution of number of rules on each learning success rates in T_d=200(B) and in T_d=400(D). C, E: a boxplot for success ratio comparing the cases of unfiltered and filtered in T_d=200 (C) and in T_d=400 (E).

3.4 Behavior of Class-IV reservoir

In addition to the above experiments, we also observed the state of the reservoir itself and evaluated it using indicators. The following two equations were introduced by Uragami[32]. First, for two different time steps t_1 and t_2, the difference in the state of the reservoir at those time steps $d(t_1, t_2)$ is expressed by the following equation:

$$d(t_1, t_2) = \frac{|A_{t_1} - A_{t_2}|}{N} \tag{16}$$

Here, A_{t_1} and A_{t_2} represent the reservoir states at times t_1 and t_2, respectively, and N represents the number of cells in the reservoir.

Based on equation (16), the following metric D(t) is derived.

$$D(t) = \min_{0 \leq \tau < t} (d(\tau, t)) \tag{17}$$

Equation (17) indicates how much the state of the reservoir at a given time t differs from that generated up to that point. The higher the value of D(t), the more the state differs from previous states. This was used to investigate the transition of the reservoir state for Class IV rules.

Figure 16 shows the spatiotemporal pattern of the reservoir state (left) and the transition of D(t) (right) calculated for four class IV rules. Looking at the transition of D(t), the values of D(t) are high in the first 20 to 30 steps, then transition to low values and finally rise again in the last few steps. The high D(t) values in the first few dozen steps ensure diverse responses to the input data. The low values (often around 0) in the middle are responses during the input of the distractor. Since the distractor is a continuous uniform input, it can be considered to be similar to generalization in

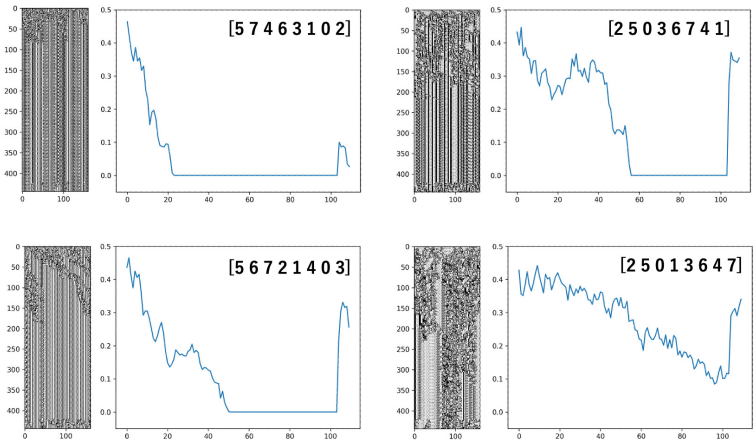


Fig. 16 The internal states of reservoir with QLCA in 4 transition rules. The left side: the spatiotemporal pattern obtained by directly outputting the state of the reservoir CA. The right side: the time evolution of the equation (17).

that it continues to generate the same reservoir state for the same input. The rise in the last few steps is due to the end of the distractor and the input of the regeneration pattern. Here, as in the beginning, diverse responses are required, but the rise in D(t) indicates that it is able to respond well to this and generate diverse reservoir states.

Looking at each of the spatiotemporal patterns on the left side of the D(t) plot, we can see that this is exactly what is happening. That is, in the first few dozen steps, diverse outputs are generated, and the patterns are complex, with tree-like foreground waves crossing over periodic background waves. However, when the distractor is input (around 100 to 400 on the vertical axis), the tree-like patterns stop propagating, and the periodic patterns are preserved as they are in the downward direction. Then, when the cue is input and the playback pattern begins (vertical axis 400 and beyond), complex patterns composed of diverse bit sequences are generated.

Figure 17 compares the internal states of the reservoir in Class III and Class IV. The blue line represents the former, while the orange line represents the latter. As shown in the legend, the transition rules are [1 5 0 7 2 4 3 6] for Class III and [6 4 7 3 5 1 0 2] for Class IV. The distractor length was 200. As shown in the figure, the blue D(t) does not change significantly over time, whereas the orange D(t) drops sharply midway through, then rises significantly in the last 10 steps or so. This indicates that the former model treated the distractor as necessary information, similar to the initial input, while the latter model regarded the distractor as noise and ignored it. Then It could judge subsequent inputs based on previous information and respond accordingly. This trend is also evident from the accuracy rates for the 5-bit memory task at $T_d = 200$: 0.36 for the former model versus 0.78 for the latter.

To confirm that this high tolerance of the reservoir to distractors is a situation specific to Class IV, we also conducted the following experiment. In the five bit memory task, we calculate the difference between the average D(t) when a distractor is input

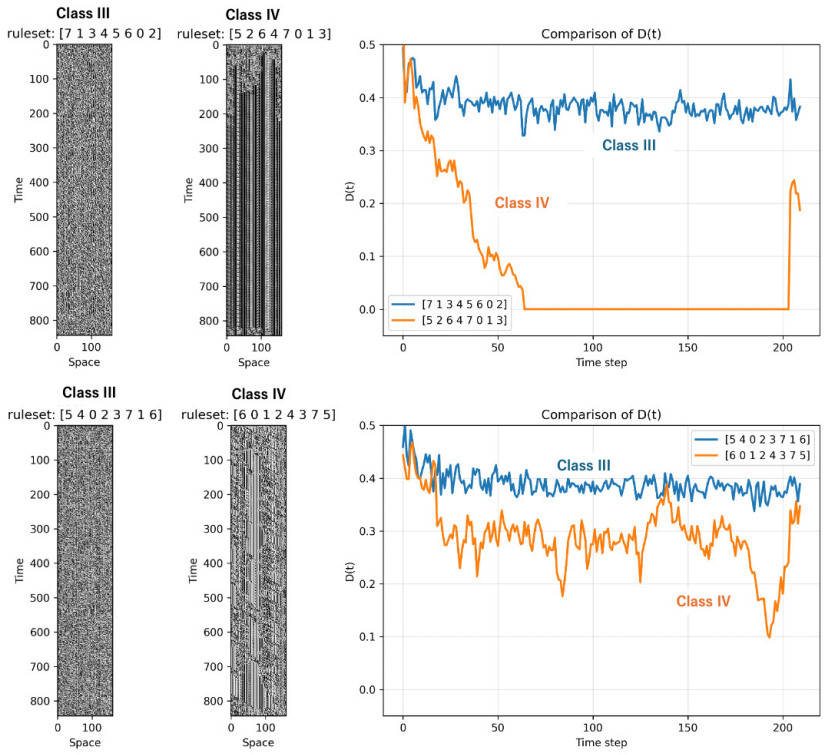


Fig. 17 A comparison of D(t) plots by 4 different rules. Blue line represents classIII rule while orange one represents classIV rule. The parameter L, R, I, and T_d were set to 20, 8, 4, and 200, respectively.

and when other information (i.e., the information to be learned) is input, using the following formula. This value provides an overview of whether the reservoir can store information without being confused by distractors.

$$C_{D(t)} = \frac{\sum_{t \neq T_d} D(t)}{T - T_d} - \frac{\sum_{t=T_d} D(t)}{T_d} \tag{18}$$

We investigate whether the value of $C_{D(t)}$ differs depending on the behavior of the QLCA, specifically whether it is class III or class IV. Figure 19 shows the results. Blue represents class IV and orange represents class III. For all classification methods, class IV tends to show higher values than class III. The p-values of Welch's t-test were 0.00004 for visual similarity, 0.03921 for entropy, and 0.00002 for mutual information, all indicating significant differences. From this we can conclude that the reservoir under the Class IV rules tends to behave more robustly against distractors and retain learned information compared to the reservoir under Class III rules.

We also investigated how D(t) changes when background noise is removed. Figure 18 shows the change in D(t) before and after applying a background noise filter to a reservoir with a Class IV rule. The orange plot represents pre-filtering, while the

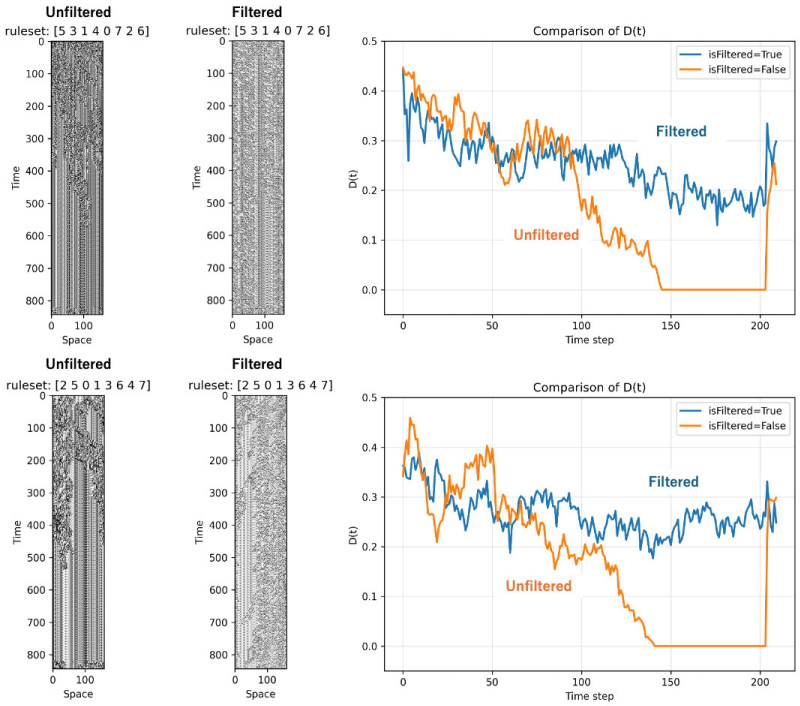


Fig. 18 A comparison of D(t) plots by filtered and unfiltered reservoir of 2 classIV rules. Blue line represents filtered rule while orange one represents unfiltered rule. The parameter L, R, I, and T_d were set to 20, 8, 4, and 200, respectively.

blue plot shows post-filtering. The graph on the right shows that the response to noise improves significantly after the filter is applied. This again demonstrates the decline in learning performance due to background wave removal mentioned in the previous section. The left side of the figure displays the internal state patterns of the reservoir before and after filter application. The left shows the state after filtering and the right shows the state before filtering. Comparing these, it is evident that while the background wave is not completely eliminated after filtering due to the filter's characteristics, its features are largely eliminated compared to before filtering. Consequently, foreground waves that previously ceased propagation midway now propagate continuously to the end in most cases after filtering, indicating that the reservoir repeatedly responds to the distractor input.

Such behavior is impossible for Class II, which generates only periodic patterns, and Class III, which generates only random (chaotic) patterns. Furthermore, considering the experiment in the previous section (Figure 15), it is thought that it is impossible even with only the foreground waves extracted by filtering to Class IV. In this section, the property of responding to diverse information while retaining the output obtained

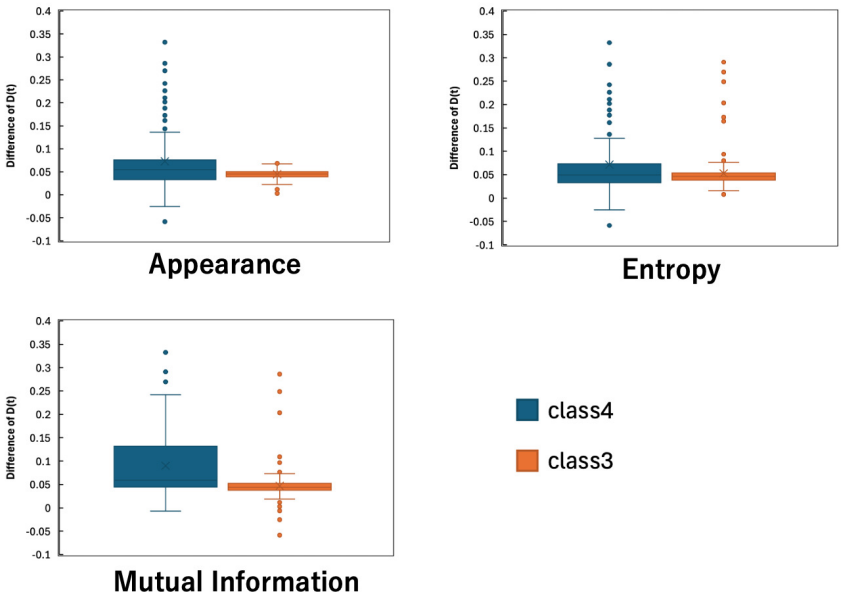


Fig. 19 Comparison of $C_{D(t)}$ (the difference between the values of D(t) when the input is a destructor and when it is not) in Class III and Class IV. The classification was performed based on visual appearance, entropy, and mutual information, respectively.

and not disrupted by continuous input of similar information is suggested to be realized by the overall processing of complexity: randomness and periodicity in Class IV patterns.

4 Discussion

The results demonstrate that QLCA-based reservoirs achieve robust performance on a broad range of rules, in contrast to ECA-based reservoirs whose success is limited to a narrow subset of Class-IV rules. This difference indicates that the critical dynamics in QLCA is not accidental, but structurally induced by the underlying quantum-logical transition relations.

A key finding of this study is that long-horizon robustness in Class-IV-like QLCA rules critically depends on the coexistence and interaction of periodic background components and propagating foreground structures. When the background component is selectively removed, performance under long distractors deteriorates significantly, even though foreground propagation remains. This suggests that foreground criticality alone is insufficient to sustain computation over extended time scales.

From an information-processing perspective, the background component can be interpreted as a stabilizing scaffold that preserves the reproducible state structure under sustained perturbations, while the foreground component provides input-sensitive propagation and separation. Their interaction enables the reservoir to suppress distractor-induced variability while rapidly re-expanding state diversity

upon cue presentation. This interpretation is consistent with the observed behavior of the proposed information-theoretic indices, which capture the balance between information retention and degradation over time.

These observations support a broader hypothesis: the non-distributive (quantum-logical) structure generically promotes a regime in which stability and sensitivity coexist. By gluing multiple local Boolean contexts, the system avoids both frozen dynamics and fully chaotic behavior, yielding a structurally enforced critical regime. Importantly, this mechanism does not rely on fine-tuning of parameters and is therefore well suited to physical, biological, and unconventional computing substrates.

Several limitations and future directions remain. The present study focuses on the five-bit memory task; extending the evaluation to standard RC benchmarks such as NARMA, chaotic time-series prediction, and multivariate classification will further clarify generality. Additionally, while the current implementation is abstract, exploring hardware-realizable substrates that can approximate QLCA-like transition relations may bridge quantum-logical structure and neuromorphic or physical reservoir computing.

5 Conclusion

We have presented a reservoir computing architecture based on a Quantum-Logic Cellular Automaton whose transition relations induce a non-distributive orthomodular lattice. Without parameter tuning, QLCA reservoirs exhibit robust critical dynamics and significantly outperform elementary cellular automata reservoirs on distractor-heavy memory tasks, particularly at long time horizons. Through pattern decomposition, we identified an interaction between periodic background waves and propagating foreground structures as a key mechanism sustaining computation near criticality. These results establish quantum-logical structure as an information-theoretic design principle for robust reservoir computing and motivate further exploration of structurally induced criticality in physical and bio-inspired computational systems.

Acknowledgements. This research was supported by Japan Society for the Promotion of Science, JSPS 23K00237.

Declarations

- Funding: JSPS 23K00237.
- Conflict of interest/Competing interests: There is no conflicts of interest.
- Ethics approval and consent to participate: Not applicable
- Consent for publication: All authors have read and approved the final manuscript for publication.
- Data availability: The experimental data are available from the contributing author upon reasonable request via email.
- Materials availability: Not applicable
- Code availability: Not applicable

- Author contribution: Conceptualization: YT, YPG; Formal analysis: YT; Investigation: YT; Writing - original draft: YT, YPG; Writing - review and editing: YT, YPG; Supervision: YPG.

References

[1] Jaeger, H.: The "echo state" approach to analysing and training recurrent neural networks. Technical report, German National Research Center for Information Technology (2001)

[2] Maass, W., Natschläger, T., Markram, H.: Real-time computing without stable states: A new framework for neural computation. Neural Computation **14**(11), 2531–2560 (2002)

[3] Dambre, J., Verstraeten, D., Schrauwen, B., *et al.*: Information processing capacity of dynamical systems. Scientific Reports **2**, 514 (2012)

[4] Langton, C.G.: Computation at the edge of chaos. Physica D **42**(1–3), 12–37 (1990)

[5] Bertschinger, N., Natschläger, T., Legenstein, R.: At the edge of chaos: Real-time computations and self-organized criticality in recurrent neural networks. Advances in neural information processing systems **17** (2004)

[6] Boedecker, J., Obst, O., Lizier, J.T., Fernando, N.K., Asada, M.: Information processing in echo state networks at the edge of chaos. PLoS ONE **7**(4), 34665 (2012)

[7] Beggs, J.M., Plenz, D.: Neuronal avalanches in neocortical circuits. Journal of Neuroscience **23**(35), 11167–11177 (2003)

[8] Lizier, J.T., Prokopenko, M., Zomaya, A.: The local information dynamics of distributed computation in complex systems. Information Sciences **178**(15), 3647–3669 (2008)

[9] Cocchi, L., Gollo, L.L., Zalesky, A., Breakspear, M.: Criticality in the brain: A synthesis of neurobiology, models and cognition. Trends in Cognitive Sciences **21**(12), 933–947 (2017)

[10] Muñoz, M.A.: Colloquium: Criticality and dynamical scaling in living systems. Reviews of Modern Physics **90**(3), 031001 (2018)

[11] Tanaka, G., Yamane, T., *et al.*: Recent advances in physical reservoir computing. Neural Networks **115**, 100–123 (2019)

[12] Nichele, S., Gundersen, M.S.: Reservoir computing using nonuniform binary cellular automata. Complex Systems **26**(3), 225–245 (2017) https://doi.org/10.25088/

ComplexSystems.26.3.225

[13] Babson, N., Teuscher, C.: Reservoir computing with complex cellular automata. Complex Systems **28**(4), 433–455 (2019) https://doi.org/10.25088/ComplexSystems.28.4.433

[14] Birkhoff, G., Neumann, J.: The logic of quantum mechanics. Annals of Mathematics **37**(4), 823–843 (1936)

[15] Abramsky, S., Coecke, B.: Categorical quantum mechanics. In: Engesser, K., Gabbay, D., Lehmann, D. (eds.) Handbook of Quantum Logic and Quantum Structures, pp. 261–325. Elsevier, ??? (2008)

[16] Heunen, C., Landsman, N.P., Spitters, B.: A topos for algebraic quantum theory. Journal of Mathematical Physics **51**(3), 053510 (2010)

[17] Busemeyer, J.R., Bruza, P.D.: Quantum Models of Cognition and Decision. Cambridge University Press, ??? (2012)

[18] Pothos, E.M., Busemeyer, J.R.: Can quantum probability provide a new direction for cognitive modeling? Behavioral and Brain Sciences **36**(3), 255–274 (2013)

[19] Khrennikov, A.: Ubiquitous Quantum Structure: From Psychology to Finance. Springer, ??? (2010)

[20] Aerts, D.: Quantum structure in cognition. Journal of Mathematical Psychology **53**(5), 314–348 (2009)

[21] Aerts, D., Sozzo, S.: Quantum structure in cognition: Why and how concepts are entangled. In: International Symposium on Quantum Interaction, pp. 116–127 (2011). Springer

[22] Bruza, P.D., Wang, Z., Busemeyer, J.R.: Quantum cognition: a new theoretical approach to psychology. Trends in cognitive sciences **19**(7), 383–393 (2015)

[23] Khrennikov, A.: Quantum-like modeling of cognition. Frontiers in Physics **3**, 77 (2015)

[24] Scholes, G.D.: Quantum-like states on complex synchronized networks. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences **480**(2295) (2024) https://doi.org/10.1098/rspa.2024.0209

[25] Gunji, Y.P., Ohzawa, Y., Tokuyama, Y., Eto, K.: Quantum logic automata generalizing the edge of chaos in complex systems. Frontiers in Complex Systems **Volume 2 - 2024** (2024) https://doi.org/10.3389/fcpxs.2024.1347930

[26] Tokuyama, Y., Ohzawa, Y., Gunji, Y.-P.: Quantum logic automata generate class iv-like patterns and 1/f noise. BioSystems **246**, 105339 (2024) https://doi.org/

10.1016/j.biosystems.2024.105339

[27] Wolfram, S.: Universality and complexity in cellular automata. Physica D: Nonlinear Phenomena **10**(1), 1–35 (1984) https://doi.org/10.1016/0167-2789(84)90245-8

[28] Gunji, Y.-P., Haruna, T.: A non-boolean lattice derived by double indiscernibility. T. Rough Sets **12**, 211–225 (2010) https://doi.org/10.1007/978-3-642-14467-7_11

[29] YP Gunji, V.B. S Shinohara: Connecting the free energy principle with quantum cognition. Frontiers in neurorobotics **16**, 910161 (2022) https://doi.org/10.3389/fnbot.2022.910161

[30] Yao, Y., Yao, B.: Covering based rough set approximations. Inf. Sci. **200**, 91–107 (2012) https://doi.org/10.1016/j.ins.2012.02.065

[31] Gunji, Y., Nakano, T., Jin, Y., Adamatzky, A., Basios, V., Khrennikov, A.: Quantum logic in laughter: Standup and sketch comedy. Quantum Mach. Intell. **8**, 22 (2026)

[32] Uragami, D., Gunji, Y.-P.: Universal criticality in reservoir computing using asynchronous cellular automata. Complex Systems **31**(1), 103–121 (2022) https://doi.org/10.25088/ComplexSystems.31.1.103