Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Complex Social Systems:
Modeling Agents, Learning, and Games HS2021

Project Report

Virtual Zombie Pandemic on Lattice: Critical Phenomenon and Reinforcement Learning

Noureddine Gueddach
Chang Sun
Yifeng Wang
Rong Zhang

Zurich
Dec 2021

Agreement for free-download

We hereby agree to make our source code for this project freely available for download from the web pages of the SOMS chair. Furthermore, we assure that all source code is written by ourselves and is not violating any copyright restrictions.

Noureddine Gueddach	Yifeng Wang
Chang Sun	Rong Zhang

Contents

1 Abstract

This report studies a virtual zombie pandemic using the SZR (survivor, zombie, removed) model. We first solved the analytical solution for the SZR model and discussed the computer simulation method. The system behavior and critical points for the SZR model and its derivatives are discussed in details. Finally, we use a reinforcement learning algorithm to train the agents to move in order to depict the behavior of zombies and survivors dynamically.

2 Individual contributions

Gueddach Noureddine studied the critical point on the hexagonal lattice, before working on the mitigation strategies using reinforcement learning.

Chang Sun proposed the Modified SZR model and implemented (Modified) SZR model with the "contact tracing" algorithm. He also performed the critical point - occupancy study for the MSZR / SZR model on a square lattice.

Yifeng Wang is the main contributor for drafting and editing this written report. His other contribution is showing the analytical solutions and verifying the code of other exploratory studies.

Rong Zhang implemented the queue-bond algorithm and studied the relation between critical point of the SZR model and the sparsity with it.

3 Introduction and Motivations

Though it has never been confirmed (to the best of our knowledge), zombie pandemic outbreak has proven themselves to be an interesting field to investigate in recent years. As a familiar topic to the public, many have investigated such virtual pandemic and modeled it [1] [2] [3] [4]. While some focus on the strategies and probabilities of survival [5], or variables [4] in the zombie pandemic itself, it is shown that the zombie models can be made in parallel to other phenomenons (e.g., rumors [6]). Besides, though the pandemic itself is virtual, it may grant us valuable insights regarding the real pandemics [7].

In our exploratory study, we investigated the SZR model introduced in [8] and its derivatives. The SZR model introduced in [8] is similar to the SIR model, except for the distinct behavior of the agents. Due to this difference, the SZR model has advantages in describing other scenarios that depart from the conventional disease (i.e., bacteria in cells, rumor spread in villages, or even zombie pandemic outbreaks). We take zombie pandemic outbreaks as a general scenario to study the SZR model in detail. We start with a simple square-shape pixel; then, we advance to hex pixel.

We will cover the critical behavior of the model and explore its correlation with α. We will also discuss the mitigation strategies for agents.

4 Description of the Model

We should give a quick recap on SIR model first. SIR model is governed by three differential equations which states the time evolution of a given variable (i.e. S, Z, and R). S stands for the stock of susceptible population, I is the stock of infected, and R is the stock of removed population.

$$\dot{S} = -\frac{\beta I S}{N}, \tag{1}$$

$$\dot{I} = \frac{\beta I S}{N} - \gamma I, \tag{2}$$

$$\dot{R} = \gamma I. \tag{3}$$

Since N is just the normalization constant, exploring other parameters inside the differential equations is worthwhile. Unsurprisingly, the ratio of two parameters $R_0 = \frac{\beta}{\gamma}$ needs special attention, as it can be used to derive the expected number of new infections, which is called the basic reproduction ratio. With the simple SIR model as a foundation, numerous models have been derived to describe the behavior of various systems. The most phenomenological use of instance is an epidemic disease and prey-predator biological system. Unfortunately, the SIR model is computationally intensive due to the disadvantage of not admitting to closed-form analytical solutions, especially on a large grid.

To address the shortcoming of the SIR model, we introduce the SZR model, which is abbreviated for "survivors", "zombie", and "removed". The SZR model is inspired by the study of zombie breakout in such a scenario that zombies will "bite" survivors and survivors will "kill" zombies. Fortunately, our SZR model can be solved analytically, giving more opportunities to explore its applications. The key distinction from the SIR model is that the state turning rate (from S to Z or Z to R) is density-dependent rather than frequency-dependent. At a given site, the turning rate is solely dependent on the number of zombies and survivors interacting. Another interesting fact is that the R in the SZR model is non-reversible because dead zombies cannot be brought back to life either into a human or a zombie form.

$$\dot{S} = -\beta S Z, \tag{4}$$

$$\dot{Z} = (\beta - \kappa) S Z, \tag{5}$$

$$\dot{R} = \kappa S Z. \tag{6}$$

Besides the zombie breakout scenario, the SZR model has many realistic applications, such as virus attacks on human cells, forest fire progression, and rumor spreading in crowds.

4.1 Analytical solution

We will discuss the analytical solution of our SZR model; with these, we can calculate and predict the final states given the initial conditions without going into a stochastic process. First, the equation can be cast into a non-dimensional form, by replacing the dimensionless time parameter $\tau = t\beta N$ and the dimensionless so-called virulence $\alpha = \kappa/\beta$ [9].

$$\frac{\dot{S}}{d\tau} = -\frac{SZ}{N}, \tag{7}$$

$$\frac{\dot{Z}}{d\tau} = (1-\alpha)\frac{SZ}{N}, \tag{8}$$

$$\frac{\dot{R}}{d\tau} = \alpha\frac{SZ}{N}. \tag{9}$$

Then, with the assumption that the initial conditions are $R(0) = 0$, $Z_0 = Z(0)$, and $S_0 = S(0)$. We have the analytical solution as

$$P \equiv Z_0 + (1-\alpha)S_0, \tag{10}$$

$$\mu \equiv \frac{S_0}{Z_0}(1-\alpha) = \frac{P}{Z_0} - 1, \tag{11}$$

$$f(\tau) \equiv \frac{P\mu}{e^{\tau P/N} + \mu}, \tag{12}$$

$$Z(\tau) = P - f(\tau), \tag{13}$$

$$S(\tau) = \frac{f(\tau)}{1-\alpha}. \tag{14}$$

To illustrate more, the sign of function P indicates whether human will win or not. The final state has the property that

$$Z_\infty = Z_0 + (1-\alpha)S_0, \text{(if human wins)} \tag{15}$$

$$S_\infty = S_0 - \frac{Z_0}{\alpha - 1}\text{(if zombie wins)} \tag{16}$$

5 Implementation

5.1 Stochastic process and Gillespie algorithm

In last sections, toy models are on the basis of continuous simulation, so the number of agent in any states can has decimals. While continuous simulation is computational efficient and accurate when all S and Z states are comparable, the stochastic simulation needs to be implemented for better emulate the entire process, especially at the beginning of the process, where the first zombie has the possibility of being eliminated, corresponding to the no-breakout scenario, or in the end, few survivors can hold their lines with removed states in between zombies. All of such exceptional scenarios have to be included in the progression of the system by SZR model. Thus, our codes are based on discrete stochastic simulations.

5.2 Queue-bond

Besides, we can further explore the system by localizing SZR states to each 2D pixel. Only one type of state can occupy a site, and the pixel can interact with neighbor pixels according to the SZR differential equations.

To be more efficiently simulate the evolution of the "killing" and "biting" processes, rather than calculating each individual "fighting" across the edge of the pixel. The key is to record only the bond between S and Z states which is the connection between local survivors and zombies. At each time step, we randomly choose a bond with a human site being infected with the probability of $1/(1+\alpha)$ and a zombie site being killed with the probability of $\alpha/(1+\alpha)$.

5.3 Modified SZR Model

A modified version derived from the original SZR model is proposed for on-lattice simulation as a more "realistic" situation to include the on site density-dependent for killing and biting rate. The differential equations of the modified model are expressed in 19, referred to as the Modified SZR (MSZR). This model is considered a more realistic case as zombies do not "bite faster" because there are more humans around it, contrasting to the original SZR model. Also, in contrast to [8], we allow multiple people on the same grid point in a lattice. The number of people on one grid point is denoted as occupancy. Each grid point is connected to 5 grid points in the square lattice: its four neighbors and itself. When the occupancy is precisely 1, this is equivalent to the square model proposed in [8]. Consequently, the number of vertices for the modified model is $4n + (n-1) = (5n-1)$ ($4n$ neighbors in adjacent grid points and $(n-1)$ neighbors on the same grid point), where n is the occupancy.

$$\dot{S} = -\beta Z, \tag{17}$$

$$\dot{Z} = \beta Z - \kappa S, \tag{18}$$

$$\dot{R} = \kappa S. \tag{19}$$

5.3.1 Contact Tracing

The queue-bond algorithm is no longer valid with the modified model: the probability of having state transition on each bond is not equal. Hence, instead, we used a contact tracking algorithm, which keeps track of all grid points having none-zero Z, with non-zero S in its neighborhood, where S in such grids is denoted S^*_{ij}. The vice versa is performed for Z. For each new step. There is a probability of $K/(K+B)$ that a zombie is killed, with $K = \kappa \sum S^*_{ij}$ and $B = \beta \sum Z^*_{ij}$. Then, a weighted sample, with weight being S^*_{ij} or Z^*_{ij}, is performed to choose which grid initiates the action (kill or bite), and a random neighbor of it will receive the action. The contact list of the grids having state modified and all its neighbor's state in the contact tracing lists are updated accordingly.

The algorithm described is checked on the vanilla SZR model as a safety measure. With replacing the probability of kill to $\kappa/(\kappa+\beta)$ and weight being $S^*_{ij} \cdot Z_{<ij>}$ or $Z^*_{ij} \cdot S_{<ij>}$, where $Z_{<ij>}$ or $S_{<ij>}$ denotes the total number of Z or S, respectively, in the neighborhood of grid point (i, j), this algorithm produces a similar critical point for the SZR model with an occupancy of 1. As the efficiency of this algorithm is one order slow than the one proposed in [8], we used a smaller lattice of 256×256 size for critical point search.

6 Critical Point

According to [8], when the inverse virulence α is small, the zombie outbreak will cover the whole lattice. When α is large, humans will be able to contain the zombie outbreak before it reaches the boundary of the lattice. At the critical point α_c, the zombie outbreak reaches the boundary, but form a self-similar pattern, or fractal. Although the random fractal is not transformed to itself by scaling like the Sierpinski carpet, the probability distribution of the size of the cluster should satisfy $P(s) = fP(s/B)$, where B denotes the scaling factor.

This phenomenon is closely related to the renormalization group, which has also been applied to theoretical condensed matter physics and high-energy physics. In the

context of condensed matter physics, for example, the renormalization group transformation is applied to deriving the relations between the critical exponents and the computing the critical point numerically of the 2D Ising model, which describes the transition from paramagnetism to ferromagnetism in many materials. By requiring the parition function to be equivalent before and after replacing blocks of spins on the Ising lattice by a single spin according to some predefined rules, the effective coupling between spins can be determined, so does the critical temperature. In the context of high-energy physics, this phenomenon is understood as a result of coarsifying the details of a theory. When theory goes to higher energy, the measurement resolution is increased, which implies subtle structure will appear. However, some sub-structures will be repeating themselves in the larger picture, just like snowflakes. In quantum field theory, the critical point is defined as the fix-point of the beta function at which the theory is scale-free. No matter how many higher-order expansions of correlation function we include, analogous to increasing scope resolution, the theory is still renormalizable. This is because the counter-terms precisely cancel the divergent terms at the critical point. Such scale-invariant property is why theorists shall be sensitive to whether the system admits a critical point. Analogously, in our case, the critical point is the α at which the SZR system has no natural scale. Pictorially, the pattern of survivors or zombies will be repetitive on small and large scales.

6.1 Critical Point Search

The search of critical points is based on the self-similarity of the percolation clusters at the critical point and the scaling behavior close to the critical point. Numerical experiments have support the assumption that the distribution of the cluster size n_s near the critical point is proportional to $s^{-\tau} f((\alpha - \alpha_c)s^{\sigma})$, where f is an analytic function. [10] The self-similarity of the percolation cluster at critical point requires $n_s \propto s^{-\tau}$. Therefore, after Taylor expansion, the probability that a cluster is greater than s in size is related to the model parameter, in this case, the inverse virulence α, by

$$P_{\geq s} \sim s^{2-\tau}(A + B(\alpha - \alpha_c)s^{\sigma}), \tag{20}$$

where τ and σ are critical exponents. An approximation to the critical point is found if the $s^{\tau-2}P_{\geq s}$ versus s^{σ} curve has zero slope within the statistical uncertainty. The critical point is determined by linear fitting of the slope $-\ \alpha$ curve. The slope of the $s^{\tau-2}P_{\geq s}$ is determined by linear fitting to the simulation data. As the Equation 20 is valid for large cluster and our simulation is on a finite lattice, we avoid the artifacts arising from the finite periodic lattice by fitting to data of clusters of moderate size.

[8] has shown that the percolation exponent τ for the vanilla SZR model is the same as that for 2D random percolation. The critical exponent τ for 2D random percolation describes the distribution of the size of clusters formed by randomly occupying a 2D lattice site, which also describes the distribution of the size of clusters formed by burned trees in the forest fire model. The study of 2D random percolation shows that [10] the critical exponents depends on the dimensionality of the lattice, but not the detail of the lattice, i.e. the critical exponents of 2D square or hexagonal lattices are identical, but different from that for 3D lattices, or Bethe lattices, etc. As the study of sparsity is an effecitve reduction of coordination number, we expect the critical exponents remain unchanged. In fact, we will see the SZR model reduces to the forest fire model at the critical sparsity, which further justifies our choice of 2D percolation exponent. Figure 1 shows a frugal replication of the result in [8]. For our study of the critical point of square lattice with low occupation number, Figure 2 shows a similar behavior as the vanilla SZR model, supporting our choice of the percolation exponents for the study of sparsity.

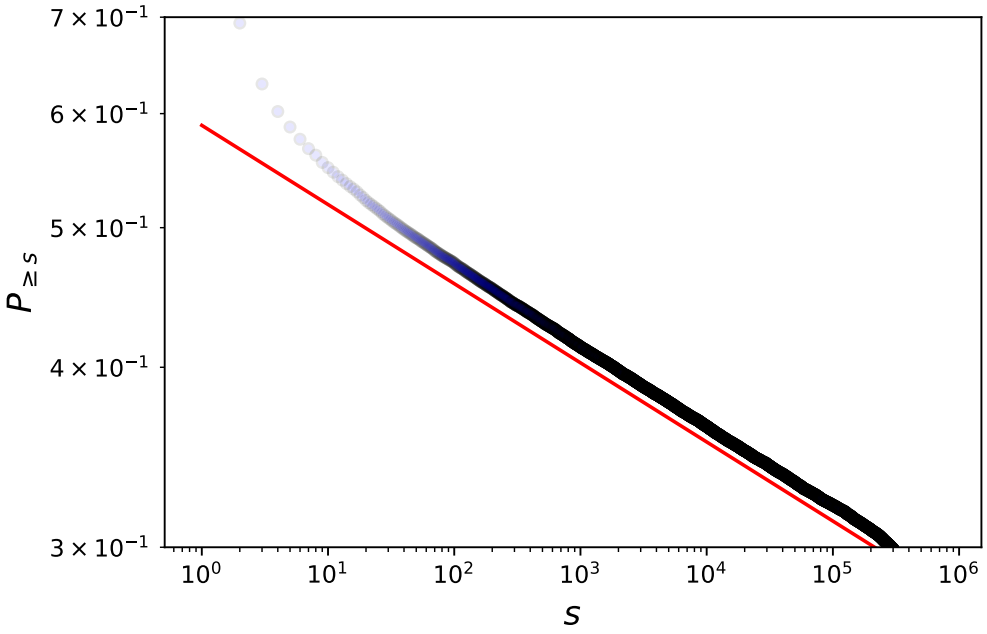


Figure 1: Red line: a line with slope $= 2 - \frac{187}{91}$. Blue dots: cumulative cluster size distribution of 50000 simulations on 1024×1024 square lattice of the vanilla SZR model at the critical point $\alpha_c = 0.43734613$.[8]

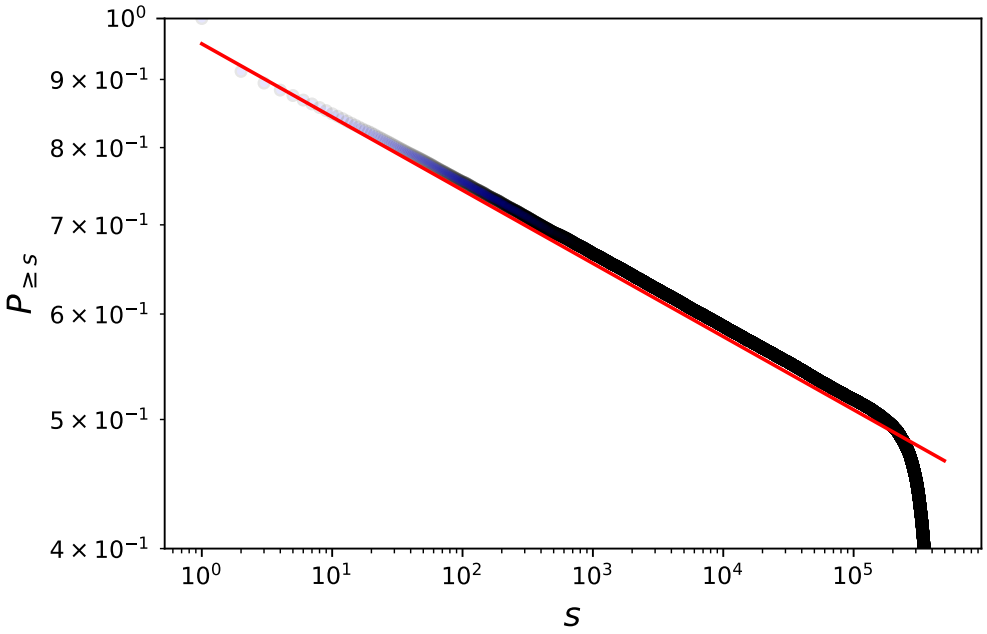
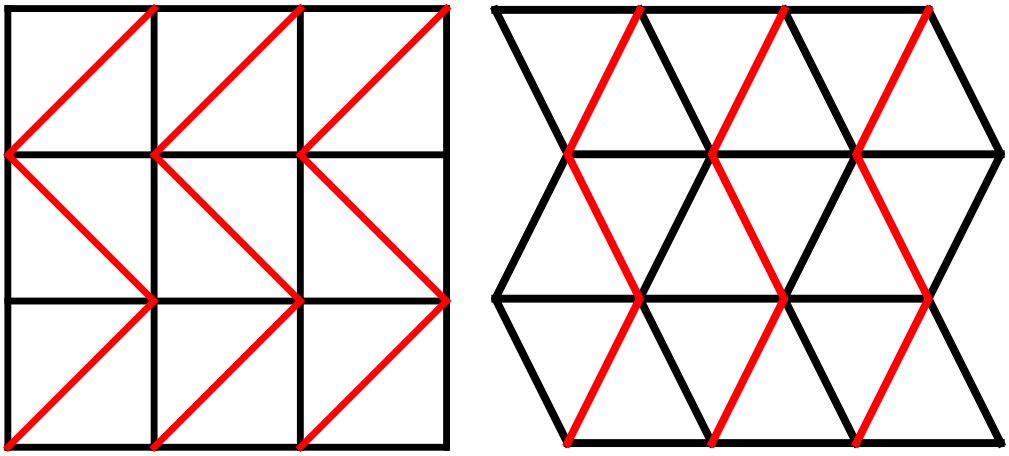


Figure 2: Red line: a line with slope $= 2 - \frac{187}{91}$. Blue dots: cumulative cluster size distribution of 50000 simulations on 1024×1024 square lattice of the SZR model with sparsity $= 0.35$ at the critical point $\alpha = 0.0805$.

In order to search for critical points of the low-occupancy lattices, bootstrapping is performed to determine the uncertainty of the slope, i.e., the linear fit is performed to samples obtained from resampling without replacement from the simulation. The standard deviation of 100 sets of samples from resampling is taken as the uncertainty of the slope estimate. We start from two initial α. We evaluate the slope and perform a linear fit to obtain a new critical point estimation. Then the slope of the new α is evaluated and added to the data points for the linear fit. Once the magnitude of the estimated slope is smaller than the uncertainty, we quadruple the sample size for the measurement of cluster size until the maximum allowed sample size is reached, which is set to be 64000. The simulations are performed on 1024×1024 square lattice. Since the accuracy of our critical point is sufficient for the phase diagrams, we did not perform a rigorous uncertainty quantification on the critical points. However, We such analysis can be performed by error propagation based on the linear fit.

The critical point for high-occupancy lattices are performed in a "quasi" binary-search way, based on the scaling postulate: One first set an upper bound α_h and a

lower bound α_l before the search. In a new iteration, α is initialized by $(\alpha_h + \alpha_l)/2$. The algorithm then simulates the same lattice with different random seeds for 1024 times, and a linear fit is performed with $x = s^\sigma$ and $y = s^{\tau-2}P_{\geq s}$, on where the curve is on a plateau. By the scaling postulate, this slope is 0 when $\alpha = \alpha_c$. Hence, when the slope is smaller than 0, $\alpha_h \leftarrow \alpha$, or $\alpha_l \leftarrow \alpha$ otherwise. When $|\text{slope}| < 3\sigma_{fit}$, the iteration breaks, and current α is deemed to be α_c. However, notice that this algorithm diverges from the conventional binary search in one way: when the slope changes in an unexpected way after updating one boundary, the last change in the boundary of the opposite direction will be undone. (e.g., if the slope decreases after decreasing α_h, the last increment of α_l be undone.) The resolution of α_c obtained this way is inferior to the one used for low-occupancy lattices, but is used based on performance considerations.



(a) Square lattice (b) Hexagonal lattice

Figure 3: Bond mapping from square lattice to hexagonal lattice

6.2 Results and Discussions on Critical Point Dependence

We extend the same methodology to the hexagonal lattice in order to verify that the latter belongs to the same universality class as the square lattice. To do so, we compute the new critical point α_c and examine it; we compute the probabilities of sites being in a cluster of size at least s (x-axis in Figure 7 b). The points forming a line on a log-log plot with the same slope (2 - τ) as the square lattice indicates that it follows the same power law and hence verifies our prediction ($\tau = 187/91$ is given by percolation theory). To actually compute the critical point, we follow the strategy described in section 5.2 and we find a value of $\alpha_{\text{c, hexagonal}} \approx 0.527825$ which

is higher than that of the square lattice ($\alpha_{\text{c, square}} = 0.43734613$). The method of embedding the hexagonal lattice in square lattice is shown in Figure 3, where the hexagonal lattice is shown on the right and the square lattice is shown on the left.

Some of the results for the critical point are shown in Figure 7: (a) shows s^{σ} plotted against $s^{\tau-2}P_{\geq s}$ ($\sigma = 36/91$ also given by percolation theory). A flat tangent to the graph indicates a scale invariant (clusters of different sizes are equally likely), which in turn indicates that we are at the critical point. Indeed, for instance, with $\alpha > \alpha_c$, we get a negative slope (and hence fewer large clusters, i.e., humans win) and vice-versa. As can be seen in (c), various α's become almost indistinguishable at the precision we operate. To get more precision, we use the graph in (d); notice that our critical point α_c has a slope of almost zero (slope k = 7.1565×10^{-7}). Finally, (e) shows the resulting lattice with the computed critical point after an outbreak.

Besides, studies on the critical point dependence on the sparsity / occupancy are done, and the results are presented in Figure 4 and Figure 5.

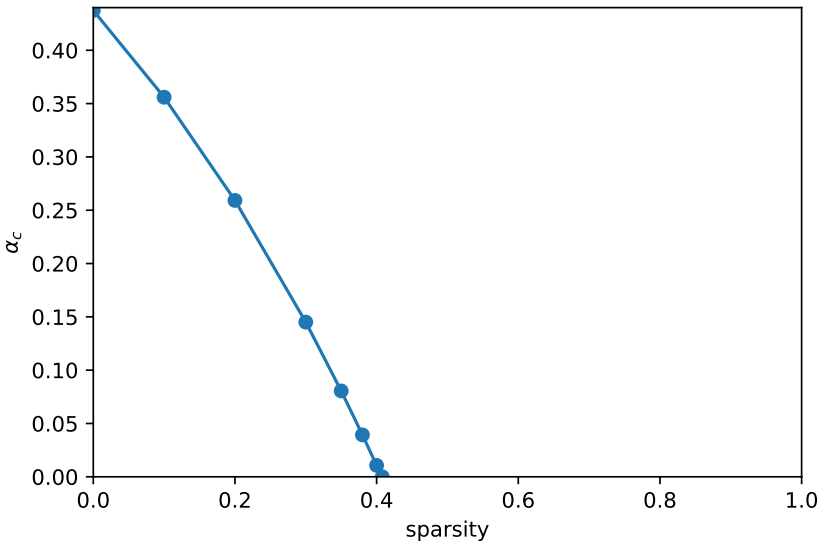


Figure 4: Critical point - Sparsity, SZR model

Sparsity is defined as the probability that a lattice site is empty. The empty state is effectively the same as the removed state, except that it is not considered to be a part of the infected cluster, unlike the removed state. In the hypothetical zombie

outbreak scenario, the empty sites play the role of randomly placed firewalls. The critical point decreases with increasing sparsity until it reaches zero at $p = 0.407$ [11], which is the percolation threshold of the forest fire model.

The forest fire model describes the scenario where each site on a lattice is occupied with trees at a probability. At the onset of the simulation, there is only one burning tree. In each time step, all the nearest neighbors occupied by trees of a burning tree catch fire, and all the burning trees burn out. In this way, at the end of the simulation, the forest fire will cover the whole connected cluster. The percolation threshold of the forest fire model is the density of the trees where an infinite cluster starts to appear on an infinite lattice.

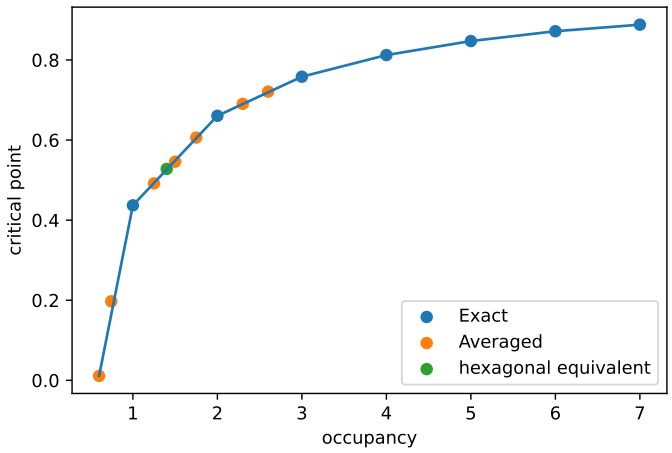


Figure 5: Critical Point - Occupancy, SZR model
An averaged occupancy n is obtained by letting the probability of each grid point having occupancy of $\lfloor n \rfloor$ be p, and the probability of having an occupancy of $\lceil n \rceil$ be $(1 - p)$, with $p = n - \lfloor n \rfloor$. The hexagonal equivalent point is evaluated by the degree of vertices (individuals): With an occupancy of $n | n \in \mathbb{N}_*$, the degree of vertices is $(5 \cdot n - 1)$. Hence, with the hexagonal model at an occupancy of 1, as all its vertices have a degree of 6, it is expected to be equivalent to a square model with an occupancy of 1.4. More details of this hexagonal model are described in section 6.2.

Though it looks that the Figure 5 critical point is nearly linear dependent on the occupancy/sparsity, the Figure 4 showed that the relationship is not exact. On the other hand, the trend of the critical point from Figure 5 suggests that it is saturating to a value < 1, which is compatible to the analytical solution without a lattice structure. This is reasoned from that when the occupancy is comparable or

exceeding to the lattice dimension, the system will be close to a "fully connected network", which is illustrated to be analytical solvable in Section 4.1. In contrast, as shown in Figure 6, the critical point seems to asymptotically converging to and upper bounded by 1/2. Interestingly, Figure 5 showed that the hexagonal model aligns well with the square model with the same number of connections for each individual. This suggests that the hexagonal model is likely be able to reduced to a square lattice with a heterogeneous occupancy at each grid point.

What is more interesting is that the MSZR seems to indeed fit into the same universality class: its behaviour around the found critical point – which is based on the assumption that it is of the same university class as the SZR model – is presented as fractals visually. Though the interaction mechanics are totally different, the critical exponent indeed seems to be more or less equal.

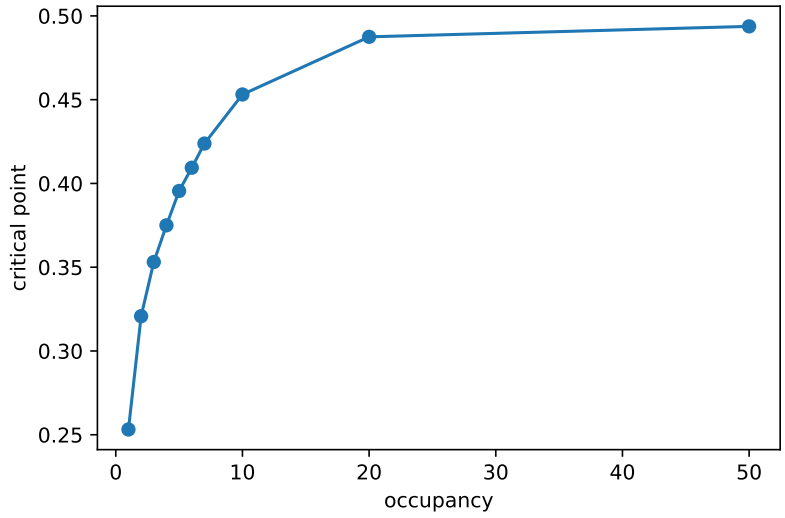


Figure 6: Critical Point - Occupancy, MSZR model

7 Learning mitigation strategies

We wanted to further explore ways to stop the spread of the epidemic by making the agents learn behaviors that help them survive. We do this on our MSZR model as it makes the bite-/kill-rate state-dependent. Namely, if a human is surrounded by other humans, he has higher chances of surviving.
To achieve this, we train a single agent using a model-free Reinforcement Learning algorithm: Deep Q-Network (DQN) [12]. The network is used to approximate the

Q-Function, which in turn induces the agent's policy.
Once training is finished, we use the same learning policy for all agents in the simulation. The zombies chase the nearest human within a perimeter or move randomly if the nearest neighbor is too far.

7.1 Reward functions

As can be seen in Figure 8 the rewards are highly stochastic. This is mainly due to two facts:

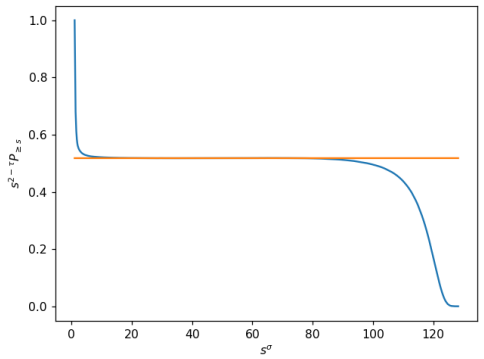
1. The intrinsic stochasticity of the model, arising from the high uncontrolled variance of outcomes: at a confrontation, both life (positive reward) and death (negative reward) are possible and are decided given a probability distribution depending on the number of allies the human has.

2. The noise introduced by our particular choice of reward function (referred to as Type-I reward). We have a large negative reward for death (**-20**), a positive reward for a kill (**2**), and a large positive reward for surviving the whole episode (**20**). This induces two natural strategies for increasing the expected reward: either running away or gambling and confronting zombies.

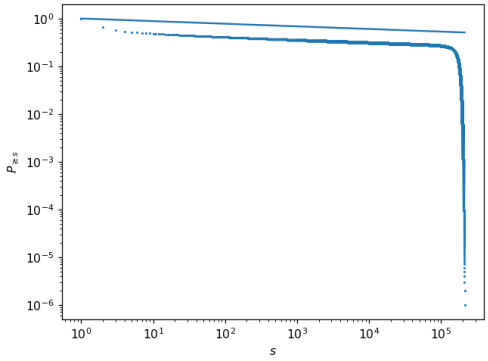
To mitigate this second point, we experiment with a modified reward function (referred to as Type-II reward) that consists in keeping a large negative reward for death (**-20**) and the positive kill reward (**2**) but removing the survival positive reward. This has two advantages:

- We now have only one strategy to increase the reward, namely confronting zombies, which reduces the variance.

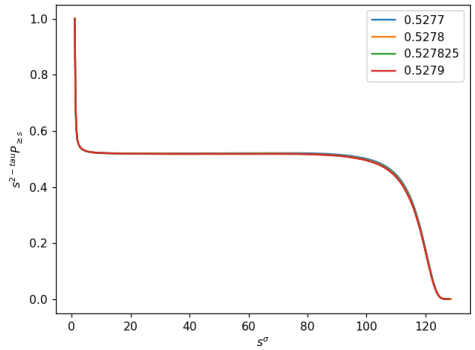
- From a 'realistic' point of view, it is a preferable scenario, as we typically want an epidemic to end as quickly as possible (minimizing deaths, economic impact) and a more aggressive strategy goes in this direction.



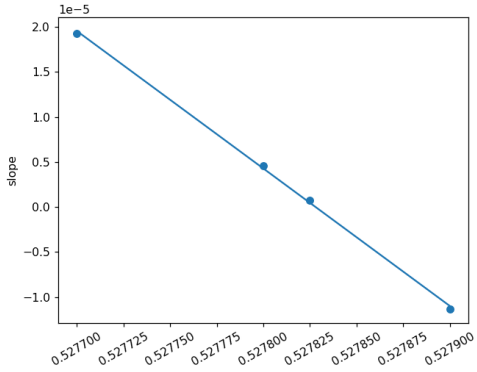
(a) Horizontal tangent at α_c



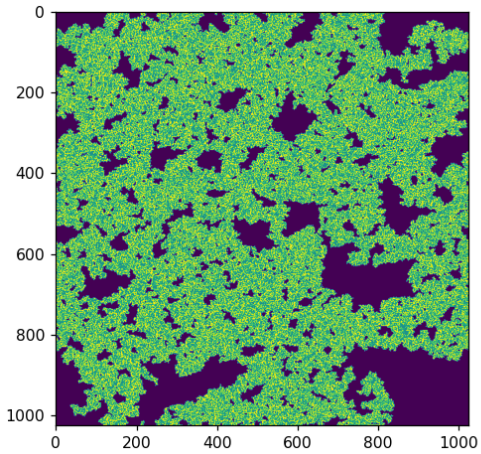
(b) Power law of the hexagonal lattice model



(c) Cluster sizes for α's near the critical point



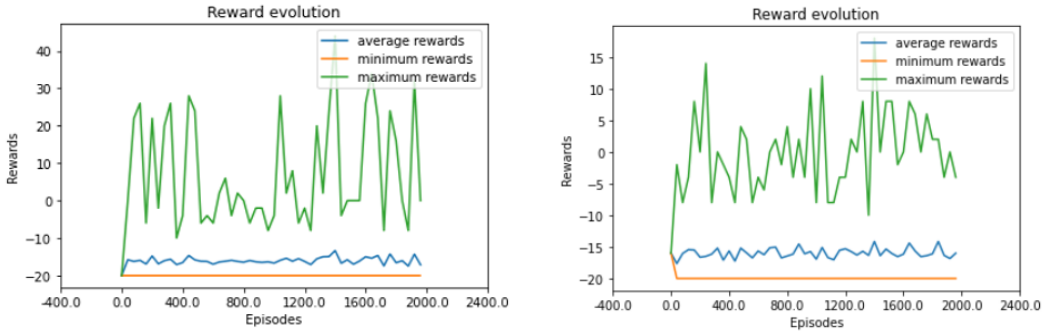
(d) Tangent slopes for α's near the critical point



(e) Outbreak in the hexagonal lattice at α_c

Figure 7: Hexagonal lattice critical point ($\alpha_c = 0.523825$)

In practice, we found that this latter strategy typically works better than the first one, as can be seen in the average number of human survivors with both strategies (Figure 9). Though the noise remains high, it tends to slightly decrease (at least for the maximum reward), as can be seen in Figure 8. A possible improvement would be to use a collaborative RL algorithm [13], using other different RL-algorithms, testing with a different number of layers, hyper-parameters, to both improve performance and reduce the reward noise. For this project, though, we limit ourselves to what we just presented. In what follows, we give more precise specifications of the algorithm.



(a) Type-I reward evolution

(b) Type-II reward evolution

Figure 8: Aggregated rewards per episode

```
Type-I reward:
        Mean:                138.99
        Standard deviation: 59.25
Type-II reward:
        Mean:                152.11
        Standard deviation: 54.00
```

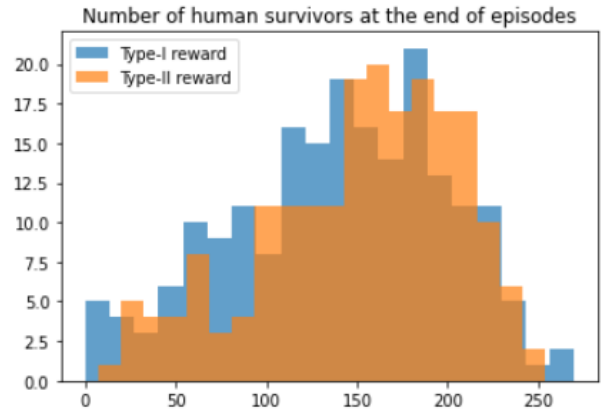


Figure 9: Number of survivors for each reward (over 200 simulations each)

7.2 The Environment

The environment is a square lattice of size (45×45). At initialization time, each cell has some probability of being initialized as either a human, a zombie, or empty. The initial population density (humans and zombies combined) is 0.2, of which 0.25 are zombies. The state size is a square around the human of size 19×19. We introduce a 'warm-up' time of 20 epochs before the zombies start moving for the humans to organize themselves.

The model is trained for 2000 episodes with a discount factor of $\gamma = 0.95$ and uses a $\epsilon-$greedy policy with ϵ decaying from 0.99 to 0.05 over 1200 episodes.

At each epoch, each agent performs an action (top, down, left, right, or stay put), followed by an action by each zombie. The reason for this order is to enforce 'thinking ahead' for the agents. Once all actions are performed, all contacts between agents and zombies are settled by the death of either party. Here are the parameters used:

$$\begin{aligned} b &= 1.35 \\ k &= 1 + (\#\text{ surrounding humans} \times 2) \\ \mathbb{P}[\text{survival}] &= \mathbb{P}\left[U \geq \frac{b}{b+k}\right],\ U \sim \text{Uniform}([0,1]) \end{aligned}$$

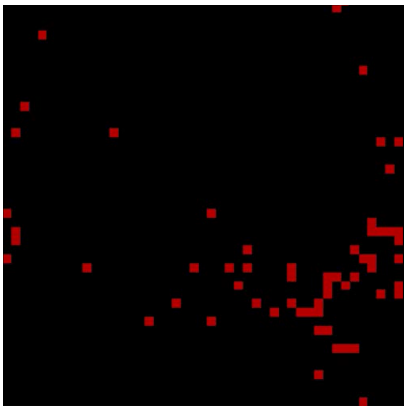
Where b is the bite rate, and k is the kill rate. Note that if a human has no surrounding allies, he has a higher chance of dying, whereas the chances of survival are greater with allies. We use a multiplicative factor of 2 for the number of surrounding humans to make the results more tangible. We use a factor of 4 during training to help the agents learn the policy.

7.3 The Deep Q-Network

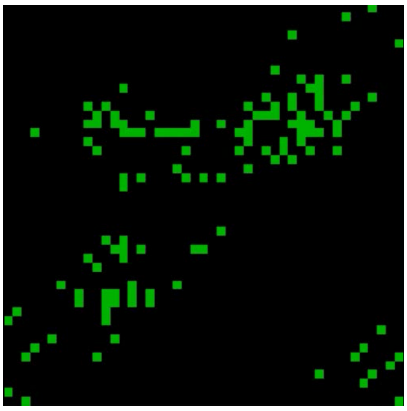
The DQN consists of 2 convolutional layers with batch normalization followed by two fully-connected layers with ReLU activation function. The network inputs a 2D state of the environment around the agent and outputs 5 Q-values, one for each possible action (move top, down, left, right, or stay put). The action with the highest value is selected.

7.4 The Deep Q-Learning Agent

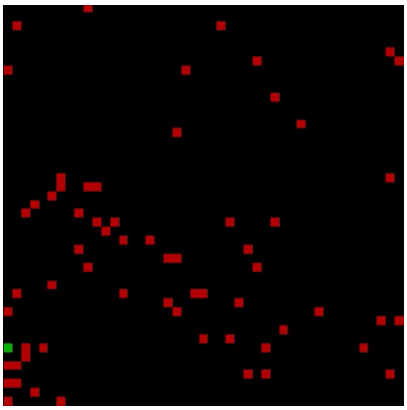
The agent uses 2 DQNs, a policy one and a target one. The policy DQN is used for selecting actions and gets optimized at each time step. The target DQN on the other hand is updated only every some fixed number of episodes and is used as a more stable error measure. Indeed, contrary to standard Q-Learning where only the Q-value of the state-action pair that we just played is changed, running back-prop on a neural network typically updates all actions, which makes the error measurement very noisy. Having a network that changes less frequently for evaluating the action (the target network) helps palliate this problem and yields a more robust approximator.
In order to improve the sample complexity, we store the transitions in a replay buffer. This allows us to bootstrap batch of samples, which helps break correlation between samples and, on the one hand, make better use of our experience by potentially reusing it multiple times.



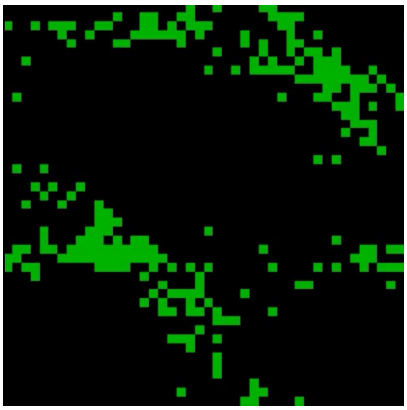
(a) Type-I reward - Final step before training



(b) Type-I reward - Final step after training



(c) Type-II reward - Final step before training



(d) Type-II reward - Final step after training

Figure 10: Training with different reward functions (zombies in red and humans in green)

7.5 Simulation Results and Discussion

It is of interest from an interpretability point of view to actually visualize an outbreak. In the simulations, we can see a clustering behavior of the humans that helps them increase their survival chances. Furthermore, as discussed earlier on, it seems that the second type of rewards we studied creates denser structures that are more efficient in maintaining a group alive. It would be interesting to develop other reward functions that drive the agents to optimal strategies more efficiently (by putting a penalty on movements for example), since in the current state, many agents seem to take sub-optimal/random actions. A possible interesting extension would be to assign some

of the humans significantly higher survival chances when confronted to a zombie (emulating police officers or armed forces) and introduce a negative reward per time step (to give an incentive to stop the epidemic as quick as possible) to see what strategies would be developed in order to best contain the spread.

8 Summary

Our project started from the work of [8], which introduced the SZR model as an alternative to more common SIR model. SZR model is special due to the existence of an analytical solution and the possibility to describe systems of our interest. We used the queue-bond algorithm on the simple SZR model where agents lives on pixel-shape site and modify the toy model, and implemented it with the contact tracking algorithm, for a more realistic situation. We have explored the critical behavior of SZR system by searching for critical points and exploiting their relationship between sparsity and occupancy. Besides, square pixel and hexagonal pixel are compared for critical points values and their relevant properties. As a supplement, we have used the learning mitigation strategies to train the agents to move for better inter-agent dynamic response, in which different types of rewards function are evaluated.

Our numerical studies confirm that the universality class of the SZR model does not depend on occupation or sparsity of the lattice. Moreover, the hexagonal model and MSZR model still seem to fit into the same universality class. The relation of the critical points of the SZR model to the human population sparsity implies the effectiveness of firewalls or other large scale obstacles in suppressing a zombie outbreak. The reinforcement learning result suggests that clustering of humans might be an effective survival strategy during the zombie outbreak.

References

[1] M. Crossley and M. Amos, "Simzombie: A case-study in agent-based simulation construction," in *Agent and Multi-Agent Systems: Technologies and Applications*, J. O'Shea, N. T. Nguyen, K. Crockett, R. J. Howlett, and L. C. Jain, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 514–523, ISBN: 978-3-642-22000-5.

[2] P. Munz, I. Hudea, J. Imad, and R. Smith, "When zombies attack!: Mathematical modelling of an outbreak of zombie infection," pp. 133–150, Jan. 2009.

[3] C. Witkowski and B. Blais, *Bayesian analysis of epidemics - zombies, influenza, and other diseases*, 2013. arXiv: 1311.6376 [q-bio.PE].

[4] F. Nuñez, C. Ravello, H. Urbina, and T. Perez-Acle, *A rule-based model of a hypothetical zombie outbreak: Insights on the role of emotional factors during behavioral adaptation of an artificial population*, 2012. arXiv: 1210.4469 [q-bio.PE].

[5] J. P. A. de Mendonça, L. M. V. Teixeira, F. Sato, and L. R. N. Ferreira, "Modeling a hypothetical zombie outbreak can save us from real-world monsters," *The Mathematical Intelligencer*, vol. 41, no. 3, pp. 72–79, Sep. 2019, ISSN: 1866-7414. DOI: 10.1007/s00283-019-09893-9. [Online]. Available: https://doi.org/10.1007/s00283-019-09893-9.

[6] M. A. Amaral and J. J. Arenzon, "Rumor propagation meets skepticism: A parallel with zombies," *EPL (Europhysics Letters)*, vol. 124, no. 1, p. 18 007, 2018.

[7] M. J. Guitton and C. Cristofari, "Does surviving the zombie apocalypse represent a good model of human behavior in response to pandemics?" en, *J Public Health Manag Pract*, vol. 20, no. 4, pp. 375–377, Jul. 2014.

[8] A. Alemi, M. Bierbaum, C. Myers, and J. Sethna, "You can run, you can hide: The epidemiology and statistical mechanics of zombies," *Physical Review E*, vol. 92, Mar. 2015. DOI: 10.1103/PhysRevE.92.052801.

[9] D. T. Gillespie, A. Hellander, and L. R. Petzold, "Perspective: Stochastic algorithms for chemical kinetics," en, *J Chem Phys*, vol. 138, no. 17, p. 170 901, Mar. 2013.

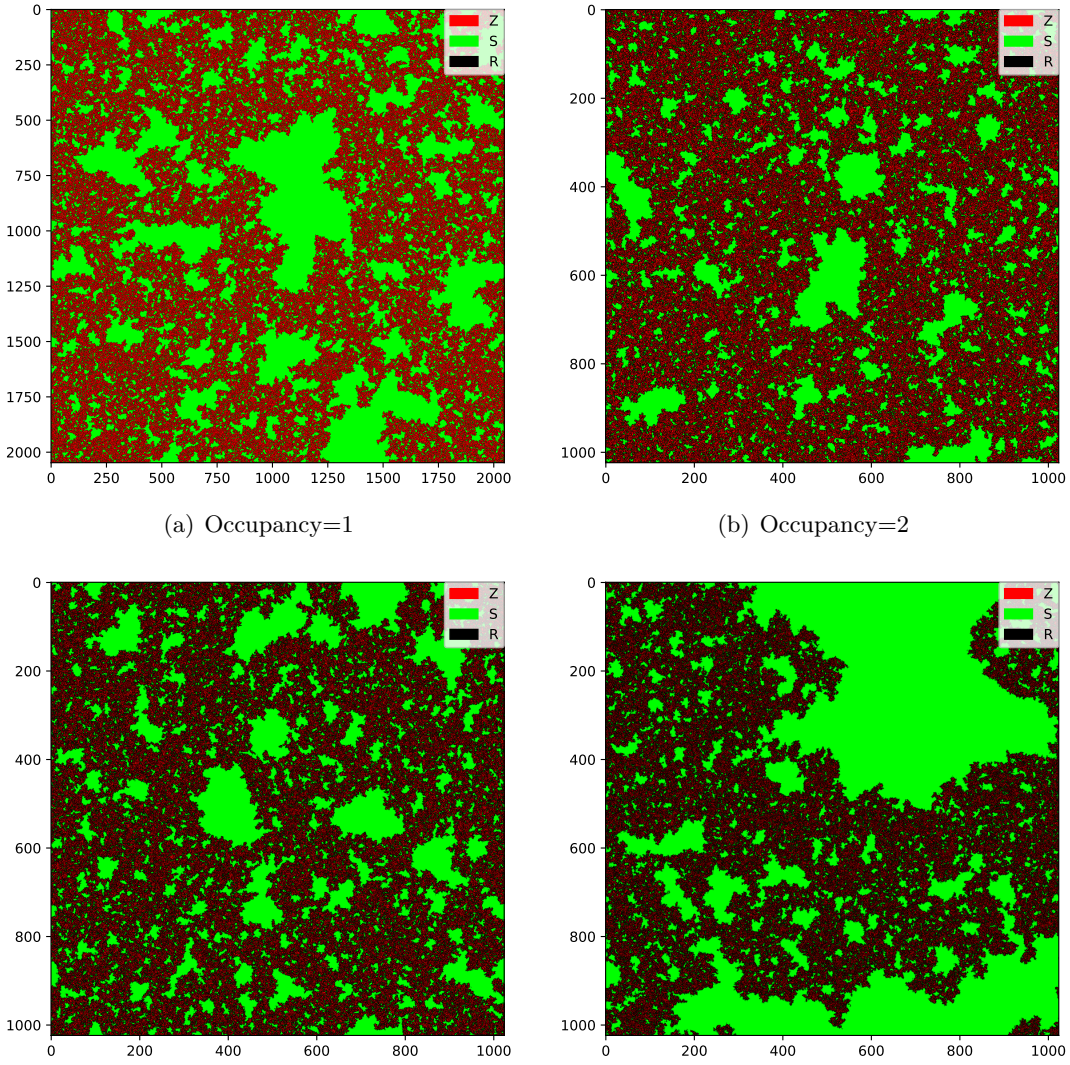
[10] D. Stauffer and A. Aharony, *Introduction To Percolation Theory*. Taylor & Francis, Dec. 2018. DOI: 10.1201/9781315274386. [Online]. Available: https://doi.org/10.1201/9781315274386.

[11] L. Böttcher, *Computational statistical physics*. Cambridge, United Kingdom New York, NY: Cambridge University Press, 2021, ISBN: 978-1108841429.

[12] V. Mnih, K. Kavukcuoglu, D. Silver, *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529–533, 2015.

[13] P. Sunehag, G. Lever, A. Gruslys, *et al.*, "Value-decomposition networks for cooperative multi-agent learning," *CoRR*, vol. abs/1706.05296, 2017. arXiv: 1706.05296. [Online]. Available: http://arxiv.org/abs/1706.05296.

9 Backup



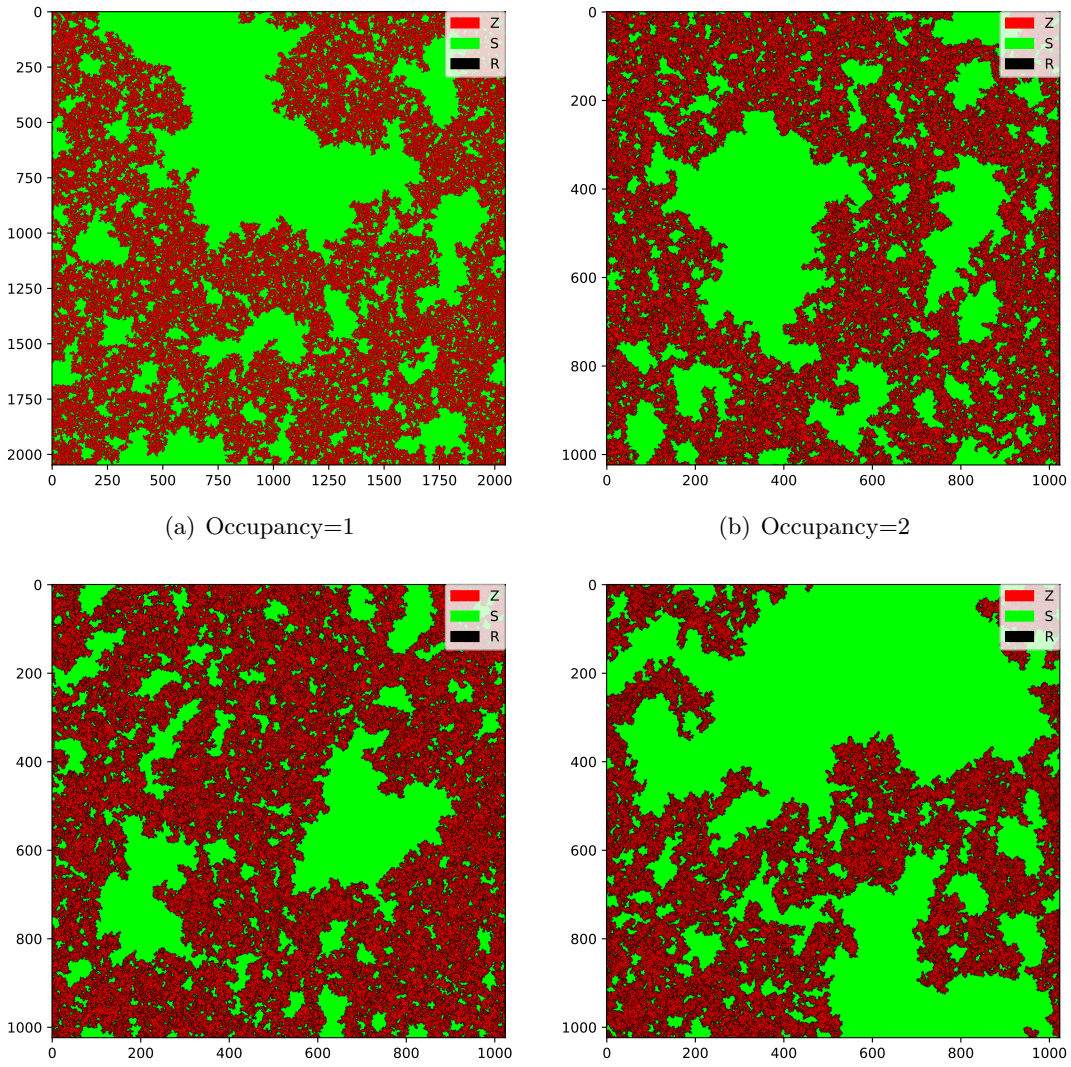
(a) Occupancy=1

(b) Occupancy=2

(c) Occupancy=3

(d) Occupancy=4

Figure 11: Critical behaviors of SZR model



(a) Occupancy=1

(b) Occupancy=2

(c) Occupancy=3

(d) Occupancy=4

Figure 12: Critical behaviors of MSZR model