On cluster approximations of cellular automata

Ilmari Karonen

January 29, 2009

M.Sc. thesis

Abstract

In this thesis I examine one commonly used class of methods for the analytic approximation of cellular automata, the so-called *local cluster approximations*. This class subsumes the well known *mean-field* and *pair approximations*, as well as higher order generalizations of these. While a straightforward method known as *Bayesian extension* exists for constructing cluster approximations of arbitrary order on one-dimensional lattices (and certain other cases), for higher-dimensional systems the construction of approximations beyond the pair level becomes more complicated due to the presence of loops. In this thesis I describe the one-dimensional construction as well as a number of approximations suggested for higher-dimensional lattices, comparing them against a number of consistency criteria that such approximations could be expected to satisfy. I also outline a general variational principle for constructing consistent cluster approximations of arbitrary order with minimal bias, and show that the one-dimensional construction indeed satisfies this principle. Finally, I apply this variational principle to derive a novel consistent expression for symmetric three cell cluster frequencies as estimated from pair frequencies, and use this expression to construct a quantitatively improved pair approximation of the well-known lattice contact process on a hexagonal lattice.

Contents

1 Cellular automata

A cellular automaton consists of a lattice of cells L, each of which, at any given time, is in a state belonging to some finite set Ω.[1] The state of a cell $x \in L$ at time t will be denoted by $\sigma_x(t)$, and these together form the global state of the automaton $\sigma(t)$.

The time evolution of the state of the cellular automaton is given by a *rule*; these rules come in various kinds, which I will enumerate below. A defining characteristic of cellular automata is that these rules are *memoryless*, such that the state of the automaton at time $t+1$ depends only on its state at time t,[2] and *local*, such that the time evolution of the state of each cell depends only on the states of the immediately adjacent cells.

Specifically, each cell $x \in L$ is equipped with a *neighborhood* (or *environment*) $N_x \subset L$, such that the time evolution of the cell's state σ_x depends only on the state of its neighborhood (which I will denote by σ_{N_x}). Opinions vary on whether or not a cell should be considered a part of its own neighborhood; since both definitions have their advantages, I shall use them both below, adopting the convention that N_x denotes the neighborhood excluding x and $\bar{N}_x = N_x \cup \{x\}$ the neighborhood including it.

The rules governing the time evolution of cellular automata can be divided into several classes. The most traditional kind are the *discrete-time*, *deterministic* rules, where the state of the cell x at time $t+1$ is uniquely determined by the state of its (inclusive) neighborhood at time t:

$$\sigma_x(t+1) = r_x(\sigma_{\bar{N}_x}(t)). \tag{1.1}$$

I will also consider *stochastic* rules, where the evolution of the automaton may involve an element of randomness. For these rules, the state of the neighborhood $\bar{N}_x$ at time t defines the *probability* of the cell x ending up in any given state at time $t+1$. Formally, for each state $a \in \Omega$,

$$\Pr[\sigma_x(t+1) = a] = r_x(a|\sigma_{\bar{N}_x}(t)). \tag{1.2}$$

This equation, applied to each cell of the lattice, defines a stochastic trajectory for the global state of the automaton. However, taking advantage of the memorylessness of the rule, equation 1.2 can be rewritten to describe the *deterministic* time evolution of the *probabilities* of the various states.

[1] While allowing infinitely many states per cell would be a straightforward generalization, the approximations discussed in this paper generally depend on the state space being finite.

[2] That is, cellular automata are *Markov processes*.

This is accomplished by summing over all the possible lattice states $\tau \in \Omega_L$, yielding

$$\Pr[\sigma_x(t+1) = a] = \sum_{\tau \in \Omega_L} \Pr[\sigma(t) = \tau]\ r_x(a|\tau_{\bar{N}_x}), \tag{1.3}$$

where $\tau_{\bar{N}_x}$ is the restriction of the global state τ to the cells in the inclusive neighborhood of x.

The final class of rules I will consider are *continuous-time* rules, where cells are allowed to change their state at any time, and do so at a rate determined by the current state of their neighborhood. These transitions are usually assumed to occur independently, but it is also possible to consider rules which allow concerted transition events involving multiple cells (e.g. *movement* events where two adjacent cells exchange states). For the case of independent transitions, the rule may be written as a set of differential equations, such that, writing $p(\tau, t) = \Pr[\sigma(t) = \tau]$ and $p_x(a, t) = \Pr[\sigma_x(t) = a]$, for all states $a \in \Omega$,

$$\frac{d}{dt} p_x(a) = \sum_{\tau \in \Omega_L} p(\tau, t) \sum_{b \in \Omega} (f_x(b \to a|\tau_{N_x}) - f_x(a \to b|\tau_{N_x})) \tag{1.4}$$

where

$$f_x(a \to b|\tau_{N_x}) = p_x(a, t)\ r_x(a \to b|\tau_{N_x}) \tag{1.5}$$

and $r_x(a \to b|\tau_{N_x})$ is the rate at which the cell x changes from state a to b when its neighborhood is in state τ_{N_x}; for simultaneous multi-cell transitions, the equations would have to be written in terms of larger clusters. In general, however, I will not specify rules explicitly like this, but will instead merely provide expressions for the fundamental transition rates $r(a \to b|\tau_{N_x})$ (or the corresponding multi-cell transition rates) from which the global evolution of the automaton, as well as the various approximations thereof discussed below, may be defined.

Of the types listed above, the deterministic cellular automata may obviously be treated as a subset of the stochastic (discrete-time) ones. The discrete- and continuous-time classes, on the other hand, are genuinely different.[3] To avoid excessive duplication of notation, in the remainder of this thesis I shall mostly restrict myself to considering the continuous-time case,

[3]While continuous-time dynamical systems may in general be treated as a subset of discrete-time ones, attempting to discretize continuous-time cellular automata would break the locality criterion, since there is no fundamental limit on the number of adjacent cells that might change state during a single timestep.

trusting the reader to be able to make the straightforward adjustments necessary to apply the described methods to discrete-time automata and only mentioning the differences between the two cases where they are of specific relevance.

1.1 Examples

At this point, it may be useful to introduce some concrete examples of different types of cellular automata.

Example 1.6 (Conway's Game of Life)**.** The prototypical example of a "classical" deterministic cellular automaton, and perhaps the most widely known of all cellular automata, is the *Game of Life* introduced by John Conway in 1970.[Gar70] It is based on a two-dimensional lattice of cells $L = \mathbb{Z}^2$, where the neighborhood of each cell consists of the four cells orthogonally and four diagonally adjacent to it. Each cell may be in one in one of two states conventionally denoted as $\Omega = \{dead, alive\}$, the rule being that a cell will be *alive* at time $t + 1$ if and only if, at time t, either

(a) it is *dead* and exactly three of its eight neighbors are *alive*, or

(b) it is *alive* and two or three of its eight neighbors are *alive*.

From this simple rule, surprisingly complex behavior emerges. Originally envisioned as a grossly simplistic spatial model of population growth, the Game of Life has attracted considerable interest among computational theorists due to its propensity to exhibit a rich variety of stationary and moving time-periodic cell patterns that interact in complex ways. The Game of Life has been shown to be *computationally universal*, such that there exist arrangements of cell states whose evolution under the rule emulates a universal Turing machine.[Ada02]

Example 1.7 (The lattice contact process)**.** Contact processes are commonly used in epidemiology to model the spread of diseases. In the simplest possible "SIS" contact process, the population consists of a large number of otherwise identical individuals, each of which is in one of two states: susceptible (S) or infected (I). Individuals meet each other at rate c; when a susceptible individual meets an infected one, it becomes infected with probability p. Infected individuals recover at rate r, becoming susceptible again. If the population is assumed to be well mixed, then, in the limit as the population size tends to infinity, the dynamics or the fraction i of infected individuals

in the population will evolve deterministically according to the differential equation

$$\frac{di}{dt} = cp(1-i)i - ri. \tag{1.8}$$

This system has a trivial steady state at $i = 0$. If $cp \leq r$, this trivial equilibrium is stable, and the system will tend towards it from any initial state $0 \leq i_0 \leq 1$, since di/dt will be strictly negative for all $i > 0$. If $cp > r$, however, the trivial equilibrium at $i = 0$ will be unstable, and the system will instead converge to a nontrivial equilibrium at

$$\tilde{\imath} = 1 - \frac{r}{cp}.$$

To introduce an explicit spatial element to the SIS model, the individuals may be assumed to reside on a suitably chosen lattice L, one individual per cell, and to interact only with individuals in their neighborhood, each cell having n neighbors.[4] This turns the SIS model into a continuous-time cellular automaton evolving according to the rule

$$\frac{d}{dt}\Pr[\sigma_x = \mathrm{I}] = cp \sum_{\tau \in \Omega_L} \Pr[\sigma(t) = \tau]\delta_{\mathrm{S}}(\tau_x)\frac{n_{\mathrm{I}}(\tau_{N_x})}{n} - r\Pr[\sigma_x(t) = \mathrm{I}], \tag{1.9}$$

where $\delta_{\mathrm{S}}(\tau_x)$ equals 1 if $\tau_x = \mathrm{S}$ and 0 otherwise, and $n_{\mathrm{I}}(\tau_{N_x})$ equals the number of cells in state I in the neighborhood N_x of x. Expressed in terms of equations 1.4 and 1.5,

$$r_x(\mathrm{S} \to \mathrm{I}|\tau_{N_x}) = cp\frac{n_{\mathrm{I}}(\tau_{N_x})}{n}, \text{ and} \tag{1.10}$$

$$r_x(\mathrm{I} \to \mathrm{S}|\tau_{N_x}) = r. \tag{1.11}$$

If the lattice is assumed to consist of infinitely many cells, then the time evolution of the mean fraction i of cells in state I again follows a deterministic trajectory. The qualitative behavior of the simple lattice contact process turns out to resemble that of the well-mixed case, at high values of r/cp converging to a trivial equilibrium at $i = 0$ and at lower values to a non-trivial equilibrium $0 < \tilde{\imath} \leq 1$. However, the locality of interactions in the spatial SIS model tends to cause clustering of the infected cells, which reduces the likelihood of S–I contacts and therefore lowers the effective infection rate. This causes the both the equilibrium density and the extinction threshold to be lower than for the well-mixed process. The actual equilibrium value depends on both the number of cells in each cell's neighborhood and on the connectivity of the lattice, including the presence of loops.

[4]Models of this type have been widely employed in the literature at least since the 1960s; see for example [Mol77].

2 Approximating cellular automata

A characteristic feature of many cellular automata—and, indeed, one of the main reasons why they are considered interesting in the first place—is their capability for self-organization, including the spontaneous emergent formation of complex patterns and long-range correlations. Unfortunately, the same features that make cellular automata useful models of complex emergent phenomena also often make them frustratingly intractable—the map may turn out to be as complex as the territory. Indeed, some have argued that the proper way to study cellular automata should be an essentially experimental one: run the automaton and see what happens. Alas, rightly or wrongly, in many fields such purely empirical studies may be seen as insufficiently rigorous, at least by themselves. Thus, one would frequently wish to examine cellular automata in a more analytical fashion, even if such rigor may come at the expense of simplifying the model.

Many such approximations of varying accuracy have been devised. The very crudest include the so called *activity* of the automaton (often denoted in cellular automata literature as λ; see e.g. [Wol02]), which is simply the probability of a cell being in a state (or one of several states) designated as "active" after one time step, summed over all possible starting states of the cell and its neighborhood. Though an extremely crude indicator of automaton behavior, it has nonetheless found some use (mainly among computational theorists working with classical, synchronous CA) due to its simplicity: the activity can be directly and very straighforwardly calculated in closed form from the rule table.

The next simplest approximation is the *mean field density* , which refines the concept of the activity by weighing the contributions of various starting neighborhoods according to their probability of occurring, under the assumption that each cell on the lattice has the same, independent "mean field" probability of being in a given state. Formally, for a continuous-time automaton, the mean field state probabilities $\tilde{p}_{\mathrm{mf}}$ are given by solving the equation

$$\sum_{b\in\Omega, E\in\Omega_N} \left(\tilde{p}_{\mathrm{mf}}(b)\, r(b \to a|E) - \tilde{p}_{\mathrm{mf}}(a)\, r(a \to b|E)\right) \prod_{y\in N} \tilde{p}_{\mathrm{mf}}(E_y) = 0, \tag{2.1}$$

where N is the neighborhood of an arbitrarily chosen cell (with all neighborhoods assumed isomorphic), for all states $a \in \Omega$.

Though it originates in the field of statistical physics, the mean field approximation finds frequent use e.g. in spatial ecology and epidemiology, since it essentially assumes the lattice to be *well mixed*, as if the cells were randomly exchanging places at a high rate. As such, it provides a useful way

of connecting cellular automata models to traditional, non-spatial models which assume a well mixed population.

Though the mean field density is a much better approximation of the actual average evolution of the cellular automaton than the crude concept of activity, it still almost entirely ignores the spatial structure of the lattice. The next step up from the mean field approximation is the *pair approximation* [vB00], also commonly employed e.g. in spatial ecology, which keeps track of the probabilities of *pairs* of adjacent cells and uses these to approximate the neighborhood probabilities in a manner similar to the mean field approximation.[5]

The pair approximation incorporates some information about the lattice structure, but still fails to account for features such as loops of interconnected cells, as well as failing entirely to model any possible emergent structures involving larger clusters of cells than pairs. Sometimes such features can have significant quantitative or even qualitative effects on the behavior of the automaton, in which case a higher-order approximation may be called for.

2.1 Cluster approximations

Cluster approximations are a general class of approximations subsuming the mean field and pair approximations. The idea is that, instead of considering the time evolution of the entire state of the lattice, or of the probabilities of all the possible global states, one only looks at one or more finite clusters of cells, and the probabilities of those cluster being in particular states. For every cluster X, one then has a set of differential equations[6], one for each possible cluster state $A \in \Omega_X$:

$$\frac{\mathrm{d}}{\mathrm{d}t} p_X(A) = \sum_{B \in \Omega_X} (r(B \to A) - r(A \to B)), \tag{2.2}$$

where $r(A \to B)$, the mean rate at which state A changes to state B, is a weighted sum of the conditional transition rates $r(A \to B|E)$ in each environment $E \in \Omega_{N_X}$ (which may be derived directly from the automaton rule):

$$r(A \to B) = \sum_{E \in \Omega_{N_X}} p_{\bar{N}_X}(AE) r(A \to B|E). \tag{2.3}$$

[5]Both the mean field and pair approximations will be described in more detail in the following sections.

[6]For discrete-time cellular automata, the set would consist of finite difference equations, with $p_X(A, t+1) - p_X(A, t)$ replacing $\frac{\mathrm{d}}{\mathrm{d}t} p_X(A)$ on the left-hand side of equation 2.2.

The equations 2.2 and 2.3 above describe the time evolution of the cellular automaton exactly. However, their usefulness is limited by the fact that the rates of change for the cluster X depend on the state probabilities $p_{\bar{N}_X}$ for the extended cluster $\bar{N}_X$, which are not known. One could, of course, write a similar set of differential equations for the extended cluster, but those equations would then involve the state probabilities for the twice-extended cluster $\bar{N}_{\bar{N}_X}$, and so on. To escape from this endless cycle, one must *close* the system of differential equations by somehow *approximating* the extended cluster probabilities $p_{\bar{N}_X}$ in terms of the original probabilities p_X.

There are, of course, many possible ways to do this closure, and indeed many have been suggested in the literature.[7] For practical and theoretical reasons, it would be desirable for the closure to satisfy the basic requirements of *Kolmogorov consistency*: the state probabilities for any cluster should be non-negative and should sum to 1. It would also be desirable for the marginal probabilities of any subclusters to be *automorphism invariant*: if there is a bijection $g : L \to L$ that maps the lattice onto itself, such that $g(N_x) = N_{g(x)}$ (and $r_x = r_{g(x)}$) for all $x \in L$, and if X and $Y = g(X)$ are both subsets of a cluster Z, then the marginal probabilities $p_X(A)$ and $p_Y(A)$ obtained by summing over p_Z should be equal for all states $A \in \Omega_X$.[8]

Even so, that still leaves a large number of possible closures. (And indeed, it will become apparent later on that few of the commonly used closures in more than one dimension even satisfy all of the requirements given above.) Many of those closures, of course, will be rather arbitrary, containing aspects that have no obvious justification; they might, for example, be arbitrarily biased towards certain cluster configurations over others. As the choice of the closure does affect the results that may be derived from the approximation, it should be clear that, by choosing a biased closure, one may obtain biased results—results that might reflect more on the assumptions and preconceptions of the approximator than on the actual behavior of the automaton itself. It would thus be desirable to have some rule for choosing, out of all the possible closures, the one that was, in some sense, the least arbitrary or biased.

[7] See for example [GV87], which gave me the original inspiration for choosing this topic for my thesis.

[8] It may not always be reasonable to demand invariance with respect to *all* the automorphisms of the lattice: for example, one might not wish to consider all sites equivalent even if their environments (and rules) are isomorphic, or, more importantly, one might well want to consider globally asymmetric states even on a symmetric lattice. In general, one would like to demand at least *shift* invariance, for the cluster approximation to make sense as a global description of the lattice, but the inclusion of other automorphisms may require additional justification. Naturally, the choice of automorphisms to consider is likely to affect other aspects of the closure construction.

I suggest that such a closure is given by the *principle of maximum entropy*. Specifically, I define the maximum entropy closure as the one that is closest to uniform, in that it minimizes the Kullback–Leibler divergence () D between the estimated extended cluster probabilities and the uniform distribution (where each cluster state $C \in \Omega_Y$ occurs with the same probability $\frac{1}{\#\Omega_Y}$, $\#\Omega_Y$ being the number of possible cluster states),

$$D_Y = \sum_{C \in \Omega_Y} p_Y(C) \log \frac{p_Y(C)}{\frac{1}{\#\Omega_Y}} = \log \#\Omega_Y + \sum_{C \in \Omega_Y} p_Y(C) \log p_Y(C), \tag{2.4}$$

subject to the consistency requirements above and the known marginal probabilities. This is equivalent to saying that it maximizes the Shannon entropy of the extended cluster probabilities, from whence the name.

Informally, the justification for choosing the maximum entropy closure is that, in choosing the extended cluster state probabilities as close to uniform as possible, given the information available, it minimizes the *a priori* bias in the approximation. In the absence of *any* information to the contrary, the most reasonable choice would be to assign equal probabilities to all the possible cluster states, so that should be the starting point from which the probability distribution can be refined according to the constraints imposed by the known marginal probabilities. Indeed, as I will demonstrate in a following section, in those cases where a fully consistent and automorphism invariant closure can be constructed on purely probabilistic grounds, based only on an assumption of conditional independence and an application of Bayes' rule, that closure turns out to be the maximum entropy one.

2.2 The mean field approximation

The simplest possible cluster approximation is one where the basis clusters are simply individual cells. Assuming—since we have no evidence to the contrary—that the state probabilities of different cells are independent, i.e. that $p_{xy}(ab) = p_x(a)p_y(b)$ for all $x \neq y$, $a \in \Omega_x$, $b \in \Omega_y$, allows equation 2.3 to be closed by substituting

$$p_{\bar{N}_x}(aE) = p_x(a) \prod_{y \in N_x} p_y(E_y), \tag{2.5}$$

where $E_y \in \Omega_y$ denotes the state assigned to the cell y by the neighborhood cluster state $E \in \Omega_{N_x}$.

If we further assume the state spaces $\Omega_x = \Omega$ and probabilities $p_x(a) = p_{\mathrm{mf}}(a)$, $a \in \Omega$, are the same for all cells $x \in L$ on the lattice, and that the neighborhoods $N_x \sim N$ for all cells are isomorphic,[9] equations 2.2 and 2.3 reduce to a finite system of differential equations

$$\frac{\mathrm{d}}{\mathrm{d}t} p_{\mathrm{mf}}(a) = \sum_{b \in \Omega, E \in \Omega_N} \left(p_{\mathrm{mf}}(b)\, r(b \to a | E) - p_{\mathrm{mf}}(a)\, r(a \to b | E) \right) \prod_{y \in N} p_{\mathrm{mf}}(E_y), \tag{2.6}$$

whose stationary points are as given by equation 2.1 earlier.

This simple cluster approximation is known as the *mean field approximation*. The concept of the mean field approximation, in a broad sense, is considerably older than the general theory of cluster approximation, or indeed even the concept of cellular automata. The term comes from statistical physics, where it is used to approximate the behavior of a particle, in a system consisting of many like it, by replacing the influences of the other individual particles by an average term representing the "mean field" excerted upon a randomly chosen particle in an ensemble of particles itself randomly chosen from among those with the same bulk thermodynamic properties.

The mean field approximation allows the time evolution of these bulk system properties—such as density, temperature, etc.—to be approximated by a closed system of equations that may be solved analytically. In cellular automata, the analogues of these bulk system properties are the mean state densities $p_{\mathrm{mf}}(a)$, whose approximate time evolution is given by equation 2.6.

[9] Many of these assumptions can of course be relaxed with appropriate modifications to equation 2.6. For example, one could consider systems where the lattice cells are divided into several types with possibly different state spaces, probabilities and neighborhood sizes.

2.3 The pair approximation

Going beyond the mean field approximation, the obvious next step is to consider clusters consisting of pairs of adjacent cells. Instead of the complete independence assumption in the mean field approximation, we will assume the state probabilities to be *conditionally independent*—that is, that, given three cells x, y and z, of which xy and yz are adjacent pairs but xz isn't,

$$p_{xyz}(abc) = p_{xy}(ab)p_{z|y}(c|b) = p_{yz}(bc)p_{x|y}(a|b) = \frac{p_{xy}(ab)\,p_{yz}(bc)}{p_y(b)}, \tag{2.7}$$

where $p_{x|y}(a|b) = \frac{p_{xy}(ab)}{p_y(b)}$ is the *conditional probability* of cell x being in state a if cell b is in state b.

Assuming that we know the pair probabilities p_{xy} for all pairs xy, the single cell probabilities can of course be easily calculated as the marginal probabilities

$$p_x(a) = \sum_{b\in\Omega_y} p_{xy}(ab). \tag{2.8}$$

Further assuming that the cells in the pair xy have no shared neighbors, we can then close equation 2.3 by substituting

$$p_{\bar{N}_{xy}}(abE) = p_{xy}(ab) \prod_{z\in N_x\setminus y} p_{z|x}(E_z|a) \prod_{z\in N_y\setminus x} p_{z|y}(E_z|b). \tag{2.9}$$

That is, we assume that the state probabilities for the cells (other than y) in the neighborhood of x depend only on the state of x but not on each other, and likewise for those in the neighborhood of y.

Again assuming, as in the previous section, that $N_x \sim N$ and $\Omega_x = \Omega$ for all $x \in L$, with $n = \#N$ being the number of neighbors per cell, and that $p_{xy}(ab) = p_{\text{pair}}(ab)$, for all adjacent pairs xy, $ab \in \Omega_{xy}$, equations 2.2 and 2.3 can be written as

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}p_{\text{pair}}(ab) = &\sum_{c\in\Omega, E\in\Omega_{N_x\setminus y}} (r_{\text{pair}}(c(bE)\to a) - r_{\text{pair}}(a(bE)\to c)) \\ &+ \sum_{d\in\Omega, F\in\Omega_{N_y\setminus x}} (r_{\text{pair}}(d(aF)\to b) - r_{\text{pair}}(b(aF)\to d)),\end{aligned} \tag{2.10}$$

where

$$r_{\text{pair}}(a(bE)\to c) = r(a\to c|bE)\, p_{\text{pair}}(ab) \prod_{z\in N_x\setminus y} p_{\text{pair}}(E_z|a). \tag{2.11}$$

If the lattice contains connected triangles of cells, such that the cells in a pair may have shared neighbors, constructing the pair approximation becomes a bit more complicated. For some applications, one might simply ignore the overlap and apply equation 2.10 as if the neighborhoods were disjoint, essentially "double-counting" the shared neighbors. However, the resulting approximation will include contributions from "impossible" pair neighborhoods $E \in N_{xy}$ that assign two different states to the same cells.

If one instead discounts such impossible configurations, one needs to divide the probabilities in equation 2.9 by a normalizing term (the sum of the approximated probabiltities of all "valid" neighborhood states) in order to preserve Kolmogorov consistency. The equivalent to equation 2.10 then becomes

(2.12)
$$\frac{\mathrm{d}}{\mathrm{d}t}p_{\mathrm{pair}}(ab) = \sum_{E\in\Omega_{N_{xy}}}\sum_{c\in\Omega_x}(r(c\to a|bE)\,p_{\bar{N}_{xy}}(cbE) - r(a\to c|bE)\,p_{\bar{N}_{xy}}(abE)) + \sum_{d\in\Omega_y}(r(d\to b|aE)\,p_{\bar{N}_{xy}}(adE) - r(b\to d|aE)\,p_{\bar{N}_{xy}}(abE)),$$

(2.13)
$$p_{\bar{N}_{xy}}(abE) = p_{xy}(ab)\frac{\prod_{z\in N_x\setminus y}p_{z|x}(E_z|a)\prod_{z\in N_y\setminus x}p_{z|y}(E_z|b)}{\sum_{F\in\Omega_{N_{xy}}}\left(\prod_{z\in N_x\setminus y}p_{z|x}(F_z|a)\prod_{z\in N_y\setminus x}p_{z|y}(F_z|b)\right)},$$

where the denominator is the normalization factor that ensures the probabilities sum to one.

Even so, however, the classical pair approximation is not entirely free of problems in the presence of triangles: even though the normalization forces the probabilities to sum to one, the approximate marginal state probabilities for the shared neighbors, $p_{z;\mathrm{approx}}(c|ab) = p_{xz}(ac)p_{yz}(bc)/C$, where C is the denominator from equation 2.13, are rarely a good approximation of the true probabilities $p_{z|xy}(c|ab)$. I will later describe a way to improve this approximation (at the cost of having to solve a fourth-order polynomial) using a result derived via the maximum entropy method.

3 Cluster approximations in one dimension

For cellular automata on one-dimensional lattices, there exists a straightforward method, known as *Bayesian extension* [GVK87] or sometimes *Markovian extension*, for systematically constructing cluster approximations of arbitrary size. The approximations constructed in this manner are Kolmogorov consistent and shift invariant; they also turn out to be the maximum entropy approximations for each cluster size.

Let $L = \{\ldots, x_{-1}, x_0, x_1, \ldots\}$ denote the lattice of the automaton and $X_{i,j} = \{x_i, \ldots, x_j\}$ a contiguous cluster of cells in it. I shall assume, without loss of generality, that the inclusive neighborhood $\bar{N}_{x_k}$ of a cell $x_k \in L$ is the contiguous cluster $X_{k-l,k+r} = \{x_{k-l}, \ldots, x_{k+r}\}$, with $l \geq 0$, $r \geq 0$, such that the neighborhood of the contiguous cluster $X_{i,j}$, $\bar{N}_{X_{i,j}} = X_{i-l,j+r}$, is itself contiguous. To simplify notation, I shall use the shorthand $p_{i,j}(\sigma) = p_{X_{i,j}}(\sigma)$ to denote the probability of the cluster $X_{i,j}$ having the state σ (or its restriction to the cluster $X_{i,j}$), and $\Omega_{i,j} = \Omega_{X_{i,j}}$ to denote the set of possible states of the cluster $X_{i,j}$.

By appealing to the shift invariance of the lattice (i.e. the fact that $g : L \to L, g(x_i) = (x_{i+k})$ is an automorphism of the lattice for all $k \in \mathbb{Z}$), it may be assumed that (at least under equilibrium conditions) the state probabilities for each cluster of given length should be the same, that is, that $p_{i,j}(\sigma) = p_{i+k,j+k}(\sigma + k)$ for all k, where $\sigma + k$ denotes the state σ shifted by k cells. This assumption is not, actually, required by either of the constructions of the closure rule given below, but it (or something like it) is needed to turn the closure into a practical cluster approximation with a finite number of equations to solve.

Assuming the state probabilities for all n-cell clusters $Y_k = X_{k,k+n-1}$ to be known, closing the equations 2.2 and 2.3 then requires estimating the state probabilities for the extended clusters $\bar{N}_{Y_k} = X_{k-l,k+n-1+r}$ in terms of these known n-cell cluster probabilities. I shall present below two approaches for this, one using probabilistic reasoning based on an assumption of *conditional independence* between distant cells, and another based on the principle of maximum entropy. Both turn out to lead to the same result:

$$p_{i,j}(\sigma) = \frac{p_{i,i+n-1}(\sigma) p_{i+1,i+n}(\sigma) \cdots p_{j-n+1,j}(\sigma)}{p_{i+1,i+n-1}(\sigma) p_{i+2,i+n}(\sigma) \cdots p_{j-n+1,j-1}(\sigma)}, \tag{3.1}$$

that is, the estimated probability of the extended cluster $X_{i,j}$ being in state σ is the product of the corresponding marginal probabilities for the n-cell subclusters, divided by the marginal probabilities for $n-1$ -cell subclusters where each pair of adjacent n-cell clusters overlap.

3.1 A probabilistic construction

Let $xYz \in L$ be a contiguous $n+1$ -cell cluster consisting of the cells x and z and the $n-1$ -cell cluster B between them. The probability of this cluster being in the state aBc, where $a \in \Omega_x$, $B \in \Omega_Y$ and $c \in \Omega_z$, is then

$$p_{xYz}(aBc) = p_{xY}(aB)p_{z|xY}(c|aB), \tag{3.2}$$

where $p_{z|xY}(c|aB)$ is the conditional probability of the cell z being in state c if the cluster xY is in state aB.

So far, this representation is still exact, and still includes the $n+1$ -cell cluster state probabilities in the guise of the conditional probabilities. I will now introduce an additional assumption, namely that the states of cells n or more cells apart are *conditionally independent* such that $p_{z|xY}(c|aB) = p_{z|Y}(c|B)$. This assumption may *not* hold, and indeed probably will not hold exactly, in practice for any given automaton, but it is the central assumption that allows the system of equations 2.2 and 2.3 to be closed. By using the assumption of conditional independence, equation 3.2 can be rewritten as

$$p_{xYz}(aBc) = p_{xY}(aB)p_{z|Y}(c|B) = \frac{p_{xY}(aB)p_{Yz}(Bc)}{p_Y(B)}. \tag{3.3}$$

The state probabilities $p_Y(B)$ for the central $n-1$ -cell cluster Y are, of course, straightforwardly obtained by summing over the n-cell cluster probabilities

$$p_Y(B) = \sum_{a\in\Omega_x} p_{xY}(aB) = \sum_{c\in\Omega_z} p_{Yz}(Bc). \tag{3.4}$$

By reapplying the same extension process to the new $n+1$ -cell cluster state probabilities just obtained, one can then easily derive state probabilities for $n+2$ -cell clusters, and so on, eventually leading to equation 3.1 as given above.[10]

3.2 A maximum entropy construction

Equation 3.1 can also be derived from the principle of maximum entropy.[11] Recall from equation 2.4 that the Kullback–Leibler divergence between $p_{i,j}$ and the uniform distribution is

$$D_{i,j} = \log \#\Omega_{i,j} + \sum_{\sigma\in\Omega_{i,j}} p_{i,j}(\sigma) \log p_{i,j}(\sigma), \tag{3.5}$$

[10] See e.g. [GVK87] for a more thorough treatment.

[11] The derivation given in this section is my own, though the idea itself is not original; [GV87] e.g. cites Brascamp (1971) for an equivalent result.

where the $\log \#\Omega_{i,j}$ term is constant and may be ignored when choosing $p_{i,j}$ so as to minimize $D_{i,j}$ subject to the marginal constraints

$$p_{k,k+n-1}(B) = \sum_{A \in \Omega_{i,k-1}} \sum_{C \in \Omega_{k+n,j}} p_{i,j}(ABC) \tag{3.6}$$

for all $k \in \{i, \ldots, j-n+1\}$ and $B \in \Omega_{k,k+n-1}$, where the n-cell cluster probabilities $p_{k,k+n-1}(B)$ on the left hand side are taken to be known.

Assigning a Lagrange multiplier $\lambda_k(B)$ to each constraint, minimizing $D_{i,j}$ is equivalent to solving

$$\frac{\mathrm{d}\Lambda}{\mathrm{d}p_{i,j}(\sigma)} = 0 \quad \forall\ \sigma \in \Omega_{i,j}, \tag{3.7}$$

where

$$\Lambda = D_{i,j} + \sum_{k=i}^{j-n+1} \sum_{B \in \Omega_{k,k+n-1}} \lambda_k(B) \left(p_{k,k+n-1}(B) - \sum_{A \in \Omega_{i,k-1}} \sum_{C \in \Omega_{k+n,j}} p_{i,j}(ABC) \right). \tag{3.8}$$

Choosing a particular cluster state σ and differentiating L with respect to its probability $p_{i,j}(\sigma)$ gives

$$\frac{d\Lambda}{dp_{i,j}(\sigma)} = 1 + \log p_{i,j}(\sigma) + \sum_{k=i}^{j-n+1} \lambda_k(\sigma_{k,k+n-1}). \tag{3.9}$$

Combined with equation 3.7, this rearranges to

$$\log p_{i,j}(\sigma) = -1 + \sum_{k=i}^{j-n+1} \lambda_k(\sigma_{k,k+n-1}), \tag{3.10}$$

which, by taking the antilogarithm of both sides, gives

$$p_{i,j}(\sigma) = e^{-1} \prod_{k=i}^{j-n+1} e^{\lambda_k(\sigma_{k,k+n-1})}. \tag{3.11}$$

For convenience, I'll define $\zeta_k(\sigma) = e^{\lambda_k(\sigma_{k,k+n-1})}$, so that equation 3.11 becomes

$$p_{i,j}(\sigma) = e^{-1} \prod_{k=i}^{j-n+1} \zeta_k(\sigma). \tag{3.12}$$

From equations 3.6 and 3.12, the marginal n and $n-1$ cell cluster probabilities $p_{k,k+n-1}$ and $p_{k,k+n-2}$ may be written as

$$p_{k,k+n-1}(B) = e^{-1}\zeta_k(B)\Sigma_{\mathrm{L}}(k,B)\Sigma_{\mathrm{R}}(k+1,B), \tag{3.13}$$

$$p_{k,k+n-2}(B) = e^{-1}\Sigma_{\mathrm{L}}(k,B)\Sigma_{\mathrm{R}}(k,B). \tag{3.14}$$

where

$$\Sigma_{\mathrm{L}}(k,B) = \left(\sum_{A\in\Omega_{i,k-1}} \prod_{h=i}^{k-1} \zeta_h(AB) \right), \text{ and} \tag{3.15}$$

$$\Sigma_{\mathrm{R}}(k,B) = \left(\sum_{C\in\Omega_{k+n-1,j}} \prod_{h=k}^{j-n+1} \zeta_h(BC) \right). \tag{3.16}$$

Dividing the product of the n-cell cluster probabilities with the $n-1$ cell ones, the Σ_{L} and Σ_{R} terms (as well as all but one of the e^{-1} factors) cancel, leaving us with

$$\frac{\prod_{k=i}^{j-n+1} p_{k,k+n-1}(\sigma_{k,k+n-1})}{\prod_{k=i+1}^{j-n+1} p_{k,k+n-2}(\sigma_{k,k+n-2})} = e^{-1} \prod_{k=i}^{j-n+1} \zeta_k(\sigma) = p_{i,j}(\sigma), \tag{3.17}$$

which is the same as the result obtained via probabilitic reasoning as equation 3.1.

4 Maximum entropy closures in more than one dimension

The principle of maximum entropy, applied above to derive the Bayesian extension rule for constructing cluster approximations on one-dimensional lattices, can also be used to derive consistent and automorphism invariant closure rules on multidimensional lattices. The basic step for applying the maximum entropy method are:

1. Enumerate the constraints imposed on the extended cluster state probabilities by Kolmogorov consistency, lattice automorphisms and the known marginal state probabilities for smaller clusters. All these constraints take the form of linear equations.

2. Simplify the resulting system of linear equations (e.g. via simple Gaussian elimination), giving expressions for the extended cluster probabilities as functions of the known smaller cluster probabilities and one or more free variables.

3. Substitute the expressions into equation 2.4 and solve it to obtain the maximum-entropy closure.

4. Plug the closure into equations 2.2 and 2.3 and solve the resulting system.

Steps 1 and 2 are straightforward; the difficulty lies in performing steps 3 and 4 analytically. As shown below, step 3 reduces to solving a system of polynomial equations, but the polynomials involved may be numerous and of high degree. Numerical solution methods can of course be employed, but this may defeat the point of constructing the approximation in the first place, which is often to obtain analytical insight into the behavior of the automaton.

An additional difficulty is that, while the one-dimensional result derived above was independent of the state space Ω, this is not necessarily true in general. Thus, increasing the number of possible states necessitates deriving the closure anew from scratch, with the number of constraints involved in the derivation increasing as a high power of the number of states.

At least for some simple clusters and small state spaces, however, closed form maximum extropy closures can indeed be found. As an example, I will derive the maximum entropy approximation of *triangle probabilities* as the function of known *pair probabilities* on an isotropic six-neighbor lattice with two states ($\Omega = \{0, 1\}$) per cell.[12] The resulting expression, though

[12] I am not currently aware of any prior work featuring this particular result; it may be original to this thesis.

not particularly useful as a closure rule on its own, may be applied e.g. to construct a better closure for the pair approximation on a hexagonal lattice.

4.1 Maximum entropy triangle probabilities

Let $T = xyz$ be a cluster of three cells, each with two possible states ($\Omega = \{0, 1\}$). I will assume the pair state probabilities $p_{xy} = p_{yx} = p_{yz} = p_{zy} = p_{zx} = p_{xz} = p_{\text{pair}}$ to be known and, due to symmetry, identical for all pairs of cells in T.[13] The goal is to compute the maximum entropy estimate of the triangle state probabilities p_T as a function of the pair probabilities p_{pair}.

Since, for this particular example, there is no risk of confusion, I'll adopt the shorthand of omitting the subscripts from the probabilities and simply writing $p(abc) = p_T(abc)$ for the triangle probabilities, $p(ab) = p_{\text{pair}}(ab)$ for the pair probabilities and $p(a) = p_x(a) = p_y(a) = p_x(a)$ for the single cell probabilities.

Since there are two possible states, 0 and 1, per cell, there are four possible pair states (00, 01, 10 and 11) and eight possible triangle states (000, 001, 010, 011, 100, 101, 110 and 111). However, there are several consistency and symmetry constraints that link the various state probabilities together:

Sum constraint:

$$p(000)+p(001)+p(010)+p(011)+p(100)+p(101)+p(110)+p(111) = 1$$

Marginal constraints:

$$\begin{aligned}
p(000) + p(001) &= p(000) + p(100) = p(000) + p(010) = p(00) \\
p(010) + p(011) &= p(001) + p(101) = p(001) + p(011) = p(01) \\
p(100) + p(101) &= p(010) + p(110) = p(100) + p(110) = p(10) \\
p(110) + p(111) &= p(011) + p(111) = p(101) + p(111) = p(11)
\end{aligned}$$

Symmetry constraints:

$$\begin{aligned}
p(001) &= p(010) = p(100) \\
p(011) &= p(110) = p(101)
\end{aligned}$$

Of course, similar symmetry and sum constraints exist for pairs and single cells, but these are clearly implied by the constraints listed above. Even so,

[13] In particular, the symmetry implies that $p_{\text{pair}}(ab) = p_{\text{pair}}(ba)$ for all $a, b \in \Omega$. In fact, for $\Omega = \{0, 1\}$, this also follows from the assumption that all single cells have identical state probabilities, since then $p_{xy}(01) = p_x(0) - p_{xy}(00) = p_y(0) - p_{xy}(00) = p_{xy}(10)$.

there are 17 constraints but only 8 unknowns $(p(000), \ldots, p(111))$, so the system is overdetermined: if a solution exists at all, some of the constraints must still be redundant. When simplifying the system of linear equations, it turns out to be convenient to express the other known values as functions of the single-cell and pair probabilities $p := p(1) = p(10) + p(11)$ and $q := p(11)$. The simplified linear system then becomes

$$\begin{aligned} p(000) &= -z + 3q - 3p + 1 \\ p(001) = p(010) = p(100) &= z - 2q + p \\ p(011) = p(101) = p(110) &= -z + q \\ p(111) &= z \end{aligned} \tag{4.1}$$

with one unknown parameter z. The Kullback–Leibler divergence as given in equation 2.4 is then

$$\begin{aligned} D = &\log 8 \\ &+ (-z + 3q - 3p + 1) \log(-z + 3q - 3p + 1) \\ &+ 3(z - 2q + p) \log(z - 2q + p) \\ &+ 3(-z + q) \log(-z + q) \\ &+ z \log z, \end{aligned} \tag{4.2}$$

and its derivative with respect to z is

$$\begin{aligned} \frac{\mathrm{d}D}{\mathrm{d}z} = &- \log(-z + 3q - 3p + 1) \\ &+ 3 \log(z - 2q + p) \\ &- 3 \log(-z + q) \\ &+ \log z. \end{aligned} \tag{4.3}$$

The equation $\frac{\mathrm{d}D}{\mathrm{d}z} = 0$ rearrages to

$$\log(-z + 3q - 3p + 1) + 3 \log(-z + q) = 3 \log(z - 2q + p) + \log z, \tag{4.4}$$

where one can take the antilogarithm of both sides to obtain the polynomial equation

$$(z - 3q + 3p - 1)(z - q)^3 = (z - 2q + p)^3 z. \tag{4.5}$$

This is a fourth degree polynomial equation, and therefore has a closed form solution in terms of radicals. The Maple computer algebra system gives the solution as $z = 1/6\,(540\,q^3p^2 - 216\,q^4p + 108\,q^4 - 324\,p^3q^2 - 324\,pq^3 +$

$324\,p^2q^2 - 108\,p^4q + 108\,p^5 - 108\,p^3q - 216\,p^6 + 648\,p^5q - 648\,p^4q^2 + 216\,p^3q^3 + 12\,(81\,p^6q^2 - 486\,p^5q^3 + 1215\,p^4q^4 + 3618\,q^5p^4 - 4068\,q^6p^3 - 27\,p^{10} - 756\,q^8p - 2943\,q^6p^4 + 864\,q^7p^3 - 108\,q^8p^2 + 2430\,q^7p^2 + 5562\,q^5p^5 - 414\,q^3p^6 - 6345\,q^4p^6 - 1296\,p^5q^4 + 540\,p^7q^2 - 1809\,p^8q^2 + 378\,p^9q - 486\,q^7p + 1215\,q^6p^2 - 1620\,q^5p^3 - 162\,p^8q + 96\,q^9 + 81\,q^8 + 12\,p^9 + 4428\,q^3p^7)^{1/2})^{1/3} - 6\,(2/3\,q^3 - pq^2 + 1/3\,p^3 - p^4 + 2\,p^3q - p^2q^2)/((540\,q^3p^2 - 216\,q^4p + 108\,q^4 - 324\,p^3q^2 - 324\,pq^3 + 324\,p^2q^2 - 108\,p^4q + 108\,p^5 - 108\,p^3q - 216\,p^6 + 648\,p^5q - 648\,p^4q^2 + 216\,p^3q^3 + 12\,(81\,p^6q^2 - 486\,p^5q^3 + 1215\,p^4q^4 + 3618\,q^5p^4 - 4068\,q^6p^3 - 27\,p^{10} - 756\,q^8p - 2943\,q^6p^4 + 864\,q^7p^3 - 108\,q^8p^2 + 2430\,q^7p^2 + 5562\,q^5p^5 - 414\,q^3p^6 - 6345\,q^4p^6 - 1296\,p^5q^4 + 540\,p^7q^2 - 1809\,p^8q^2 + 378\,p^9q - 486\,q^7p + 1215\,q^6p^2 - 1620\,q^5p^3 - 162\,p^8q + 96\,q^9 + 81\,q^8 + 12\,p^9 + 4428\,q^3p^7)^{1/2})^{1/3}) - p^2 + pq + q$. Though verbose, this is nonetheless a closed-form solution than can be substituted into the equations 4.1 to give expressions for the maximum entropy estimates of the triangle state probabilities in terms of the pair probabilities.

5 Application to pair approximations on a hexagonal lattice

As noted earlier, the classical method of constructing pair approximations to cellular automata presents a conceptual problem when applied no lattices featuring connected triangles of cells: what to do with the shared neigbors of the central pair? Whereas cells that are adjacent to only one cell in the central pair can be reasonably taken to be conditionally independent of the other cell in the pair, such an assumption makes little sense for cells that are direct neighbors to both of the cells in the central pair.

The result derived in the previous section provides a seemingly natural answer to this problem, at least for cellular automata involving only two states. Letting x and y denote the cells in the central pair, and z their shared neighbor, let the approximated conditional probability $p_{z|xy}(c|ab)$ be given by $p_T(abc)/p_{pair}(ab)$, where p_{pair} are the known pair probabilities and p_T the maximum entropy triangle probabilities as derived above. The conditional probabilities for the remaining cells in the neighborhood of the central pair, which are adjacent to only one cell in the pair, may be approximated by $p_{w|y}(d|b) = p_{pair}(bd)/p_{single}(b)$ as in the classical pair approximation.

Note that this is essentially a "semiclassical" approximation: it still assumes that the neighbors of the central pair are conditionally independent of each other, even when they're actually themselves neighbors to each other. Thus, it does not constitute a fully automorphism invariant maximum entropy approximation of the extended pair probabilities, being instead a sort of

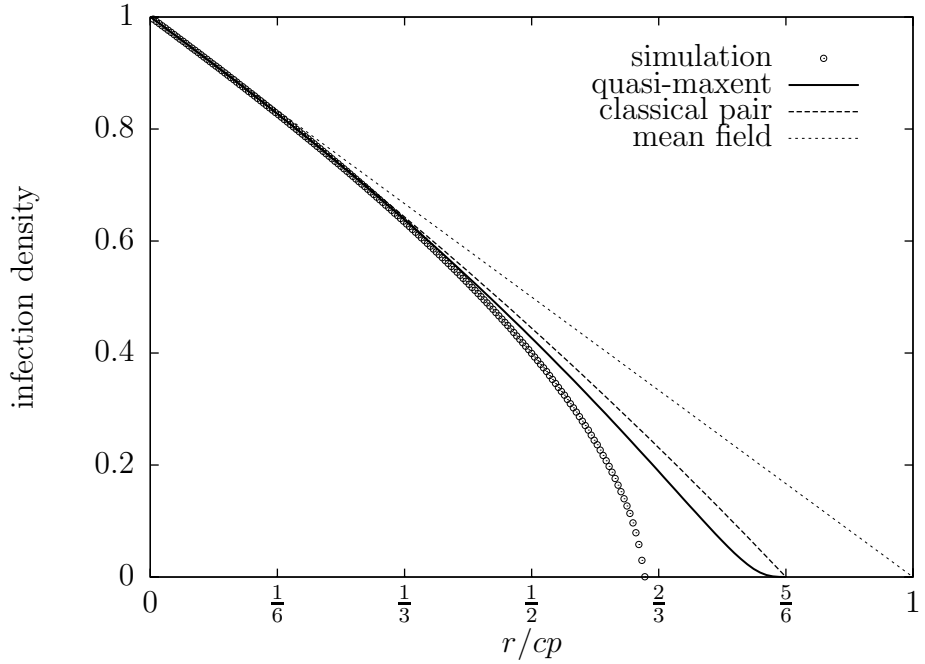

Figure 1: Equilibrium infection density as a function of the scaled recovery rate r/cp in the lattice contact process on a hexagonal lattice, with three approximations compared to simulation results.

a hybrid of the classical and maximum entropy approximations. Depending on the exact lattice geometry, it might be possible extend the approximation to take into account *some* of the inter-neighbor connections by also considering connected triangles containing one cell in the central pair and two of its neighbors; however, for most natural lattice geometries, such as the hexagonal lattice I will consider below, it is not possible to do this for *all* of the connections within the extended cluster without running into the same sort of consistency problems as the ones that arose with the classical pair approximation in the first place.

5.1 Testing with the lattice contact process

Of course, the value of any approximation lies in how well it approximates the actual system being investigated. To test this in practice, I have applied the "quasi-maximum-entropy" pair approximation described above to the spatial contact process (as described in section 1.1) on a uniform hexagonal lattice, and compared it with the classical pair approximation (where shared neigh-

bors are treated as two independent cells) as well as with simulation results.

At least for this particular example, the results are mixed: while using the maximum entropy triangle approximation does bring the approximated equilibrium density closer to the simulation results at intermediate values of r/cp, both approximations still predict the same extinction threshold $r/cp = 5/6$, which significantly exceeds the value $r/cp \approx 0.65$ observed in simulations. Indeed, it turns out that both the classical and quasi-maxent pair approximations yield, for this model, the exact same value $p(1|1) = 1 - r/cp$ for the conditional infection density in the neighborhood of an infected cell. It is thus not surprising that they predict the same extinction threshold, since it is this conditional density which determines the average infection rate in the limit where infected cells are rare.

In fact, in some ways the quasi-maxent approximation turns out to describe the behavior of the actual automaton *worse* than the classical one: whereas the classical pair approximation correctly predicts a discontinuity in the derivative of the equilibrium density as a function of r/cp at the extinction threshold, as observed also in simulations, this discontinuity disappears when the maximum entropy triangle probabilities are incorporated in the approximation. Thus, paradoxically, the quasi-maxent pair approximation appears to be, at least for this particular automaton, quantitatively better but qualitatively worse than the classical one.

6 Further work

The limited success of the consistent quasi-maxent pair approximation derived in section 5 raises rather more questions than it answers. In particular, it would be interesting to construct the full maximum entropy closure for the pair approximation on a hexagonal lattice, involving all the 19 nearest neighbor pairs among the 10 cells in the neighborhood of the central pair, and see whether it gives better results or whether moving to a larger base cluster is necessary for further improvement. Also, it might be worth carrying out further tests on cellular automata for which the classical pair approximation does less well than for the simple lattice contact process, to see if this would increase the difference between the classical and quasi-maxent pair approximations.

Ultimately, the "holy grail" I set out to seek, when I first chose this topic for my thesis, would have been a straightforward procedure for deriving fully consistent maximum-entropy closures for arbitrarily large clusters in more than one dimension. Alas, this has proven to be more difficult than I at first anticipated, and it now appears unlikely to me that such closures could in general be derived analytically for arbitrary cluster sizes. Nonetheless, some hope remains that, even if the analytic approach is doomed to fail, the procedure set out in section 4 could still be successfully applied numerically.

Of course, the question then would be whether such a numerical approach, lacking in analytic insight, would actually be any more useful for investigating the behavior of cellular automata than simply carrying out Monte Carlo simulations on the automaton itself. Yet it is well known that many cellular automata can exhibit very slow convergence and persistently lingering bias, particularly in critical regions of their parameter space. Even a purely numerical approximation could be quite useful, if it allowed one to essentially take a short cut past this slow convergence.

It has now been more than a year and a half since I started work on this thesis. Although there are surely many reason for the slow and sporadic nature of my progress with it, one psychological obstacle that has kept me from simply finishing it and moving on long ago has been the lack of a neat and satisfying conclusion. Alas, it has become apparent to me that none is likely to be forthcoming within a reasonable time, at least not in quite the manner I'd originally envisioned. Thus, I'm forced to conclude this thesis on a somewhat flat note, leaving (to mix my metaphors) several threads unravelled, and simply submit it more or less as it stands—hoping that I may be able to answer some of the questions left open above in later work, and that, though hardly perfect, this thesis, as it is, may nonetheless suffice for its intended purpose.

Symbols and definitions

$\mathbb{N}$	The set of natural numbers: $\{0, 1, 2, 3, \ldots\}$
$\mathbb{Z}$	The set of integers: $\{\ldots, -2, -1, 0, 1, 2, \ldots\}$
$\mathbb{R}$	The set of real numbers.
L	The lattice of a cellular automaton.
$x, y, z \in L$	Individual cells.
$X, Y, Z \subset L$	Clusters of cells.
$\#X$	The number of cells in the cluster X.
$xy := \{x, y\}$	A pair of cells.
$XY := X \cup Y$	The union of two (usually disjoint) clusters.
N_x	The neighborhood of (= the set of cells adjacent to) the cell x, excluding x itself.
$\bar{N}_x := N_x \cup \{x\}$	The neighborhood of the cell x, including x itself.
$N_X := \bar{N}_x \setminus X$	The neighborhood of the cluster X, excluding X itself.
$\bar{N}_X := \bigcup_{x \in X} \bar{N}_x$	The neighborhood of the cluster X, including X itself.
$\sigma = \sigma(t) = \sigma_L(t)$	The state of the entire lattice at time t.
$\sigma_x = \sigma_x(t)$	The state of the cell $x \in L$ at time t.
$\sigma_X = \sigma_X(t)$	The state of the cluster $X \subset L$ at time t.
σ_{xy}	The state of the pair of cells x, y.
σ_{XY}	The state of the combined cluster $XY = X \cup Y \subset L$.
Ω	The set of possible states of a single cell.
Ω_X	The set of possible states of the cluster X.
$\Pr[Q]$	The probability of the event Q.
$\Pr[Q\|R]$	The probability of the event Q given R.

$p_x(a) = p_x(a,t)$ The probability $\Pr[\sigma_x(t) = a]$ of the cell x being in the state a at time t.

$p_X(A) = p_X(A,t)$ The probability $\Pr[\sigma_X(t) = A]$ of the cluster X being in the state A at time t.

$p_{X|Y}(A|B)$ The conditional probability $\Pr[\sigma_X = A|\sigma_Y = B]$ of the cluster X being in state A if the cluster Y is in state B.

$r_x(a \to b|C)$ The rate at which the cell x changes from state a to state b when its neighborhood N_x is in state C. (The subscript may be omitted.)

$r_X(A \to B|C)$ The rate at which the cluster X changes from state A to state B when its neighborhood N_X is in state C. (The subscript may be omitted.)

c Rate at which infected (state 1) cells contact their neighbors in the lattice contact process.

p Probability of infection upon contact between infected (state 1) and susceptible (state 0) cells in the lattice contact process.

r Recovery rate $r(1 \to 0)$ of infected cells in the lattice contact process.

References

[Ada02] Andrew Adamatzky, *Collision-based computing*, Springer, Berlin, 2002.

[Gar70] Martin Gardner, *Mathematical Games: The fantastic combinations of John Conway's new solitaire game "Life"*, Scientific American **223** (1970), 120–123.

[GV87] Howard A. Gutowitz and Jonathan D. Victor, *Local structure theory in more than one dimension*, Complex Systems **1** (1987), 57–68.

[GVK87] Howard A. Gutowitz, Jonathan D. Victor, and Bruce W. Knight, *Local structure theory for cellular automata*, Physica D **28** (1987), 18–48.

[Mol77] Denis Mollison, *Spatial contact models for ecological and epidemic spread*, J. Roy. Stat. Soc. B **39** (1977), 283–326.

[vB00] Minus van Baalen, *Pair approximations for different spatial geometries*, The Geometry of Ecological Interactions: Simplifying Spatial Complexity (Dieckmann, Law, and Metz, eds.), Cambridge University Press, 2000, pp. 359–387.

[Wol02] Stephen Wolfram, *A New Kind of Science*, Wolfram Media, 2002.