POLITECNICO DI TORINO – UNIVERSITY OF LJUBLJANA

Master Degree course in Physics of Complex Systems

Master Degree Thesis

Exact Solutions of two integrable Cellular Automata with localized impurity

Supervisors
Prof. Tomaž Prosen, *University of Ljubljana*
Prof. Fabrizio Dolcini, *Politecnico di Torino*

Candidate
Dario Di Gregorio

Academic Year 2024-2025

Abstract

We study a $1+1$ dimensional integrable reversible cellular automaton (RCA54) with an impurity cell governed by a local rule different from the one that governs all the other cells. Building upon prior works, using matrix product and patch state ansatz techniques, we derive exact analytical solutions for the non-equilibrium steady states and identify eigenstates of the evolution operator associated with -1 eigenvalues, which correspond to discrete time-crystal phases. Coupling the system to stochastic boundary baths yields a Markov chain with deterministic bulk and stochastic boundary dynamics, where boundary parameters uniquely determine steady and time-crystal states. Our results provide exact insights into impurity dynamics in integrable discrete systems, revealing degenerate steady-state manifolds and emergent complex behaviour from simple local interactions. This work highlights the potential of integrable RCAs as discrete models for exploring non-equilibrium and impurity phenomena in mathematical physics and opens promising directions for future theoretical studies of impurity problems using discrete integrable models.

Contents

List of Figures

Chapter 1

Introduction

Cellular Automata (CA) are mathematical models consisting of grids made up of individual cells, where each cell can be in one of a limited number of possible states. The state of each cell changes over discrete time steps according to a fixed set of rules, based on the states of neighboring cells. The concept was introduced by John von Neumann in the 1950s during his work on self-replicating systems [1]. Despite being known for a long time and having been used for decades, it is just recently that cellular automata had their deserved attention. They have proven to be remarkably effective in modelling complex, dynamical phenomena across a wide range of disciplines. Indeed, despite their simplicity, these systems can model complex and often unpredictable behaviour that mirrors the richness of natural systems. In this sense, cellular automata are useful as computational and discrete analogues of physical systems, challenging the fundamental assumptions of continuous models. This feature makes Cellular Automata especially interesting in scenarios where continuous differential equations are inadequate or unamenable, such as in quantum gravity or special relativity [2–5]. CA have proven to be very effective in studying thermodynamics and statistical mechanics, offering interesting computational perspectives on the second law of thermodynamics. Furthermore, evolutions of lattice gas automata – such as the Lattice Boltzmann Method – have been applied to fluid dynamics and turbulent flows simulations [6]. These advances contributed to strengthen the link between discrete computational approaches and continuous physical theories. Another relevant field in which CA have been proven to be a powerful analytic tool is quantum theory. Indeed, they are believed to possibly reconcile quantum mechanics with a discrete spacetime structure. The idea that quantum behaviour could possibly emerge from deterministic CA at the Planck scale has been explored by Gerard 't Hooft [7], leading him to reinterpret quantum probabilities as epistemic rather than fundamental. In fact, Cellular Automata evolved into a fundamental tool to explore the structure of physical laws, allowing analyses about the nature of emergence, computation and relativity. CA are used also in studying the emergence phenomenon. For instance, John Conway's Game of Life is emblematic and is a proof of how simple and local rules can lead to the emergence of complex structures and long range correlations. Finally, CA challenge also our concept of universe itself, since they question the assumption universe is described by continuous mathematics [8]. In fact, no CA has yet successfully reproduced all known physical

laws, but their proved usefulness in gaining information on various and complex systems continues to increase the interest towards these discrete, abstract computational systems.

A particularly interesting class of CA is the one of reversible cellular automata (RCA), where each configuration has a unique predecessor in time. These discrete dynamical systems can uniquely evolve backward in time and exhibit a large number of conserved quantities that constrain dynamics, often enabling one to find exact solutions. In particular, the relevance of reversibility stems from the time-symmetry shown by fundamental physical laws that – unlike thermodynamic or biological systems – are fundamentally reversible, both in classical mechanics and quantum theory. The simplicity of RCA models makes them attractive as toy models for physical systems where evolution is bijective and information loss is forbidden. Moreover, while solvability of PDEs is typically not possible when discretisation is introduced, integrable RCAs can retain exact solvable structures even in a fully discrete setting [9]. This proves how integrability can manifest also when spacetime is no longer assumed to be continuous. In addition to their possible generalisation into Quantum Cellular Automata (QCA) – especially reversible and unitary ones – integrable RCA can be used to clarify the structure of conserved dynamics in discrete systems and contribute to the formulation of discrete, integrable field theories. Moreover, their reversibility can be useful in studying how time-symmetric laws govern the evolution of complex many-body systems.

In this thesis we focus on a $1+1$ dimensional (time + space) integrable RCA54 – i.e., an integrable RCA evolving according to a dynamical law known as "rule 54". Moreover, the model is enriched by the presence of an impurity, i.e., a cell that obeys a specific rule, different from all other cells. We show that it is possible to solve this system analytically and provide exact solutions through matrix product states techniques. In particular, following the works [9, 10], we investigate how to achieve a good understanding of equilibrium and non-equilibrium statistical physics of the aforementioned RCA with impurity by means of a matrix product ansatz and a patch state ansatz. The former has been used to retrieve the analytical solutions of the non-equilibirum steady states, while the latter allowed us to find the eigenstates that represent time-crystal states. By coupling the system boundaries to stochastic thermal baths that inject and absorb solitons, and subsequently integrating out the reservoir degrees of freedom, the resulting dynamics of the system can be described as an exact Markov chain. In this model, the cells in the bulk – both the ones governed by rule 54 and the impurity cell – are updated deterministically, while the cells near the boundaries evolve stochastically. The set of parameters characterizing the boundary driving uniquely determines both the non-equilibrium stationary state (NESS) that the system approaches at long times and the time-crystal state, approached under specific circumstances (see sections 4.2 and 5.2).

Chapter 2

Definition of *Cellular Automaton*

A one-dimensional cellular automaton consists of a line of sites, with each site carrying a value 0 or 1 (or in general $0, \ldots, k-1$). The value a_i of the site at each position i is updated in discrete time steps according to an identical deterministic rule ϕ depending on a neighbourhood of sites around it [11], *i.e.*,

$$a_i^{(t+1)} = \phi[a_{i-r}^{(t)}, a_{i-r+1}^{(t)}, \ldots, a_{i+r}^{(t)}] \tag{2.1}$$

Even for small k and the "interaction range" r set to $r = 1$, the general behaviour can be very complex. Depending on the kind and the locality of ϕ, i.e., on its characteristics, it can produce simple or extremely complicated patterns. Empirical studies suggest that the patterns can evolve according to four qualitative forms [11]; the pattern can

1. disappear with time
2. evolve to a fixed finite size
3. grow indefinitely at a fixed speed
4. grow and contract irregularly

In a cellular automaton, if each of the k possible values can be assigned to each site with an independent equal probability we can notice self-organisation, such that ordered structures emerge from disordered initial states configurations and in some cases complexity is evident.

Remark. Different initial states with the same cellular automaton rule yield patterns that differ in detail but are similar in form and statistics; different rules yield very different patterns [11,12].

Nevertheless, another empirical study [11] suggests that four qualitative classes may be identified, yielding four characteristic limiting forms,

1. spatially homogeneous state
2. sequence of simple stable or periodic structures
3. chaotic aperiodic behaviour
4. complicated localized structures of which some are propagating

Cellular automata of the same class exhibit qualitatively similar behaviour, regardless of their differences.

Remark. Because of such universality, general results on these classes become applicable to a wide variety of systems modelled by cellular automata.

Cellular automata provide complementary models describing the discrete evolution of many (identical) components. Cellular automata models are useful in highly nonlinear regimes and in systems in which discrete thresholds occur. They are also particularly suitable when growth inhibition effects are important [11]. Some cellular automata can be used to model

- growth of dendritic crystals [13, 14]
- stochastic self-propagation of star formations [15]
- nonlinear chemical reaction-diffusion systems [16]
- fluid mechanics [6, 17]

It can be proven that if a probabilistic noise is added to the time evolution rule (2.1) the cellular automata resulting can be identified as a generalised Ising model [11, 18]. Moreover, according to the previous statement it is possible to show that phase transitions may occur if the rule ϕ retains some deterministic components or in dimensions $d > 1$.

2.1 Reversible Cellular Automata

Definition. *A rule is said to be* reversible *if it is backward deterministic, i.e., if each configuration or set of m configurations for rules that are m-th order in time, has a unique predecessor [18].*

The reversed rules are typically noted with an R before the name of the rule.

Construction of a reversible CA (Fredkin) The idea is to take any rule ϕ involving k states per cell, note the value it returns and subtract from it, in modulo k arithmetic, the value that the central cell assumed at time $t \sim 1$,

$$C^{t+1} = \phi(\text{neighbourhood at time } t) - C^{t-1} \,(\text{mod}(k)) \tag{2.2}$$

ϕ is a function that takes into account the value of a certain number of digits – one for each neighbour and the value of the considered, evolving, cell – and returns a single digit, that is the evolved state of the considered cell, so ϕ is never reversible.

Remark. Notice that a cell is an element of the CA governed by a certain rule and which can assume k states. A *site* is an element of the discrete physical system modelled by the CA. Later in this thesis the two terms will be used as synonyms since the CA cells will exactly correspond to the discrete elements of the physical system.

It is worth noticing this relation can be solved uniquely for C^{t-1}, even if ϕ is not invertible, and this feature makes the new rule reversible. For $k = 2$ the subtraction modulo 2 is simply the XOR (exclusive OR) Boolean operation ($\oplus$) [18]. The new rule is now second order in time, but it can be made first order by endowing each cell with k^2 states. The rules obtained with (2.2) are also *time-reversal invariant*, this means that a sequence of configurations can be obtained in reverse order with the *exact same* rule, simply by inverting the last two configurations. It is worth noticing that not all reversible rules are time-reversal invariant.

Relation with equilibrium The time-reversible invariance can be seen only if the system is in a non-equilibrium state, since in that case we are able to "see the arrow of time" by looking at the evolution of the system, i.e., by looking at the relaxation process towards the equilibrium. Once the equilibrium is reached, the former arrow disappears and time-dependence is no more relevant. It reappears, however, if two consecutive configurations are inverted, in an exact realisation of Loschmidt's paradox[1] [18].

Phase-space The second order dependence on time makes two consecutive configurations act somewhat like a point in a phase-space, so many of the results of classical mechanics still hold. The absence of mergers, for example, together with the discrete structure of the phase-space make the density of points invariant in a reversible evolution so the Liouville's theorem still holds.

2.1.1 Some remarks about Irreversibility

Information conservation As previously stated, not all reversible rules are time-reversal invariant, which means that even if the system evolves in what seems a totally random regime, there exists a determinism of the backward-forward dynamics. In "real physics"[2], as Vichniac pointed out: "because of the macroscopic nature of possible measurements" typically a diffusion into many body correlations is irreversible; in cellular automata we can access easily macroscopic degrees of freedom, so we can extract – out of the many-cell correlations – the encoding of an initial condition by applying the reverse rule [18].

Absence of attractors The fascinating self-organising behaviour discovered by Wolfram [12] is not possible for reversible rules. The self-organising effect requires *mergers*[3] in the evolution

[1]The paradox states that it should be impossible to deduce an irreversible process from time-symmetric dynamics. In this sense, the time-reversal symmetry of (almost) all known low-level fundamental physical processes seems to be in conflict with the second law of thermodynamics which describes the behaviour of macroscopic systems, hence the paradox.

[2]According to Stan Ulam's definition, "imaginary physics" are those views according to which a CA is a mere computational tool or an original model for some actual physical phenomena, while "real physics" is the object of the approach that sees a CA as a fully discrete dynamical system.

[3]The term *merger* is typically referred to a phenomenon or event where two or more patterns or structures come together and combine into a single structure or behaviour during the evolution of the cellular automaton.

of the cellular automaton and these mergers are the ones which permit the irreversible rules to circumvent the second law of thermodynamics and *create order from chaos* [18]. On the other hand, when we are looking for analogies with mechanics, the time-reversal invariance becomes crucial.

Chapter 3

Definition of the model

We consider a deterministic dynamics defined on a $1+1$ (space and time) dimensional lattice whose points are labelled by pairs $(x,t) \in \mathbb{Z}^2$ and a field variable s_x^t which can only assume boolean values, such that $s_x^t \in \mathbb{Z}_2 = \{0,1\}$. We restrict the dynamics to an even/odd staggered sublattice – inside the light-cone, i.e., the cone of fastest particles' maximum velocity – of $\mathbb{Z}^2$ labelled by points (x,t) such that $x+t = 0 \pmod 2$. In this sublattice odd and even cells update alternately. In this thesis work I shall consider a system with an even number n of sites where the update of a single cell at time $t+1$ depends on the state of the neighbours and the cell being updated itself at time t (rectangular updating); in our work, instead, the update of a cell at time $t+1$ depends on the states of the first neighbours at time t and its previous state at time $t-1$. This is expressed by the following evolution law

$$s_x^{t+1} = \chi(s_{x-1}^t, s_x^{t-1}, s_{x+1}^t) \tag{3.1}$$

where $\chi : \mathbb{Z}_2^3 \to \mathbb{Z}_2$ is a certain binary function that is the rule of the cellular automaton. In our case, the function χ is defined as

$$s_2' = \chi(s_1, s_2, s_3) \equiv s_1 + s_2 + s_3 + s_1 s_2 \pmod 2 \tag{3.2}$$

and is known in the CA literature as the "rule 54"[1].

Pictorially, this updating rule can be simply represented using diamond-shaped cells, thus a single line of cells acquires the form of a saw. In particular, the 54 rule can be represented as follows

[1]The numbering of cellular automaton rules follows a convention introduced by Stephen Wolfram (1983) in his classification of elementary CA [12]. Each rule defines the output for all 8 possible combinations of three neighboring binary cells (left, center, right). These outputs form an 8-bit binary sequence, which is then interpreted as a decimal number from 0 to 255. In this case, Rule 54 corresponds to the binary sequence 00110110, which produces a specific set of outputs for each neighborhood configuration. This numerical labeling provides a compact and systematic way to reference and study the behaviour of CA rules.

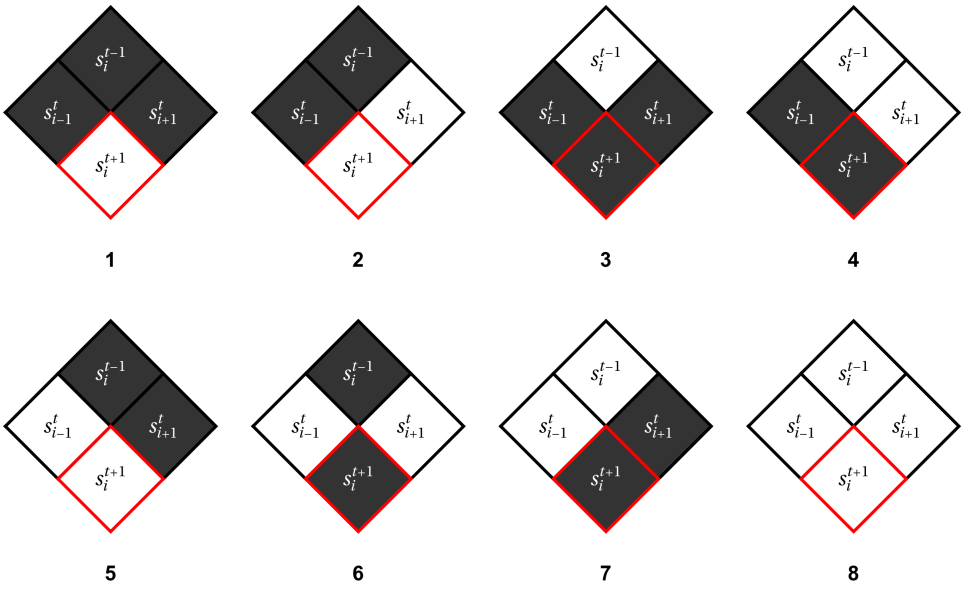


Figure 3.1: Rule 54

where the red encircled diamond is the updated cell (here time is directed downwards), the dark grey ones are cells in state 1 and the white ones are in 0.
As discussed before, we define the system on a finite lattice $\mathbb{Z}_n$ of even size n, then we couple the system edges to a stochastic reservoir of solitons. We then obtain a perfect Markov chain model in which – by removing the degrees of freedom of the reservoirs – the bulk cells are updated deterministically and the evolution of the cells near the boundaries is stochastic. The set of all configurations of the lattice – following the notation in [9] – is

$$\overline{s} = (s_1, s_2, s_3 \ldots, s_n) \equiv (s_1^1, s_2^0, s_3^1, \ldots, s_n^0) \in \mathcal{C}_n$$

where $\mathcal{C}_n = \mathbb{Z}_2^{\times n}$ and the alternation of 1 and 0 as time steps is due to the saw dynamics discussed before.

3.1 Definition of *Impurity* and Nomenclature

Definition (Impurity in a Cellular Automaton)**.** *Given a RCA with a defined global rule, an impurity is a specific cell located in the bulk that, at every time step or within a specified time interval, behaves according to a rule different from the global one.*

One of the most relevant issues that arise when dealing with a cellular automaton in which an impurity is inserted is the behaviour of the automaton around it. First, we aim to introduce a formal definition of the impurity and discuss its nomenclature. The location constraint arises from the possibility that the impurity may be large enough to cover a significant area of the cellular automaton; however, if such an area reaches the stochastic boundaries, we are no longer

dealing with an impurity, but with a discontinuity instead. Since an impurity can be located in any cell or group of adjacent cells of the CA support, and since each impurity can have its own rule, for the sake of clarity, we shall adopt the following nomenclature. Let us denote the Reversible Cellular Automaton (RCA) with a global rule α as RCAα, and a generic CA with impurity with the same rule as IRCA$_\alpha$. The impurity rule β can then be expressed as IβRCAα. We also introduce a convention according to which this notation means the impurity is located at the centre of the CA, for any lattice size. To specify the location of the impurity, we could use the cell a; in our case, we work with a one-dimensional RCA54 with a single impurity, and the general name for this configuration could be I$_a\beta$RCA54.

3.2 Interpretation of the model

The presented dynamics can be interpreted as a soliton gas [19, 20] where the time evolution of a single particle is the translation to the left or to the right, forward or backward in time, by d units in T time steps. Indeed, the model describes the dynamics of solitons that, in the absence of scattering processes, travel with velocity $v = \pm 1$ where the velocity is defined as $v = d/T$ [21]. According to Scott *et al.*, «a soliton $\phi_s(x - vt)$ is a solitary wave solution of a wave equation which asymptotically preserves its shape and velocity upon collision with other solitary waves. That is, given any solution $\phi(x, t)$ composed only of solitary waves for large negative time, [...] such solitary waves will be called solitons if they emerge from the interaction with no more than a phase shift» [22]. The idea of soliton can be naturally extended to CA as a wave identifying a sequence of cells with non-zero values, which propagates with a constant and finite velocity [19]. Solitons may scatter pair-wise or against the impurity. If the first scattering produces a time-shift of both solitons in time or space (Figure 4.1d), the scattering depends on the nature of the impurity and should be analysed case-by-case. Nevertheless, as the diagram above can be read as the description of all the possible local configurations of the dynamics and it is possible to derive from it the pair-wise scattering dynamics, it is also possible to derive the soliton-impurity scattering dynamics by combining the diagrams above with the ones related to the impurity rule, where the impurity will be the updating central cell. Observing the diagrams of rule 54 it is possible to notice that diagram 8 represents the dynamics of empty space, diagrams 2, 7 and 4, 5 show the free-soliton dynamics – left and right movers respectively – and the others display the scattering dynamics between two oppositely-moving solitons at different time steps. In particular, the two solitons firstly merge (diagram 3), then temporarily disappear (diagram 6) to reappear in the next step (diagram 1) [9].

3.3 Boundary driving and general equilibrium states

Since the dynamics is not completely deterministic, due to the presence of the stochastic updates of pairs of boundary cells (s_1, s_2) and (s_{n-1}, s_n), a probabilistic description is needed. The probability distribution over $\overline{s}$ is the *macroscopic* state $\boldsymbol{p}$ and can be interpreted as a 2^n-dimensional vector

$$\boldsymbol{p} = (p_{\overline{s}})_{\overline{s} \in \mathcal{C}_n} \in \left(\mathbb{R}^2\right)^{\otimes n}$$

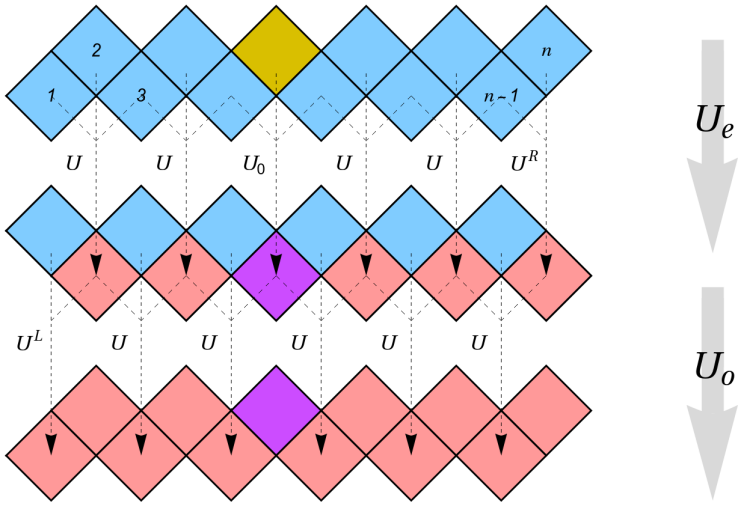


Figure 3.2: The two half-time steps in which we divided the single-period evolution operator are made by two site boundary stochastic maps, acting on the left U^L and on the right U_R end of the chain, and a three-site bulk deterministic maps. The rule 150 maps are denoted by U and the impurity one is denoted by U_0. The blue and red diamonds denote the sites before and after the update, the yellow and violet cells are the impurity cell before and after the update according to the local impurity rule.

where $p_{\overline{s}} \geq 0$ is the probability of the single configuration $\overline{s}$. We also assume that the boundary updates are Markovian – i.e., the probability of s_1 (as of s_n) to change the state depends only on the cells among the first space-time neighbourhood, namely (s_1, s_2) and (s_{n-1}, s_n). Following [9], we define the time evolution of the probability state $\boldsymbol{p}(t)$ starting from an initial probability state $\boldsymbol{p}(0)$, as

$$\boldsymbol{p}(t) = \mathcal{U}^t \boldsymbol{p}(0) \tag{3.3}$$

where the single-period propagator $\mathcal{U}$ is a $2^n \times 2^n$ matrix. Due to the temporal structure of the dynamics, we can divide this operator into two half-time steps, such that

$$\mathcal{U} = \mathcal{U}_e \mathcal{U}_o$$

where $\mathcal{U}_e$ is the *even* cells propagator and $\mathcal{U}_o$ is the *odd* cells one. This factorisation is shown in Figure 3.2. The process can be now parallelised, so we can update odd and even sites separately, according to

$$\begin{aligned} \mathcal{U}_e &= U_{1,2,3} U_{2,3,4} \dots U_{n-3,n-2,n-1} U^R_{n-1,n} \\ \mathcal{U}_o &= U^L_{1,2} U_{3,4,5} \dots U_{n-2,n-1,n} \end{aligned} \tag{3.4}$$

where $U_{j-1,j,j+1} = \mathbb{1}_{2^{j-1}} \otimes U \otimes \mathbb{1}_{2^{n-j-1}}$ and U is an 8×8 matrix. The former acts non-trivially on three adjacent sites. In particular, $U_{j-1,j,j+1}$ makes the cell at position j evolve depending on

the values of the other two neighbouring cells $j-1$ and $j+1$, according to rule 54. Specifically U can be constructed as

$$U_{(t,t',t''),(s,s',s'')} = \delta_{t,s}\delta_{t',\chi(s,s',s'')}\delta_{t'',s''}$$

where $\chi(s,s',s'')$ is the rule 54 update function. Similarly, we can construct an operator U^0 that acts only on the impurity cell and makes it evolve according to the impurity rule

$$U^0_{(t,t',t''),(s,s',s'')} = \delta_{t,s}\delta_{t',\phi(s,s',s'')}\delta_{t'',s''}$$

where $\phi(s,s',s'')$ is now the impurity rule. Concerning the boundaries, their update is determined by the operators

$$U^L_{1,2} = U^L \otimes \mathbb{1}_{2^{n-1}},$$
$$U^R_{n-1,n} = \mathbb{1}_{2^{n-1}} \otimes U^R$$

U^L and U^R are the ones in [10, 23] and they are stochastic, this implies that $\mathcal{U}$ is a stochastic matrix that conserves the probability. Moreover, the boundary matrices satisfy the following commutation relations

$$\left[U^L_{1,2}, U_{2,3,4}\right] = 0, \quad \left[U_{1,2,3}, U^R_{3,4}\right] = 0$$

These relations imply that the boundary operators act essentially on the boundary cell, while the adjacent one serves primarily as a control cell. Another constraint to the boundary operators comes from the existence of a non-equilibrium steady state (NESS) – an eigenvector $\boldsymbol{p}_0$ corresponding to an eigenvalue 1, that is

$$\mathcal{U}\boldsymbol{p}_0 = \boldsymbol{p}_0 \tag{3.5}$$

The two families of boundary driving operators are known as the *Bernoulli driving* [23] and the *Conditional driving* [10]. The use of these two classes of boundary driving is motivated by the fact they preserve the integrability of the system, in this thesis we use the *Conditional driving* only, that has the form

$$U^L = \begin{pmatrix} \alpha & 0 & \alpha & 0 \\ 0 & \beta & 0 & \beta \\ 1-\alpha & 0 & 1-\alpha & 0 \\ 0 & 1-\beta & 0 & 1-\beta \end{pmatrix} \qquad U^R = \begin{pmatrix} \gamma & \gamma & 0 & 0 \\ 1-\gamma & 1-\gamma & 0 & 0 \\ 0 & 0 & \delta & \delta \\ 0 & 0 & 1-\delta & 1-\delta \end{pmatrix} \tag{3.6}$$

where $\alpha, \beta, \gamma, \delta \in [0,1]$ are some driving rates parametrising the left and the right bath. The boundary driving operators can be interpreted as mechanisms that inject or absorb soliton-like excitations at the system boundaries with certain probabilities. A schematic representation of this process is shown in Figure 3.3.

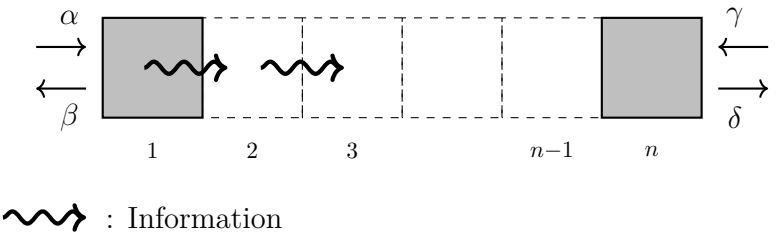


Figure 3.3: Schematic representation of soliton-like information flow in a reversible cellular automaton driven by stochastic boundary dynamics.

3.4 Existence of finite-size solutions

The probability vector can also be represented using infinite-size matrix product ansatz or other approximate or asymptotic methods; the current focus is on identifying finite-dimensional, closed-form solutions that arise directly from the spectral structure of $\mathcal{U}$. It is possible to determine whether a finite solution exists in our model, by looking at the eigenvalues and eigenvectors of the $\mathcal{U}$ operator. In particular, looking at the spectrum, it is possible to construct the latter generating all the U and U_0 operators and multiplying them. To deal with the boundary operators, it is crucial to set some values for the boundary parameters $\alpha, \beta, \gamma, \delta$ (see the code in Appendix 8). The existence of eigenvalues 1 of the evolution operator $\mathcal{U}$ indicates the presence of a NESS, one per every eigenvalue, while the existence of -1 eigenvalues is typically associated with the presence of oscillating time-crystal states [24]. In fact, any eigenvalue of the form $e^{2\pi i \frac{m}{\tau}}$ would imply a time-crystal state with time periodicity $t = \tau$. Our case is $\tau = 2$, which means a 2-periodic time crystal state is present. The presence of more than one 1 or -1 eigenvalue – i.e., degeneracy in the eigenvalue spectrum – indicates the existence of more NESS and time-crystal states. The eigenvalue equation in the latter case would be

$$\mathcal{U}\boldsymbol{p}_0 = (+1)\boldsymbol{p}_0 \qquad \text{(NESS)} \tag{3.7}$$

$$\mathcal{U}\boldsymbol{p}_{tc} = (-1)\boldsymbol{p}_{tc} \qquad \text{(time-crystal)} \tag{3.8}$$

that means that, expanding $\boldsymbol{p}(0)$ in its components, $\boldsymbol{p}(0) = \sum_{k=1}^{2^n} c_k u_k$, equation (3.3) can be written as

$$\begin{aligned} \boldsymbol{p}(t) &= \mathcal{U}^t \boldsymbol{p}(0) = \\ &= \sum_{k=1}^{2^n} \lambda_k^t c_k u_k \simeq \\ &\simeq \sum_{i:\lambda_i=1} c_i u_i + \sum_{i:\lambda_i=-1} (-1)^t c_i u_i \end{aligned} \tag{3.9}$$

where $i = 1, \ldots, 2^n$. In general, one may find eigenvectors that lie on the unit circle but are not 1, they would correspond to persistently oscillating eigenvectors. Some analogous eigenvectors have been found for quantum Markov Processes [25–27]. Observing the eigenvalue spectrum we can thus determine if the system allows a time-crystal state, while a stochastic evolution matrix

will always imply the existence of a NESS. On the other hand, the eigenvectors, lying in an Hilbert space, can be decomposed by means of the Schmidt decomposition, and by looking at the dimension of the decomposition (Schmidt rank) it is possible to determine if the solution related to a certain eigenvalue is possible or not.

3.4.1 Schmidt decomposition in classical systems

In order to understand the concept of Schmidt decomposition and why it is useful in this case, we can start thinking about a quantum many-body state $|\psi\rangle$. Here, quantum correlations are quantified by the entanglement entropy and it can be shown that the latter is rigorously connected to the required product state dimension [28]. This can be seen by considering a chain-system of length N and dividing it into a left and a right block, namely $L = \{1, \dots, \ell\}$ and $R = \{\ell + 1, \dots, N\}$, then expanding the original state as

$$|\psi\rangle = \sum_{i,j} \psi_{i,j} |i\rangle_L |j\rangle_R$$

where $\psi_{i,j}$ is the $k^\ell \times k^{n-\ell}$ matrix of amplitudes. Then a singular-value-decomposition (SVD) is performed to expose correlations of the state $|\psi\rangle$ as $\sum_{i,j} \psi_{i,j} = UDV^\dagger$, where U, V are unitary matrices and D is the diagonal matrix of all real non-negative entries. This results in the aforementioned Schmidt decomposition of $|\psi\rangle$, as

$$|\psi\rangle = \sum_{\nu=1}^{X} \rho_\nu |L_\nu\rangle |R_\nu\rangle$$

where $X = \min\{k^\ell, k^{n-\ell}\}$ is the Schmidt rank, $\{L_\nu\}$ and $\{R_\nu\}$ are the Schmidt states and $\{\rho_\nu\}$ are the Schmidt coefficients. As a result of the SVD, the Schmidt coefficients are always real and non-negative. Moreover, if the state is normalised, the sum of the squares of the Schmidt coefficients is equal to one. The two Schmidt states come from the columns of two unitary matrices of the SVD while the coefficients are the singular values specified by the diagonal matrix of the SVD arranged in nonincreasing order with ν.

In classical systems a similar reasoning can be done, since the state of a system of n sites and k local configurations is described by a probability vector

$$|\pi\rangle = \sum_j \pi_j |j\rangle$$

where we have k^N non-negative components $\{\pi_j\}$ that correspond to the probabilities that the system is in each configuration $\{j\}$. Bipartitioning the system as before we can expand $|\pi\rangle$ as

$$|\pi\rangle = \sum_{i,j} \pi_{i,j} |i\rangle_L |j\rangle_R$$

Through this result, following [29], we may draw an analogy to the Schmidt form for quantum states decomposing π as follows

$$\pi_{i,j} = \sum_{\mu=1}^{X} p_\mu \pi_L(i|\mu) \pi_R(j|\mu)$$

where $\{p_\mu\}$ is a set of probabilities arranged in nonincreasing order with μ, mixing the conditional marginal probability distributions $\pi_{L,R}$. Indeed, we can rewrite the state $|\pi\rangle$ in an elegant way, resembling the Schmidt decomposition

$$|\pi\rangle = \sum_{\nu=1}^{X} p_\nu |L_\nu\rangle |R_\nu\rangle \tag{3.10}$$

where $\{\pi_L(i|\nu)\}$ and $\{\pi_R(j|\nu)\}$ are elements of $|L_\nu\rangle$ and $|R_\nu\rangle$ respectively. As in the quantum case, a repeated bipartite decomposition of the system as in (3.10) can be used to build an approximate product ansatz $|\varpi\rangle$ having – as shown in [29] – an error bounded by

$$|| \, |\pi\rangle - |\varpi\rangle || \leq \sum_{\ell=1}^{N-1} \sum_{j=\epsilon}^{X} p_j^{[\ell]}$$

where $p_j^{[\ell]}$ is the probability of mixing the conditional marginal probability distributions for a bipartite system in ℓ, and ϵ is the required dimension of the product ansatz to describe a system having arbitrarily strong correlations. The power of this decomposition lies in the fact that we need a matrix or patch dimension – i.e., the dimension of the matrix or the patch we use to model the system – typically $\epsilon \ll X$ [30] or – at most, for strongly interacting systems – the Schmidt rank itself. Thus, given an eigenvalue λ of the evolution operator, the dimensions of the eigenvectors related to λ indicate the dimensions of our solutions, e.g., a 3-dimensional eigenstate indicates we can use a 3-state patch or a 3×3 matrix to represent the system using a patch state ansatz (PSA) or a matrix product ansatz (MPA) respectively. For instance, in the model I150RCA54 of cellular automaton with impurity that will be discussed in next chapter, the smallest dimension of an eigenvector associated with the eigenvalue $\lambda = 1$ is $X = 3$, while the smallest one of the eigenvectors associated with $\lambda = -1$ is $X = 2$. This implies we can find a NESS probability state-vector $\boldsymbol{p}_0$ with dimension $X = 3$ and a time crystal probability state-vector $\boldsymbol{p}_{tc}$ with dimension $X = 2$.

Chapter 4

The I150RCA54 model for RCA with impurity

4.1 Action of the rule 150 bulk impurity in a RCA54

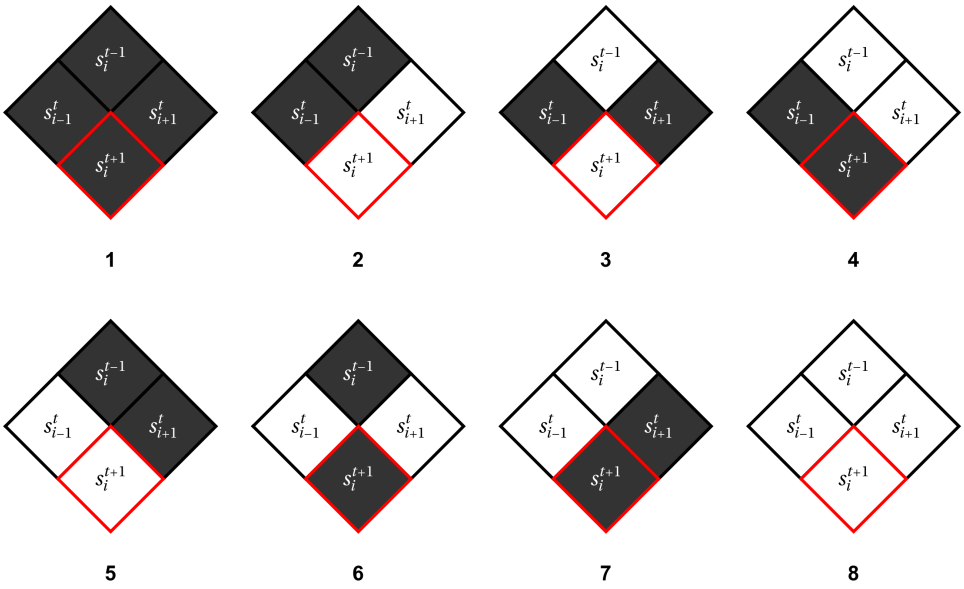
Here we focus on the model I150RCA54 of a RCA with impurity, that means we assume a $1+1$ dimensional RCA with global rule

$$\chi(s_1, s_2, s_3) = s_1 + s_2 + s_3 + s_1 s_3 \pmod 2 \tag{4.1}$$

that is rule 54, and an impurity located at the centre of the RCA with the non-interacting rule 150

$$\phi(s_1, s_2, s_3) = s_1 + s_2 + s_3 \pmod 2 \tag{4.2}$$

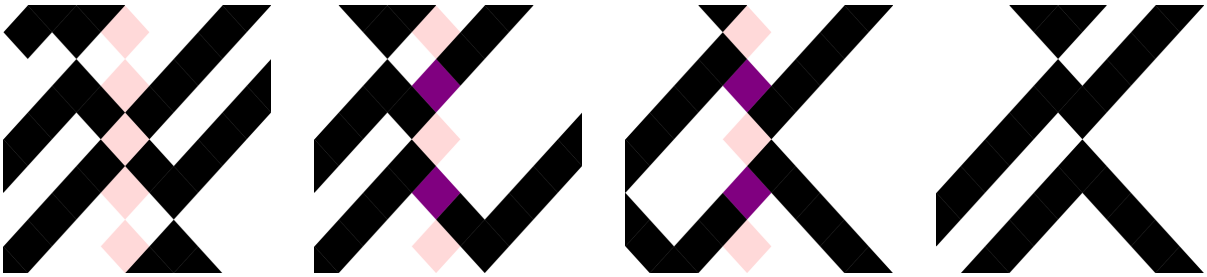
As above, the red encircled diamond is the updated cell (time is directed down), the dark grey ones are cells in state 1 and the white ones are in 0. Firstly, note that a cellular automaton with impurity remains reversible if and only if both its global and its impurity rules are reversible. Rule 54 is indeed reversible, and so it is rule 150, so the entire CA with impurity is therefore reversible. Discussing the dynamics, the left and the right movers across the impurity are here locally shown in diagrams 2, 7 and 4, 5 respectively (Figures 4.1b, 4.1c), while the last diagram represents the empty space in and around the impurity. According to the definitions of left- and right-movers given in [9], it is straightforward to observe that if a mover encounters the impurity alone, it will pass through without altering its path. This means that any interaction between two movers that occurs before (i.e., to the left of) or after (i.e., to the right of) the impurity will be exactly analogous to the interactions governed by the global rule.



Rule 150

The only interesting case is when the two movers meet at the impurity. Diagram 3 shows that two solitons that meet at the impurity do not merge but disappear instead. The very same diagram shows, combined with the rule 54 ones 3.1, that they reappear in the next time step, with no time loss (Figure 4.1a). In this scenario, the impurity is "overstepped", and the two movers continue their path as if they had never met. While this result might seem trivial at first glance, its explanations could be due to profoundly different mechanisms. Indeed, it is possible that the two solitons simply sum at the impurity and our model yields a state 0 because of how it has been defined; however, it is also possible that when the two solitons scatter simultaneously at the impurity they behave like two masses of a Newton's pendulum, that is each soliton is reflected by the impurity because of the presence of the other soliton on the other side; a third option is that an absorption-emission phenomenon occurs. Specifically, the two movers are absorbed by the cell and then re-emitted in the following time step. The question arises as to what kind of mechanism physically allows for the absorption to occur only when both movers are present. This happens only when the impurity is in the 0 state, otherwise the local dynamics is the one shown in the first and the sixth diagrams – i.e., if there are no solitons the impurity will be again in state 1 (diagram 6) but if two solitons simultaneously hit the impurity, immediately after it will emit two solitons (diagram 1) keeping its 1 value, going back to diagram 6 and then to diagram 1. This oscillation leads to a *spontaneous emission* occurring every two time steps: the gas of solitons behaves like a time-crystal.

4.2 Triggered 150 bulk impurity and spontaneous emission



(a) (b) (c) (d)

Figure 4.1: The dark-purple cell in this simulation is the impurity one in state 1 and is governed by the non-interacting rule 150. (a) "Simultaneous" vertex between a left-mover and a right-mover meeting at the non-triggered impurity (light-purple diamond); in this case, the impurity behaves as if it is skipped by the two movers. (b, c) "Sequential" vertex between a right-mover and a left-mover in sequence; in this case the former reaches the impurity a step before the latter and it results in "jump" of the movers. This case is analogous to the pure-54 rule (d); in fact, when a single mover encounters the impurity at a given time step, it passes through without altering its path. The black diamonds represent the bulk sites in state 1, the light-purple diamonds represent the impurity site in state 0, and the dark-purple diamonds represent the impurity site in state 1.

As discussed above, a far more interesting behaviour can be found by triggering the impurity rule at a certain time t^*. We assume the cell with impurity acquires value 1 when a right-mover and a left-mover meets simultaneously at the impurity. This will lead to the behaviour shown in Figure 4.2. The action of this rule, localized at the impurity, together with rule 54 that governs its neighbourhood, makes a spontaneous emission phenomenon to occur, which is rather interesting from many aspects: it is a two-particle, symmetric, pulsating, and harmonic emission. When a right and a left mover meet at the impurity (see Figure 4.2a) and the impurity cell acquires value 1 under triggering, a spontaneous emission begins and it is indeed quite consistent since, once started, no way to extinguish it has been found yet, neither by turning off the boundary emission mechanism after a certain time, nor by increasing it to make the interactions frequent enough to destabilize it. Furthermore, as stated, the emission is not casual but happens every two time-steps (pulsating, harmonic behaviour), independently of what happens in the neighbourhood of the impurity, and consists in a right and a left mover (two-particle) simultaneous (symmetric) emission. Both mechanisms are illustrated in Figure 4.2, where the first simultaneous interceptions of the impurity results in a nonrelevant mechamism (Figure 4.1a), while at t^* a symmetric emission is triggered. Due to the very small emission and high absorption probabilities, as well as the low chemical potential of the boundaries, the figure clearly represents the behavioural structure of the system.

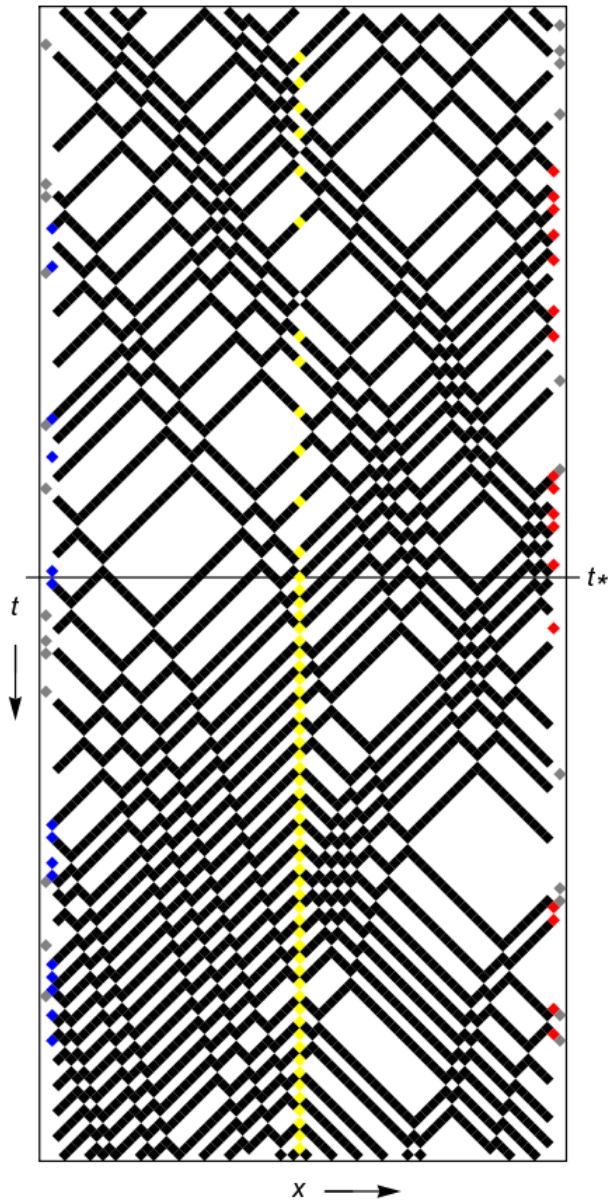


Figure 4.2: Once the impurity is triggered - at time t^* - by the meeting of a left-mover and a right-mover and the appearing of an impurity-cell of value 1, it begins emitting particles every two even-time steps (a), i.e., after two updates of the central (even) cell and not just one. This behavior is of particular interest, because it is the reason why the Perron-Frobenius theorem does not hold in this scenario and results in two degenerate non-equilibrium steady states (NESS). Moreover, the non-interacting behaviour of rule 150 is clearly visible when meeting between a right-mover and a left-mover happens without any trigger. In this case, the impurity behaves as if it is skipped by the two movers (see Figure 4.1a). This figure has been generated by simulating a boundary-driven I150RCA54 with a probability of information injection of 0.1 from the boundaries, an absorption probability of 0.6, and a chemical potential of 0.1 for both the left and right boundaries. The very low probability of emission and the small values of the chemical potentials have been chosen to make the figure more understandable, avoiding confusion that could arise from a large number of interactions.

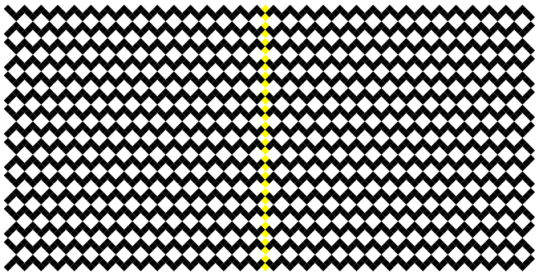
(a)

4.2.1 Spontaneous emission and stochastic drive

The stochastic boundaries play the role of a thermal excitation due to a system-bath coupling. In the absence of the stochastic drive, it is reasonable to suppose the system ends up evolving in a conservative way, i.e., maintaining the same amount of information in it. Interestingly,

however, the spontaneous emission doesn't care if the driving has been turned off, but continues undisturbed. This behaviour can be easily seen by looking at the non-driven cellular automata that is, in fact, the bulk of the system decoupled from the boundaries. The system continues emitting anyway, as shown in Figure 4.3a. It is worth noticing that, once begun, nothing else can happen at the impurity but the spontaneous two-particle emission.

(a)



(b)

(c)

Figure 4.3: Once the impurity is triggered and, the symmetric spontaneous emission begins (a). This behaviour under triggering is the one that breaks the holographic ergodicity, since starting from the "final" configuration we could not retrieve the initial one (b). We can force the system to go exactly backwards by imposing a local initial condition at the second time step, such that the configuration $\underline{s}'(1) \equiv \underline{s}(t_+ - 1)$, (c). This figure has been generated by simulating a non-boundary-driven I150RCA54.

4.3 NESS: Matrix-Product-Ansatz (MPA)

Our analysis in this section follows the methodology outlined in [9] to find the exact solutions for the two NESS our system exhibits. In particular, we look for a probability vector $\boldsymbol{p}_0 = (p_{\bar{s}})$

that, under the dynamical evolution (3.3), is mapped into itself, thereby satisfying the equation

$$\boldsymbol{p}_0 = \mathcal{U}\boldsymbol{p}_0 \tag{4.3}$$

In order to do it, we use the MPA in the auxiliary space $\mathcal{V}_a$. Following the conventions in [9], we denote

- row and column vectors using the Dirac bra-ket notation;
- vectors in physical space in bold Roman letters;
- the subscript of an operator or a vector in the physical space is now the site position in the tensor product physical space (in our case $\left(\mathbb{R}^2\right)^{\otimes n}$) on which it acts nontrivially;
- physical state components using a binary index and in corresponding non-bold font;
- matrices (operators) acting on physical space with capital Roman letters.

According to (3.4), we can decompose the eigenvalue problem as, assuming the impurity is on an even site,

$$\mathcal{U}_e\boldsymbol{p} = \Lambda_R\boldsymbol{p}', \quad \mathcal{U}_o\boldsymbol{p}' = \Lambda_L\boldsymbol{p}, \quad \Lambda_R\Lambda_L = 1 \tag{4.4}$$

Following [9,10,31], we consider a so-called *staggered* MPA, in the form

$$\boldsymbol{p} = \langle \boldsymbol{l}_1|\boldsymbol{A}_2\boldsymbol{A}'_3\boldsymbol{A}_4\boldsymbol{A}'_5\ldots\boldsymbol{A}'_{i-1}\boldsymbol{W}_i\boldsymbol{B}'_{i+1}\ldots\boldsymbol{B}'_{n-3}\boldsymbol{B}_{n-2}|\boldsymbol{r}_{n-1,n}\rangle \tag{4.5}$$

$$\boldsymbol{p}' = \langle \boldsymbol{l}'_{1,2}|\boldsymbol{A}_3\boldsymbol{A}'_4\boldsymbol{A}_5\boldsymbol{A}'_6\ldots\boldsymbol{A}_{i-1}\boldsymbol{W}'_i\boldsymbol{B}_{i+1}\ldots\boldsymbol{B}_{n-3}\boldsymbol{B}'_{n-2}\boldsymbol{B}_{n-1}|\boldsymbol{r}'_n\rangle \tag{4.6}$$

where $\boldsymbol{A}_j$ and $\boldsymbol{B}_j$ denote respectively a matrix on the left and on the right side of the impurity and $\boldsymbol{W}_i$ is the impurity matrix. Similar ansatz have been used with good results to exactly solve integrable one-dimensional quantum chains [32,33]. Because of the presence of the impurity we shall henceforth refer as *bulk* to those sections of the chain defined by only matrices $\boldsymbol{A}$ and $\boldsymbol{B}$. Since the system is a good model for solitons' behaviour, we assume that we can logically identify three regions in the chain. On the one hand, we have the two bulk-boundary sections, where information travels according to the global rule of the chain and in which the solutions retrieved may be the same as in the homogeneous (without impurity) case. On the other hand, the impurity section includes the impurity cell and the ones in its neighbourhood. In this scenario we require that operators A_s and A'_s and some unitary operator S satisfy the bulk algebra

$$U_{1,2,3}\boldsymbol{A}_1 S\boldsymbol{A}_2\boldsymbol{A}'_3 = \boldsymbol{A}_1\boldsymbol{A}'_2\boldsymbol{A}_3 S \tag{4.7}$$

$$U_{1,2,3}\boldsymbol{B}_1 S\boldsymbol{B}_2\boldsymbol{B}'_3 = \boldsymbol{B}_1\boldsymbol{B}'_2\boldsymbol{B}_3 S \tag{4.8}$$

that can be translated to component-wise equations

$$\begin{aligned} A_s S A_{\chi(s,s',s'')} A'_{s''} &= A_s A'_{s'} A_{s''} S \\ B_s S B_{\chi(s,s',s'')} B'_{s''} &= B_s B'_{s'} B_{s''} S \end{aligned} \qquad s, s', s'' \in \{0,1\}$$

where χ is the rule 54 function. Since $S^2 = \mathbb{1}$ and $\mathcal{U}^2 = \mathbb{1}$ and the two sets of matrices are mutually bijective, we can also write *dual* bulk relations

$$U_{1,2,3}\boldsymbol{A}'_1\boldsymbol{A}_2\boldsymbol{A}'_3S = \boldsymbol{A}'_1S\boldsymbol{A}'_2\boldsymbol{A}_3$$
$$U_{1,2,3}\boldsymbol{B}'_1\boldsymbol{B}_2\boldsymbol{B}'_3S = \boldsymbol{B}'_1S\boldsymbol{B}'_2\boldsymbol{B}_3$$

The boundaries will satisfy the following boundary equations[1], that must be satisfied by the Λ_L and Λ_R "partial" eigenvalues

$$U^L_{1,2}\langle \boldsymbol{l}'_{1,2}| = \Lambda_L\langle \boldsymbol{l}_1|\boldsymbol{A}_2S \tag{4.9}$$
$$U_{1,2,3}\langle \boldsymbol{l}_1|\boldsymbol{A}_2\boldsymbol{A}'_3 = \langle \boldsymbol{l}'_{1,2}|\boldsymbol{A}_3S \tag{4.10}$$
$$U^R_{1,2}|\boldsymbol{r}_{1,2}\rangle = \Lambda_R\boldsymbol{B}'_1S|\boldsymbol{r}'_2\rangle \tag{4.11}$$
$$U_{1,2,3}\boldsymbol{B}'_1\boldsymbol{B}_2|\boldsymbol{r}'_3\rangle = \boldsymbol{B}'_1S|\boldsymbol{r}_{2,3}\rangle \tag{4.12}$$

The remaining equations have to be satisfied by both the bulk and the impurity matrices, we refer to them as *impurity equations*. The first issue we encounter is that in this case we are not sure that S now propagates exactly in the same way it did in the bulk. The second issue is which and how many equations we need to satisfy. In order to solve the first problem we define an auxiliary matrix $\boldsymbol{Z}$, while the second problem can be solved by evaluating the so-called *cancellation cycle*.

[1] These equations could be computationally difficult, but they can be simplified, for instance as

$$U^L_{1,2}U_{1,2,3}\langle \boldsymbol{l}_1|\boldsymbol{A}_2\boldsymbol{A}'_3 = \Lambda_L\langle \boldsymbol{l}_1|\boldsymbol{A}_2S\boldsymbol{A}_3S$$
$$U^R_{2,3}U_{1,2,3}\boldsymbol{B}'_1\boldsymbol{B}_2|\boldsymbol{r}'_3\rangle = \Lambda_R\boldsymbol{B}'_1S\boldsymbol{B}'_2S|\boldsymbol{r}'_3\rangle$$

which in components read

$$\sum_{t,t'} U^L_{(s,s'),(t,t')}l_tA_{\chi(t,t',v)}A'_v = \Lambda_Ll_sA_uSA_vS$$
$$\sum_{t,t'} U^R_{(u,v),(t,t')}B'_sB_{\chi(s,t,t')}r'_{t'} = \Lambda_RB'_sSB'_uSr'_v$$

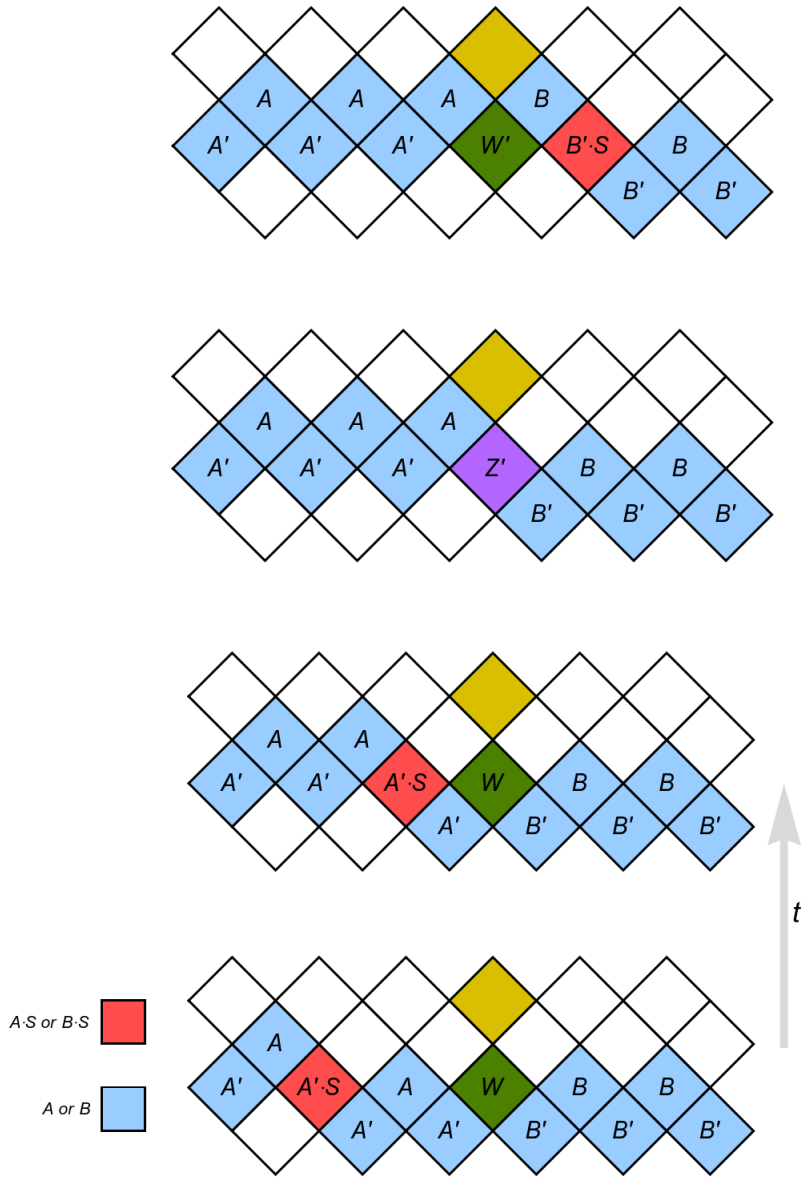


Figure 4.4: We assume that cell updates occur sequentially rather than simultaneously. In the bulk, cells containing solitons are represented in light blue, while empty bulk cells are white. The impurity cell is represented in green when occupied by a soliton (state 1) and in dark yellow when empty (state 0). The S operator propagates together with the solitons, and its interaction with the MPA is represented by a change in cell color: red in the bulk (indicating $\boldsymbol{A}^{(\prime)} \cdot S$ or $\boldsymbol{B}^{(\prime)} \cdot S$) and purple at the impurity. Since the behaviour of S at the impurity is non-trivial, we cannot use the product $\boldsymbol{W}^{(\prime)} \cdot S$; instead, we introduce a separate matrix $\boldsymbol{Z}^{(\prime)}$ to represent this case.

Practically, we try to understand how the system evolves around the impurity by evolving all the matrices from left to right and back in a portion of bulk cells around the impurity. The first steps of this cycle are shown in Figure 4.4.
The following additional equations are obtained

$$U_{1,2,3}\boldsymbol{A}'_1 S \boldsymbol{A}'_2 \boldsymbol{W}_3 = \boldsymbol{A}'_1 \boldsymbol{A}_2 \boldsymbol{Z}'_3 \tag{4.13}$$
$$U_{1,2,3}\boldsymbol{Z}'_1 \boldsymbol{B}'_2 \boldsymbol{B}_3 = \boldsymbol{W}'_1 \boldsymbol{B}_2 \boldsymbol{B}'_3 S \tag{4.14}$$
$$U^0_{1,2,3}\boldsymbol{A}_1 \boldsymbol{W}'_2 \boldsymbol{B}_3 S = \boldsymbol{A}_1 S \boldsymbol{W}_2 \boldsymbol{B}'_3 \tag{4.15}$$

Due to the properties of S and U, we could write *dual* impurity relations, as done for the bulk ones, and they will follow again from the cancellation cycle.

Statement. Provided equations (4.7), (4.8) (4.9), (4.10), (4.11), (4.12) and (4.13), (4.14), (4.15) are satisfied, the probability vectors in (4.5) and (4.6) solve (4.4), for $\Lambda_R\Lambda_L = 1$.

Proof. Plugging the MPAs in (4.5) into (4.4) and applying (4.7), (4.8), (4.9), (4.10), (4.11), (4.12) and (4.15), for the even-site equation, we have

$$\begin{aligned}
\mathcal{U}_e\boldsymbol{p} &= U_{1,2,3}\dots U_{n-5,n-4,n-3}U_{n-3,n-2,n-1}U^R_{n-1,n}\langle \boldsymbol{l}_1|\boldsymbol{A}_2\boldsymbol{A}'_3\dots\boldsymbol{A}'_{i-1}\boldsymbol{W}_i\boldsymbol{B}'_{i+1}\dots\boldsymbol{B}'_{n-3}\boldsymbol{B}_{n-2}|\boldsymbol{r}_{n-1,n}\rangle = \\
&= \Lambda_R U_{1,2,3}\dots U_{n-5,n-4,n-3}U_{n-3,n-2,n-1}\langle \boldsymbol{l}_1|\boldsymbol{A}_2\boldsymbol{A}'_3\dots\boldsymbol{A}'_{i-1}\boldsymbol{W}_i\boldsymbol{B}'_{i+1}\dots\boldsymbol{B}'_{n-3}\boldsymbol{B}_{n-2}\boldsymbol{B}'_{n-1}S|\boldsymbol{r}'_n\rangle = \\
&= \Lambda_R U_{1,2,3}\dots U_{n-5,n-4,n-3}\langle \boldsymbol{l}_1|\boldsymbol{A}_2\boldsymbol{A}'_3\dots\boldsymbol{A}'_{i-1}\boldsymbol{W}_i\boldsymbol{B}'_{i+1}\dots\boldsymbol{B}'_{n-3}S\boldsymbol{B}'_{n-2}\boldsymbol{B}_{n-1}|\boldsymbol{r}'_n\rangle = \\
&= \dots = \\
&= \Lambda_R U_{1,2,3}\dots U^0_{i-1,i,i+1}\langle \boldsymbol{l}_1|\boldsymbol{A}_2\boldsymbol{A}'_3\dots\boldsymbol{A}'_{i-1}\boldsymbol{W}_i\boldsymbol{B}'_{i+1}S\dots\boldsymbol{B}_{n-3}\boldsymbol{B}'_{n-2}\boldsymbol{B}_{n-1}|\boldsymbol{r}'_n\rangle = \\
&= \Lambda_R U_{1,2,3}\dots U_{i-2,i-1,i}\langle \boldsymbol{l}_1|\boldsymbol{A}_2\boldsymbol{A}'_3\dots\boldsymbol{A}'_{i-1}S\boldsymbol{W}'_i\boldsymbol{B}_{i+1}\dots\boldsymbol{B}_{n-3}\boldsymbol{B}'_{n-2}\boldsymbol{B}_{n-1}|\boldsymbol{r}'_n\rangle = \\
&= \dots = \\
&= \Lambda_R U_{1,2,3}\langle \boldsymbol{l}_1|\boldsymbol{A}_2\boldsymbol{A}'_3 S \boldsymbol{A}'_4\boldsymbol{A}_5\dots\boldsymbol{A}_{i-1}\boldsymbol{W}'_i\boldsymbol{B}_{i+1}\dots\boldsymbol{B}_{n-3}\boldsymbol{B}'_{n-2}\boldsymbol{B}_{n-1}|\boldsymbol{r}'_n\rangle = \\
&= \Lambda_R\langle \boldsymbol{l}'_{1,2}|\boldsymbol{A}_3\boldsymbol{A}'_4\dots\boldsymbol{A}_{i-1}\boldsymbol{W}'_i\boldsymbol{B}_{i+1}\dots\boldsymbol{B}_{n-3}\boldsymbol{B}'_{n-2}\boldsymbol{B}_{n-1}|\boldsymbol{r}'_n\rangle = \\
&= \Lambda_R\boldsymbol{p}'
\end{aligned}$$

The odd-sites equation in (4.4) is proven analogously, using (4.9), (4.10), (4.11), (4.12), (4.13) and (4.14)

$$\begin{aligned}
\mathcal{U}_o\boldsymbol{p}' &= U^L_{1,2}U_{2,3,4}\dots U_{n-2,n-1,n}\langle \boldsymbol{l}'_{1,2}|\boldsymbol{A}_3\boldsymbol{A}'_4\dots\boldsymbol{A}'_{i-2}\boldsymbol{A}_{i-1}\boldsymbol{W}'_i\boldsymbol{B}_{i+1}\boldsymbol{B}'_{i+1}\dots\boldsymbol{B}_{n-3}\boldsymbol{B}'_{n-2}\boldsymbol{B}_{n-1}|\boldsymbol{r}'_n\rangle = \\
&= U^L_{1,2}U_{2,3,4}\dots U_{n-2,n-1,n}\langle \boldsymbol{l}'_{1,2}|\boldsymbol{A}_3\boldsymbol{A}'_4\dots\boldsymbol{A}'_{i-2}\boldsymbol{A}_{i-1}\boldsymbol{W}'_i\boldsymbol{B}_{i+1}\boldsymbol{B}_{i+2}\dots\boldsymbol{B}_{n-3}\boldsymbol{B}'_{n-2}S|\boldsymbol{r}_{n-1,n}\rangle = \\
&= \dots = \\
&= U^L_{1,2}U_{2,3,4}\dots U_{i-2,i-1,i}U_{i,i+1,i+2}\langle \boldsymbol{l}'_{1,2}|\boldsymbol{A}_3\boldsymbol{A}'_4\dots\boldsymbol{A}'_{i-2}\boldsymbol{A}_{i-1}\boldsymbol{W}'_i\boldsymbol{B}_{i+1}\boldsymbol{B}'_{i+2}S\dots\boldsymbol{B}'_{n-3}\boldsymbol{B}_{n-2}|\boldsymbol{r}_{n-1,n}\rangle = \\
&= U^L_{1,2}U_{2,3,4}\dots U_{i-2,i-1,i}\langle \boldsymbol{l}'_{1,2}|\boldsymbol{A}_3\boldsymbol{A}'_4\dots\boldsymbol{A}'_{i-2}\boldsymbol{A}_{i-1}\boldsymbol{Z}'_i\boldsymbol{B}'_{i+1}\boldsymbol{B}_{i+2}\dots\boldsymbol{B}'_{n-3}\boldsymbol{B}_{n-2}|\boldsymbol{r}_{n-1,n}\rangle = \\
&= U^L_{1,2}U_{2,3,4}\dots\langle \boldsymbol{l}'_{1,2}|\boldsymbol{A}_3\boldsymbol{A}'_4\dots\boldsymbol{A}'_{i-2}S\boldsymbol{A}'_{i-1}\boldsymbol{W}_i\boldsymbol{B}'_{i+1}\boldsymbol{B}_{i+2}\dots\boldsymbol{B}'_{n-3}\boldsymbol{B}_{n-2}|\boldsymbol{r}_{n-1,n}\rangle = \\
&= \dots = \\
&= U^L_{1,2}\dots\langle \boldsymbol{l}'_{1,2}|S\boldsymbol{A}'_3\boldsymbol{A}_4\dots\boldsymbol{A}_{i-2}\boldsymbol{A}'_{i-1}\boldsymbol{W}_i\boldsymbol{B}'_{i+1}\boldsymbol{B}_{i+2}\dots\boldsymbol{B}'_{n-3}\boldsymbol{B}_{n-2}|\boldsymbol{r}_{n-1,n}\rangle = \\
&= \Lambda_L\langle \boldsymbol{l}_1|\boldsymbol{A}_2\underbrace{SS}_{=\mathbb{I}}\boldsymbol{A}'_3\boldsymbol{A}_4\dots\boldsymbol{A}_{i-2}\boldsymbol{A}'_{i-1}\boldsymbol{W}_i\boldsymbol{B}'_{i+1}\boldsymbol{B}_{i+2}\dots\boldsymbol{B}'_{n-3}\boldsymbol{B}_{n-2}|\boldsymbol{r}_{n-1,n}\rangle = \\
&= \Lambda_L\boldsymbol{p}
\end{aligned}$$

□

4.3.1 Bulk solutions

Due to the "plane-wave" assumption, for which we split the chain into three regions, we recover the bulk solutions as given in [9]. Indeed, we have

$$\boldsymbol{A}_0 = \begin{pmatrix} 1 & 0 & 0 \\ \xi & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} \qquad \boldsymbol{B}_0 = \begin{pmatrix} 1 & 0 & 0 \\ \zeta & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

$$\boldsymbol{A}_1 = \begin{pmatrix} 0 & \xi & 0 \\ 0 & 0 & 1 \\ 0 & 0 & \omega \end{pmatrix} \qquad \boldsymbol{B}_1 = \begin{pmatrix} 0 & \zeta & 0 \\ 0 & 0 & 1 \\ 0 & 0 & \eta \end{pmatrix}$$

such that $A'(\xi,\omega) = A(\omega,\xi)$ and $B'(\zeta,\eta) = B(\eta,\zeta)$ for $s = \{0,1\}$. The only difference is that in this case the spectral parameters are different for matrices on the right and on the left of the impurity. In particular parameters ξ and ω are used for $\boldsymbol{A}$ matrices, while ζ and η for $\boldsymbol{B}$ ones. A more detailed discussion regarding the second class of impurity solutions will introduce another "configuration" of matrices where the spectral parameters must be corrected in order to represent the new MPA.

4.3.2 Impurity solutions

The impurity equations return three classes of solutions on a three-dimensional auxiliary vector space $\mathcal{V}_a = \mathbb{C}^3$, where only two of them are non-trivial. What we called the "third class" solution gives indeed a zero probability vector. The three classes are here presented in a parametrized form, where both left-side and righ-side spectral parameters appear. Notice that all these results are written for the specific case of conditional driving (3.6).

Remark. The cancellation cycle can also be reversed, but it can be noticed that the solutions of the reversed cancellation cycle systems of equations belong to the same class of their respective direct cancellation cycle equations. That means a solution of the first class, for instance, is retrieved using the same free parameters in both the reversed and in the direct cycles. Moreover, the classes of solutions are the same for both the reversed and the direct cycle.

First Class

$$\boldsymbol{W}_0 = x \begin{pmatrix} \frac{(\zeta\eta-1)}{(\zeta-1)\eta} & 0 & 0 \\ 0 & \frac{\zeta}{\eta} & \frac{\zeta(\eta-1)}{\eta(\zeta-1)} \\ \frac{(\zeta\eta-1)}{(\zeta-1)\eta} & 0 & 0 \end{pmatrix} \qquad \boldsymbol{W}_1 = x \begin{pmatrix} 0 & -\frac{\zeta}{(\zeta-1)\eta} & \frac{\zeta}{\zeta-1}x \\ 0 & \frac{\zeta}{(\zeta-1)\eta} & -\frac{1}{(\zeta-1)\eta} \\ 0 & \frac{\zeta}{\zeta-1} & -\frac{1}{\zeta-1} \end{pmatrix}$$

$$\boldsymbol{W}_0' = x \begin{pmatrix} \frac{(\zeta\eta-1)}{(\zeta-1)\eta} & 0 & 0 \\ 0 & \frac{(\eta-1)}{\zeta-1} & 1 \\ \frac{(\zeta\eta-1)}{(\zeta-1)\eta} & 0 & 0 \end{pmatrix} \qquad \boldsymbol{W}_1' = x \begin{pmatrix} 0 & -\frac{1}{\zeta-1} & \frac{\zeta}{\zeta-1} \\ 0 & \frac{1}{\zeta-1} & -\frac{1}{(\zeta-1)\eta} \\ 0 & \frac{\zeta}{\zeta-1} & -\frac{\zeta}{(\zeta-1)\eta} \end{pmatrix}$$

where x is an arbitrary multiplicative factor that implies the solution uniqueness. This solutions exist for $\xi = \zeta$ and $\omega = \eta$.

Second Class To find this class one must "invert" the equations, i.e., we have to transpose a branch of the cellular automaton. It implies that in one of the two branches – e.g., $\boldsymbol{A}$'s one – every prime matrix becomes its respective non prime ($\boldsymbol{A} \to (\boldsymbol{A'})^T, \boldsymbol{A'} \to (\boldsymbol{A})^T$) and every matrix has to be transposed. Indeed also the multiplication of operator S is now a left-multiplicator: $\boldsymbol{A} \cdot S \to S \cdot (\boldsymbol{A'})^T$. In this case we transposed the left branch, but an equivalent solution is found when transposing the right branch. The solutions obtained belong to a class of matrices that has the following structure.

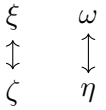
$$\boldsymbol{W}_0 = x \begin{pmatrix} 1 & -\frac{(\zeta-1)\zeta\eta}{\zeta\eta-1} & -\frac{\zeta(\eta-1)\eta}{\zeta\eta-1} \\ -\frac{\zeta(\eta-1)\eta}{\zeta\eta-1} & \frac{(\zeta-1)\zeta(\eta-1)\eta}{(\zeta\eta-1)^2} & \frac{\zeta(\eta-1)^2\eta}{(\zeta\eta-1)^2} \\ -\frac{(\zeta-1)\zeta\eta}{\zeta\eta-1} & \frac{(\zeta-1)^2\zeta\eta}{(\zeta\eta-1)^2} & \frac{(\zeta-1)\zeta(\eta-1)\eta}{(\zeta\eta-1)^2} \end{pmatrix}$$

$$\boldsymbol{W}_1 = x \begin{pmatrix} 0 & 0 & 0 \\ 0 & \frac{2\zeta\eta}{\zeta\eta-1} & -\frac{\eta(\zeta\eta+1)}{\zeta\eta-1} \\ 0 & -\frac{\zeta(\zeta\eta+1)}{\zeta\eta-1} & \frac{2\zeta\eta}{\zeta\eta-1} \end{pmatrix}$$

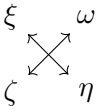
$$\boldsymbol{W}'_0 = x \begin{pmatrix} 1 & -\frac{\zeta(\eta-1)\eta}{\zeta\eta-1} & -\frac{(\zeta-1)\zeta\eta}{\zeta\eta-1} \\ -\frac{(\zeta-1)\zeta\eta}{\zeta\eta-1} & \frac{(\zeta-1)\zeta(\eta-1)\eta}{(\zeta\eta-1)^2} & \frac{(\zeta-1)^2\zeta\eta}{(\zeta\eta-1)^2} \\ -\frac{\zeta(\eta-1)\eta}{\zeta\eta-1} & \frac{\zeta(\eta-1)^2\eta}{(\zeta\eta-1)^2} & \frac{(\zeta-1)\zeta(\eta-1)\eta}{(\zeta\eta-1)^2} \end{pmatrix}$$

$$\boldsymbol{W}'_1 = x \begin{pmatrix} 0 & 0 & 0 \\ 0 & \frac{2\zeta\eta}{\zeta\eta-1} & -\frac{\zeta(\zeta\eta+1)}{\zeta\eta-1} \\ 0 & -\frac{\eta(\zeta\eta+1)}{\zeta\eta-1} & \frac{2\zeta\eta}{\zeta\eta-1} \end{pmatrix}$$

The solution is unique up to the overall multiplicative factor x. It is worth noticing that the transposition creates a different *cancellation cycle* that must be taken carefully into account. These solutions exist for $\xi = \eta$ and $\zeta = \omega$. Notice that an exchange happened in corresponding parameters, compared to the First Class solution. This exchange can be simply explained as an effect of the transposition.



First Class parameters' coupling



Second Class parameters' coupling

Third Class As previously mentioned, this class leads to a trivial $\boldsymbol{p} = 0$ probability state. We report it for completeness.

$$W_0 = \begin{pmatrix} 0 & -\frac{a}{\eta} & a \\ 0 & -\frac{\xi}{\omega\eta}b & \frac{\xi}{\omega}b \\ 0 & -\frac{a}{\eta} & a \end{pmatrix} \qquad W_1 = \begin{pmatrix} 0 & -\frac{\xi}{\eta}c & \xi c \\ 0 & -\frac{d}{\omega\eta} & \frac{d}{\omega} \\ 0 & -\frac{d}{\eta} & d \end{pmatrix}$$

$$W_0' = \begin{pmatrix} 0 & e & -\frac{1}{\eta}e \\ 0 & f & -\frac{1}{\eta}f \\ 0 & g & -\frac{1}{\eta}g \end{pmatrix} \qquad W_1' = \begin{pmatrix} 0 & h & -\frac{1}{\eta}h \\ 0 & m & -\frac{1}{\eta}m \\ 0 & n & -\frac{1}{\eta}n \end{pmatrix}$$

The exact same solutions can be retrieved by computing with only (ξ, ζ) and (ξ, η) as free parameters, however, these solutions have been found by using ξ, ω and ζ as independent. If one uses (ξ, ω, η) the result is similar, but with ζ that appears instead of η and an exchange of signs and coefficients.

$$W_0 = \begin{pmatrix} 0 & a & -\frac{a}{\zeta} \\ 0 & \frac{\xi b}{\omega} & -\frac{\xi b}{\zeta\omega} \\ 0 & a & -\frac{a}{\zeta} \end{pmatrix} \qquad W_1 = \begin{pmatrix} 0 & \xi c & -\frac{\xi c}{\zeta} \\ 0 & \frac{d}{\omega} & -\frac{d}{\zeta\omega} \\ 0 & d & -\frac{d}{\zeta} \end{pmatrix}$$

$$W_0' = \begin{pmatrix} 0 & -\frac{e}{\zeta} & e \\ 0 & -\frac{f}{\zeta} & f \\ 0 & -\frac{g}{\zeta} & g \end{pmatrix} \qquad W_1' = \begin{pmatrix} 0 & -\frac{h}{\zeta} & h \\ 0 & -\frac{m}{\zeta} & m \\ 0 & -\frac{n}{\zeta} & n \end{pmatrix}$$

4.3.3 Boundary vectors and parameters values

The boundary vectors $\langle \boldsymbol{l}|$ and $|\boldsymbol{r}\rangle$ for the non-transposed case are the same retrieved in [9], except for the right spectral parameters, since $|\boldsymbol{r}\rangle = |\boldsymbol{r}\rangle(\zeta, \eta)$ and $\langle \boldsymbol{l}| = \langle \boldsymbol{l}|(\xi, \omega)$, where ξ and ω

are also the same and relations $\xi = \zeta,\ \omega = \eta$ hold. That is

$$\langle \boldsymbol{l}_0| = \begin{pmatrix} (\beta-1)\beta\Lambda_L(\alpha+\beta-\Lambda_L-1)(\beta\Lambda_L-\alpha) \\ (\beta-1){\Lambda_L}^2(\alpha-\Lambda_L)\left(-\alpha^2+\alpha(-\beta+\Lambda_L+1)+(\beta-1)\beta(\Lambda_L+1)\right) \\ -\beta_1\left({\Lambda_L}^2\left(\beta_1\left(\alpha+\beta^2\right)+\beta\right)+\Lambda_L\left[(2\alpha+1)\beta^2+(\alpha-1)\alpha\beta_1-\beta\right]-\alpha\beta(\alpha+\beta_1)\right) \end{pmatrix}^T$$

$$\langle \boldsymbol{l}_1| = \begin{pmatrix} -(\beta-1)^2\Lambda_L(\alpha+\beta-\Lambda_L-1)(\beta\Lambda_L-\alpha) \\ (\beta-1){\Lambda_L}^2(\alpha-\Lambda_L)\left(\alpha^2+\alpha(\beta-\Lambda_L-2)-\beta((\beta-2)\Lambda_L+\beta)+\beta\right) \\ (\beta-1)^2\left({\Lambda_L}^2(\alpha+(\beta-1)\beta)-\alpha\Lambda_L(\alpha+2\beta-1)+\alpha(\alpha+\beta-1)\right) \end{pmatrix}^T$$

$$\langle \boldsymbol{l}'_{0,0}| = \begin{pmatrix} \beta\Lambda_L(\alpha+\beta-\Lambda_L-1)^2(\beta\Lambda_L-\alpha) \\ (\beta-1){\Lambda_L}^2(\alpha-\Lambda_L)(\alpha+\beta-\Lambda_L-1)(\beta\Lambda_L-\alpha) \\ (\beta-1)^2\Lambda_L(\alpha+\beta-\Lambda_L-1)(\alpha-\beta\Lambda\mathrm{L}) \end{pmatrix}^T$$

$$\langle \boldsymbol{l}'_{0,1}| = \begin{pmatrix} 0 \\ -\left({\Lambda_L}^2(\alpha-\Lambda_L)(\alpha+\beta-\Lambda_L-1)(\alpha-\beta\Lambda_L)\right) \\ -((\alpha+\beta-1)(\alpha-\Lambda_L)(\alpha+\beta-\Lambda_L-1)(\alpha-\beta\Lambda_L)) \end{pmatrix}^T$$

$$\langle \boldsymbol{l}'_{1,0}| = \begin{pmatrix} -\left((\beta-1)\Lambda_L(\alpha+\beta-\Lambda_L-1)^2(\beta\Lambda_L-\alpha)\right) \\ -\left((\beta-1){\Lambda_L}^2(\alpha-\Lambda_L)(\alpha+\beta-\Lambda_L-1)(\beta\Lambda_L-\alpha)\right) \\ (\beta-1)^2\Lambda_L(\alpha+\beta-\Lambda\mathrm{L}-1)(\beta\Lambda_L-\alpha) \end{pmatrix}^T$$

$$\langle \boldsymbol{l}'_{1,1}| = \begin{pmatrix} 0 \\ 0 \\ (\alpha-1)(\alpha+\beta-\Lambda_L-1)^2(\alpha-\beta\Lambda_L) \end{pmatrix}^T$$

$$\langle \boldsymbol{r}'_0| = \begin{pmatrix} \gamma(\delta-1)\Lambda_R \\ (\delta-1)\delta \\ (\delta-1)\delta{\Lambda_R}^2 \end{pmatrix}$$

$$\langle \boldsymbol{r}'_1| = \begin{pmatrix} -((\gamma-1)(\delta-1)\Lambda_R) \\ -(\delta-1)^2 \\ -(\delta-1)^2{\Lambda_R}^2 \end{pmatrix}$$

$$\langle \boldsymbol{r}_{0,0}| = \begin{pmatrix} \gamma(\delta-1)\Lambda_R \\ \delta(\gamma-\delta\Lambda_R+\delta-1) \\ (\delta-1)\delta{\Lambda_R}^2 \end{pmatrix}$$

$$\langle \boldsymbol{r}_{0,1}| = \begin{pmatrix} (\delta-1)\Lambda_R(\Lambda_R-\gamma) \\ -((\delta-1)(\gamma-\delta\Lambda_R+\delta-1)) \\ -(\delta-1)^2{\Lambda_R}^2 \end{pmatrix}$$

$$\langle \boldsymbol{r}_{1,0}| = \begin{pmatrix} \delta\Lambda_R(\gamma-\Lambda_R) \\ \gamma(\delta-1)\Lambda_R \\ \gamma{\Lambda_R}^2(\gamma-\Lambda_R) \end{pmatrix}$$

$$\langle \boldsymbol{r}_{1,1}| = \begin{pmatrix} \Lambda_R(\gamma(-\delta)+\gamma+\delta-1) \\ -((\gamma-1)(\delta-1)\Lambda_R) \\ -\Big((\gamma-1){\Lambda_R}^2(\gamma-\Lambda\mathrm{R})\Big) \end{pmatrix}$$

with

$$\begin{aligned} \omega =& \frac{\Lambda_L(\alpha-\Lambda_L)}{\beta-1} \\ \xi =& \frac{\alpha+\beta-1-\beta\Lambda_L}{(\beta-1)\Lambda_L^2} \end{aligned} \tag{4.16}$$

For the transposed case we obtain different left boundaries solutions, the right boundaries are exatcly the same as in the non-transposed case.

$$\begin{aligned} \langle l_0|_T &= \left(K, \tfrac{\beta K}{\alpha\Lambda_L}, \tfrac{\beta\Lambda_L K}{\alpha}\right) \\ \langle l_1|_T &= \left(\tfrac{K-\alpha K}{\alpha}, \tfrac{K-\beta K}{\alpha\Lambda_L}, \tfrac{\Lambda_L K-\beta\Lambda_L K}{\alpha}\right) \\ \langle l'_{00}|_T &= \left(K, -\tfrac{M(-\alpha+\beta\Lambda_L-\beta+1)}{(\beta-1)\Lambda_L^2}, M\right) \\ \langle l'_{01}|_T &= \left(-\tfrac{N\left(\Lambda_L^2-\alpha\Lambda_L\right)}{\beta-1}, K, -\tfrac{K\left(\Lambda_L^2-\alpha\Lambda_L\right)}{\beta-1}\right) \\ \langle l'_{10}|_T &= \left(\tfrac{(\alpha\Lambda_L-\Lambda_L^2)F}{\beta-1}, -\tfrac{T(\alpha+\beta-\beta\Lambda_L-1)}{(\beta-1)\Lambda_L^2}, T\right) \\ \langle l'_{11}|_T &= \left(G, G, \tfrac{(\alpha\Lambda_L-\Lambda_L^2)G}{\beta-1}\right) \end{aligned}$$

where K, M, N, G and T are free constants and α, β, γ, δ are the boundary parameters. Instead of having all these free parameters we can choose to set them. To set K we can consider to simplify the mathematical form of the entries, that is $K = \Lambda_L\alpha(1-\beta)$. We can notice that K is now proportional to the probability the soliton has been introduced into the system and it has not yet escaped from the left boundary, $K \propto \alpha(1-\beta)$. A sign in front of them is redundant and we retrieve the same solutions as in the case without impurity [9] by inserting it. $G = K$ can be also a good choice and $M = T = (\beta-1)\Lambda_L^2$ simplify the results. N can be put to zero since it only appears once and this assumption does not compromise the integrity of the solution, i.e.,

the final probability vector does not depend on N. We obtain, then

$$\langle \boldsymbol{l}_0|_T = \begin{pmatrix} \alpha(\beta-1){\Lambda_L}^3 \\ (\beta-1)\beta{\Lambda_L}^2 \\ (\beta-1)\beta{\Lambda_L}^4 \end{pmatrix}^T$$

$$\langle \boldsymbol{l}_1|_T = \begin{pmatrix} {\Lambda_L}^3(\alpha(-\beta)+\alpha+\beta-1) \\ -(\beta-1)^2{\Lambda_L}^2 \\ -(\beta-1)^2{\Lambda_L}^4 \end{pmatrix}^T$$

$$\langle \boldsymbol{l}'_{0,0}|_T = \begin{pmatrix} \alpha(\beta-1){\Lambda_L}^3 \\ \beta{\Lambda_L}^5(\alpha-\Lambda_L) \\ (\beta-1)\beta{\Lambda_L}^4 \end{pmatrix}^T$$

$$\langle \boldsymbol{l}'_{0,1}|_T = \begin{pmatrix} \beta(\alpha-\beta\Lambda_L+\beta-1) \\ \alpha(\beta-1){\Lambda_L}^3 \\ \alpha\Lambda_L(\alpha-\beta\Lambda_L+\beta-1) \end{pmatrix}^T$$

$$\langle \boldsymbol{l}'_{1,0}|_T = \begin{pmatrix} -((\beta-1)(\alpha-\beta\Lambda_L+\beta-1)) \\ (\beta-1){\Lambda_L}^5(\Lambda_L-\alpha) \\ -(\beta-1)^2{\Lambda_L}^4 \end{pmatrix}^T$$

$$\langle \boldsymbol{l}'_{1,1}|_T = \begin{pmatrix} -\left((\alpha-1)(\beta-1){\Lambda_L}^3\right) \\ -\left((\alpha-1)(\beta-1){\Lambda_L}^3\right) \\ -((\alpha-1)\Lambda_L(\alpha-\beta\Lambda_L+\beta-1)) \end{pmatrix}^T$$

Solving the left boundary equations (4.9) and (4.10) for parameters ξ, ω and the above left boundary vectors, we obtain

$$\begin{aligned} \xi =& \frac{\Lambda_L(\alpha-\Lambda_L)}{\beta-1} \\ \omega =& \frac{\alpha+\beta-1-\beta\Lambda_L}{(\beta-1)\Lambda_L^2} \end{aligned} \tag{4.17}$$

Similarly, solving the right boundary equations (4.11) and (4.12) for ζ, η and the above right boundary vectors, we get

$$\begin{aligned} \zeta =& \frac{(\gamma-\Lambda_R)\Lambda_R}{\delta-1} \\ \eta =& \frac{\gamma+\delta-1-\delta\Lambda_R}{(\delta-1)\Lambda_R^2} \end{aligned} \tag{4.18}$$

Since the above results hold for $\xi = \eta$ and $\omega = \zeta$, we can equate them obtaining the same eigenvalue equation found in [9], that can be expressed in terms of $\Lambda = \Lambda_R\Lambda_L$. The resulting equation is

$$(\Lambda-1)\left(\Lambda^3+(1-\alpha\gamma)\Lambda^2+\beta\delta\Lambda-(\alpha+\beta-1)(\gamma+\delta-1)(\Lambda+1)\right)=0$$

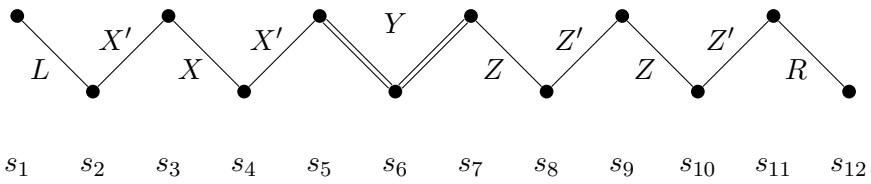
with solutions $\Lambda = 1$, $\Lambda = \frac{1}{3}$. The latter is another eigenvector of Markov chain, which is not related to NESS, but has a simple MPS structure. The important properties are $\Lambda = \Lambda_L \Lambda_R = 1$ is always a solution, whose corresponding eigenvector is the NESS, and the eigenvalues do not depend on the system size n. In summary, we found two independent eigenvectors of the Markov chain with eigenvalue 1, which means there are two distinct non-equilibrium steady states. This clearly points to non-ergodicity, since ergodic systems are characterized by a unique steady state. Therefore, the system does not settle into a single long-term behaviour, a hallmark of non-ergodic dynamics. This implies the Markov chain can evolve toward different steady states depending on the initial conditions.

4.4 Time Crystal: Patch State Ansatz (PSA)

A time-crystal NESS solution is characterised by a vector $\boldsymbol{p}_{tc}$ that, under the dynamical evolution (3.3) is mapped into itself, up to a minus sign. Thereby, the vector satisfies the equation

$$(-1)\boldsymbol{p}_{tc} = \mathcal{U}\boldsymbol{p}_{tc} \tag{4.19}$$

To retrieve the the eigenvector $\boldsymbol{p}_{tc}$ it is better to solve equations (4.4) using the so-called *patch-state-ansatz* (PSA). This ansatz has previously been used to find the NESS solutions related to the $+1$ eigenvalue, and it is now used to retrieve the solutions of the -1 eigenvalue related time-crystal NESS. To do this, we avail ourself of a two-state patch as shown below.



where the black nodes identify the states and the edges the patches, the double lines identify the three-site impurity patch.
For the conditional driving (3.6), the time-crystal NESS solution $\boldsymbol{p}_{tc}, \boldsymbol{p}'_{tc} \in \mathbb{R}^{2^n}$ of the fixed point equation (4.4) can be written in the form

$$\begin{aligned} p_{s_1,s_2,\ldots,s_n} &= L_{s_1,s_2} X'_{s_2,s_3} X_{s_3,s_4} \ldots X'_{s_{i-2},s_{i-1}} Y_{s_{i-1},s_i,s_{i+1}} Z_{s_{i+1},s_{i+2}} \ldots Z'_{s_{n-2},s_{n-1}} R_{s_{n-1},s_n} \\ p_{s_1,s_2,\ldots,s_n} &= L'_{s_1,s_2} X_{s_2,s_3} X'_{s_3,s_4} \ldots X_{s_{i-2},s_{i-1}} Y'_{s_{i-1},s_i,s_{i+1}} Z'_{s_{i+1},s_{i+2}} \ldots Z_{s_{n-2},s_{n-1}} R'_{s_{n-1},s_n} \end{aligned}$$

where $L^{(\prime)}_{s,s'}$, $X^{(\prime)}_{s,s'}$ and $Z^{(\prime)}_{s,s'}$ are strictly positive components of some rank 2 tensors with binary indices $s, s' \in \{0,1\}$, while $Y^{(\prime)}_{s,s',u}$ are components of a rank-2 tensor with binary indices $s, s', u \in \{0,1\}$. This means all patches depend on two consecutive sites except the impurity one, that lies on three sites: the impurity state and the two adjacent ones. We can again express the evolution

equations for the bulk, the boundaries and the impurity as follows

$$\left.\begin{aligned} &U_{1,2,3}X_{1,2}X'_{2,3} = X'_{1,2}X_{2,3} \\ &U_{1,2,3}Z_{1,2}Z'_{2,3} = Z'_{1,2}Z_{2,3} \end{aligned}\right\} \text{bulk equations}$$

$$\left.\begin{aligned} &U_{1,2,3}L_{1,2}X'_{2,3} = L'_{1,2}X_{2,3} \\ &U_{1,2,3}Z_{1,2,}R'_{2,3} = Z'_{1,2}R_{2,3} \\ &U^{L}_{1,2}L'_{1,2} = L_{1,2}\Lambda_L \\ &U^{R}_{1,2}R_{1,2} = R'_{1,2}\Lambda_R \end{aligned}\right\} \text{boundary equations}$$

$$\left.\begin{aligned} &U^{0}_{1,2,3}Y_{1,2,3} = \Lambda_I Y'_{1,2,3} \end{aligned}\right\} \text{impurity equation}$$

for $\Lambda_R\Lambda_I\Lambda_L = 1$. In the diagram we set  as the prime bulk tensor and  as the non-prime one, that means the bulk evolution operator acts as

$$U\left(\right) =$$ 

The boundary operator, being L represented as a non-prime bulk tensor and R a prime one (for the sake of geometry) is such that

$$U^{L}\left(\right) =$$  and $U^{R}\left(\right) =$ 

Regarding the impurity evolution operator, it is worth noticing that, since Y cannot be broken, it has to be propagated simultaneously by the impurity evolution operator and the adjacent two U, i.e.,

$$U\left(\right) =$$ 

leading to the following rewriting of the impurity equation

$$\tilde{U}^{0}_{(1,)2,3,4(,5)}X'_{1,2}Y'_{2,3,4}Z_{4,5} = U_{1,2,3}U^{0}_{2,3,4}U_{3,4,5}X'_{1,2}Y'_{2,3,4}Z_{4,5} = \Lambda_I X_{1,2}Y_{2,3,4}Z'_{4,5}$$

where $U_{1,2,3}$ and $U_{3,4,5}$ must be performed simultaneously, as if they were a single evolution operator.

4.5 Solutions of the PSA and oscillating behaviour

Solving the above equations we find the rank-2 tensors with parametrised entries, in particular we find solutions for both $+1$ and -1 eigenvalues, implying that such a solution represents the oscillating behaviour of the time crystal. Importantly, the impurity tensor has in fact three binary indices, which means we have 2^3 entries. However, the resulting tensors $Y^{(\prime)}_{\pm}$ are rank-2 tensors. In order to simplify the visualisation of the solutions we grouped the entries as follows

$$M^{(\prime)} = \begin{pmatrix} M^{(\prime)}_{0,0} & M^{(\prime)}_{0,1} \\ M^{(\prime)}_{1,0} & M^{(\prime)}_{1,1} \end{pmatrix}$$

for $M \in \{X, Z, L, R\}$, while for Y we used

$$Y^{(\prime)} = \begin{pmatrix} Y^{(\prime)}_{0,1,0} & Y^{(\prime)}_{0,1,1} \\ Y^{(\prime)}_{1,1,0} & Y^{(\prime)}_{1,1,1} \end{pmatrix}$$

This representation of $Y^{(\prime)}$ is possible since all the elements with middle index equal to 0 vanish, since the impurity must be triggered for the system to reach the time-crystal NESS.

$$\text{left side solutions} \begin{cases} X = \begin{pmatrix} 1 & \xi \\ \omega & 1 \end{pmatrix} \\ X' = \begin{pmatrix} 1 & \frac{1}{\omega} \\ \frac{1}{\xi} & 1 \end{pmatrix} \\ L = \ell_0 \begin{pmatrix} 1 & \xi \\ \frac{1-\alpha}{\alpha} & \xi \end{pmatrix} \\ L' = \ell_0 \begin{pmatrix} 1 & \frac{1}{\omega} \\ \frac{1-\alpha}{\alpha} & \frac{1}{\omega}\frac{1-\beta}{\beta} \end{pmatrix} \end{cases}$$

$$\text{right side solutions} \begin{cases} Z = \begin{pmatrix} 1 & \zeta \\ \eta & 1 \end{pmatrix} \\ Z' = \begin{pmatrix} 1 & \frac{1}{\eta} \\ \frac{1}{\zeta} & 1 \end{pmatrix} \\ R = r_0 \begin{pmatrix} 1 & \frac{1-\delta}{\delta} \\ \eta & \frac{1-\gamma}{\gamma}\eta \end{pmatrix} \\ R' = r_0 \begin{pmatrix} \zeta & \zeta\frac{1-\gamma}{\gamma} \\ 1 & \frac{1-\delta}{\delta} \end{pmatrix} \end{cases}$$

$$\text{impurity solutions} \begin{cases} \left.\begin{array}{l} Y_- = y\Lambda_I \begin{pmatrix} \eta\omega\Lambda_I & 0 \\ 0 & 1 \end{pmatrix} \\ Y'_- = y \begin{pmatrix} \eta\omega\Lambda_I & 0 \\ 0 & 1 \end{pmatrix} \end{array}\right\} \text{for } \Lambda_I = -\frac{\beta\delta}{\alpha\gamma} \\ \left.\begin{array}{l} Y_+ = y\Lambda_I \begin{pmatrix} \eta\omega\Lambda_I & 0 \\ 0 & 1 \end{pmatrix} \\ Y'_+ = y \begin{pmatrix} \eta\omega\Lambda_I & 0 \\ 0 & 1 \end{pmatrix} \end{array}\right\} \text{for } \Lambda_I = +\frac{\beta\delta}{\alpha\gamma} \end{cases}$$

where

$$\xi = \frac{\beta^2}{\alpha^2}\frac{1}{\omega},\ \omega = \frac{\ell_0}{\ell_1},\ \zeta = \frac{\gamma^2}{\delta^2}\frac{1}{\eta}$$

with ℓ_0, ℓ_1 and η free parameters, and also

$$\Lambda_L = \frac{\alpha}{\beta},\ \Lambda_R = \frac{\gamma}{\delta} \implies \Lambda_L \Lambda_R \Lambda_I = \pm 1$$

One can immediately notice the only nonzero entries for $Y_{\pm}^{(\prime)}$ are the ones in which the central binary index – that corresponds to the binary state of the impurity – is 1. This means the impurity plays a role only if its state is 1, that can be seen as the presence of information inside the cell, otherwise we have no response for the interaction. This is coherent with the results obtained in the simulations (see Figure 4.2 and Figure 4.1), in which the impurity cell had to be triggered to start an oscillating (and emitting) pattern. Notice also that the emitting pattern appears only if the impurity is in state 1 and its first neighbours are both in state 1, this corresponds to the $Y_{1,1,1}^{(\prime)}$ case. An interesting result is the presence of another $\Lambda = +1$ solution. This suggest that there is one specific combination of two 3×3 solutions (belonging to class I and class II discussed earlier) that reduces to 2×2 PSA with $\Lambda = +1$.

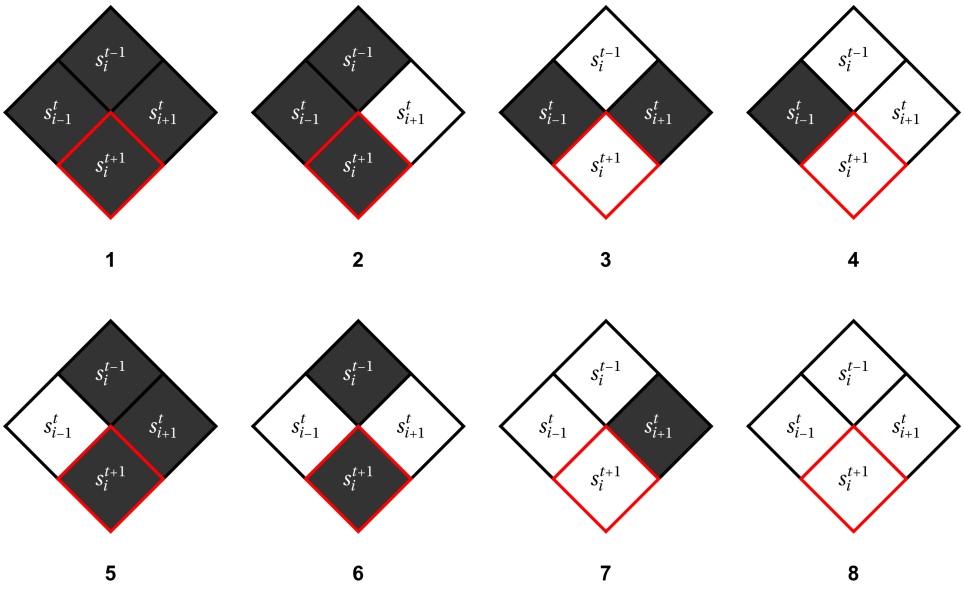
Chapter 5

The I204RCA54 model for RCA with impurity

5.1 Action of the rule 204 bulk impurity in a RCA54

Here we focus on the I204RCA54, that means we assume a 1+1 dimensional RCA with global rule (4.1) that is rule 54, and an impurity located at the centre of the RCA with the non-interacting rule 204

$$\phi(s_1, s_2, s_3) = s_2 \pmod 2 \tag{5.1}$$



Rule 204

As for the I150RCA54 diagrams, the red encircled diamond is the updated cell (time is directed down), the dark grey ones are cells in state 1 and the white ones are in 0. Since both rule 54 and rule 204 are reversible, so the entire cellular automaton with impurity results to be reversible. In this case the dynamics is more difficult to be understood by looking at the diagrams above, indeed, the "reflection" of the soliton on the impurity when the former collides with the latter cannot be simply recognised using only these diagrams. Nevertheless, it is possible to notice that – as expected – the impurity state does not change because of scatterings and it must be changed by an external action. Numbering the above diagrams from the left, the emitting behaviour is determined by the alternation of diagrams 1 and 6, while the last diagram represents the empty space in and around the impurity. According to the definitions of left- and right-movers given in [9], it is straightforward to observe that if a mover encounters the impurity alone, it will be reflected (just combine diagrams 4 with 4 and 6 with 6 in sequence). The same holds for two movers that scatters with the impurity simultaneously. This means that any interaction between two movers that occurs before (to the left of) or after (to the right of) the impurity will be exactly analogous to the interactions governed by the global rule. The interesting property is the mirroring behaviour of the impurity, that reflects completely the incoming soliton.
This reflection leaves the impurity as untouched, while the soliton goes back with the same initial velocity (see Figure 5.1a).

5.2 Triggered 204 bulk impurity and spontaneous emission

As for the rule 150 impurity case, a spontaneous emission is retrieved when the impurity is triggered – i.e., it is in state 1. The emission's properties are exactly the same, except in this case we do not need for two solitons to meet at the impurity at the same time it is triggered. In fact, a single trigger suffices to activate the spontaneous emission, as shown in Figure 5.1b.

5.3 NESS: Matrix-Product-Ansatz (MPA)

Using a similar theoretical approach as for the rule 150 impurity, we find the exact solutions for the NESS by means of the MPA method. In order not to make the understanding more difficult, we will follow exactly the same notation as before, except for the impurity because of some necessary changes. In particular, following the discussion in section 3.4.1, we can perform a bi-partition of the chain and notice that the rank of the needed impurity matrix, being the impurity rule non interacting in this case, is a rank-1 matrix. Therefore we can write it as

$$W_i = |d_i\rangle\langle g_i|$$

where the index i is the impurity index and specifies – as for the other matrices – the site of the matrix and the state of the site itself, that can be 0 or 1. Indeed, a rank-1 matrix in this context corresponds to a product state, meaning that it can be expressed as a tensor product of individual states.

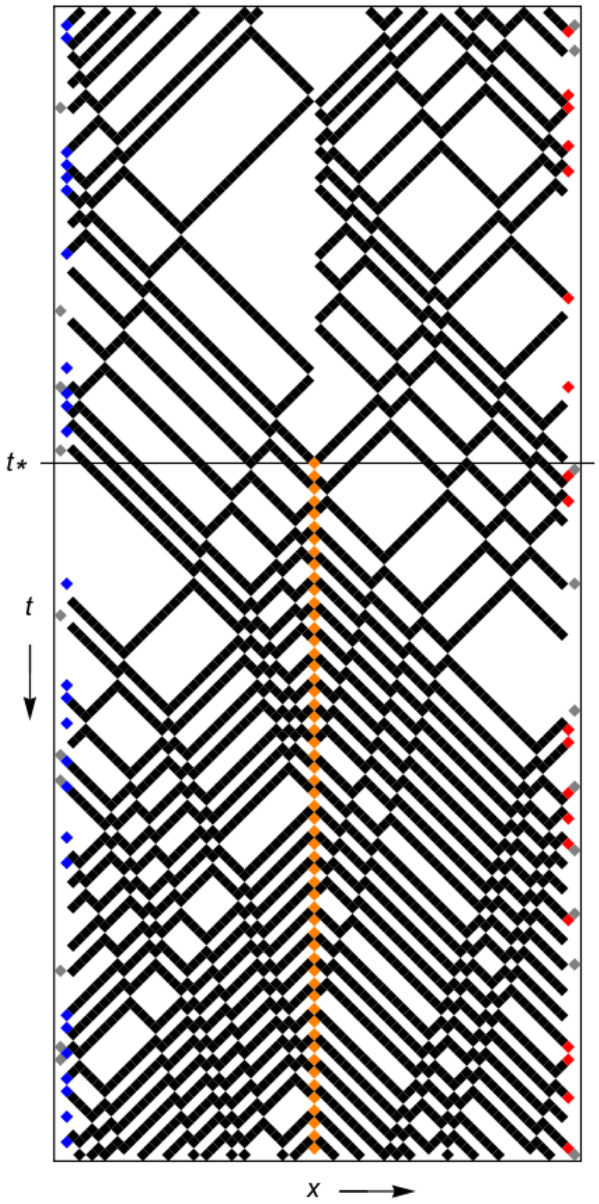
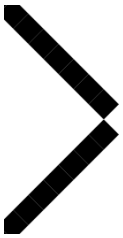


Figure 5.1: The impurity – if in a zero state – behaves as a mirror, reflecting the solitons when they collide with it (a). The state of the impurity is frozen, that means only an external trigger can modify it. Once the impurity is triggered - at time t^* - it will maintain its state, and it starts emitting particles (b) as in the rule 150 impurity case, that is every two even-time steps, i.e., after two updates of the central (even) cell and not just one. This figure has been generated by simulating a boundary-driven I204RCA54 with a probability of information injection of 0.1 from the boundaries, an absorption probability of 0.6, and a chemical potential of 0.1 for both the left and right boundaries. The very low probability of emission and the small values of the chemical potentials have been chosen to make the figure more understandable, avoiding confusion that could arise from a large number of interactions.

(a)



(b)

Calling r the Schmidt rank and λ_i the Schmidt coefficients, we can use the entanglement entropy to represent a certain degree of correlation in the system, that is the entropy is zero for non-interacting states and nonzero for entangled ones. The entropy reads

$$S = -\sum_{i=1}^{r} \lambda_i^2 \log(\lambda_i^2)$$

If the Schmidt rank is $r = 1$ – i.e., we have only one non-zero λ_i – the many-body state is a product of single states, since the entanglement entropy – i.e., the interaction degree – is $S = 0$. This means we can disentangle the left branch of the chain $\langle \boldsymbol{l}_1 | \boldsymbol{A}_2 \dots \boldsymbol{A}'_{i-1} | d_i \rangle$ from the right branch $\langle g_i | \boldsymbol{B}'_{i+1} \dots \boldsymbol{B}_{n-2} | \boldsymbol{r}_{n-1,n} \rangle$, where i is the impurity site. Under this assumption, we are solving two different and independent problems, one on the left and the other on the right of the impurity site, which is in a definite state 1 or 0. The MPA matrices must satisfy the bulk algebra equations (4.7) and (4.8), the boundary equations (4.9), (4.10), (4.11), (4.12) (all satisfied by the same results presented in the chapter before) and the following impurity equations

$$\begin{aligned} U^0_{0,1,2} \langle g_1 | \boldsymbol{B}'_2 &= \Lambda_I^L \langle g'_1 | \boldsymbol{B}_2 S \\ U^0_{1,2,0} \boldsymbol{A}'_1 | d_2 \rangle &= \Lambda_I^R \boldsymbol{A}_I S | d'_2 \rangle \end{aligned}$$

where the 0 index is nothing but an auxiliary index to consider the component of the staggered evolution operator $\mathcal{U}$. From the Schmidt rank analysis – explained in section 3.4.1 – we should obtain three degenerate NESS. Proceeding as before, we find two nontrivial solutions for the impurity states, the third solution can be obtained by transpose one of the two branches (the left or the right one) in which we split the product of matrices. It is worth noticing that, being the solutions separated, setting Λ_L and Λ_R as 1 is permitted since they are both free parameters for the bulk and boundary solutions we are using. In this way we can also avoid putting them among the variables and $\Lambda_I^{(L,R)}$ can give us the result 1 (as for the MPA) or -1, carrying all the meaning of this eigenvalue. All MPA solutions provide for $\Lambda_I^{(L,R)} = 1$. Moreover, the solution is computationally simpler, since we can completely split the sequence of matrices writing the evolution equation as

$$\begin{aligned} \mathcal{U}_e \boldsymbol{p} &= U_{1,2,3} \dots U^0_{i-1,i,i+1} \dots U_{n-3,n-2,n-1} \langle \boldsymbol{l}_1 | \boldsymbol{A}_2 \dots \boldsymbol{A}'_{i-1} | d_i \rangle \langle g_i | \boldsymbol{B}'_{i+1} \dots \boldsymbol{B}_{n-2} | \boldsymbol{r}_{n-1,n} \rangle = \\ &= \underbrace{U_{1,2,3} \dots U^0_{i-1,i,0} \langle \boldsymbol{l}_1 | \boldsymbol{A}_2 \dots \boldsymbol{A}'_{i-1} | d_i \rangle}_{\text{left branch}} \underbrace{U^0_{0,i,i+1} \dots U_{n-3,n-2,n-1} \langle g_i | \boldsymbol{B}'_{i+1} \dots \boldsymbol{B}_{n-2} | \boldsymbol{r}_{n-1,n} \rangle}_{\text{right branch}} \end{aligned}$$

where the subscript i indicates the impurity site.

First Solution In the following expressions, the subscript indicates the state of the impurity (0 or 1).

$$|d_0\rangle = \begin{pmatrix} -\frac{1-\beta}{1-\alpha}x_1 - x_2 \\ -x_1 \\ x_3 \end{pmatrix} \qquad |d_0'\rangle = \begin{pmatrix} \frac{1-\beta}{1-\alpha}x_1 - x_3 \\ x_1 \\ x_2 \end{pmatrix}$$

$$|d_1\rangle = \begin{pmatrix} x_5 \\ -\frac{\alpha-1}{\beta-1}(x_4 - x_6) \\ x_6 \end{pmatrix} \qquad |d_1'\rangle = \begin{pmatrix} x_4 \\ -\frac{\alpha-1}{\beta-1}(x_7 - x_5) \\ x_7 \end{pmatrix}$$

$$\langle g_0| = \begin{pmatrix} y_7 \\ y_8 \\ -\frac{\gamma-1}{\delta-1}(y_8 + y_2) - (y_1 + y_3 + y_7) \end{pmatrix} \qquad \langle g_0'| = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}$$

$$\langle g_1| = \begin{pmatrix} -\frac{1-\delta}{1-\gamma}y_5 - y_6 \\ -y_5 \\ y_9 \end{pmatrix} \qquad \langle g_1| = \begin{pmatrix} \frac{1-\delta}{1-\gamma}y_5 - y_9 \\ y_5 \\ y_6 \end{pmatrix}$$

Second Solution

$$|d_0\rangle = \begin{pmatrix} \frac{1-\beta}{1-\alpha}x_1 + x_2 \\ x_1 \\ x_3 \end{pmatrix} \qquad |d_0'\rangle = \begin{pmatrix} \frac{1-\beta}{1-\alpha}x_1 + x_3 \\ x_1 \\ x_2 \end{pmatrix}$$

$$|d_1\rangle = \begin{pmatrix} x_5 \\ \frac{\alpha-1}{\beta-1}(x_4 - x_6) \\ x_6 \end{pmatrix} \qquad |d_1'\rangle = \begin{pmatrix} x_4 \\ -\frac{\alpha-1}{\beta-1}(x_7 - x_5) \\ x_7 \end{pmatrix}$$

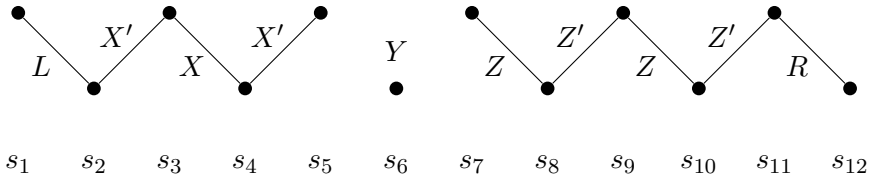
$$\langle g_0| = \begin{pmatrix} y_7 \\ y_8 \\ -\frac{\gamma-1}{\delta-1}(y_8 - y_2) + (y_1 + y_3 - y_7) \end{pmatrix} \qquad \langle g_0'| = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}$$

$$\langle g_1| = \begin{pmatrix} \frac{1-\delta}{1-\gamma}y_5 + y_6 \\ y_5 \\ y_9 \end{pmatrix} \qquad \langle g_1| = \begin{pmatrix} \frac{1-\delta}{1-\gamma}y_5 + y_9 \\ y_5 \\ y_6 \end{pmatrix}$$

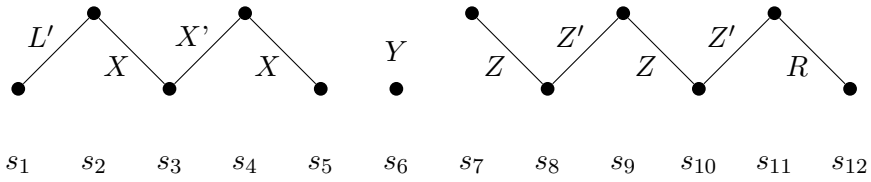
All x_i and y_i are free parameters, while we have used the boundary solutions of ξ, ω, ζ and η to write the solutions above. In particular, the bulk and the boundary solutions are the ones discussed in the previous sections and in [9, 10].

5.4 Time Crystal: Patch-State-Ansatz (PSA)

The patch tensor for the impurity, in this case, is made by a single site, since there is interaction neither between the left side and the impurity nor between the latter and the right side of the chain. The rule 204 is, indeed, completely non-interacting and this feature allows us to disconnect the rest of the chain from the impurity when constructing the PSA. We know the chain enriched with the rule 150 impurity can be described through a two-site patch tensor for every bulk site and a three-site patch tensor for the impurity; if the bulk can be always described by a two-site patch, the rule 204 impurity site can be described – as discussed – by a single site. That means, graphically,



Here it is simple to notice that the impurity state is completely uncorrelated and will be described by a single variable $Y \in \mathbb{R}$ that can assume any value. In order to maintain the probability normalisation we can set $Y = 1$. The solutions for $X^{(\prime)}$ and $Z^{(\prime)}$ patch tensors are the same used for the PSA of the rule 150 impurity and retrieved in [9]. From the Schmidt decomposition, we retrieve two time-crystal eigenvectors, implying that there should be two distinct solutions related to the -1 eigenvalue. The second solution of this framework can be obtained following the same idea used for the MPA: we transpose one of the two branches of the chain, as below



The same procedure could be done on the Z branch of the chain without altering the solution, since it is symmetric. Graphically, the position of the Y state does not follow the classical alternation, but this is irrelevant in this case, since there is no link between Y and all the other states.

Chapter 6

Observables

To quantify the statistical structure of the system, we compute the *average density* and the *two-point correlator*, both evaluated over the full ensemble of possible configurations. These quantities serve as one- and two-point observables, respectively, providing a window into local densities and spatial structure in the system. Since the system lacks homogeneity, all quantities retain their full site dependence and are computed without assumptions of translational invariance.

6.1 Average density in stationary states of the closed system

Called the space of all configurations $\mathcal{C}$, each configuration $C \in \mathcal{C}$ is specified by a binary occupation string $C = (s_1, s_2, \ldots, s_L)$, where $s_j(C) \in \{0,1\}$ denotes the state of site j in configuration C. The probability of configuration C is denoted $p(C)$, normalised such that

$$\sum_{C \in \mathcal{C}} p(C) = 1 \tag{6.1}$$

According to the MPA, we define the two weights for each time slice $w(C)$ and $w'(C)$ as in (4.5) and (4.6) respectively[1], thus

$$p^{(\prime)}(C) = \frac{w^{(\prime)}(C)}{\sum_{C \in \mathcal{C}} w^{(\prime)}(C)}$$

such that (6.1) is fulfilled; at the same time the definition of a partition sum $\mathcal{Z}^{(\prime)} = \sum_{C \in \mathcal{C}} w^{(\prime)}(C)$ is possible.
Considering the existence of two classes of impurity solutions, we have to define two different

[1] A question may arise whether the previous definitions of probability regarding (4.5) and (4.6) are correct or not. Previously we referred to the two expressions as "probabilities", since their role is indeed the one of a probability, despite normalisation was not taken into account. This definition mismatch can be easily solved by considering weights $w^{(\prime)}(C)$ as non-normalised probabilities.

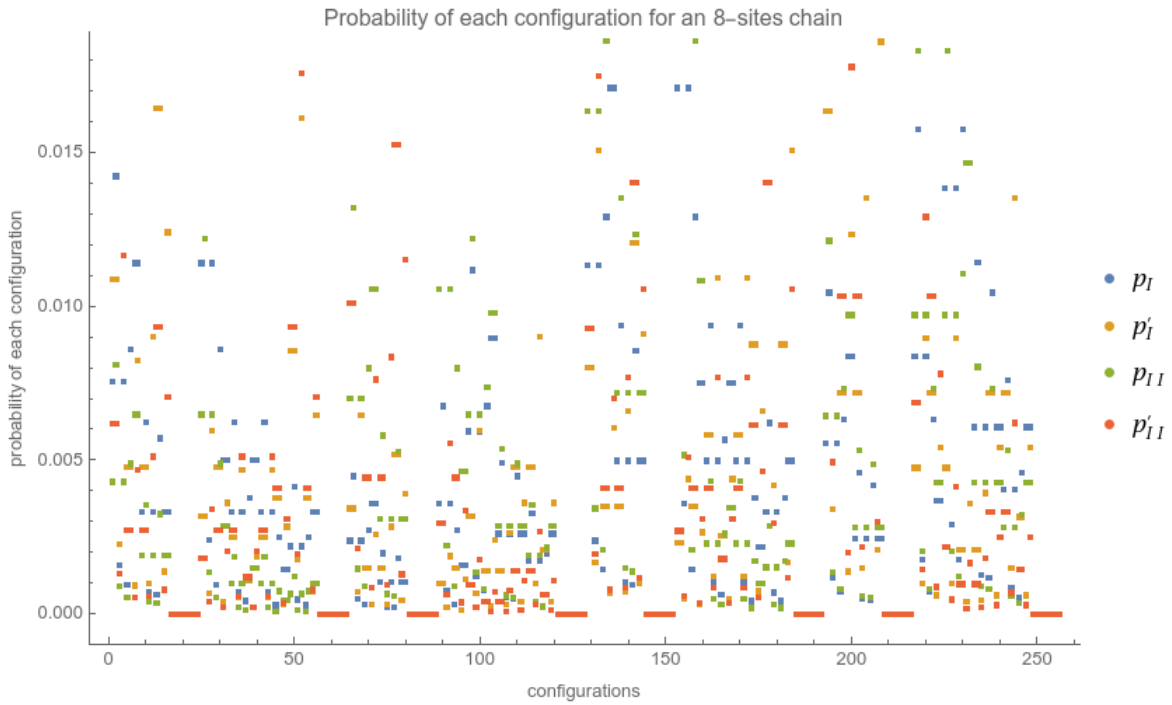


Figure 6.1: (a) The image shows the probabilities $p_\mu^{(\prime)}$ of every possible configuration for a chain of 8 sites with a rule 150 impurity on site 4. The configurations are listed as (0,0,0,0,0,0,0,0), (0,0,0,0,0,0,0,1), ..., (1,1,1,1,1,1,1,1).

probabilities, one for each impurity class

$$p_{\mu,\underline{s}} = p_{\mu,\,s_1,\ldots,s_n} = \frac{1}{\mathcal{Z}}\langle \boldsymbol{l}_{s_1}|\boldsymbol{A}'_{s_2}\boldsymbol{A}_{s_3}\ldots\boldsymbol{A}_{s_{i-1}}\boldsymbol{W}'^{\,\mu}_{s_i}\boldsymbol{B}_{s_{i+1}}\ldots\boldsymbol{B}'_{s_{n-2}}|\boldsymbol{r}_{s_{n-1},s_n}\rangle$$
$$p'_{\mu,\underline{s}} = p'_{\mu,\,s_1,\ldots,s_n} = \frac{1}{\mathcal{Z}'}\langle \boldsymbol{l}'_{s_1,s_2}|\boldsymbol{A}'_{s_3}\ldots\boldsymbol{A}'_{s_{i-1}}\boldsymbol{W}^{\mu}_{s_i}\boldsymbol{B}'_{s_{i+1}}\ldots\boldsymbol{B}'_{s_{n-1}}|\boldsymbol{r}'_{s_n}\rangle$$

where the index $\mu = 1{,}2$ indicates the considered impurity solution's class. A numerical result is shown in Figure 6.1. We define the average density at site j as the expected occupation at that site, that is

$$\rho_j = \sum_{C\in\mathcal{C}} s_j(C)\,p(C) \tag{6.2}$$

By evaluating that we notice two important features

1. the impurity site shows always a smaller density (expected occupation) than all the others while the boundaries' densities are strongly influenced by the boundary parameters, as expected;

2. varying the boundary parameters there are no strong changes in the overall "spatial structure" – i.e., the impurity site has always the smallest density among the others. This implies the correlation matrix has always a particular structure itself.

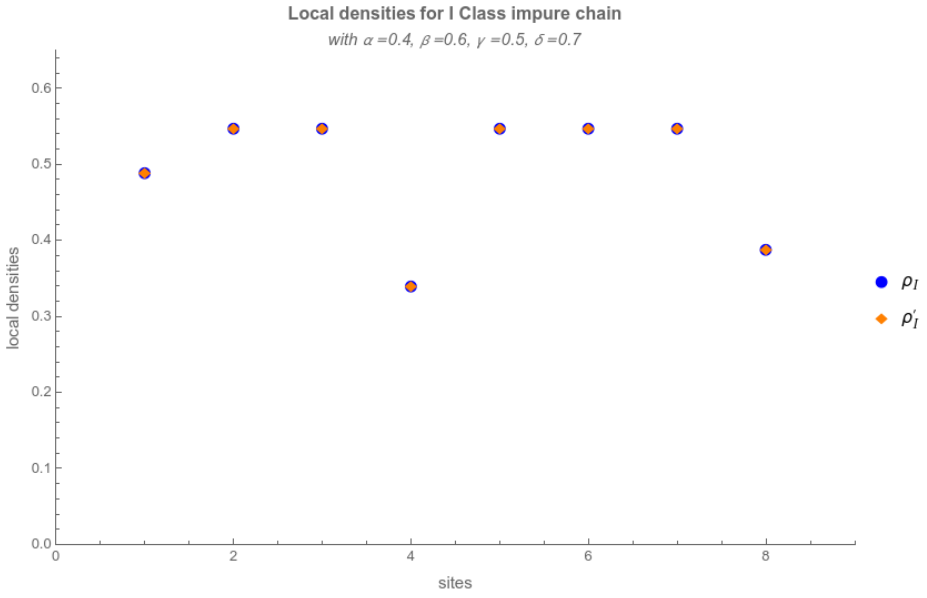
Various numerical results, varying the boundary parameters, are presented in Figure 6.2.

6.2 Two-Points correlators

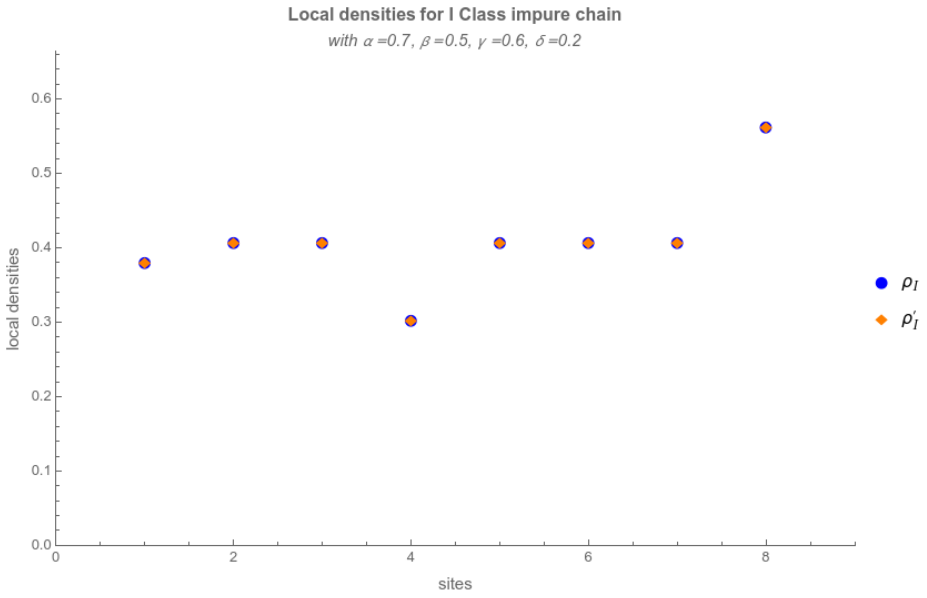
The knowledge – and with it, the definition – of the local average density allows us to evaluate the two points correlator. To measure spatial correlations between different sites, we define the two-point correlator as

$$G_{\ell,j} = \sum_{C \in \mathcal{C}} s_\ell(C)\, s_j(C)\, p(C)$$

which captures the joint probability that both site ℓ and site j are simultaneously occupied, averaged over the configuration ensemble. Figure 6.3 shows the numerical results for a chain with a rule 150 impurity. As expected, two-point correlators involving the impurity – here, the fourth site – take noticeably lower values compared to those between bulk sites, clearly visible in the corresponding rows and columns.

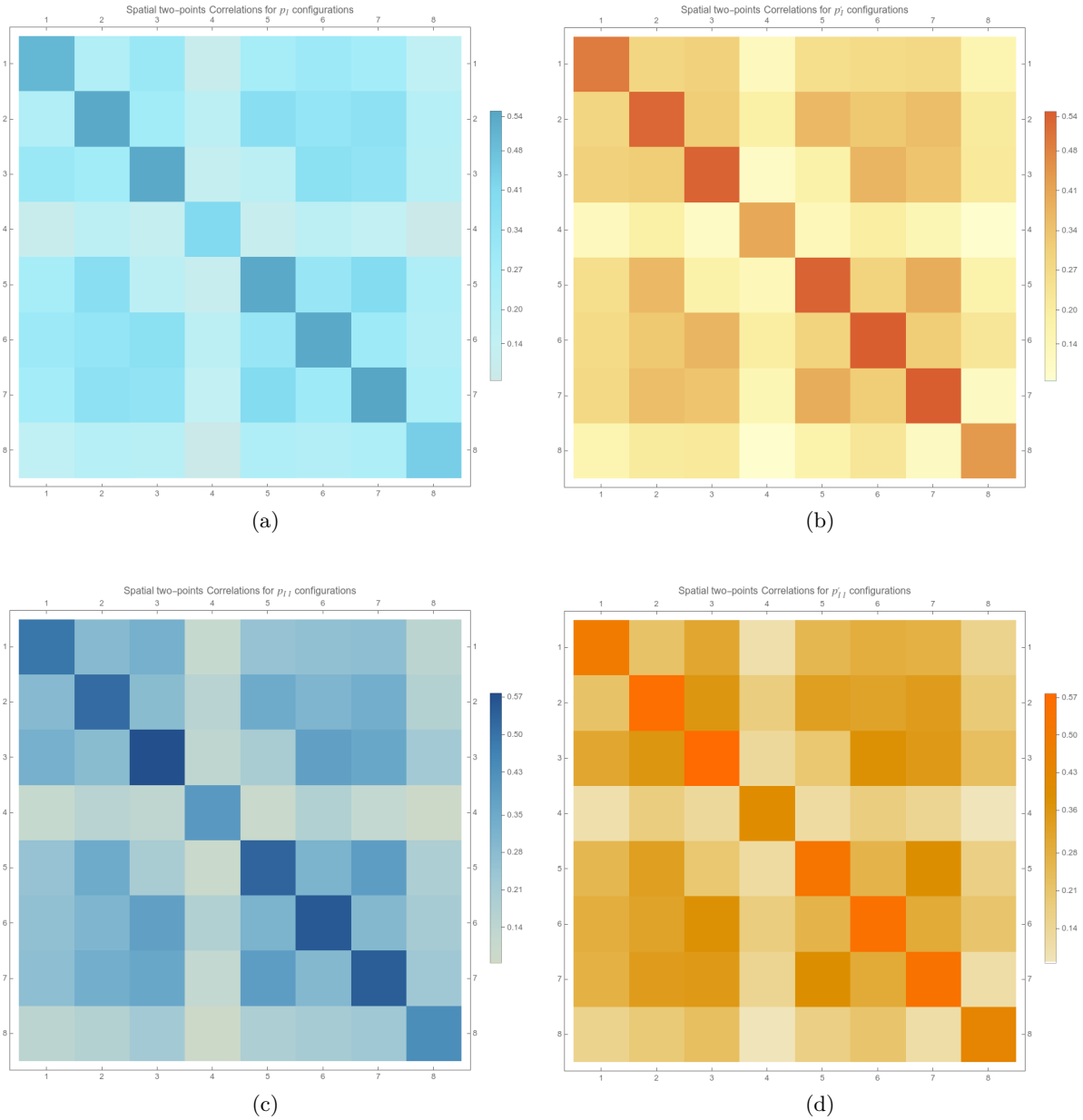


(a)



(b)

Figure 6.2: The average density plot shows the "depression" on the rule 150 impurity site with I class impurity solution. This behaviour was expected, according to the kind of interactions a mover can have with this impurity, well displayed in Figures 4.1a, 4.1b and 4.1c, where the first interaction might be the major contribute to this decreasing of density. The other states, on the contrary, exhibit higher average density.



(a) (b) (c) (d)

Figure 6.3: Two-points correlator for I (a,b) and II (c,d) class impurity chain solution, for p (a,c) and p' (b,d) time slices. The influence of the impurity is evident in the lower values across the row and the column that represents the impurity site.

Chapter 7

RCA as simple models for solitons dynamics

The results obtained here can be compared – with some degree of equivalence – to other results concerning the soliton behaviour in the presence of an impurity. Following the work by Bobenko *et al.* [20], we can draw a parallel between this theoretical approach and the results in [34–36]. Before discussing the similarities between the two models, we have to point out some important details. When two solitons meet at the impurity our model produces a 0 impurity state, while in fact the real value is the sum of the two solitons' values $1+1$, that in our specific case is 0 due to the modulo 2. Moreover, it is known that two scattering solitons with the same amplitude can either "resonate", if they have different velocities, and can be distinguished no more, or they can merge, if they have the same velocity, into one. Since the solitons emerging from the scattering exhibit the same velocity as before the scattering and experience a phase shift in the bulk – under rule 54 – we can conclude that the state 0 or 1 of the cell has nothing to do with the amplitude of the solitons and is, instead, a tag of the absence (0 state) or presence (1 state) of the mover in that cell. Secondly, the dynamics of our model can, in principle, continue indefinitely. It does not display dissipative effects, nor does it show signs of slowdowns or resonances, which often characterize real soliton systems. Nevertheless, the model could offer a useful framework for exploring and understanding soliton behaviour. The soliton-impurity interaction has been widely studied recently, and the correlation between the soliton behaviour and some impurity properties has been deeply analysed. In particular, it has been pointed out that if a soliton scatters with a highly interacting [34, 35] or heavy mass [36] impurity, it will be reflected. This reflection can be a total reflection (completely elastic) – meaning that 100% of the soliton has been reflected and the final velocity is the same as the initial one – or partially reflected – meaning some energy has been dissipated and the emerging velocity is smaller than the impact one, or a small resonant phenomenon occurred and a delay in the reflection happens. Considering that in our model energy doesn't exist, since the model is not invariant under continuous time-translations, but only discrete ones, only the total reflection can occur in this model and this behaviour appears when a mover encounters a rule 204 impurity. The absence of transmission phenomena, together with the presence of reflection phenomena in the rule 204 impurity case is suggestive that an absolutely non-interacting rule (such as rule

204) can model, with good approximation, highly interacting or heavy impurities, in particular when we can neglect reflection delays – e.g., for long timescales – or small energy dissipations. A good transmission is retrieved when dealing with rule 150 impurity, instead. If we can neglect the small velocity oscillation that occurs when the soliton path crosses the impurity, the rule 150 impurity framework provides a good tool to analyse the solitons' behaviour interacting with light or weakly interacting impurities. When considering excited impurities, instead, we still need to understand whether the emission found in the IRCA models can model a real emission – often spontaneous, typical of light impurities and entailed by resonances – or a combined emission-reflection-trapping mechanism. Since trapping mechanism has not been retrieved in these IRCA scenarios, we favour the interpretation that they model a real emission that, in a real situation, will damp after a certain time due to the depletion of energy (partially re-input in the system, partially dissipated). If the excitation is artificially induced, then the IRCA model is a very good theoretical tool to analyse an emerging emitting behaviour.

Chapter 8

Conclusions

In this Master's Thesis work we have investigated the dynamics of the interacting reversible cellular automaton (RCA54). In particular, we have focussed on the behaviour of RCA54 in presence of an impurity, denoted as IRCA54. While exactly solvable models of non-equilibrium statistical mechanics based on RCA have been previously studied, the effect of an impurity had not been analysed until now. Impurity problems represent an important paradigm of condensed matter physics. They also represent a special class of integrable and exactly solvable systems, such as the Kondo spin impurity problem, for which the numerical renormalisation group method has first been developed, or Anderson impurity problem. However, closed form solutions are difficult to obtain even within the context of integrable impurity systems. This is particularly true of real-time dynamical problems. In this context the study of interacting RCA, such as the RCA54, have recently proven very useful, as many paradigms of non-equilibrium statistical physics can be understood and calculated exactly and in closed form.

In this work, we have provided explicit results on the non-equilibrium statistical mechanics of IRCA54, showing that not only does it exhibit the main features of continuous integrable systems, but it also helps elucidate how complexity can emerge from minimal, local interactions. Generalising previous results, this work shows how it is possible to retrieve an exact solution that allows the understanding of spontaneous emission, time-crystal behaviour and scattering. We exactly computed the steady state (associated with the eigenvalue $+1$ eigenvector of a Markov chain) corresponding to boundary driven I150RCA54 and I204RCA54 using the *conditional* boundary driving, previously presented in [9, 10]. Inspecting the numerical Schmidt spectra for small chains we noticed a constant rank matrix product steady state which is twice or even thrice (as for rule 204 impurity) degenerate, prompting us to analytically construct a steady state manifold. This manifold is a family of steady states, among which one or more degenerate eigenstates related to the -1 eigenvalue of the evolution operator – which have been proven to represent time-crystal behaviours of the system – are present.

Building on these exact results, we turned our attention to the role of impurities and localized structures within RCA dynamics, uncovering surprising parallels with soliton behaviour in continuous systems. Although RCA is an abstract, "mute" model with no physical notion of

energy or momentum, the behaviour of soliton-like structures in the presence of impurities can be meaningfully interpreted in terms of information transmission. Reflections, transmissions, and emissions correspond to distinct modes of information flow, making RCA a valuable tool for modeling not just qualitative dynamics, but also information-theoretic analogs of soliton behaviour in complex media. For the first time, we have obtained exact solutions for RCA models with impurities, demonstrating how specific impurity rules (notably rule 204 and rule 150) give rise to emergent behaviours that closely mirror soliton-impurity interactions observed in continuous systems. The correspondence between total reflection and rule 204, and partial transmission under rule 150, provides a discrete analogue to classical results involving heavy and light impurities, respectively. Moreover, the study of IRCA-induced emissions opens a novel direction, suggesting a possible analogy with spontaneous emission in resonant soliton systems – even in the absence of dissipation or energy – thereby expanding the interpretative scope of these models. In conclusion, we have shown that integrable RCA and IαRCA do more than replicate known physical phenomena in a simplified setting. They provide a powerful lens through which we can not only understand, but also reconsider, the foundations of dynamical systems. In particular, IαRCA models – where α is the number of the rule that governs the impurity cell – offer a discrete formal language in which fundamental aspects of the theoretical physics of systems with impurities may be re-expressed and explored with new clarity.

Code to determine the existence of finite solutions

(*Definition of boundary matrices*)[1]

$$\mathbf{PL} = \begin{pmatrix} \alpha & 0 & \alpha & 0 \\ 0 & \beta & 0 & \beta \\ 1-\alpha & 0 & 1-\alpha & 0 \\ 0 & 1-\beta & 0 & 1-\beta \end{pmatrix};$$

$$\mathbf{PR} = \begin{pmatrix} \gamma & \gamma & 0 & 0 \\ 1-\gamma & 1-\gamma & 0 & 0 \\ 0 & 0 & \delta & \delta \\ 0 & 0 & 1-\delta & 1-\delta \end{pmatrix};$$

```
P0 = Table[(*150, 3x3 + 5x5 + 2x2MPA − state-nonergodicwithtimecrystal*)
{i1, i2, i3} = IntegerDigits[i,2,3];
{j1, j2, j3} = IntegerDigits[j,2,3];
If[(i1==j1)&&(i3==j3)&&(j2==Mod[i1 + i2 + i3,2]),1,0], {i,0,7}, {j,0,7}];

MatrixForm[P0]
```

[1]Notice that in codes letters U and P are used indistindiscriminately to indicate the evolution matrices of the staggered evolution operator $\mathcal{U}$.

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

```
P2 = Table[(*204, 3x3MPA − state-nonergodicwith2timecrystal*)
{i1, i2, i3} = IntegerDigits[i,2,3];
{j1, j2, j3} = IntegerDigits[j,2,3];
If[(i1==j1)&&(i3==j3)&&(j2==Mod[i2,2]),1,0], {i,0,7}, {j,0,7}];

MatrixForm[P2]
```

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

```
P = SparseArray[{{1,1} → 1 − a, {2,4} → 1, {3,3} → 1 − d, {1,3} → d,
    {3,1} → a, {4,2} → 1, {5,7} → 1, {6,8} → 1, {7,5} → 1, {8,6} → 1}, {8,8}];[2]

MatrixForm[P]
```

[2] Value a and d are set to 0, they are necessary only to consider probabilistic update rules.

$$\begin{pmatrix} 1-a & 0 & d & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ a & 0 & 1-d & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

```
TP:=KroneckerProduct


SA:=SparseArray


(* We Place Free Impurity in the middle of "odd" time step *)


DO[m_]:=
(dum = SA[IdentityMatrix[2^(2m)]];
Do[dum = dum.SA[TP[IdentityMatrix[2^(2k - 1)], If[k==(m - 1)/2, P2, P],
     IdentityMatrix[2^(2m - 2k - 2)]]], {k, m - 2}];
dum = dum.SA[TP[IdentityMatrix[2^(2m - 3)], P]])


DE[m_]:=
(dum = SA[IdentityMatrix[2^(2m)]];
Do[dum = dum.SA[TP[IdentityMatrix[2^(2m - 2k - 2)], P,
     IdentityMatrix[2^(2k - 1)]]], {k, m - 2}];
dum = dum.SA[TP[P, IdentityMatrix[2^(2m - 3)]]])


PropExt[m_, α_, β_, γ_, δ_]:=
```

$$\left(\mathbf{PL} = \begin{pmatrix} \alpha & 0 & \alpha & 0 \\ 0 & \beta & 0 & \beta \\ 1-\alpha & 0 & 1-\alpha & 0 \\ 0 & 1-\beta & 0 & 1-\beta \end{pmatrix};\ \mathbf{PR} = \begin{pmatrix} \gamma & \gamma & 0 & 0 \\ 1-\gamma & 1-\gamma & 0 & 0 \\ 0 & 0 & \delta & \delta \\ 0 & 0 & 1-\delta & 1-\delta \end{pmatrix};\right.$$

$$(\mathbf{Uo} = \mathbf{DO}[m].\mathbf{SA}[\mathbf{TP}[\mathbf{PL}, \mathbf{IdentityMatrix}[2\text{^}(2m-2)]]]).$$

$$\left.(\mathbf{Ue} = \mathbf{DE}[m].\mathbf{SA}[\mathbf{TP}[\mathbf{IdentityMatrix}[2\text{^}(2m-2)], \mathbf{PR}]])\right)$$

$a = 0.0;$

$d = 0.0;$

$\alpha = N[1/2];$

$\beta = N[2/3];$

$\gamma = N[2/5];$

$\delta = N[3/7];$

$U = \mathbf{PropExt}[\mathbf{chain\ dimension}, \alpha, \beta, \gamma, \delta];$

$\mathbf{Dimensions}[U]$

$\{\mathbf{eval}, \mathbf{evec}\} = \mathbf{Eigensystem}[U];$

$\mathbf{Take}[\mathbf{eval}]$

(*For example*)

$\{1. + 0.i, 1. + 0.i, -1. + 0.i, -1. + 0.i, 1. + 0.i, \ldots\}$

(*To finally evaluate the Schmidt rank dimension we use*)

$\mathbf{SingularValueList}[\mathbf{Partition}[\mathbf{evec}[[\mathbf{\#\ eigenv\ in\ list}]], 2\text{^}[\mathbf{length\ of\ the\ chain}]]]$

Code to find MPA Solutions (for rule 150 impurity)

Definition of 54 Rule

```
Clear[χ]
```

```
χ[0,0,0] = 0;
χ[0,0,1] = 1;
χ[0,1,0] = 1;
χ[0,1,1] = 0;
χ[1,0,0] = 1;
χ[1,0,1] = 1;
χ[1,1,0] = 0;
χ[1,1,1] = 0;
```

Definition of 150 Rule

```
Clear[ϕ]
```

```
ϕ[0,0,0] = 0;
ϕ[0,0,1] = 1;
ϕ[0,1,0] = 1;
ϕ[0,1,1] = 0;
```

```
ϕ[1,0,0] = 1;
ϕ[1,0,1] = 0;
ϕ[1,1,0] = 0;
ϕ[1,1,1] = 1;
```

S Matrix Definition

```
MatrixForm[S = {{1,0,0}, {0,0,1}, {0,1,0}}];
S.Transpose[S]==IdentityMatrix[3]
```

Unknown Impurity Matrices W and Z

```
Clear[Zp, W, Wp, Z, nvar, dim1, dim]
dim = 3;
dim1 = 3;
nvar = 0;
MatrixForm[Zp[0] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];
MatrixForm[Zp[1] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];
MatrixForm[W[0] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];
MatrixForm[W[1] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];
MatrixForm[Wp[0] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];
MatrixForm[Wp[1] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];
MatrixForm[Z[0] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];
MatrixForm[Z[1] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];
```

Direct Cancellation Cycle Bulk Matrices

```
Clear[A, Ap, B, Bp, ξ, ζ, ω, η]
```

```
MatrixForm[A[0] = {{1,0,0}, {ξ,0,0}, {1,0,0}}]
MatrixForm[Ap[0] = {{1,0,0}, {ω,0,0}, {1,0,0}}]
MatrixForm[A[1] = {{0, ξ,0}, {0,0,1}, {0,0, ω}}]
MatrixForm[Ap[1] = {{0, ω,0}, {0,0,1}, {0,0, ξ}}]
MatrixForm[AA[0] = {{1,0,0}, {ξ,0,0}, {1,0,0}}.S]
MatrixForm[AA[1] = {{0, ξ,0}, {0,0,1}, {0,0, ω}}.S]
MatrixForm[AAp[0] = {{1,0,0}, {ω,0,0}, {1,0,0}}.S]
MatrixForm[AAp[1] = {{0, ω,0}, {0,0,1}, {0,0, ξ}}.S]
MatrixForm[B[0] = {{1,0,0}, {ζ,0,0}, {1,0,0}}]
MatrixForm[Bp[0] = {{1,0,0}, {η,0,0}, {1,0,0}}]
MatrixForm[B[1] = {{0, ζ,0}, {0,0,1}, {0,0, η}}]
MatrixForm[Bp[1] = {{0, η,0}, {0,0,1}, {0,0, ζ}}]
MatrixForm[BB[0] = {{1,0,0}, {ζ,0,0}, {1,0,0}}.S]
MatrixForm[BBp[0] = {{1,0,0}, {η,0,0}, {1,0,0}}.S]
MatrixForm[BB[1] = {{0, ζ,0}, {0,0,1}, {0,0, η}}.S]
MatrixForm[BBp[1] = {{0, η,0}, {0,0,1}, {0,0, ζ}}.S]
```

Direct System Equations

Pure Bulk System

```
MatrixForm[bulkeq = Union[Flatten[Table[{s1, s2, s3} = IntegerDigits[s,2,3];
{AAp[s1].Ap[χ[s1, s2, s3]].A[s3] − Ap[s1].A[s2].AAp[s3],
A[s1].Ap[χ[s1, s2, s3]].AA[s3] − AA[s1].A[s2].Ap[s3],
BBp[s1].Bp[χ[s1, s2, s3]].B[s3] − Bp[s1].B[s2].BBp[s3],
```

B[s1].Bp[χ[s1, s2, s3]].BB[s3] − BB[s1].B[s2].Bp[s3]}, {s,0,7}]]]]

Direct Cancellation Cycle Impurity System

MatrixForm[eqs = Union[Flatten[Table[{s1, s2, s3} = IntegerDigits[s,2,3];

{Zp[s1].Bp[χ[s1, s2, s3]].B[s3] − Wp[s1].B[s2].BBp[s3],

AAp[s1].Ap[χ[s1, s2, s3]].W[s3] − Ap[s1].A[s2].Zp[s3],

A[s1].Wp[ϕ[s1, s2, s3]].BB[s3] − AA[s1].W[s2].Bp[s3]

}, {s,0,7}]]]];

Direct solutions

First Class Solutions

sol1 = Solve[eqs==0, Table[x[j], {j,1, nvar}] ∼ Join ∼ {ξ, η, ω}]; [3]

MatrixForm[W[0]/.sol1[[34]]]

MatrixForm[W[1]/.sol1[[34]]]

MatrixForm[Wp[0]/.sol1[[34]]]

[3] Computationally, this class can be retrieved computing with (ξ, η) or (ω, ζ) as independent parameters. This is simply verified by the assumption $\xi = \zeta,\ \omega = \eta$. The computation with ξ, ζ and (ω, η) returns the third class solutions. These first class solutions have been evaluated by using ξ, ω and η as independent parameters. If one uses (ξ, ω, ζ) the result is similar but with a different displacement of some entries and the factor $(\eta - 1)$ instead of $(\zeta - 1)$, as follows

$$\boldsymbol{W}_0 = x \begin{pmatrix} \frac{(\zeta\eta-1)}{(\eta-1)\eta} & 0 & 0 \\ 0 & \frac{(\zeta-1)\zeta}{(\eta-1)\eta} & \frac{\zeta}{\eta} \\ \frac{(\zeta\eta-1)}{(\eta-1)\eta} & 0 & 0 \end{pmatrix} \qquad \boldsymbol{W}_1 = x \begin{pmatrix} 0 & -\frac{\zeta}{(\eta-1)\eta} & \frac{\zeta}{\eta-1} \\ 0 & \frac{\zeta}{(\eta-1)\eta} & -\frac{1}{(\eta-1)\eta} \\ 0 & \frac{\zeta}{\eta-1} & -\frac{1}{\eta-1} \end{pmatrix}$$

$$\boldsymbol{W}_0' = x \begin{pmatrix} \frac{(\zeta\eta-1)}{(\eta-1)\eta} & 0 & 0 \\ 0 & 1 & \frac{(\zeta-1)}{\eta-1} \\ \frac{(\zeta\eta-1)}{(\eta-1)\eta} & 0 & 0 \end{pmatrix} \qquad \boldsymbol{W}_1' = x \begin{pmatrix} 0 & -\frac{1}{\eta-1} & \frac{\zeta}{\eta-1} \\ 0 & \frac{1}{\eta-1} & -\frac{1}{(\eta-1)\eta} \\ 0 & \frac{\zeta}{\eta-1} & -\frac{\zeta}{(\eta-1)\eta} \end{pmatrix}$$

```
MatrixForm[Wp[1]/.sol1[[34]]]
MatrixForm[Zp[0]/.sol1[[34]]]
MatrixForm[Zp[1]/.sol1[[34]]]
```

Transposed Cancellation Cycle Bulk Matrices

```
Clear[F, G, FF, GG, Fp, Gp, FFp, GGp, ξ, ζ, ω, η]
```

$\xi = \eta;$

$\omega = \zeta;$

```
MatrixForm[Fp[0] = Transpose[{{1,0,0}, {ξ,0,0}, {1,0,0}}]]
MatrixForm[F[0] = Transpose[{{1,0,0}, {ω,0,0}, {1,0,0}}]]
MatrixForm[Fp[1] = Transpose[{{0, ξ,0}, {0,0,1}, {0,0, ω}}]]
MatrixForm[F[1] = Transpose[{{0, ω,0}, {0,0,1}, {0,0, ξ}}]]
MatrixForm[G[0] = {{1,0,0}, {ζ,0,0}, {1,0,0}}]
MatrixForm[Gp[0] = {{1,0,0}, {η,0,0}, {1,0,0}}]
MatrixForm[G[1] = {{0, ζ,0}, {0,0,1}, {0,0, η}}]
MatrixForm[Gp[1] = {{0, η,0}, {0,0,1}, {0,0, ζ}}]
MatrixForm[FFp[0] = S.Transpose[{{1,0,0}, {ξ,0,0}, {1,0,0}}]]
MatrixForm[FF[0] = S.Transpose[{{1,0,0}, {ω,0,0}, {1,0,0}}]]
MatrixForm[FFp[1] = S.Transpose[{{0, ξ,0}, {0,0,1}, {0,0, ω}}]]
MatrixForm[FF[1] = S.Transpose[{{0, ω,0}, {0,0,1}, {0,0, ξ}}]]
MatrixForm[GG[0] = {{1,0,0}, {ζ,0,0}, {1,0,0}}.S]
MatrixForm[GGp[0] = {{1,0,0}, {η,0,0}, {1,0,0}}.S]
MatrixForm[GG[1] = {{0, ζ,0}, {0,0,1}, {0,0, η}}.S]
MatrixForm[GGp[1] = {{0, η,0}, {0,0,1}, {0,0, ζ}}.S]
```

Transposed System Equations

Transposed Pure Bulk System

```
MatrixForm[bulkeqT = Union[Flatten[Table[{s1, s2, s3} = IntegerDigits[s,2,3];
{FFp[s1].F[χ[s1, s2, s3]].Fp[s3] − F[s1].Fp[s2].FFp[s3],
F[s1].Fp[χ[s1, s2, s3]].FFp[s3] − FFp[s1].F[s2].Fp[s3],
GGp[s1].Gp[χ[s1, s2, s3]].G[s3] − Gp[s1].G[s2].GGp[s3],
G[s1].Gp[χ[s1, s2, s3]].GG[s3] − GG[s1].G[s2].Gp[s3]}, {s,0,7}]]]]
```

Transposed Cancellation Cycle Impurity System

```
MatrixForm[eqsT = Union[Flatten[Table[{s1, s2, s3} = IntegerDigits[s,2,3];
{Zp[s1].Gp[χ[s1, s2, s3]].G[s3] − Wp[s1].G[s2].GGp[s3],
FFp[s1].F[χ[s1, s2, s3]].W[s3] − F[s1].Fp[s2].Zp[s3],
Fp[s1].Wp[ϕ[s1, s2, s3]].GG[s3] − FF[s1].W[s2].Gp[s3]
}, {s,0,7}]]]];
```

Left and Right Boundaries of I Class Solution

```
Clear[χ]

χ[0,0,0] = 0;
χ[0,0,1] = 1;
χ[0,1,0] = 1;
χ[0,1,1] = 0;
χ[1,0,0] = 1;
χ[1,0,1] = 1;
χ[1,1,0] = 0;
```

$\chi[1,1,1] = 0;$

$$\mathbf{PL} = \begin{pmatrix} \alpha & 0 & \alpha & 0 \\ 0 & \beta & 0 & \beta \\ 1-\alpha & 0 & 1-\alpha & 0 \\ 0 & 1-\beta & 0 & 1-\beta \end{pmatrix};$$

$$\mathbf{PR} = \begin{pmatrix} \gamma & \gamma & 0 & 0 \\ 1-\gamma & 1-\gamma & 0 & 0 \\ 0 & 0 & \delta & \delta \\ 0 & 0 & 1-\delta & 1-\delta \end{pmatrix};$$

$\mathbf{MatrixForm}[S = \{\{1,0,0\},\{0,0,1\},\{0,1,0\}\}]$

$\mathbf{MatrixForm}[A[0] = \{\{1,0,0\},\{\xi,0,0\},\{1,0,0\}\}]$

$\mathbf{MatrixForm}[\mathbf{Ap}[0] = \{\{1,0,0\},\{\omega,0,0\},\{1,0,0\}\}]$

$\mathbf{MatrixForm}[A[1] = \{\{0,\xi,0\},\{0,0,1\},\{0,0,\omega\}\}]$

$\mathbf{MatrixForm}[\mathbf{Ap}[1] = \{\{0,\omega,0\},\{0,0,1\},\{0,0,\xi\}\}]$

$\mathbf{MatrixForm}[B[0] = \{\{1,0,0\},\{\zeta,0,0\},\{1,0,0\}\}]$

$\mathbf{MatrixForm}[\mathbf{Bp}[0] = \{\{1,0,0\},\{\eta,0,0\},\{1,0,0\}\}]$

$\mathbf{MatrixForm}[B[1] = \{\{0,\zeta,0\},\{0,0,1\},\{0,0,\eta\}\}]$

$\mathbf{MatrixForm}[\mathbf{Bp}[1] = \{\{0,\eta,0\},\{0,0,1\},\{0,0,\zeta\}\}]$

$$\mathbf{MatrixForm}\left[W[0] = \begin{pmatrix} \frac{(-1+\zeta\eta)y[1]}{(-1+\zeta)\eta} & 0 & 0 \\ 0 & \frac{\zeta y[1]}{\eta} & \frac{\zeta(-1+\eta)y[1]}{(-1+\zeta)\eta} \\ \frac{(-1+\zeta\eta)y[1]}{(-1+\zeta)\eta} & 0 & 0 \end{pmatrix}\right]$$

$$\mathbf{MatrixForm}\left[W[1] = \begin{pmatrix} 0 & -\frac{\zeta y[1]}{(-1+\zeta)\eta} & \frac{\zeta y[1]}{-1+\zeta} \\ 0 & \frac{\zeta y[1]}{(-1+\zeta)\eta} & -\frac{y[1]}{(-1+\zeta)\eta} \\ 0 & \frac{\zeta y[1]}{-1+\zeta} & -\frac{y[1]}{-1+\zeta} \end{pmatrix}\right]$$

$$\texttt{MatrixForm}\left[\texttt{Wp}[0] = \begin{pmatrix} \frac{(-1+\zeta\eta)y[1]}{(-1+\zeta)\eta} & 0 & 0 \\ 0 & \frac{(-1+\eta)y[1]}{-1+\zeta} & y[1] \\ \frac{(-1+\zeta\eta)y[1]}{(-1+\zeta)\eta} & 0 & 0 \end{pmatrix}\right]$$

$$\texttt{MatrixForm}\left[\texttt{Wp}[1] = \begin{pmatrix} 0 & -\frac{y[1]}{-1+\zeta} & \frac{\zeta y[1]}{-1+\zeta} \\ 0 & \frac{y[1]}{-1+\zeta} & -\frac{y[1]}{(-1+\zeta)\eta} \\ 0 & \frac{\zeta y[1]}{-1+\zeta} & -\frac{\zeta y[1]}{(-1+\zeta)\eta} \end{pmatrix}\right]$$

```
dim1 = 1;

dim = 3;

nvar = 0;

MatrixForm[left[0] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];

MatrixForm[left[1] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];

MatrixForm[leftp[0,0] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];

MatrixForm[leftp[0,1] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];

MatrixForm[leftp[1,0] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];

MatrixForm[leftp[1,1] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];

MatrixForm[rightp[0] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];

MatrixForm[rightp[1] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];

MatrixForm[right[0,0] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];

MatrixForm[right[0,1] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];

MatrixForm[right[1,0] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];

MatrixForm[right[1,1] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];

nvar
```

Left Boundary

```
MatrixForm[Leqs = Union[Flatten[Table[{s1, s2, s3} = IntegerDigits[s,2,3];

{Sum[PL[[2s1 + s2 + 1,2t1 + t2 + 1]] * left[t1].Ap[χ[t1, t2, s3]].A[s3].S, {t1,0,1},
```

```
{t2,0,1}] − ΛL * left[s1].Ap[s2].S.Ap[s3]}, {s,0,7}]]]//FullSimplify

Lsolution = Solve[Leqs==0, Table[x[k], {k,1, nvar}] ∼ Join ∼ {ξ, ω}]

Lsol = FullSimplify[Lsolution]

MatrixForm[l[0] = left[0]/.Lsol[[−2]]]

MatrixForm[l[1] = left[1]/.Lsol[[−2]]]

MatrixForm[Leqs2 = Union[Flatten[Table[{s1, s2, s3} = IntegerDigits[s,2,3];

{leftp[s1, s2].Ap[s3].S − l[s1].Ap[χ[s1, s2, s3]].A[s3]}, {s,0,7}]]]]

Lsolution2 = FullSimplify[Solve[Leqs2==0, Table[x[k], {k,1, nvar}]]]

MatrixForm[lp[0,0] = leftp[0,0]/.Lsolution2[[1]]]

MatrixForm[lp[0,1] = leftp[0,1]/.Lsolution2[[1]]]

MatrixForm[lp[1,0] = leftp[1,0]/.Lsolution2[[1]]]

MatrixForm[lp[1,1] = leftp[1,1]/.Lsolution2[[1]]]
```

4

Right Boundary

```
MatrixForm[Reqs = Union[Flatten[Table[{s1, s2, s3} = IntegerDigits[s,2,3];

{Sum[PR[[2s2 + s3 + 1,2t2 + t3 + 1]] * A[s1].Ap[χ[s1, t2, t3]].Transpose[rightp[t3]],

{t2,0,1}, {t3,0,1}] − ΛR * A[s1].S.A[s2].S.Transpose[rightp[s3]]}, {s,0,7}]]]

Rsolution = Solve[Reqs==0, Table[x[k], {k,1, nvar}] ∼ Join ∼ {ξ, ω}]
```

[4]Notice that solving the Boundary Equations using Λ_R or Λ_L as variables is the same of solving them without this additional variable. For at the end, the two Λ s are completely free. Since we are looking at the NESS and this implies $\Lambda_R \cdot \Lambda_L$=1, we can put both Λ_L and Λ_R equal to 1. However, here we continue to use them explicitly just to understand better the behaviour of the solutions. Regarding the free entries we can set the free entries of the vectors as follows: For x[1] and x[19] we can say we need the probability a particle is inside the system, i.e., α(1-β) or γ(1-δ), and then we multiply it by its associated eigenvalue, that is $\Lambda_L\alpha$(1-β) and $\Lambda_R\gamma$(1-δ). A sign in front of them is pleonastic and by inserting it we retrieve the same solutions as in the non-impure case. x[4] = x[1] can be good. x[3]=x[6]=(β-1)Λ_L simplify the results. While x[2]=x[5]=0 is motivated by the fact they appear only one time among the vectors.

```
Rsol = FullSimplify[Rsolution]

MatrixForm[rp[0] = rightp[0]/.Rsol[[-1]]]
MatrixForm[rp[1] = rightp[1]/.Rsol[[-1]]]

MatrixForm[Reqs2 = Union[Flatten[Table[{s1, s2} = IntegerDigits[s,2,2];
{Sum[PR[[2s1 + s2 + 1,2t1 + t2 + 1]] * Transpose[right[t1, t2]], {t1,0,1}, {t2,0,1}]-
ΛR * A[s1].S.Transpose[rp[s2]]}, {s,0,3}]]]]

Rsolution2 = Solve[Reqs2==0, Table[x[k], {k,24, nvar}]]

MatrixForm[r[0,0] = right[0,0]/.Rsolution2[[1]]]
MatrixForm[r[0,1] = right[0,1]/.Rsolution2[[1]]]
MatrixForm[r[1,0] = right[1,0]/.Rsolution2[[1]]]
MatrixForm[r[1,1] = right[1,1]/.Rsolution2[[1]]]
```

Step-equations check

```
FullSimplify[MatrixForm[REQS = Union[Flatten[Table[{s1, s2, s3, s4, s5, s6} =
IntegerDigits[s,2,6];
{l[s1].Ap[s2].A[s3].Ap[s4].Sum[PR[[2s5 + s6 + 1,2t1 + t2 + 1]] * Transpose[r[t1, t2]],
{t1,0,1}, {t2,0,1}]
-ΛR * l[s1].Ap[s2].A[s3].Ap[s4].A[s5].S.Transpose[rp[s6]]}, {s,0,63}]]]]

FullSimplify[MatrixForm[Union[Flatten[Table[{s1, s2, s3, s4, s5, s6} =
IntegerDigits[s,2,6];
{l[s1].Ap[χ[s1, s2, s3]].A[s3].S.A[s4].Ap[s5].Transpose[rp[s6]]-
lp[s1, s2].Ap[s3].A[s4].Ap[s5].Transpose[rp[s6]]}, {s,0,63}]]]]
```

Left and Right Boundaries of II Class Solution

```
Clear[α, β, γ, δ, ξ, ω, ζ, η, ΛL, ΛR, x, PL, PR, F, Fp, G, Gp, FF, FFp, GG, GGp]
```

$$\mathtt{PL} = \begin{pmatrix} \alpha & 0 & \alpha & 0 \\ 0 & \beta & 0 & \beta \\ 1-\alpha & 0 & 1-\alpha & 0 \\ 0 & 1-\beta & 0 & 1-\beta \end{pmatrix};$$

$$\mathtt{PR} = \begin{pmatrix} \gamma & \gamma & 0 & 0 \\ 1-\gamma & 1-\gamma & 0 & 0 \\ 0 & 0 & \delta & \delta \\ 0 & 0 & 1-\delta & 1-\delta \end{pmatrix};$$

$$\mathtt{MatrixForm}\left[Z[0] = \begin{pmatrix} y[2] & -\frac{(-1+\zeta)\zeta\eta y[2]}{-1+\zeta\eta} & -\frac{\zeta(-1+\eta)\eta y[2]}{-1+\zeta\eta} \\ -\frac{\zeta(-1+\eta)\eta y[2]}{-1+\zeta\eta} & \frac{(-1+\zeta)\zeta(-1+\eta)\eta y[2]}{(-1+\zeta\eta)^2} & \frac{\zeta(-1+\eta)^2\eta y[2]}{(-1+\zeta\eta)^2} \\ -\frac{(-1+\zeta)\zeta\eta y[2]}{-1+\zeta\eta} & \frac{(-1+\zeta)^2\zeta\eta y[2]}{(-1+\zeta\eta)^2} & \frac{(-1+\zeta)\zeta(-1+\eta)\eta y[2]}{(-1+\zeta\eta)^2} \end{pmatrix}\right]$$

$$\mathtt{MatrixForm}\left[Z[1] = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \frac{2\zeta\eta y[2]}{-1+\zeta\eta} & -\frac{\eta(1+\zeta\eta)y[2]}{-1+\zeta\eta} \\ 0 & -\frac{\zeta(1+\zeta\eta)y[2]}{-1+\zeta\eta} & \frac{2\zeta\eta y[2]}{-1+\zeta\eta} \end{pmatrix}\right]$$

$$\mathtt{MatrixForm}\left[\mathtt{Zp}[0] = \begin{pmatrix} y[2] & -\frac{\zeta(-1+\eta)\eta y[2]}{-1+\zeta\eta} & -\frac{(-1+\zeta)\zeta\eta y[2]}{-1+\zeta\eta} \\ -\frac{(-1+\zeta)\zeta\eta y[2]}{-1+\zeta\eta} & \frac{(-1+\zeta)\zeta(-1+\eta)\eta y[2]}{(-1+\zeta\eta)^2} & \frac{(-1+\zeta)^2\zeta\eta y[2]}{(-1+\zeta\eta)^2} \\ -\frac{\zeta(-1+\eta)\eta y[2]}{-1+\zeta\eta} & \frac{\zeta(-1+\eta)^2\eta y[2]}{(-1+\zeta\eta)^2} & \frac{(-1+\zeta)\zeta(-1+\eta)\eta y[2]}{(-1+\zeta\eta)^2} \end{pmatrix}\right]$$

$$\mathtt{MatrixForm}\left[\mathtt{Zp}[1] = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \frac{2\zeta\eta y[2]}{-1+\zeta\eta} & -\frac{\zeta(1+\zeta\eta)y[2]}{-1+\zeta\eta} \\ 0 & -\frac{\eta(1+\zeta\eta)y[2]}{-1+\zeta\eta} & \frac{2\zeta\eta y[2]}{-1+\zeta\eta} \end{pmatrix}\right]$$

```
MatrixForm[Fp[0] = Transpose[{{1,0,0}, {ξ,0,0}, {1,0,0}}]];
MatrixForm[F[0] = Transpose[{{1,0,0}, {ω,0,0}, {1,0,0}}]];
MatrixForm[Fp[1] = Transpose[{{0,ξ,0}, {0,0,1}, {0,0,ω}}]];
```

```
MatrixForm[F[1] = Transpose[{{0,ω,0},{0,0,1},{0,0,ξ}}]];
MatrixForm[G[0] = {{1,0,0},{ζ,0,0},{1,0,0}}];
MatrixForm[Gp[0] = {{1,0,0},{η,0,0},{1,0,0}}];
MatrixForm[G[1] = {{0,ζ,0},{0,0,1},{0,0,η}}];
MatrixForm[Gp[1] = {{0,η,0},{0,0,1},{0,0,ζ}}];
MatrixForm[FFp[0] = S.Transpose[{{1,0,0},{ξ,0,0},{1,0,0}}]];
MatrixForm[FF[0] = S.Transpose[{{1,0,0},{ω,0,0},{1,0,0}}]];
MatrixForm[FFp[1] = S.Transpose[{{0,ξ,0},{0,0,1},{0,0,ω}}]];
MatrixForm[FF[1] = S.Transpose[{{0,ω,0},{0,0,1},{0,0,ξ}}]];
MatrixForm[GG[0] = {{1,0,0},{ζ,0,0},{1,0,0}}.S];
MatrixForm[GGp[0] = {{1,0,0},{η,0,0},{1,0,0}}.S];
MatrixForm[GG[1] = {{0,ζ,0},{0,0,1},{0,0,η}}.S];
MatrixForm[GGp[1] = {{0,η,0},{0,0,1},{0,0,ζ}}.S];
dim1 = 1;
dim = 3;
nvar = 36;
MatrixForm[left[0] = Table[x[++nvar],{i,1,dim1},{j,1,dim}]];
MatrixForm[left[1] = Table[x[++nvar],{i,1,dim1},{j,1,dim}]];
MatrixForm[leftp[0,0] = Table[x[++nvar],{i,1,dim1},{j,1,dim}]];
MatrixForm[leftp[0,1] = Table[x[++nvar],{i,1,dim1},{j,1,dim}]];
MatrixForm[leftp[1,0] = Table[x[++nvar],{i,1,dim1},{j,1,dim}]];
MatrixForm[leftp[1,1] = Table[x[++nvar],{i,1,dim1},{j,1,dim}]];
MatrixForm[rightp[0] = Table[x[++nvar],{i,1,dim1},{j,1,dim}]];
MatrixForm[rightp[1] = Table[x[++nvar],{i,1,dim1},{j,1,dim}]];
MatrixForm[right[0,0] = Table[x[++nvar],{i,1,dim1},{j,1,dim}]];
MatrixForm[right[0,1] = Table[x[++nvar],{i,1,dim1},{j,1,dim}]];
MatrixForm[right[1,0] = Table[x[++nvar],{i,1,dim1},{j,1,dim}]];
```

```
MatrixForm[right[1,1] = Table[x[++nvar], {i,1, dim1}, {j,1, dim}]];

nvar

MatrixForm[LeqsT = Union[Flatten[Table[{s1, s2, s3} = IntegerDigits[s,2,3];
{Sum[PL[[2s1 + s2 + 1,2t1 + t2 + 1]] * left[t1].F[χ[t1, t2, s3]].Fp[s3],
{t1,0,1}, {t2,0,1}] − ΛL * left[s1].FFp[s2].FFp[s3]}, {s,0,7}]]]

LsolT = Solve[LeqsT==0, Table[x[k], {k,1, nvar}] ~ Join ~ {ξ, ω}]

MatrixForm[lT[0] = left[0]/.LsolT[[−1]]]
MatrixForm[lT[1] = left[1]/.LsolT[[−1]]]

ζ = (−1+γ+δ−δΛR)/((−1+δ)ΛR^2);
η = ((γ−ΛR)ΛR)/(−1+δ);
ω = ((α−ΛL)ΛL)/(−1+β);
ξ = (−1+α+β−βΛL)/((−1+β)ΛL^2);

Clear[F, Fp, G, Gp, FF, FFp, GG, GGp]

MatrixForm[G[0] = {{1,0,0}, {ζ,0,0}, {1,0,0}}];
MatrixForm[Gp[0] = {{1,0,0}, {η,0,0}, {1,0,0}}];
MatrixForm[G[1] = {{0, ζ,0}, {0,0,1}, {0,0, η}}];
MatrixForm[Gp[1] = {{0, η,0}, {0,0,1}, {0,0, ζ}}];
MatrixForm[GG[0] = {{1,0,0}, {ζ,0,0}, {1,0,0}}.S];
MatrixForm[GGp[0] = {{1,0,0}, {η,0,0}, {1,0,0}}.S];
MatrixForm[GG[1] = {{0, ζ,0}, {0,0,1}, {0,0, η}}.S];
MatrixForm[GGp[1] = {{0, η,0}, {0,0,1}, {0,0, ζ}}.S];
MatrixForm[Fp[0] = Transpose[{{1,0,0}, {ξ,0,0}, {1,0,0}}]];
MatrixForm[F[0] = Transpose[{{1,0,0}, {ω,0,0}, {1,0,0}}]];
MatrixForm[Fp[1] = Transpose[{{0, ξ,0}, {0,0,1}, {0,0, ω}}]];
MatrixForm[F[1] = Transpose[{{0, ω,0}, {0,0,1}, {0,0, ξ}}]];
```

```
MatrixForm[FFp[0] = S.Transpose[{{1,0,0}, {ξ,0,0}, {1,0,0}}]];
MatrixForm[FF[0] = S.Transpose[{{1,0,0}, {ω,0,0}, {1,0,0}}]];
MatrixForm[FFp[1] = S.Transpose[{{0, ξ,0}, {0,0,1}, {0,0, ω}}]];
MatrixForm[FF[1] = S.Transpose[{{0, ω,0}, {0,0,1}, {0,0, ξ}}]];

MatrixForm[LeqsT2 = Union[Flatten[Table[{s1, s2, s3} = IntegerDigits[s,2,3];
{leftp[s1, s2].FF[s3] − lT[s1].Fp[χ[s1, s2, s3]].F[s3]}, {s,0,7}]]]]

LsolT2 = FullSimplify[Solve[LeqsT2==0, Table[x[k], {k,37, nvar}]]]

MatrixForm[lTp[0,0] = leftp[0,0]/.LsolT2[[1]]]
MatrixForm[lTp[0,1] = leftp[0,1]/.LsolT2[[1]]]
MatrixForm[lTp[1,0] = leftp[1,0]/.LsolT2[[1]]]
MatrixForm[lTp[1,1] = leftp[1,1]/.LsolT2[[1]]]
```

Left Boundary check

```
FullSimplify[MatrixForm[Union[Flatten[Table[{s1, s2, s3, s4, s5, s6} =
IntegerDigits[s,2,6];
{lT[s1].Fp[χ[s1, s2, s3]].FFp[s3].Zp[s4].B[s5].Bp[s6].Transpose[r[s7, s8]]−
lTp[s1, s2].Fp[s3].Zp[s4].B[s5].Bp[s6].Transpose[r[s7, s8]]}, {s,0,63}]]]]
```

Code to find PSA Solutions (for rule 150 impurity)

[5]

The update rules

$\chi[0,0,0] = 0;$

$\chi[0,0,1] = 1;$

$\chi[0,1,0] = 1;$

$\chi[0,1,1] = 0;$

$\chi[1,0,0] = 1;$

$\chi[1,0,1] = 1;$

$\chi[1,1,0] = 0;$

$\chi[1,1,1] = 0;$

$\phi[0,0,0] = 0;$

$\phi[0,0,1] = 1;$

$\phi[0,1,0] = 1;$

$\phi[0,1,1] = 0;$

$\phi[1,0,0] = 1;$

$\phi[1,0,1] = 0;$

[5]Here we show the results of the calculations in order to make the discussion clearer.

$\phi[1,1,0] = 0;$

$\phi[1,1,1] = 1;$

The boundary matrices

$\texttt{MatrixForm}[\texttt{PL} = \{\{\alpha,0,\alpha,0\},\{0,\beta,0,\beta\},\{1-\alpha,0,1-\alpha,0\},\{0,1-\beta,0,1-\beta\}\}]$

$$\begin{pmatrix} \alpha & 0 & \alpha & 0 \\ 0 & \beta & 0 & \beta \\ 1-\alpha & 0 & 1-\alpha & 0 \\ 0 & 1-\beta & 0 & 1-\beta \end{pmatrix}$$

$\texttt{MatrixForm}[\texttt{PR} = \{\{\gamma,\gamma,0,0\},\{1-\gamma,1-\gamma,0,0\},\{0,0,\delta,\delta\},\{0,0,1-\delta,1-\delta\}\}]$

$$\begin{pmatrix} \gamma & \gamma & 0 & 0 \\ 1-\gamma & 1-\gamma & 0 & 0 \\ 0 & 0 & \delta & \delta \\ 0 & 0 & 1-\delta & 1-\delta \end{pmatrix}$$

Bulk & Boundary Solutions

Bulk Matrices and Boundary Vectors

$\texttt{vars} = 0;$

$X[0,0] = x[\texttt{++vars}];$

$\texttt{Xp}[0,0] = x[\texttt{++vars}];$

$X[0,1] = x[\texttt{++vars}];$

$\texttt{Xp}[0,1] = x[\texttt{++vars}];$

$X[1,0] = x[\texttt{++vars}];$

$\texttt{Xp}[1,0] = x[\texttt{++vars}];$

```
X[1,1] = x[++vars];
Xp[1,1] = x[++vars];

Y[0,0,0] = x[++vars];
Yp[0,0,0] = x[++vars];
Y[0,0,1] = x[++vars];
Yp[0,0,1] = x[++vars];
Y[0,1,0] = x[++vars];
Yp[0,1,0] = x[++vars];
Y[0,1,1] = x[++vars];
Yp[0,1,1] = x[++vars];
Y[1,0,0] = x[++vars];
Yp[1,0,0] = x[++vars];
Y[1,0,1] = x[++vars];
Yp[1,0,1] = x[++vars];
Y[1,1,0] = x[++vars];
Yp[1,1,0] = x[++vars];
Y[1,1,1] = x[++vars];
Yp[1,1,1] = x[++vars];

Z[0,0] = x[++vars];
Zp[0,0] = x[++vars];
Z[0,1] = x[++vars];
Zp[0,1] = x[++vars];
Z[1,0] = x[++vars];
Zp[1,0] = x[++vars];
Z[1,1] = x[++vars];
```

```
Zp[1,1] = x[++vars];

L[0,0] = x[++vars];
Lp[0,0] = x[++vars];
L[0,1] = x[++vars];
Lp[0,1] = x[++vars];
L[1,0] = x[++vars];
Lp[1,0] = x[++vars];
L[1,1] = x[++vars];
Lp[1,1] = x[++vars];

R[0,0] = x[++vars];
Rp[0,0] = x[++vars];
R[0,1] = x[++vars];
Rp[0,1] = x[++vars];
R[1,0] = x[++vars];
Rp[1,0] = x[++vars];
R[1,1] = x[++vars];
Rp[1,1] = x[++vars];
vars
```

Bulk Solutions

```
BeqsX = {};
Do[{s1, s2, s3} = IntegerDigits[i,2,3];
{AppendTo[BeqsX, X[s1, χ[s1, s2, s3]]Xp[χ[s1, s2, s3], s3]−
    Xp[s1, s2]X[s2, s3]]}, {i,0,7}]
```

```
BulkSolutionX = Solve[BeqsX==0, Table[x[i], {i,1, vars}]]


BeqsZ = {};
Do[{s1, s2, s3} = IntegerDigits[i,2,3];
{AppendTo[BeqsZ, Z[s1, χ[s1, s2, s3]]Zp[χ[s1, s2, s3], s3] - Zp[s1, s2]Z[s2, s3]]}, {i,0,7}]


BulkSolutionZ = Solve[BeqsZ==0, Table[x[i], {i,1, vars}]]


X[0,0] = 1;
X[1,1] = 1;
X[0,1] = ξ; (*where ξ = x[7]x[8]/x[4], ω = x[7]x[8]/x[3]*)
X[1,0] = ω;


Z[0,0] = 1;
Z[1,1] = 1;
Z[0,1] = ζ; (*where ζ = x[31]x[32]/x[28], ω = x[31]x[32]/x[27]*)
Z[1,0] = η;


Xp[0,0] = 1;
Xp[1,1] = 1;
Xp[0,1] = 1/ω;
Xp[1,0] = 1/ξ;


Zp[0,0] = 1;
Zp[1,1] = 1;
Zp[0,1] = 1/η;
Zp[1,0] = 1/ζ;
```

Boundary Solutions

```
eqsL = {};
eqsR = {};
Do[{s1, s2, s3} = IntegerDigits[i,2,3];
{AppendTo[eqsL, Sum[PL[[2s1 + s2 + 1,2t1 + t2 + 1]]Lp[t1, t2], {t1,0,1}, {t2,0,1}]−
    ΛL * L[s1, s2]],
AppendTo[eqsR, Sum[PR[[2s1 + s2 + 1,2t1 + t2 + 1]]R[t1, t2], {t1,0,1}, {t2,0,1}]−
    ΛR * Rp[s1, s2]],
AppendTo[eqsL, L[s1, χ[s1, s2, s3]]Xp[χ[s1, s2, s3], s3] − Lp[s1, s2]X[s2, s3]],
AppendTo[eqsR, Z[s1, χ[s1, s2, s3]]Rp[χ[s1, s2, s3], s3] − Zp[s1, s2]R[s2, s3]]
}, {i,0,7}]

MatrixForm[equationsL = FullSimplify[Union[eqsL]]]

MatrixForm[equationsR = FullSimplify[Union[eqsR]]]

BulkSolutionX = Solve[equationsL==0, Table[x[i], {i,1, vars}] ~ Join ~ {ΛL, ξ, ω}]

X[0,0]/.BulkSolutionX[[3]]
X[0,1]/.BulkSolutionX[[3]]
X[1,0]/.BulkSolutionX[[3]]
X[1,1]/.BulkSolutionX[[3]]
Xp[0,0]/.BulkSolutionX[[3]]
Xp[0,1]/.BulkSolutionX[[3]]
Xp[1,0]/.BulkSolutionX[[3]]
Xp[1,1]/.BulkSolutionX[[3]]
L[0,0]/.BulkSolutionX[[3]]
L[0,1]/.BulkSolutionX[[3]]
L[1,0]/.BulkSolutionX[[3]]
```

L[1,1]/.BulkSolutionX[[3]]

Lp[0,0]/.BulkSolutionX[[3]]

Lp[0,1]/.BulkSolutionX[[3]]

Lp[1,0]/.BulkSolutionX[[3]]

Lp[1,1]/.BulkSolutionX[[3]]

1

$\frac{\beta^2 x[36]}{\alpha^2 x[34]}$

$\frac{x[34]}{x[36]}$

1

1

$\frac{x[36]}{x[34]}$

$\frac{\alpha^2 x[34]}{\beta^2 x[36]}$

1

$x[34]$

$\frac{\beta^2 x[36]}{\alpha^2}$

$\frac{x[34]-\alpha x[34]}{\alpha}$

$\frac{\beta x[36]-\beta^2 x[36]}{\alpha^2}$

$x[34]$

$x[36]$

$\frac{x[34]-\beta x[34]}{\beta}$

$\frac{x[36]-\alpha x[36]}{\alpha}$

```
BulkSolutionZ = Solve[equationsR==0, Table[x[i], {i,1, vars}]] ~ Join ~ {ΛR, ζ, η}]
```

```
Z[0,0]/.BulkSolutionZ[[-1]]
```

```
Z[0,1]/.BulkSolutionZ[[-1]]
```

```
Z[1,0]/.BulkSolutionZ[[-1]]
```

```
Z[1,1]/.BulkSolutionZ[[-1]]
```

```
Zp[0,0]/.BulkSolutionZ[[-1]]
```

```
Zp[0,1]/.BulkSolutionZ[[-1]]
```

```
Zp[1,0]/.BulkSolutionZ[[-1]]
```

```
Zp[1,1]/.BulkSolutionZ[[-1]]
```

```
R[0,0]/.BulkSolutionZ[[-1]]
```

```
R[0,1]/.BulkSolutionZ[[-1]]
```

```
R[1,0]/.BulkSolutionZ[[-1]]
```

```
R[1,1]/.BulkSolutionZ[[-1]]
```

```
Rp[0,0]/.BulkSolutionZ[[-1]]
```

```
Rp[0,1]/.BulkSolutionZ[[-1]]
```

```
Rp[1,0]/.BulkSolutionZ[[-1]]
```

```
Rp[1,1]/.BulkSolutionZ[[-1]]
```

$$1$$

$$\frac{\gamma^2}{\delta^2 \eta}$$

$$\eta$$

$$1$$

$$1$$

$$\frac{1}{\eta}$$

$$\frac{\delta^2 \eta}{\gamma^2}$$

1

$\frac{\gamma^2 x[46]}{\delta^2 \eta}$

$-\frac{\gamma^2(-1+\delta)x[46]}{\delta^3 \eta}$

$\frac{\gamma^2 x[46]}{\delta^2}$

$\frac{\gamma x[46]-\gamma^2 x[46]}{\delta^2}$

$\frac{\gamma^2 x[46]}{\delta^2 \eta}$

$\frac{\gamma x[46]-\gamma^2 x[46]}{\delta^2 \eta}$

$x[46]$

$\frac{x[46]-\delta x[46]}{\delta}$

Impurity Solutions

Impurity Equations and Solutions

```
eqsI = {};
Do[{s1, s2, s3, s4, s5} = IntegerDigits[i,2,5];
AppendTo[eqsI, X[s1, χ[s1, s2, s3]]Yp[χ[s1, s2, s3], s3, χ[s3, s4, s5]]Zp[χ[s3, s4, s5], s5]−
    Xp[s1, s2]Y[s2, s3, s4]Z[s4, s5]],
{i,0,31}]
Do[{s1, s2, s3, s4, s5} = IntegerDigits[i,2,5];
AppendTo[eqsI, Y[s1, ϕ[s1, s2, s3], s3] − ΛI * Yp[s1, s2, s3]],
{i,0,31}]

MatrixForm[equationsI = FullSimplify[eqsI/.BulkSolutionZ[[−1]]
    /.BulkSolutionX[[3]]]]
```

```
SolutionI = Solve[equationsI==0, Table[x[i], {i,1, vars}] ~ Join ~ {ΛI}]
```

```
Y[0,0,0]/.SolutionI[[1]]
```

```
Y[0,0,1]/.SolutionI[[1]]
```

```
Y[0,1,0]/.SolutionI[[1]]
```

```
Y[0,1,1]/.SolutionI[[1]]
```

```
Y[1,0,0]/.SolutionI[[1]]
```

```
Y[1,0,1]/.SolutionI[[1]]
```

```
Y[1,1,0]/.SolutionI[[1]]
```

```
Y[1,1,1]/.SolutionI[[1]]
```

$$0$$

$$0$$

$$\frac{\beta^2\delta^2\eta x[24]x[36]}{\alpha^2\gamma^2 x[34]}$$

$$0$$

$$0$$

$$0$$

$$0$$

$$-\frac{\beta\delta x[24]}{\alpha\gamma}$$

```
Yp[0,0,0]/.SolutionI[[1]]
```

```
Yp[0,0,1]/.SolutionI[[1]]
```

```
Yp[0,1,0]/.SolutionI[[1]]
```

```
Yp[0,1,1]/.SolutionI[[1]]
```

```
Yp[1,0,0]/.SolutionI[[1]]
```

```
Yp[1,0,1]/.SolutionI[[1]]
```

Yp[1,1,0]/.SolutionI[[1]]

Yp[1,1,1]/.SolutionI[[1]]

0

0

$-\frac{\beta\delta\eta x[24]x[36]}{\alpha\gamma x[34]}$

0

0

0

0

$x[24]$

Y[0,0,0]/.SolutionI[[2]]

Y[0,0,1]/.SolutionI[[2]]

Y[0,1,0]/.SolutionI[[2]]

Y[0,1,1]/.SolutionI[[2]]

Y[1,0,0]/.SolutionI[[2]]

Y[1,0,1]/.SolutionI[[2]]

Y[1,1,0]/.SolutionI[[2]]

Y[1,1,1]/.SolutionI[[2]]

0

0

$\frac{\beta^2\delta^2\eta x[24]x[36]}{\alpha^2\gamma^2 x[34]}$

0

0

0

0

$\frac{\beta\delta x[24]}{\alpha\gamma}$

Yp[0,0,0]/.SolutionI[[2]]

Yp[0,0,1]/.SolutionI[[2]]

Yp[0,1,0]/.SolutionI[[2]]

Yp[0,1,1]/.SolutionI[[2]]

Yp[1,0,0]/.SolutionI[[2]]

Yp[1,0,1]/.SolutionI[[2]]

Yp[1,1,0]/.SolutionI[[2]]

Yp[1,1,1]/.SolutionI[[2]]

0

0

$\frac{\beta\delta\eta x[24]x[36]}{\alpha\gamma x[34]}$

0

0

0

0

$x[24]$

Bibliography

[1] E. W. Weisstein, "Cellular automaton." From *MathWorld*-A Wolfram Resource. https://mathworld.wolfram.com/CellularAutomaton.html.

[2] C. Wetterich, "Cellular automaton for spinor gravity in four dimensions," *Journal of Physics: Conference Series*, vol. 2533, p. 012016, jun 2023.

[3] M. A. Smith, *Cellular automata methods in mathematical physics*. Phd thesis, Massachusetts Institute of Technology, Dept. of Physics, 1994.

[4] T. Ostoma and M. Trushyk, "Special relativity derived from cellular automata theory: The origin of the universal speed limit," 1998.

[5] T. Ostoma and M. Trushyk, "Special relativity theory and cellular automata: Light as a cellular automata process," *arXiv: General Physics*, 1999.

[6] J. A. Somers, "Direct simulation of fluid flow with cellular automata and the lattice-boltzmann equation," *Applied Scientific Research*, vol. 51, pp. 127–133, Jun 1993.

[7] G. 't Hooft, *The Cellular Automaton Interpretation of Quantum Mechanics*. Springer Cham, 2016.

[8] M. Klaus and C. Leon, *The Universe as Automaton*. Springer Berlin, Heidelberg, 2011.

[9] B. Buča, K. Klobas, and T. Prosen, "Rule 54: exactly solvable model of nonequilibrium statistical mechanics," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2021, p. 074001, jul 2021.

[10] T. Prosen and B. Buča, "Exact matrix product decay modes of a boundary driven cellular automaton," *Journal of Physics A: Mathematical and Theoretical*, vol. 50, p. 395002, sep 2017.

[11] S. Wolfram, "Cellular automata as models of complexity," *Nature*, vol. 311, pp. 419–424, 1984.

[12] S. Wolfram, "Statistical mechanics of cellular automata," *Rev. Mod. Phys.*, vol. 55, pp. 601–644, Jul 1983.

[13] Y. Yu, Y. Li, F. Lin, and W. Yan, "A multi-grid cellular automaton model for simulating dendrite growth and its application in additive manufacturing," *Additive Manufacturing*, vol. 47, p. 102284, 2021.

[14] H. Yin and S. D. Felicelli, "A cellular automaton model for dendrite growth in magnesium alloy az91," *Modelling and Simulation in Materials Science and Engineering*, vol. 17, p. 075011, aug 2009.

[15] W. L. Freedman and B. F. Madore, "Time evolution of disk galaxies undergoing stochastic self-propagating star formation," *Astrophysical Journal*, vol. 265, pp. 140–147, Feb. 1983.

[16] B. F. Madore and W. L. Freedman, "Computer simulations of the belousov-zhabotinsky reaction," *Science*, vol. 222, pp. 615–616, Nov. 1983.

[17] C. Ramos, F. Carapau, and P. Correia, *Cellular Automata Describing Non-equilibrium Fluids with Non-mixing Substances*, pp. 229–245. Cham: Springer International Publishing, 2022.

[18] G. Y. Vichniac, "Simulating physics with cellular automata," *Physica D: Nonlinear Phenomena*, vol. 10, no. 1, pp. 96–116, 1984.

[19] J. K. Park, K. Steiglitz, and W. P. Thurston, "Soliton-like behavior in automata," *Physica D: Nonlinear Phenomena*, vol. 19, no. 3, pp. 423–432, 1986.

[20] A. Bobenko, M. Bordemann, C. Gunn, and U. Pinkall, "On two integrable cellular automata," *Communications in Mathematical Physics*, vol. 158, pp. 127–134, Nov 1993.

[21] M. J. Ablowitz, J. M. Keiser, and L. A. Takhtajan, "Class of stable multistate time-reversible cellular automata with rich particle content," *Phys. Rev. A*, vol. 44, pp. 6909–6912, Nov 1991.

[22] A. Scott, F. Chu, and D. McLaughlin, "The soliton: A new concept in applied science," *Proceedings of the IEEE*, vol. 61, no. 10, pp. 1443–1483, 1973.
[23] T. Prosen and C. Mejía-Monasterio, "Integrability of a deterministic cellular automaton driven by stochastic boundaries," *Journal of Physics A: Mathematical and Theoretical*, vol. 49, p. 185003, apr 2016.
[24] B. Buča, "Unified theory of local quantum many-body dynamics: Eigenoperator thermalization theorems," *Phys. Rev. X*, vol. 13, p. 031013, Aug 2023.
[25] D. Bluvstein, A. Omran, H. Levine, A. Keesling, G. Semeghini, S. Ebadi, T. T. Wang, A. A. Michailidis, N. Maskara, W. W. Ho, S. Choi, M. Serbyn, M. Greiner, V. Vuletić, and M. D. Lukin, "Controlling quantum many-body dynamics in driven rydberg atom arrays," *Science*, vol. 371, no. 6536, pp. 1355–1359, 2021.
[26] N. Maskara, A. A. Michailidis, W. W. Ho, D. Bluvstein, S. Choi, M. D. Lukin, and M. Serbyn, "Discrete time-crystalline order enabled by quantum many-body scars: Entanglement steering via periodic driving," *Phys. Rev. Lett.*, vol. 127, p. 090602, Aug 2021.
[27] P. Frey and S. Rachel, "Realization of a discrete time crystal on 57 qubits of a quantum computer," *Science Advances*, vol. 8, no. 9, p. eabm7652, 2022.
[28] G. Vidal, "Efficient classical simulation of slightly entangled quantum computations," *Phys. Rev. Lett.*, vol. 91, p. 147902, Oct 2003.
[29] K. Temme and F. Verstraete, "Stochastic matrix product states," *Phys. Rev. Lett.*, vol. 104, p. 210502, May 2010.
[30] T. H. Johnson, S. R. Clark, and D. Jaksch, "Dynamical simulations of classical stochastic systems using matrix product states," *Phys. Rev. E*, vol. 82, p. 036702, Sep 2010.
[31] B. Buča, J. P. Garrahan, T. Prosen, and M. Vanicat, "Exact large deviation statistics and trajectory phase transition of a deterministic boundary driven cellular automaton," *Phys. Rev. E*, vol. 100, p. 020103, Aug 2019.
[32] F. C. Alcaraz and M. J. Lazo, "The bethe ansatz as a matrix product ansatz," *Journal of Physics A: Mathematical and General*, vol. 37, p. L1, dec 2003.
[33] F. C. Alcaraz and M. J. Lazo, "Exact solutions of exactly integrable quantum chains by a matrix product ansatz," *Journal of Physics A: Mathematical and General*, vol. 37, p. 4149, mar 2004.
[34] M. O. D. Alotaibi and L. D. Carr, "Scattering of a dark-bright soliton by an impurity," *Journal of Physics B: Atomic, Molecular and Optical Physics*, vol. 52, p. 165301, jul 2019.
[35] C. Jia and Z. Liang, "Interaction between an impurity and nonlinear excitations in a polariton condensate," *Entropy*, vol. 24, no. 12, 2022.
[36] K. Forinash, M. Peyrard, and B. Malomed, "Interaction of discrete breathers with impurity modes," *Phys. Rev. E*, vol. 49, pp. 3400–3411, Apr 1994.