Cellular Automata Methods in Mathematical Physics

by

Mark Andrew Smith

B.S. (Physics with Astrophysics Option and Mathematics) and
B.S. (Computer Science with Scientific Programming Applications
Option), New Mexico Institute of Mining and Technology,
Socorro, New Mexico (1982)

Submitted to the Department of Physics
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 1994

Author . . . .
Department of Physics
May 16th, 1994

Certified by . . . .
Tommaso Toffoli
Principal Research Scientist
Thesis Supervisor

Accepted by . . . .
George F. Koster, Professor of Physics
Chairman, Departmental Graduate Committee

Cellular Automata Methods

in Mathematical Physics

by

Mark Andrew Smith

Submitted to the Department of Physics

on May 16th, 1994, in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

Abstract

Cellular automata (CA) are fully discrete, spatially-distributed dynamical systems which can serve as an alternative framework for mathematical descriptions of physical systems. Furthermore, they constitute intrinsically parallel models of computation which can be efficiently realized with special-purpose cellular automata machines. The basic objective of this thesis is to determine techniques for using CA to model physical phenomena and to develop the associated mathematics. Results may take the form of simulations and calculations as well as proofs, and applications are suggested throughout.

We begin by describing the structure, origins, and modeling categories of CA. A general method for incorporating dissipation in a reversible CA rule is suggested by a model of a lattice gas in the presence of an external potential well. Statistical forces are generated by coupling the gas to a low temperature heat bath. The equilibrium state of the coupled system is analyzed using the principle of maximum entropy. Continuous symmetries are important in field theory, whereas CA describe discrete fields. However, a novel CA rule for relativistic diffusion based on a random walk shows how Lorentz invariance can arise in a lattice model. Simple CA models based on the dynamics of abstract atoms are often capable of capturing the universal behaviors of complex systems. Consequently, parallel lattice Monte Carlo simulations of abstract polymers were devised to respect the steric constraints on polymer dynamics. The resulting double space algorithm is very efficient and correctly captures the static and dynamic scaling behavior characteristic of all polymers. Random numbers are important in stochastic computer simulations; for example, those that use the Metropolis algorithm. A technique for tuning random bits is presented to enable efficient utilization of randomness, especially in CA machines. Interesting areas for future CA research include network simulation, long-range forces, and the dynamics of solids. Basic elements of a calculus for CA are proposed including a discrete representation of one-forms and an analog of integration. Eventually, it may be the case that physi-

cists will be able to formulate cellular automata rules in a manner analogous to how they now derive differential equations.

Thesis Supervisor: Tommaso Toffoli

Title: Principal Research Scientist

Acknowledgements

The most remarkable thing about MIT is the people one meets here, and it is with great pleasure that I now have the opportunity to thank those who have helped me, directly or indirectly, to complete my graduate studies with a Ph.D. First, I would like to thank my thesis advisor, Tommaso Toffoli, for taking me as a student and for giving me financial support and the freedom to work on whatever I wanted. I would also like to thank my thesis committee members Professors Edmund Bertschinger and Felix Villars for the time they spent in meetings and reading my handouts. Professor Villars showed a keen interest in understanding my work by asking questions, suggesting references, arranging contacts with other faculty members, and providing detailed feedback on my writing. I deeply appreciate the personal interest he took in me and my scientific development. Professor George Koster, who must have the hardest job in the physics department, helped me with numerous administrative matters.

The Information Mechanics Group in the Laboratory for Computer Science has been a refuge for me, and its members have been a primary source of strength. Special thanks is due to Norman Margolus who has always been eager to help whenever I had a problem. He and Carol Collura have made me feel at home here by treating me like family. Norm also blazed the thesis trail for me and my fellow graduate students in the group: Joe Hrgovčić, Mike Biafore, Milan Shah, David Harnanan, and Raissa D'Souza. Being here gave me the opportunity to interact with parade of visiting scientists: Charles Bennett, Gérard Vichniac, Bastien Chopard, Jeff Yepez, PierLuigi Pierini, Bob Fisch, Attilia Zumpano, Fred Commoner, Vincenzo D'Andrea, Leonid Khalfin, Asher Perez, Andreas Califano, Luca de Alfaro, and Pablo Tamayo. I also had the chance to get to know several bright undergraduate students who worked for the group including Rebecca Frankel, Ruben Agin, Jason Quick, Sasha Wood, Conan Dailey, Jan Maessen, and Debbie Fuchs. Finally, I was able to learn from an eclectic group of engineering staff: Tom Cloney, David Zaig, Tom Durgavich, Ken Streeter, Doug Faust, and Harris Gilliam. My gratitude goes out to the entire group for making my time here interesting and enjoyable as well as educational.

I would like to express my appreciation to several members of the support staff: Peggy Berkovitz, Barbara Lobbregt, and Kim Wainwright in physics; Be Hubbard, Joanne Talbot, Anna Pham, David Jones, William Ang, Nick Papadakis, and Scott Blomquist in LCS. Besides generally improving the quality of life, they are the ones who are really responsible for running things and have helped me in countless ways.

I am indebted to Professor Yaneer Bar-Yam of Boston University for giving me some exposure to the greater physics community. He was the driving force behind the work on polymers (chapter 5) which led to several papers and conference presentations. The collaboration also gave me the opportunity to work with Yitzhak Rabin, a polymer theorist from Israel, as well as with Boris Ostrovsky and others in the polymer center at Boston University. Yaneer has also shown me uncommon strength and kindness which I can only hope to pick up.

My first years at MIT were ones of great academic isolation, and I would have dropped out long ago if it were not for the fellowship of friends that I got to know at the Thursday night coffee hour in Ashdown House—it was an hour that would often go on for many. The original gang from my first term included Eugene Gath, John Baez, Monty McGovern, Bob Holt, Brian Oki, Robin Vaughan, Glen Kissel, Richard Sproat, and Dan Heinzen. Later years included David Stanley, Arnout Eikeboom, Rich Koch, Vipul Bhushan, Erik Meyer, and most recently, Nate Osgood and Lily Lee from outside Ashdown. I thank you for the intellectual sustenance and fond memories you have given me. Coffee hour was initiated by housemasters Bob and Carol Hulsizer, and I have probably logged 2000 hours in the dining room that now bears their name. Many thanks to them and the new housemasters, Beth and Vernon Ingram, for holding this and other social activities which have made Ashdown House such an enjoyable place to live.

Finally, I owe my deepest gratitude to my family for their continual love, support, and encouragement. I dedicate this thesis to them.

Support was provided in part by the National Science Foundation, grant no. 8618002-IRI, in part by DARPA, grant no. N00014-89-J-1988, and in part by ARPA, grant no. N00014-93-1-0660.

To Mom, Dad, Sarah, David, and Pamela

Contents

Chapter 1

Cellular Automata Methods in Mathematical Physics

The objective of this thesis is to explore and develop the potential of cellular automata as mathematical tools for physical modeling. Cellular automata (CA) have a number of features that make them attractive for simulating and studying physical processes. In addition to having computational advantages, they also offer a precise framework for mathematical analysis. The development proceeds by constructing and then analyzing CA models that resemble a particular physical situation or that illustrate some physical principle. This in turn leads to a variety of subjects of interest for mathematical physics. The paragraphs below serve as a preview of the major points of this work.

The term cellular automata refers to an intrinsically parallel model of computation that consists of a regular latticework of identical processors that compute in lockstep while exchanging information with nearby processors. Chapter 2 discusses CA in more detail, gives some examples, and reviews some of the ways they have been used. Cellular automata started as mathematical constructs which were not necessarily intended to run on actual computers. However, CA can be efficiently implemented as computer hardware in the form of cellular automata machines, and these machines have sparked a renewed interest in CA as modeling tools. Unfortunately, much of the resulting development in this direction has been limited to "demos" of potential application areas, and many of the resulting models have been left wanting supporting analysis. This thesis is an effort to improve the situation by putting the field on a more methodical, mathematical tack.

A problem that comes up in many contexts when designing CA rules for physical modeling is the question of how to introduce forces among the constituent parts of a system under consideration. This problem is exacerbated if one wants to obtain an added measure of faithfulness to real physical laws by making the dynamics reversible. The model presented in chapter 3 gives a technique for generating statistical forces which are derived from an external potential energy function. In the process, it clarifies the how and why of dissipation in the face of microscopic reversibility. The chapter also serves as a paradigm for modeling in statistical mechanics and thermodynamics using cellular automata machines and shows how one has control over all aspects of a problem—from conception and implementation, through theory and experiment, to application and generalization. Finally, the compact representation of the potential energy function that is used gives one way to represent a physical field and brings up the general problem of representing arbitrary fields.

Many field theories are characterized by symmetries under one or more Lie groups, and it is a problem to reconcile the continuum with the discrete nature of CA. At the finest level, a CA *cannot* reflect the continuous symmetries of a conventional physical law because of the digital degrees of freedom and the inhomogeneous, anisotropic nature of a lattice. Therefore, the desired symmetry must either arise as a suitable limit of a simple process or emerge as a collective behavior of an underlying complex dynamics. The maximum speed of propagation of information in CA suggests a connection with relativity, and chapter 4 shows how a CA model of diffusion can display Lorentz invariance despite having a preferred frame of reference. Kinetic theory can be used to derive the properties of lattice gases, and a basic Boltzmann transport argument leads to a manifestly covariant differential equation for the limiting continuum case. The exact solution of this differential equation compares favorably with a CA simulation, and other interesting mathematical properties can be derived as well.

One of the most important areas of application of high-performance computation is that of molecular dynamics, and polymers are arguably the most important kinds of molecules. However, it has proven difficult to devise a Monte Carlo updating scheme that can be run on a parallel computer, while at the same time, CA are *inherently* parallel. Hence, it is desirable to find CA algorithms for simulating polymers, and this is the starting point for chapter 5. The resulting parallel lattice Monte Carlo polymer dynamics should be of widespread interest in fields ranging from biology and chemical engineering to condensed matter theory. In addition to the theory of scaling, there are a number of mathematical topics which are related to this area such as the study of the geometry and topology of structures in complex fluids. Another is the generation and utilization of random numbers which is important, for example, in the control of chemical reaction rates. Many modifications to the basic method are possible and a model of pulsed field gel electrophoresis is given as an application.

The results obtained herein still represent the early stages of development of CA as general-purpose mathematical modeling tools, and chapter 6 discusses the future prospects along with some outstanding problems. Cellular automata may be applied to any system that has an extended space and that can be given a dynamics, and this includes a very broad class of problems indeed! Each problem requires significant creativity, and the search for solutions will inevitably generate new mathematical concepts and techniques of interest to physicists. Additional topics suggested here are the simulation of computer networks as well as the problems of long-range forces and the dynamics of solids. Besides research on specific basic and applied problems, many mathematical questions remain along with supporting work to be done on hardware and software.

The bulk of this document is devoted to describing a variety of results related to modeling with CA, while chapter 7 summarizes the main contributions and draws some conclusions as sketched below. The appendices are effectively supplementary chapters containing details which are somewhat outside the main line of development, although they should be considered an integral part of the work. They primarily consist of calculations and proofs, but the accompanying discussions are important as well.

Inventing CA models serves to answer the question of what CA are naturally capable of doing and how, but it is during the process of development that the best—the unanticipated—discoveries are made. The simplicity of CA means that one is forced to capture essential phenomenology in abstract form, and this makes for an interesting way to find out what features of a system are really important. In the course of presenting a number of modeling techniques, this thesis exemplifies a methodology for generating new topics in mathematical physics. And in addition to giving results on previously unsolved problems, it identifies related research problems which range in difficulty from straightforward to hard to impossible. This in turn encourages continued progress in the field. In conclusion, CA have a great deal of unused potential as mathematical modeling tools for physics; furthermore, they will see more and more applications in the study of complex dynamical systems across many disciplines.

Chapter 2

Cellular Automata as Models of Nature

This chapter introduces cellular automata (CA) and reviews some of the ways they have been used to model nature. The first section describes the basic format of CA, along with some possible variations. Next, the origins and history of CA are briefly summarized, continuing through to current trends. The last section gives a series of prior applications which illustrate a wide range of phenomena, techniques, concepts, and principles. The examples are organized into categories, and special features are pointed out for each model.

2.1 The Structure of Cellular Automata

Cellular automata can be described in several ways. The description which is perhaps most useful for physics is to think of a CA as an entirely discrete version of a physical field. Space, time, field variables, and even the dynamical laws can be completely formulated in terms of operations on a finite set of symbols. The points (or *cells*) of the space consist of the vertices of a regular, finite-dimensional lattice which may extend to infinity, though in practice, periodic boundary conditions are often assumed. Time progresses in finite steps and is the same at all points in space. Each point has dynamical state variables which range over a finite number of values. The time evolution of each variable is governed by a local, deterministic dynamical law (usually called a *rule*): the value of a cell at the next time step depends on the current state of a finite number of "nearby" cells called the *neighborhood*. Finally, the rule acts on all points simultaneously in parallel and is the same throughout space for all times.

Another way to describe CA is to build on the mathematical terminology of dynamical systems theory and give a more formal definition in terms of sets and mappings between them. For example, the state of an infinite two-dimensional CA on a Cartesian lattice can be defined as a function, $S : Z \times Z \rightarrow s$, where Z denotes the set of integers, and s is a finite set of cell states. Mathematical definitions can be used to capture with precision the meaning of intuitive terms such as "space" and "rule" as well as important properties such as symmetry and reversibility. While such formulations are useful for giving rigorous proofs of things like ergodicity and equivalences between definitions, they will not be used here in favor of the physical notion of CA. Formalization of physical results can always be carried out later by those so inclined.

The final way to describe CA is within their original context of the theory of (automated) computation. Theoretical computing devices can be grouped into equivalence classes known as *models of computation*, and the elements of these classes are called *automata*. Perhaps the simplest kind of automaton one can consider is a finite state machine or *finite automaton*. Finite automata have inputs, outputs, a finite amount of state (or memory), and feedback from output to input; furthermore, the state changes in some well-regulated way, such as in response to a clock (see figure 2-1). The output depends on the current state, and the next state depends on the inputs as well as the current state.[1] A *cellular automaton*, then, is a regular array of identical finite automata whose inputs are taken from the outputs of neighboring automata. CA have the advantage over other models of computation in that they are parallel, much as is the physical world. In this sense, they are better than serial models for describing processes which have independent but simultaneous activities occurring throughout a physical space.

[1]Virtually any physical device one wishes to consider (e.g., take any appliance) can be viewed as having a similar structure: inputs, outputs, and internal state. Computers are special in that these

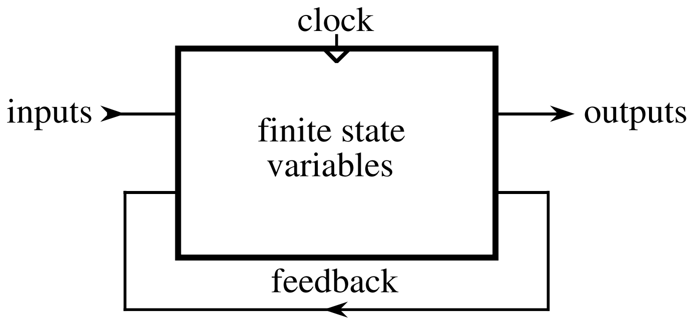


Figure 2-1: A block diagram of a general finite automaton. The finite state variables change on the ticks of the clock in response to the inputs. The state of the automaton determines the outputs, some of which are fed back around to the input. Automata having this format constitute the cells of a cellular automaton.

New cellular models of computation may be obtained by relaxing the constraints on the basic CA format described above. Many variations are possible, and indeed, such variations are useful in designing CA for specific modeling purposes. For example, one of the simplest ways to generalize the CA paradigm would be to allow different rules at each cell. Newcomers to CA often want to replace the finite state variables in each cell with variables that may require an unbounded or infinite amount of information to represent exactly, e.g., true integers or real numbers. Continuous variables would also open up the possibility of continuous time evolution in terms of differential equations. While allowing arbitrary field variables may be desirable for the most direct attempts to model physics, it violates the "finitary" spirit of CA—and is impossible to achieve in an actual digital simulation in any case. In practice, the most common CA variation utilizes temporally periodic changes in the dynamical rule or in the lattice. In these cases, it is often useful to depict the system with a spacetime diagram (see figure 2-2), where time increases downward for typographical reasons. Such cyclic rules are useful for making a single complex rule out of several simpler ones, which in turn is advantageous for both conceptual and computational reasons. Another generalization that turns out to be very powerful is to allow nondeterministic or stochastic dynamics. This is usually done by introducing random variables as

three media consist, in essence, of pure information.

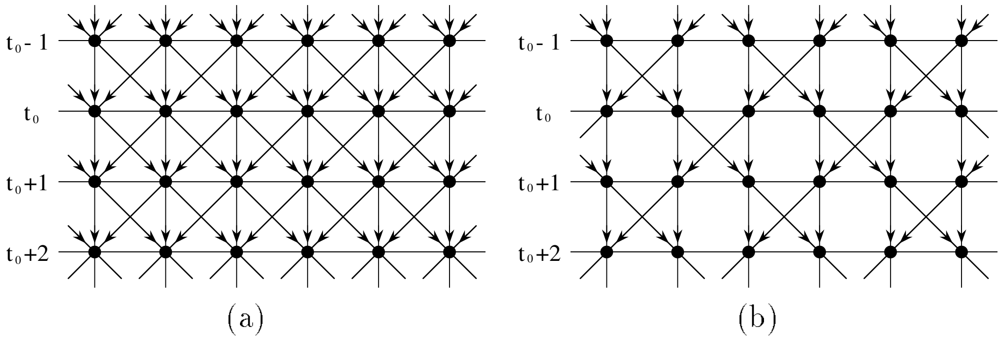


Figure 2-2: Spacetime schematics of one-dimensional CA showing six cells over four time steps. Arrows indicate causal links in the dynamics of each automaton. (a) Completely uniform CA format wherein the dynamical rule for any cell depends on itself and its nearest neighbors. (b) A partitioning CA format wherein the rule for the cells varies in time and space. Alternating pairs of cells only depend on the pair itself.

inputs to the transition function, but could also be accomplished by letting a random interval of time pass between each update of a cell.

It turns out that some of the variations given above do not, in fact, yield new models of computation because they can be embedded (or simulated) in the usual CA format by defining a suitable rule and possibly requiring special initial conditions. In the case of rules that vary from site to site, it is a simple matter to add an extra variable to each cell that specifies which of several rules to use. If these secondary variables have a cyclic dynamics, they effectively implement periodic changes in the primary rule. These techniques can be combined to generate any regular spacetime pattern in the lattice or in the CA rule. A cyclic rule can also be embedded without using extra variables by composing the individual simple steps into a single step of a more complex CA. Similarly, at any given moment in time, spatially periodic structures in the lattice can be composed into unit cells of an ordinary Cartesian lattice. These spatial and temporal patterns can be grouped into unit cells in spacetime in order to make the rule and the lattice time-invariant (see figure 2-3). By adding an extra dimension to a CA it would be even be possible to simulate true integer-valued states, though the simulation time would increase as the lengths of the integers grew, and consequently, the integer variables would no longer evolve simultaneously. Finally, real-valued and stochastic CA can be simulated approximately by using floating point numbers and pseudorandom number generators respectively.

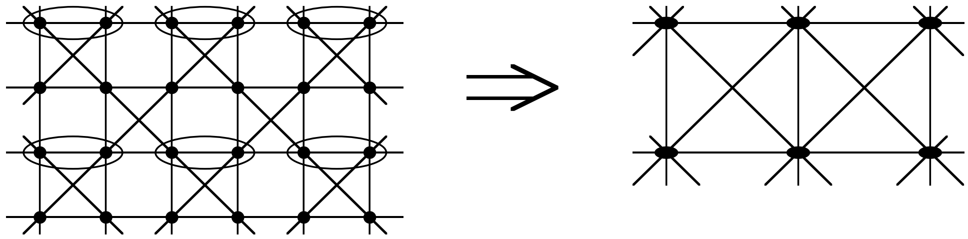


Figure 2-3: Spacetime diagrams showing how a partitioning CA format (left) can be embedded in a uniform CA format (right). The embedding is accomplished by grouping pairs of cells into single cells and composing pairs of time steps into single time steps.

The equivalences described here are important because they form the beginnings of CA modeling techniques.

The concepts introduced above can be illustrated with a simple, familiar example. Figure 2-4 shows a portion of a spacetime diagram of a one-dimensional CA that generates Pascal's triangle. Here the cells are shown as boxes, and the number in a cell denotes its state. Successive rows show individual time steps and illustrate how a lattice may alternate between steps. All the cells are initialized to 0, except for a 1 in a single cell. The lattice boundaries (not shown) could be periodic, fixed, or extended to infinity. The dynamical rule adds the numbers in two adjacent cells to create the value in the new intervening cell on the next time step. The binomial coefficients are thus automatically generated in a purely local, uniform, and deterministic manner. In a suitable large-scale limit, the values in the cells approach a Gaussian distribution. The pattern resembles the propagator for the one-dimensional diffusion equation and could form the basis of a physical model. Unfortunately, the growth is exponential, and as it stands, the rule cannot be carried out indefinitely because the cells have a limited amount of state. The description of the rule must be modified to give only allowed states, depending on the application. One possibility would be to have the cells saturate at some maximum value, but in figure 2-4, the rule has been modified to be addition modulo 100. This is apparent in the last two rows, where the rule merely drops the high digits and keeps the last two. This dynamics is an example of a *linear* rule which means that any two solutions can be superimposed by addition

0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	1	1	0	0	0	0	
0	0	0	0	1	2	1	0	0	0	0
0	0	0	1	3	3	1	0	0	0	
0	0	0	1	4	6	4	1	0	0	0
0	0	1	5	10	10	5	1	0	0	
0	0	1	6	15	20	15	6	1	0	0
0	1	7	21	35	35	21	7	1	0	
0	1	8	28	56	70	56	28	8	1	0
1	9	36	84	26	26	84	36	9	1	
1	10	45	20	10	52	10	20	45	10	1

Figure 2-4: A spacetime diagram showing the cells of a one-dimensional CA that generates Pascal's triangle. The automaton starts out with a single 1 in a background of 0's, and thereafter, each cell is the sum (mod 100) of the two cells above it. The last two rows show the effect of truncating the high digits.

(mod 100) to give another solution. The only other rules in this thesis that are truly linear are those that also show diffusive behavior, though a few nonlinear rules will exhibit some semblance of linearity.

2.2 A Brief History of Cellular Automata

John von Neumann was a central figure in the theory and development of automated computing machines, so it is not surprising that he did the earliest work on CA as such. Originally, CA were introduced around 1950 (at the suggestion of Slanislaw Ulam) in order to provide a simple model of self-reproducing automata [94]. The successful (and profound) aim of this research was to show that certain essential features of biology can be captured in computational form. Much of von Neumann's work was completed and extended by Burks [9]. This line of CA research was followed through the 60's by related studies on construction, adaptation, and optimization as well as by investigations on the purely mathematical properties of CA.

A burst of CA activity occurred in the 70's with the introduction of John Conway's game of "life" [33, 34]. *Life* was motivated as a simple model of an ecology containing cells which live and die according to a few simple rules. This most familiar example of a CA displays rich patterns of activity and is capable of supporting many intricate structures (see figure 2-7). In addition to its delightful behavior, the popularity of this model was driven in part by the increasing availability of computers, but the computers of the day fell well short of what was to come.

The next wave of CA research—and the most relevant one for purposes of this thesis—was the application of CA to physics, and in particular, showing how essential features of physics can be captured in computational form. Interest in this subject was spawned largely by Tommaso Toffoli [84] and Edward Fredkin. Steven Wolfram was responsible for capturing the wider interest of the physics community with a series of papers in the 80's [99], while others were applying CA to a variety of problems in other fields [27, 38]. An important technological development during this time was the introduction of special hardware in the form of cellular automata machines [89] and massively parallel computers. These investigations set the stage for the development of lattice gases [32, 55], which have become a separate area of research in themselves [24, 25].

In addition to lattice gases, one of the largest areas of current interest in CA involves studies in complexity [66, 97]. CA constitute a powerful paradigm for pattern formation and self-organization, and nowhere is this more pronounced than in the study of artificial life [52, 53]. Given the promising excursions into physics, mathematics, and computer science, it is interesting that CA research should keep coming back to the topic of life. No doubt this is partly because of CA's visually captivating, "lifelike" behavior, but also because of peoples' interest in "playing God" by simulating life processes.

2.3 A Taxonomy of Cellular Automata

This section presents a series of exemplary CA rules which serve as a primer on the use and capabilities of CA as modeling tools.[2] They are intended to build up the reader's intuition for the behavior of CA, and to provide a framework for thinking about modeling with CA. No doubt he or she will start to have questions and ideas of his or her own upon seeing these examples. Each rule will only be described to the extent needed for understanding the basic ideas and significant features of the model. Most of the examples are discussed in greater depth in reference [89].

The examples have been grouped into seven modeling categories which cover a range of applications, and each section illustrates a definite dynamical theme. Perhaps a more rigorous way to classify CA rules is by what type of neighborhood they use and whether they are reversible or irreversible. It also turns out that the aforementioned modeling categories are partially resolved by this scheme. CA rules that have a standard neighborhood format (i.e., the same for every cell in space) are useful for models in which activity in one cell excites or inhibits activity in a neighboring cell, whereas partitioning neighborhood formats are useful for models that have tokens that move as conserved particles. Reversible rules try to represent nature at a "fundamental" level, whereas irreversible rules try to represent nature at a coarser level. These distinctions should become apparent in the sections below.

All of the figures below show CA configurations taken from actual simulations performed on CAM-6, a programmable cellular automata machine developed by the Information Mechanics Group [59, 60]. Each configuration lies on a 256×256 grid with periodic boundary conditions, and the values of the cells are indicated by various shades of gray (0 is shown as white, and higher numbers are progressively darker). CAM-6 updates this state space 60 times per second while generating a real-time video display of the dynamics. Unfortunately it isn't possible watch the dynamics in

[2] All of these rules were either invented or extended by members of the Information Mechanics Group in the MIT Laboratory for Computer Science. This group has played a central role in the recent spread of interest in CA as modeling tools and in the development of cellular automata machines.

print, so two subsequent frames are often shown to give a feeling for how the systems evolve. A pair of frames may alternatively represent different rule parameters or initial conditions. The CA in question are all two dimensional unless otherwise noted.

2.3.1 Chaotic Rules

Many CA rules display a seemingly random behavior which may be regarded as chaotic. In fact, almost any rule chosen at random falls into this category. A consequence of this fact is that in order to model some particular natural phenomenon, one must adopt some methods or learning strategies to explicitly *design* CA rules, since most rules will not be of any special interest to the modeler. One design technique that is used over and over again is to run two or more rules in parallel spaces and couple them together. This technique in turn makes it possible to put chaotic rules to good use after all since they can be used to provide a noise source to another rule, though the quality of the resulting random numbers is a matter for further research.

The rules described in this section use a standard CA format called the *von Neumann* neighborhood which can be pictured as follows: . The von Neumann neighborhood of the cell marked with the dot includes itself and its nearest neighbors. Note that the neighborhoods of nearby cells will overlap. Given a neighborhood, a CA rule can be specified in a tabular form called a *lookup table* which lists the new value of a cell for every possible assignment of the current values in its neighborhood. Thus, the number of entries in a lookup table must be k^n, where k is the number of states per cell, and n is the number of cells in the neighborhood.

Figure 2-5 shows the behavior of a typical (i.e., random) rule that uses the von Neumann neighborhood ($n = 5$) and has one bit per cell ($k = 2$). The lookup table therefore consists of $2^5 = 32$ bits. For this example, the table was created by generating a 32-bit random number: 2209261910 (written here in base 10). The initial condition is a 96×96 square of 50% randomness in which each cell has been independently initialized to 1 with a probability of 50% (0 otherwise). The resulting

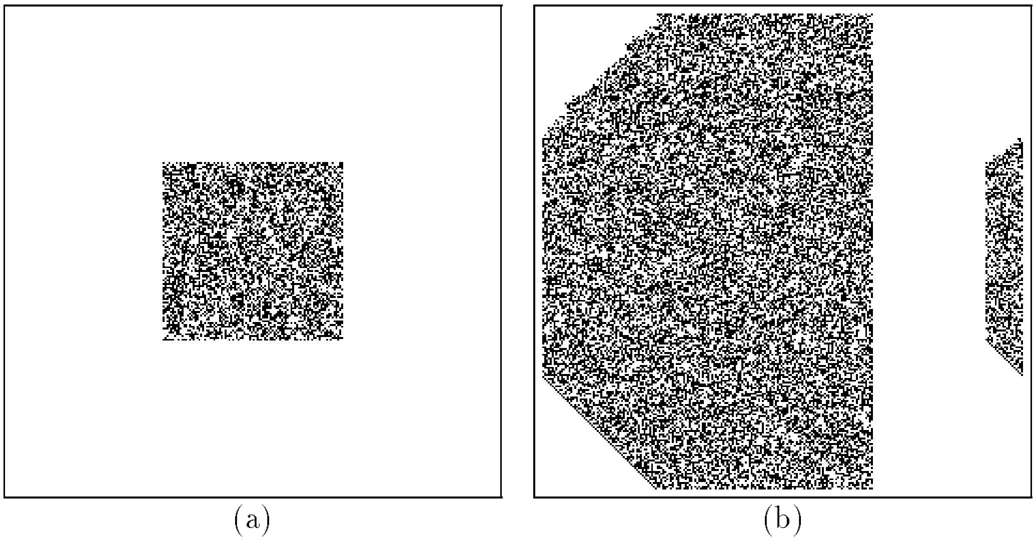


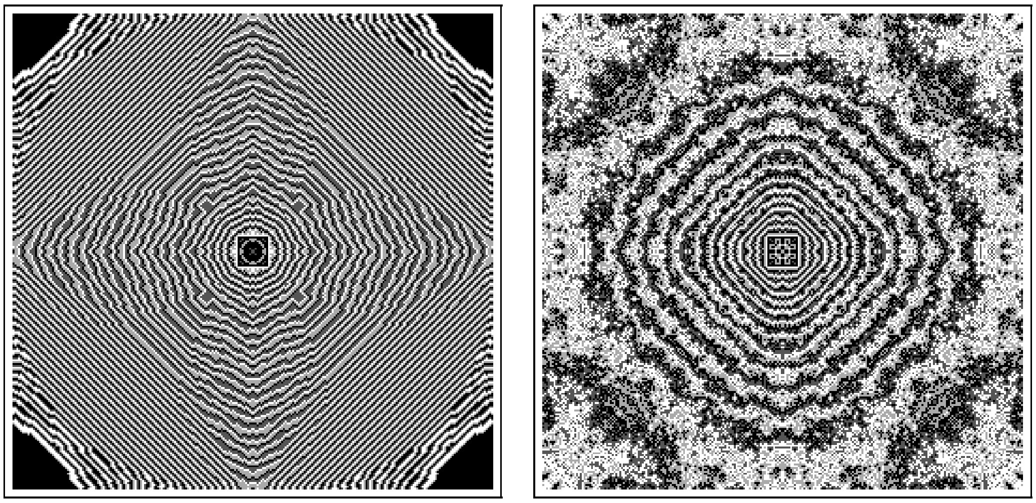
Figure 2-5: Typical behavior of a rule whose lookup table has been chosen at random. Starting from a block of randomness (a), the static spreads at almost the maximum possible speed (b).

disturbance expands at nearly the speed of light,[3] and after 100 steps, it can be seen wrapping around the space. Note that the disturbance does not propagate to the right, which means that the rule doesn't depend on the left neighbor if there is a background of 0's. Furthermore, the interior of the expanding disturbance appears to be just as random as the initial square. As a final comment, it can be verified by finding a state with two predecessors that this CA is indeed irreversible, as are most CA.

Reversible rules can also display chaotic behavior, and in some sense, they must. The initial state of a system constitutes a certain amount of disorder, and since the dynamics is reversible, the information about this initial state must be "remembered" throughout the evolution. However, if the dynamics is nontrivial and there are not too many conservation laws, this information will get folded in with itself many times and the apparent disorder will increase. In other words, the *entropy* will increase, and there is a "second law of thermodynamics" at work.[4] Figure 2-6 shows how

[3]The causal nature of CA implies that a cell can only be effected by perturbations in its neighborhood on a single time step. The speed of light is a term commonly used to refer to the corresponding maximum speed of information flow.

[4]The notion of entropy being used here will be discussed in more detail in the next chapter.



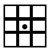
(a) (b)

Figure 2-6: Typical behavior of a simple, symmetrical, reversible rule. Starting from a small, isolated, solid block (not shown), waves propagate outward and interfere with each other nonlinearly as they wrap around the space several times (a). After twice the time, the waves continue to become noiser and longer in wavelength (b).

this folding takes place and how the disorder typically increases in the context of a simple, nonlinear, reversible (second-order) rule. Initially, the system contained a solid 16×16 square (whose outline is still visible) in an empty background: obviously, a highly ordered state. Waves start to emanate from the central block, and after 1200 steps, the nonlinear interactions have created several noisy patches. After another 1200 steps, the picture appears even noiser, though certain features persist. The resulting bands of gray shift outward every step, and constitute superluminal phase waves (or beats) stemming from correlated information which has spread throughout the system.

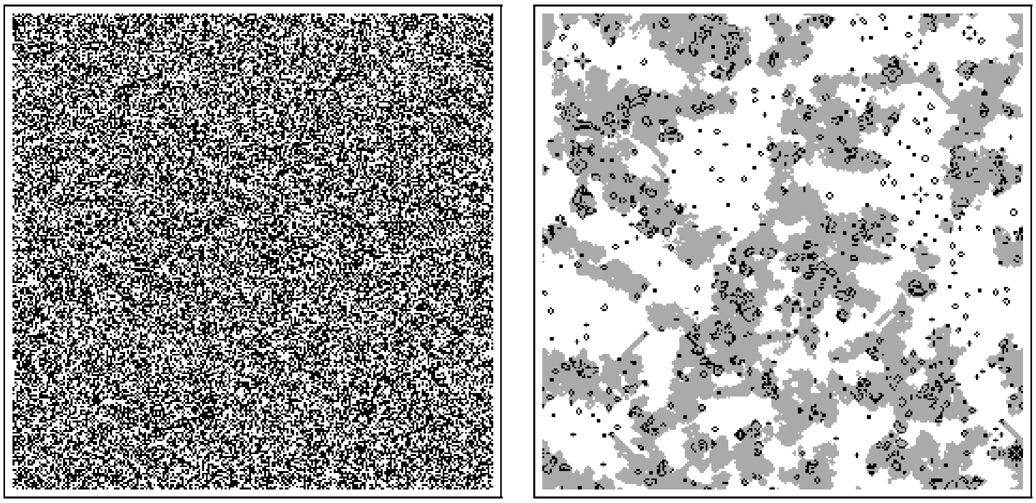
2.3.2 Voting Rules

The dynamics of many CA can be described as a *voting* process where a cell tends to take on the values of its neighbors, or in some circumstances, opposing values [93]. Voting rules are characteristically irreversible because the information in a cell is often erased without contributing to another cell, and irreversibility is equivalent to destroying information. In practice, it is usually pretty clear when a rule is irreversible because frozen or organized states appear when there were none to begin with, and this means there must have been a many-to-one mapping of states. As we shall see, voting rules are examples of *pattern-forming* rules.

The first two examples in this section use a standard CA format called the *Moore* neighborhood which consists of a cell along with its nearest and next nearest neighbors: . As before, the dot marks the cell which depends on the neighborhood, and the neighborhoods of nearby cells will overlap. The third example uses the other standard format, the von Neumann neighborhood.

We begin our survey of CA models with the most well-known example which is Conway's game of life. It was originally motivated as an abstract model of birth and death processes and is very reminiscent of cells in a Petri dish. While not strictly a voting rule, it is similar in that it depends on the total count of living cells surrounding a site (here, 1 indicates a living cell, and 0 indicates a dead cell or empty site). If an empty site has exactly 3 living neighbors, there is a birth. If a living cell has fewer than 2 or more than 4 living neighbors, it dies of loneliness or overcrowding respectively. All other cases remain unchanged. Figure 2-7(a) shows the default CAM-6 pattern of 50% randomness which is used as a standard in many of the examples below. Starting from this initial condition for 300 steps (with a trace of the activity taken for the last 100 steps) yields figure 2-7(b). The result is a number of stable structures, oscillating "blinkers," propagating "gliders," and still other areas churning with activity. It has been shown that this rule is already sufficiently complex to simulate *any* digital computation [4].

Figure 2-8 shows the behavior of some true voting rules with simple yes (1)/no (0) voting. Pattern (a) is the result of a simulation where each cell votes to go along with a simple majority (5 out of 9) of its neighbors. Starting from the standard random configuration and running for 35 steps gives a stable pattern of voters. At this point the rule is modified so that the outcome is inverted in the case of 4 or 5 votes. This has the effect of destabilizing the boundaries and effectively generating a surface tension. Figure 2-8(b) compares the system 200 and 400 steps after the rule is changed and shows how the boundaries anneal according to their curvature. The self-similar scaling

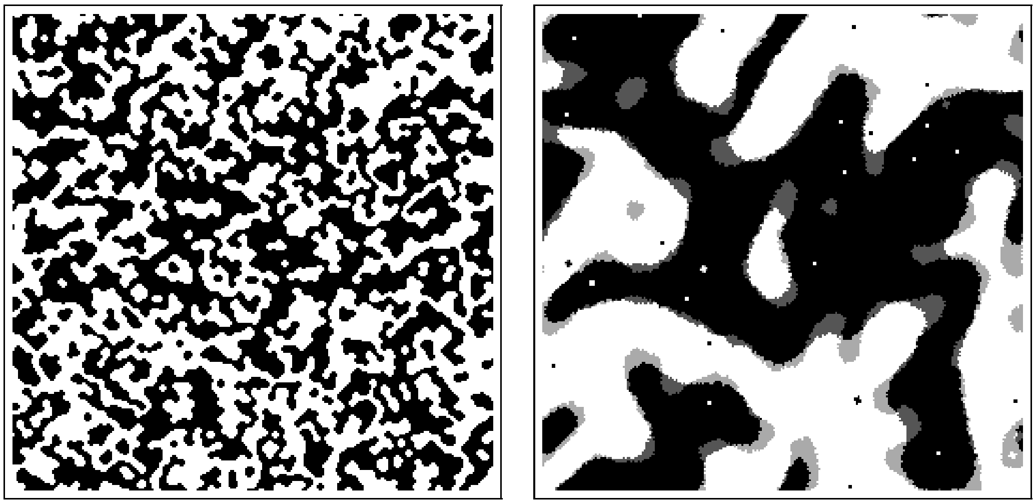


(a) (b)

Figure 2-7: Conway's game of life. The initial condition (a) is the standard pattern of randomness used in many of the examples below. After a few hundred steps (b), complex structures have spontaneously formed. The shaded areas indicate regions of the most recent activity.

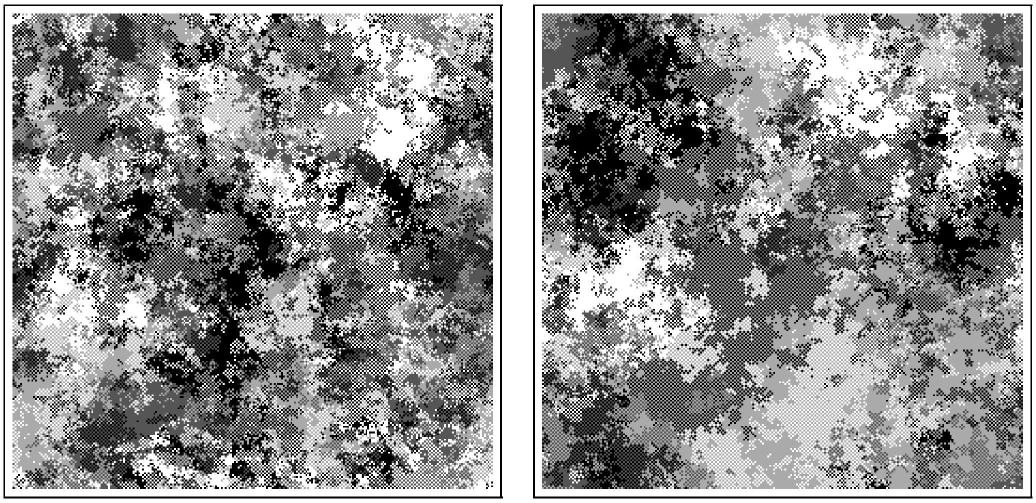
of the patterns is evident.

The final example of a voting rule is also our first example of a stochastic CA. The rule is for each cell to take on the value of one of its four nearest neighbors at random. Two bits from a chaotic CA (not shown) are used to pick the direction, and two bits of state specify one of four values in a cell. Since the new value of a cell always comes from a neighbor, the system actually decouples into two separate systems in a checkerboard fashion. Figure 2-9 shows how this dynamics evolves from a standard random pattern. After 1000 steps, the clustering of cell values is well underway, and after 10000 steps, even larger domains have developed. The domains typically grow according to power laws, $\xi \sim t^{\alpha/2}$ where $\alpha \leq 1$, and in an infinite space, they would eventually become distributed over all size scales [18]. This rule has also been suggested as a model of genetic drift, where each cell represents an individual, and the value in the cell represents one of four possible genotypes [50].



(a) (b)

Figure 2-8: Deterministic voting rules. Starting from the standard random pattern, simple majority voting quickly leads to stable voting blocks (a). By having the voters vacillate in close elections, the boundaries are destabilized and contract as if under surface tension (b). The shaded areas indicate former boundaries.



(a) (b)

Figure 2-9: A random voting rule with four candidates. Starting from a random mix of the four possible cell values, each cell votes to go along with one of its four nearest neighbors at random. After 1000 rounds of voting (a), distinct voting blocks are visible. After 10000 rounds (b), some regions have grown at the expense of others.

2.3.3 Reversible Rules

This section discusses a class of reversible CA that are based on *second order* rules, i.e., the next state depends on two consecutive time steps instead of only one. They use standard neighborhood formats, but the dynamical laws are all of the form $s_{t+1} = f(\{s\}_t) - s_{t-1}$, where s_t denotes the state of a cell at time t, $f(\{s\}_t)$ is any function of the cells in its neighborhood at time t, and subtraction is taken modulo an integer. Clearly this is reversible because $s_{t-1} = f(\{s_t\}) - s_{t+1}$ (in fact, it is time-reversal invariant). The example of a reversible chaotic rule in section 2.3.1 was also of this form, but the examples given here are more well-behaved due to the presence of special conservation laws. Another class of reversible rules based on partitioning neighborhoods will be discussed in the next section.

Figure 2-10 shows the behavior of a one-dimensional, second-order reversible rule with a neighborhood five cells wide. This particular rule has a (positive) conserved energy which is given by the number of differences between the current and past values of neighboring cells. The dynamics supports a wide variety of propagating structures which spread from regions of disorder into regions of order. Any deterministic CA on a finite space must eventually enter a cycle because it will eventually run out of new states. If in addition the rule is reversible, the system must eventually return to its initial state because each state can only have one predecessor. Usually the recurrence time is astronomical because there are so many variables and few conservation laws, but for the specific system shown in figure 2-10, the period is a mere 40926 steps.

Cellular automata are well-suited for doing dynamical simulations of Ising models [19, 20, 93], where each cell contains one spin (1 = up, -1 = down). Many variations are possible, and figure 2-11 shows one such model that is a reversible, second-order CA which conserves the usual Ising Hamiltonian,

$$H = -J \sum_{<i,j>} s_i s_j, \tag{2.1}$$

where J is a coupling constant, and $< \cdots >$ indicates a sum over nearest neighbors. The second-order dynamics uses a so-called "checkerboard" updating scheme, where

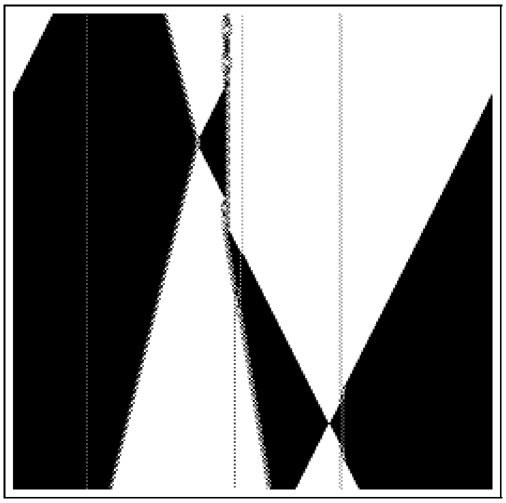


Figure 2-10: A spacetime diagram of a one-dimensional, second-order rule in which conservation laws keep the system from exploding into chaos. Several kinds of propagating "particles," "bound states," and interactions are apparent.

the the black and red squares represent even and odd time steps respectively, so only half of the cells can change on any one step. The easiest way to describe the rule on one sublattice is to say that a spin is flipped if and only if it conserves energy, i.e., it has exactly two neighbors with spin up and two with spin down. The initial condition is a pattern of 8% randomness, which makes the value of H near the critical value. The spins can only start flipping where two dots are close enough to be in the von Neumann neighborhood of another cell, but after 10000 steps, fairly large patches of the minority phase have evolved. A separate irreversible dynamics records a history of the evolution (shown as shaded areas).

The energy in the Ising model above resides entirely in the boundaries between the phases, and it flows as a locally conserved quantity. It is not surprising then, that it is possible to describe the dynamics in terms of an equivalent (nonlinear) dynamics on the corresponding bond-energy variables. These energy variables form closed loops around the magnetic domains, and as long as the loops don't touch—and thereby interact nonlinearly—they happen to obey the one-dimensional wave equation in the plane of the CA. This wave behavior is captured in a modified form by the second-order, reversible CA in figure 2-12. The initial condition consists of of several normal

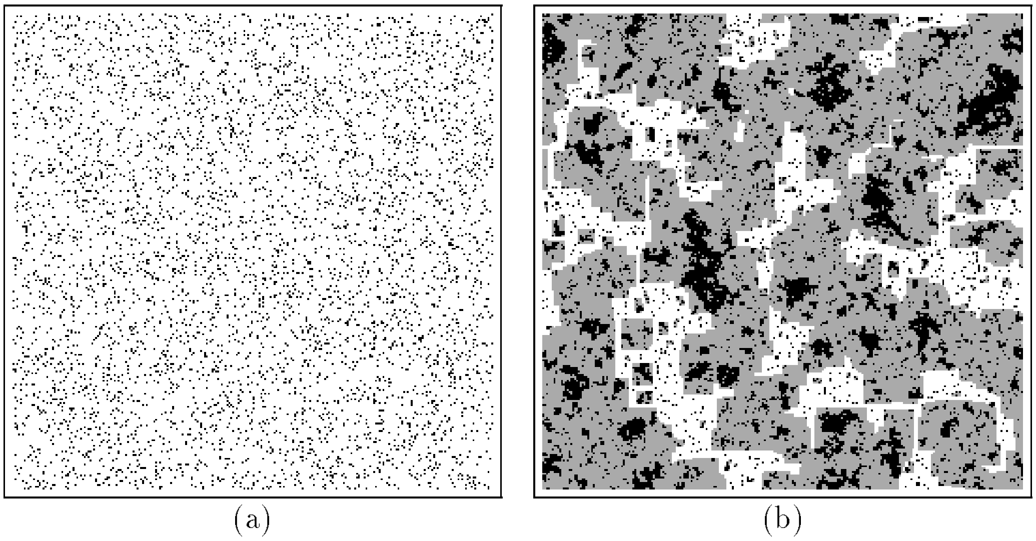
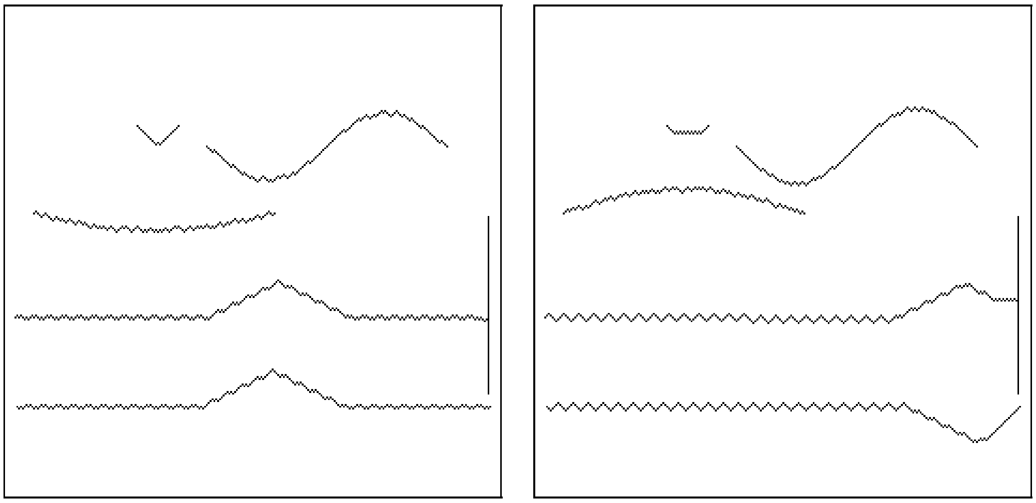


Figure 2-11: A reversible, microcanonical simulation of an Ising model. The initial state has 8% of the spins up at random giving the system a near-critical energy (a). After roughly one relaxation time, the spins have aligned to form some fairly large magnetic domains (b). The shaded areas mark all the sites where the domains have been.

modes and two pulses traveling to the right on a pair of strings. All of the kinks in the strings travel right or left at one cell per time step, and therefore the frequency of oscillation of the normal modes is given by $\nu = 1/\lambda$. The rule also supports open and closed boundary conditions, so the pulses may or may not be inverted upon reflection depending on the impedance. This rule is especially interesting in that it contains a linear dynamics in a nonlinear rule and illustrates how a field amplitude can be represented by the positions of particles in the plane of the CA.

2.3.4 Lattice Gases

The most important CA for physics, and those of the greatest current interest, are the lattice gases. Their primary characteristic is that they have distinct, conserved particles, and in most cases of interest, the particles have a conserved momentum. Lattice gases are characteristically reversible, though sometimes strict reversibility is violated in practice. The Ising bond-energy variable dynamics alluded to above, as well as many other CA, can be thought of as non-momentum conserving lattices



(a) (b)

Figure 2-12: A reversible, two-dimensional rule which simulates the one-dimensional wave equation in the plane of the CA. An initial condition showing normal modes as well as localized pulses (a). After 138 steps, the modes are out of phase, and the pulses have undergone reflection (b).

gases.[5] However, for our purposes, lattice gases will be reversible and momentum conserving unless stated otherwise.

The rules in this section (and in the following section) use a partitioning CA format called the *Margolus* neighborhood: ⊞. In this case, all of the cells in the neighborhood depend on all of the others, and distinct neighborhoods *don't* overlap, i.e., they partition the space. This is very important because it makes it easy to make reversible rules which conserve particle number. One merely has to enforce reversibility and particle conservation on a single partition and these constraints will also hold globally. The cells in different partitions are coupled together by redrawing the partitions between steps (see for example, figures 2-2(b) and 3-4). Thus, the dynamical laws in a partitioning CA depend on both space and time.

The simplest example of a nontrivial lattice gas that one can imagine is the HPP lattice gas which contains identical, single-speed particles moving on a Cartesian lattice. The only interactions are head-on, binary collisions in which particles are deflected through 90°. Note that this is a highly nonlinear interaction (consider

[5] In fact, the term *lattice gas* originally referred to the energy variables of the Ising model.

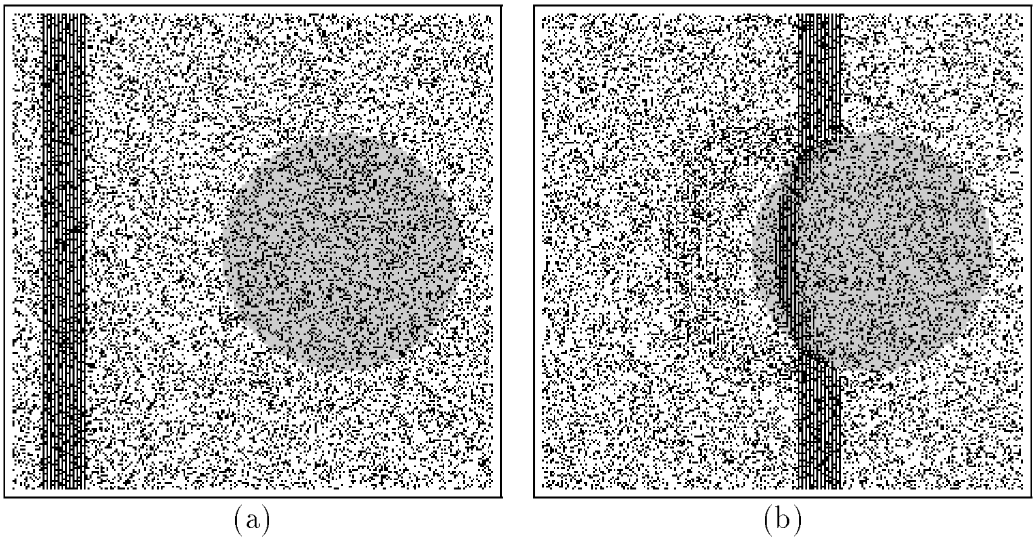
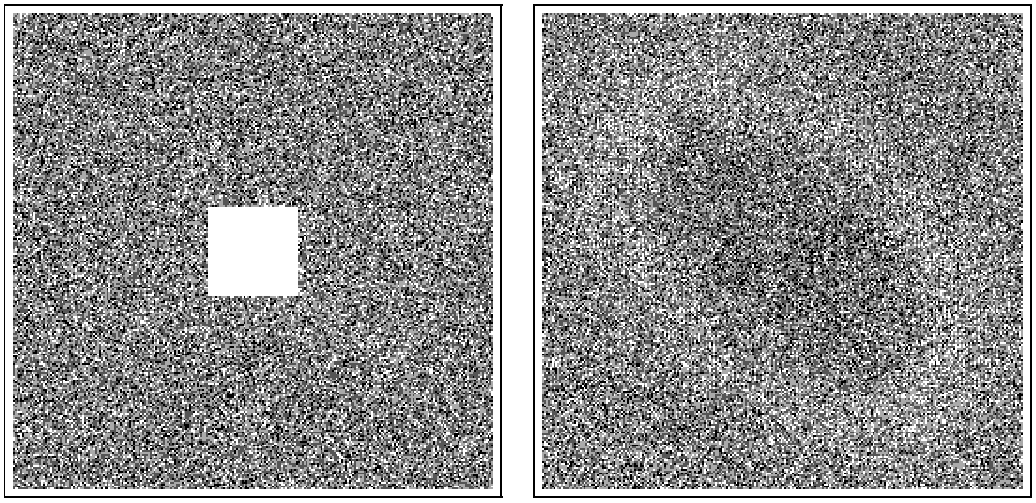


Figure 2-13: A primitive lattice gas (HPP) which exhibits several interesting phenomena. A wave impinging on a region of high index of refraction (a). The wave undergoes reflection as well as refraction (b).

the particles separately). One result of this (reversible) dynamics is to damp out inhomogeneities in the gas, and this dissipation can be viewed as an inevitable increase of entropy. Unfortunately, isotropic fluid flow is spoiled by spurious conservation laws such as the conserved momentum along each and every coordinate line. However, the speed of sound is a constant, $1/\sqrt{2}$, independent of direction.

Figure 2-13 shows an implementation of the HPP model including an interesting modification. Here, the particles move diagonally, and horizontal and vertical "soliton" waves can propagate without dissipation at a speed of $1/\sqrt{2}$. These waves are maintained as a result of a *moving invariant* in which the number of particles along a moving line is conserved. The shaded area marks a region where the HPP rule operates only half the time. This effectively makes a lens with an index of refraction of $n = 2$, and there is an associated impedance mismatch. When the wave hits the lens, both a reflected and refracted wave result.

A better lattice gas (FHP) is obtained by switching to a triangular lattice and adding three-body collisions. This breaks the spurious conservation laws and gives a fully isotropic viscosity tensor. In the appropriate limit, one correctly recovers the incompressible Navier-Stokes equations. It is difficult to demonstrate hydrodynamic



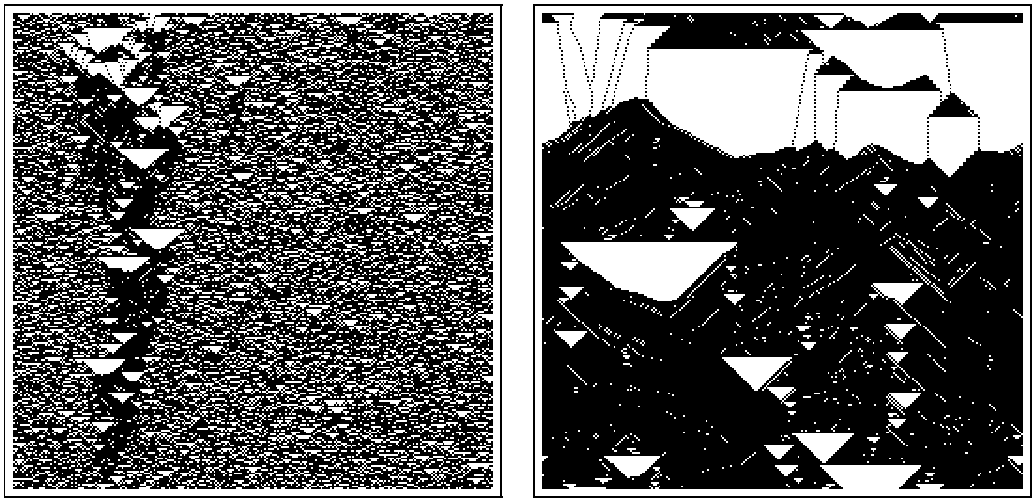
(a) (b)

Figure 2-14: A sophisticated lattice gas (FHP) which supports realistic hydrodynamics. An initial condition with a vacuum surrounded by gas in equilibrium (a). The resulting rarefaction moves away at the speed of sound (b). The asymmetry is due to the embedding of the triangular lattice in a square one.

effects on CAM-6, but sound waves can be readily observed. Figure 2-14 shows an equilibrated gas with a 48×48 square removed. The speed of sound is again $1/\sqrt{2}$ in all directions, and after 100 steps, the disturbance has traveled approximately 71 lattice spacings. The wave appears elliptical because the triangular lattice has been stretched by a factor of $\sqrt{2}$ along the diagonal in order to fit it into the Cartesian space of the machine.

2.3.5 Material Transport

Complex phenomena often involve the dissipative transport of large amounts of particulate matter (such as molecules, ions, sand, or even living organisms); furthermore, CA are well-suited to modeling the movement and deposition of particles and the subsequent growth of patterns. This section gives two examples of CA which are abstract models of such transport phenomena. The models are similar to lattice gases in that they have conserved particles, but they differ in that they are characteristically irreversible and don't have a conserved momentum. In order to conserve particles, it is again useful to use partitioning neighborhoods.

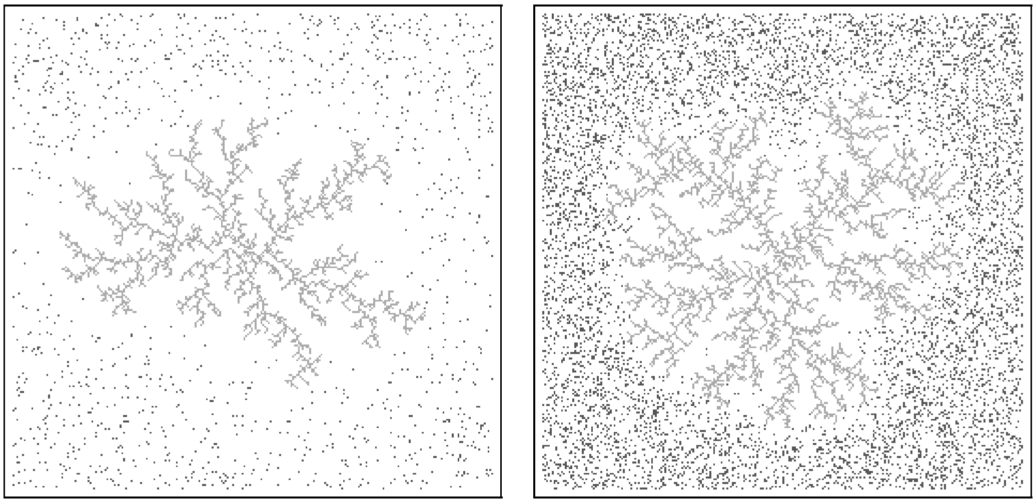


(a) (b)

Figure 2-15: A stylized model of packing sand. Shortly after starting from a near-critical density configuration, most of the material has stabilized except for a single expanding cave-in (a). Eventually, the collapse envelopes the entire system and leaves large caverns within densely packed material (b).

The detailed growth of materials depends on the mode of transport of the raw materials. One such mode would be directed ballistic motion which occurs, for example, in deposition by molecular-beam epitaxy. Consideration of the action of gravity leads to abstract models for the packing of particulate material as shown in figure 2-15. The rule has been designed so that particles slide down steep slopes and fall straight down when there are no particles on either side. The result of this dynamics is the annealing of small voids and the formation of large ones. Eventually the system settles into a situation where a ceiling extends across the entire space. This model also exhibits some interesting "critical" phenomena. If the initial condition is denser than about 69%, the system rapidly settles down to a spongy state. If the initial density is less than about 13%, all of the particles end up falling continuously.

Another important mode of transport of small particles is diffusion. Figure 2-16 shows a model of diffusion limited aggregation in which randomly diffusing particles stick to a growing dendritic cluster. The cluster starts out with a single seed, and the initial density of diffusing particles is adjustable. The cluster in 2-16(a) took 10000 steps to grow in a gas with an initial density of 6%. However, with an initial density of 17%, the cluster in 2-16(b) formed in only 1000 steps. Fractal patterns such as



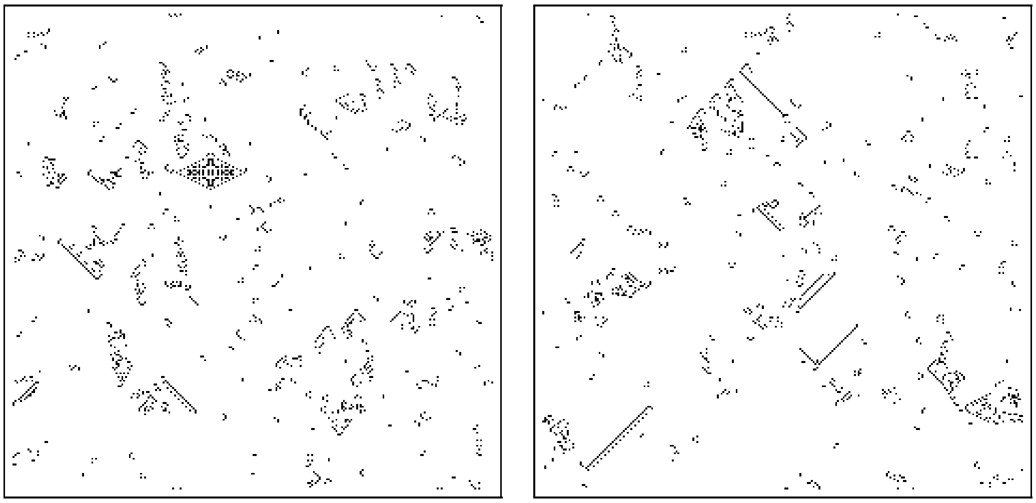
(a) (b)

Figure 2-16: A simulation of diffusion limited aggregation. Diffusing particles preferentially stick to the tips of a growing dendritic cluster (a). With a higher density of random walkers, the growth is more rapid and the cluster has a higher fractal dimension (b).

these are actually fairly commonly in dissipative CA models. While new abstract physical mechanisms would have to be constructed, the visual appearance of these patterns suggests the possibility of making CA models of crystal formation, electrical discharge, and erosion.

2.3.6 Excitable Media

A characteristic feature of a broad class of CA models is the existence of attractive, excitable states in which a cell is ready to "fire" in response to some external stimulus. Once a cell fires, it starts to make its way back to an excitable state again. The triggering stimulus is derived from a standard CA neighborhood, and the dynamics is characteristically irreversible. Such rules are similar to the voting rules in that the cells tend to follow the behavior of their neighbors, but they are different in that the excitations do not persist. Rather, the cells must be restored to their rest state during a recovery period. The examples given here differ from each other in the recovery mechanism and in the form of the stimulus. Like many of the irreversible CA above, they form patterns, but unlike the ones above, the patterns must oscillate. Many



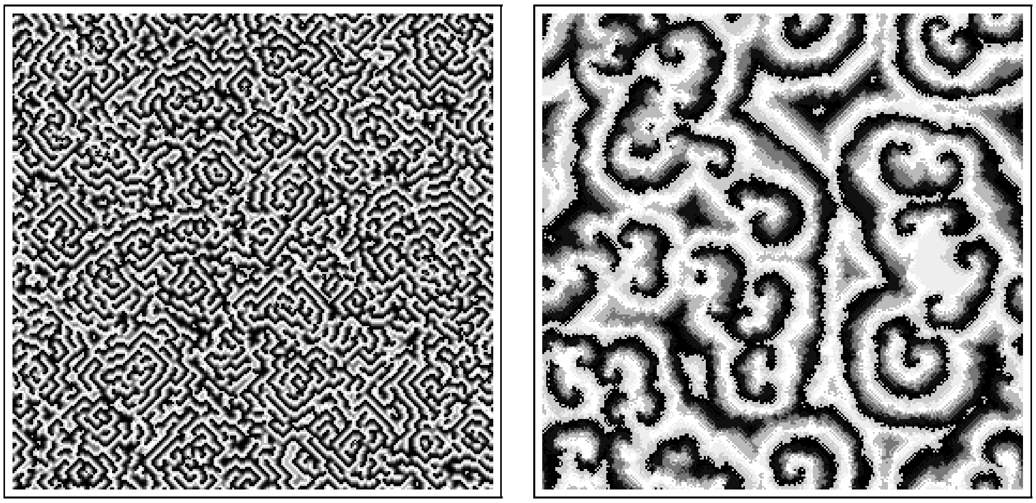
(a) (b)

Figure 2-17: A model of neural activity in which cells must undergo a refractory period after being stimulated to fire. Starting from the standard random pattern, the system quickly settles down to a low density of complex, propagating excitations (a). After only 100 more steps, the pattern has completely changed (b).

models of excitable media are biological in nature, and even the game of life could perhaps be moved to this category.

The best example of excitable media found in nature is undoubtedly brain tissue, since very subtle stimuli can result in profound patterns of activity. At the risk of drastically oversimplifying the real situation, neural cells have the property that they fire as a result of activity in neighboring cells, and then they must rest for a short time before they can fire again. This behavior can be readily captured in a simple CA. A cell in the resting phase fires if there are exactly two active cells (out of 8) in its neighborhood. A cell which has just fired must wait at least one step before it can fire again. Figure 2-17 shows the behavior of this rule 100 and 200 steps after starting from the standard random pattern. The system changes rapidly and often has brief blooms of activity, though the overall level fluctuates between about two and three percent. This rule was specifically designed to prevent stationary structures, and it is notable for the variety of complex patterns it produces.

Many physically interesting systems consist of a spatially-distributed array or field of coupled oscillators. A visually striking instance of such a situation is given by the oscillatory Zhabotinsky redox reaction, in which a solution cycles through a series of

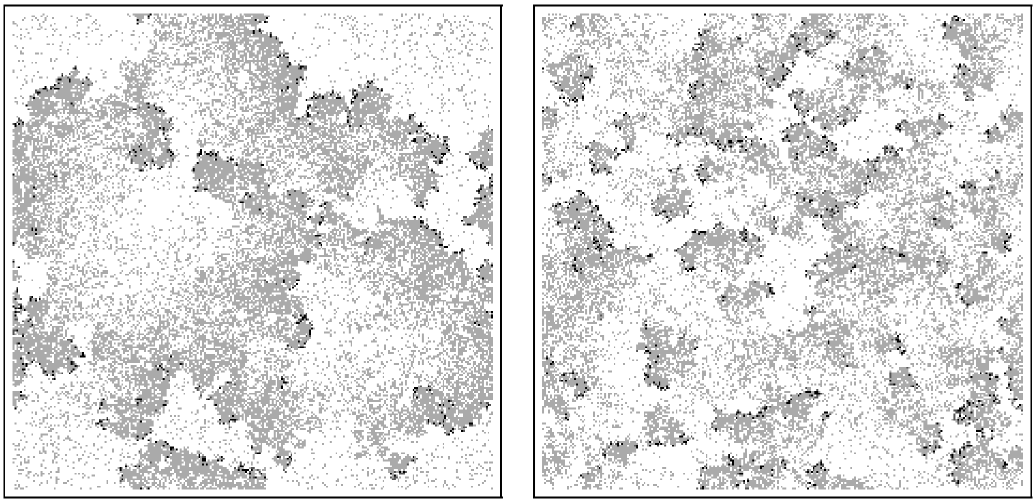


(a) (b)

Figure 2-18: A model of oscillatory chemical reactions. When enough reactants are in the neighborhood of a cell, the next product in the cycle is formed. With a low reaction threshold, the resulting patterns are small, and the oscillations fall into a short cycle (a). A non-monotonic threshold creates large patterns which continually change (b).

chemical species in a definite order. Since there must be enough reactants present to go from one species to the next, the oscillators try to become phase locked by waiting for their neighbors. In a small reaction vessel, the solution will oscillate synchronously, but in a large vessel, some regions may be out of phase with the rest; furthermore, there may be topological constraints which prevent the whole system from ever becoming synchronized. Figure 2-18 shows two CA simulations of this phenomenon with different reaction thresholds. The resulting spiral patterns are a splendid example of self-organization. Similar phase locked oscillations can also be observed in fireflies, certain marine animals, and groups of people clapping.

The final example of an excitable CA is a stochastic predator-prey model. The fact that there are tens-of-thousands of degrees of freedom arranged in a two-dimensional space means that the behavior cannot be captured by a simple pair of equations. The prey reproduce at random into empty neighboring cells, while the predators reproduce into neighboring cells that contain prey. Restoration of the empty cells is accomplished by having the predators randomly die off at a constant rate. Figure 2-19 shows the behavior of this dynamics for two different death rates. Variations on this



(a) (b)

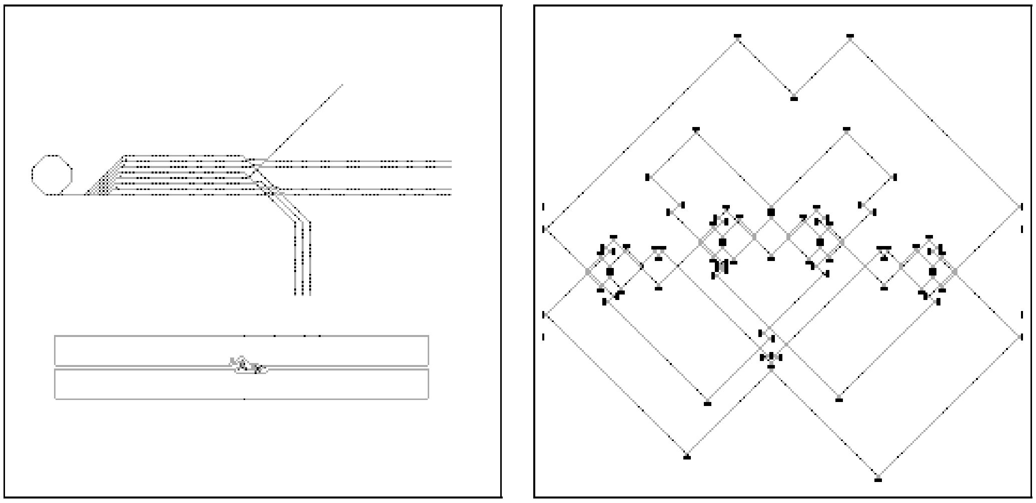
Figure 2-19: Predator-prey simulations. The prey reproduces at random into empty areas, while the predators flourish behind the waves of prey (a). Death among the predators occurs at random at a constant rate. With a higher death rate, there are more places for the prey to live (b).

rule could possibly be used to model grass fires, epidemics, and electrical activity in the heart.

2.3.7 Conventional Computation

In addition to computational modeling, CA can be used to compute in a more general sense. For example, they are well-suited to certain computing tasks in vision and image processing. Running algorithms that have a local, parallel description is an important area of application for CA, but the main point of this section is to demonstrate that CA are capable of performing *universal* computation. In other words, CA can be programmed to simulate *any* digital computer. Universal computers must be able to transport and combine information at will, and for CA, this means gaining control over propagating structures and interactions. Partitioning neighborhoods give us the necessary control. Finally, computation is characteristically an irreversible process, though complex calculations can be done reversibly as well.

Modern digital circuits are written on to silicon chips, and in a similar manner one can write circuits into the state space of a CA. With the right dynamical rule,



(a) (b)

Figure 2-20: Digital circuit simulations. (a) An irreversible rule showing a universal set of logic components along with an example of a counting circuit. (b) A reversible permutation generator built using the billiard-ball model.

the CA will behave in exact correspondence with the physical chip. Figure 2-20(a) shows a pair of circuits which can be used with a special irreversible digital logic rule. The upper circuit shows all of the components necessary to simulate an arbitrary digital circuit: wires, signals, fanouts, clocks, crossovers, AND gates and NOT gates. An example of what these components can do is given in the lower circuit which is a serial adder that has been wired up to act as a counter.

Figure 2-20(b) shows a circuit in the so-called billiard ball model, which is an example of a reversible, non-momentum conserving lattice gas. Pairs of point particles travel through empty space (the lines merely mark where the particles have traveled) and effectively act like finite-sized billiard balls during collisions with reflectors and other pairs of particles. Collisions between balls serve to modulate two signals and can be used as logic gates. This rule can be used to simulate an arbitrary reversible digital circuit.

With these computational rules, we have reached the opposite end of a spectrum of CA models in comparison to the chaotic rules. In regard to the classifications developed above, both kinds can be either reversible or irreversible. A difference which is somewhat incidental but also with significance is that the chaotic rules use the standard CA neighborhood format while the computational rules use the partitioning format. However, the main way that they differ is in that information is channeled in the computational rules while in the chaotic rules it is not. This fine control over information makes it possible to design highly complex yet structured systems such as computers and, hopefully, models of arbitrary natural processes.

The examples above show that CA are capable of modeling many aspects of nature including those of special interest to physics. However, these and similar models could use further simulation and data analysis coupled with more in-depth analytical work. Furthermore, there is a need to develop and codify the techniques involved as well as to apply them to specific external problems. These considerations form a basic motivation for the subject of this thesis. Consequently, the following chapters present new CA models of particular interest to physics while analyzing their behaviors and addressing the general problem of developing novel mathematical methods.

Chapter 3

Reversibility, Dissipation, and Statistical Forces

3.1 Introduction

3.1.1 Reversibility and the Second Law of Thermodynamics

An oft cited property of the physical world is that of time-reversal invariance. This means that any physical process run in reverse would be a possible evolution of another process.[1] In any event, the fundamental laws of physics are (as far as we know) reversible—that is, the equations of motion could theoretically be integrated backwards from final conditions to initial conditions—and the concept of reversibility has even been elevated to the status of a principle to which new dynamical theories must conform.

We are interested in capturing as many principles of physics in our CA models as possible, and therefore, we strive to construct CA dynamics which are reversible. In other words, the dynamics should map the states of the system in a one-to-one and onto manner [90]. In addition, we would like the inverse mapping to also be a local CA rule and to resemble the forward rule as much as possible. One way to

[1] The validity of this statement requires a suitable time-reversal transformation on all the variables in the system and neglects the break in this symmetry which may accompany known CP violation.

think about reversibility from an information mechanical viewpoint is that the system always contains the same information: it is only transformed (or scrambled) into new forms throughout the computation. In the case of invertible CA running on digital computing machines, this interpretation of reversibility is very direct. If the inverse rule is known, then we have complete control over the dynamics and can explicitly run the system backwards at will.

An important consequence of reversibility is the "second law of thermodynamics." Without quibbling over caveats, it states that in any transformation of a closed, reversible system, the entropy must increase or stay the same. The notion of entropy used here will be discussed at greater length below, but for now it suffices to think of it as a measure of the disorder of the system. The intuitive reason for the tendency for increasing disorder is that there are *vastly* more disordered states than ordered ones so that the fraction of time spent in disordered states is likely to be greater. Reversibility is important here because it prevents a many to one mapping of states; otherwise, the system could preferentially evolve towards the relatively few ordered states.

Another important consequence of reversibility is that it helps prevent spurious dynamical effects due to frozen degrees of freedom. What can happen in an irreversible rule is that certain features of the system become locked in and artificially restrict the dynamics. This cannot happen under a reversible dynamics because there are no attractive fixed points. In fact, a reversible dynamics on a finite set of states will always return the system, eventually, to its initial condition. This recurrence is stronger than the Poincaré recurrence theorem in classical mechanics because it will return exactly to its starting point, not only approximately.

3.1.2 Potential Energy and Statistical Forces

One of the most fundamental concepts of physics is energy. Its interest lies in the fact that it is conserved, it takes on many forms interchangeably, and that it plays the role of the generator of dynamics. Considerations of energy transport and exchange thoroughly pervade almost every aspect of physics. Thus, many of our CA formulations of physical systems will be designed to reflect properties and effects conferred by energy.

Perhaps the most basic expression of energy is that of potential energy. Potentials are often used to describe the statics of physical systems, and they express the essential physics in both the Lagrangian and Hamiltonian formulations of dynamics. Potentials act on a system by generating forces which in turn alter the movement of particles. Ultimately, these forces arise from interactions of particles with other particles in the system, but we can simplify the situation further and consider the case of externally imposed scalar potentials on individual particles. The more general case will be reconsidered in chapter 6.

A single, classical particle in a potential well would follow some definite orbit, while many such particles would distribute themselves according to a Boltzmann distribution. At present, we do not know how to have individual particles follow non-trivial, smooth trajectories in a cellular space. However, the advantages of parallelism inherent in CA are best suited to many-body systems, so we choose to develop some of the statistical properties of a gas of particles in the potential instead. Situations such as this arise, for example, in the case of an atmosphere in a gravitational field or an electron gas in a Coulomb potential.

The physical phenomenon that we wish to capture then, is the microscopically reversible concentration of a gas in the bottom of a potential well. Note however, that for a gas consisting of particles on a lattice, this implies a *decrease* in the entropy, and herein lies a paradox. How is it possible to have increasing order in a system in which entropy must increase, and in particular, within the closed environment of a bounded CA array? How can we have a reversible dynamics which conserves energy and information while still exhibiting dissipation? Dissipation in this context refers to the loss of detailed information about the dynamical history of the system, and in the case of particles falling down a potential gradient, the concomitant loss of energy. Thus, we would like to find a simple reversible mechanism with which one can model a gas settling into a potential well, and in the process, discover the essential features of dissipation in reversible systems.

The solution to the paradox involves invoking *statistical forces.* In other words, we actually use the second law of thermodynamics to our advantage by arranging the system so that an increase in entropy brings about the very behavior that we seek. This requires coupling the system to another having low entropy so that the joint system is far from equilibrium. The approach to equilibrium is manifested as a statistical force: particles are "pushed" by an overwhelming probability to move to the lower potential. The force continues until the transition probabilities match and equilibrium is restored. Throughout the process, the overall dynamics remains strictly deterministic and reversible, and the total entropy increases.

3.1.3 Overview

This chapter presents a CA simulation that explicitly demonstrates dissipative interactions in the context of a microscopically reversible dynamics. This is done by introducing a representation of an external scalar potential which in turn can effectively generate forces. Hence, it gives one approach to solving the problem of how to include forces in CA models of physical systems. The model incorporates conservation of energy and particles, and by virtue of reversibility, it also can be said to conserve information. The dynamics allows us to follow the process of dissipation in detail and to examine the associated statistical phenomena. Moreover, the model suggests a general technique for turning certain irreversible physical processes into reversible ones.

The rest of the chapter is organized as follows:

Section 3.2 presents a CA model of a gas in the presence of an external potential well. The model provides a very clean demonstration of some key ideas in statistical mechanics and thermodynamics. A simplified analysis is presented which introduces counterparts of entropy, temperature, occupation number, and Fermi statistics, among others. The analysis also serves as a point of departure for a further discussion of microcanonical randomness.

Section 3.3 gives the details of the implementation of the algorithm on CAM-6. While the details themselves are not unique, they are illustrative of the techniques which can be employed when developing physical models for CA. Details of the implementation lead to practical considerations for the design and programming of cellular automata machines, and could have important consequences for the design of future hardware and software.

The deviation of the simplified analysis from the results of a trial run of the system indicates the presence of additional conserved quantities which invalidate a naïve application of the ergodic hypothesis. Section 3.4 reveals these quantities and the analysis is revised to take broken ergodicity and finite size effects into account.

Section 3.5 presents the statistical results of a long run of the system and compares them to the theory of section 3.4. Together, these sections demonstrate the interplay of theory and experiment that is readily possible when using cellular automata machines. Precise measurements of the system agree closely with the revised calculations.

Section 3.6 indicates some of the ways that the model can be modified, extended, or improved. Possible applications of the method are given for a number of problems. Some of the implications for modeling with CA are discussed along with issues pertaining to mechanistic interpretations of physics. Finally, a number of open problems are outlined.

3.2 A CA Model of Potentials and Forces

3.2.1 Description of the Model

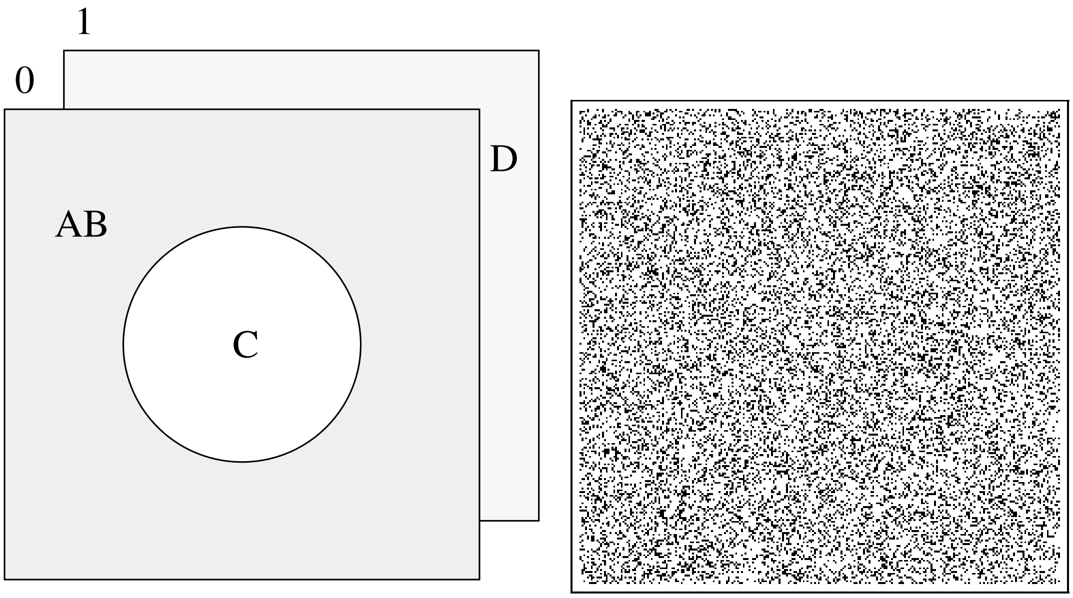
Any CA which has some form of particle conservation may be referred to as a *lattice gas*. The most common examples are simulations of physical substances, though one can imagine examples from fields ranging from biology to economics. A bit in a lattice gas usually represents a single particle state of motion, including position and velocity. Each of these states may contain either zero or one particle, so an exclusion principle is automatically built in. In addition to this basic lattice gas format, other properties such as reversibility or energy/momentum conservation may also be imposed. The problem when designing a lattice gas is to specify a rule that yields the desired phenomena while maintaining the constraints. For example, in hydrodynamic applications, one is often interested in choosing collision rules which minimize viscosity [24, 25]. However in this case, the goal is just to demonstrate a reversible dynamics which generates a statistical force corresponding to an external scalar potential.

The specific CA model treated in this chapter consists of a pair of two dimensional lattice gases that are coupled via interaction with a potential well. The former will be thought of as the system proper (consisting of featureless particles), and the latter will be the heat bath (consisting of energy tokens). The tokens are sometimes called demons following Creutz, who originated the general idea of introducing additional degrees of freedom to enable microcanonical Monte Carlo simulation [19, 20]. The particles and the energy tokens move from cell to cell under an appropriate CA rule as described below.

The lattice gases can be viewed as residing in parallel spaces (0 and 1), and the system logically consists of three regions (AB, C, and D) as shown in figure 3-1(a).[2] The circle indicates the boundary of the potential well where the coupling of the lattice gases takes place. By definition, a particle in the central region (C) has zero potential, a particle in the exterior region (AB) has unit potential, and the demons (D) represent unit energy. The particles are conserved while the demons are created and destroyed to conserve the total energy when particles cross the boundary.

The dynamics should be nontrivial and have the following properties, but it need not be otherwise restricted. First and foremost, it must be reversible. Second, the gas particles are to be conserved between regions AB and C. Third, the total energy, consisting of the number of particles outside the well plus the number of demons, should be conserved. Finally, the key part of the rule is that a particle will be permitted to cross the boundary of the potential well only if it can exchange energy with the heat bath appropriately: a particle falling into the well must release one unit of energy in the form of a demon, and a particle leaving the well must absorb a demon.

[2]The first region will later be subdivided into two regions, A and B; hence, the notation, AB.



(a) (b)

Figure 3-1: (a) A system comprised of a lattice gas on plane 0 coupled to a heat bath of energy demons (D) on plane 1. The gas particles feel a potential which is low in the central region (C) and high in the outer region (AB). (b) The initial state of the lattice gas in plane 0 has an occupation number of 25%.

There are other choices of properties that create a similar effect (see section 3.6), but this one will illustrate the point.

Given only these assumptions on the dynamics, what can we say about the evolution of the system? According to the second law of thermodynamics, the total entropy of an isolated, reversible system such as ours must, in general increase or stay the same. In fact, it will (again, in general) increase unless there are conservation laws which prevent it from doing so. The notion of entropy used here is discussed next and is consistent with Jaynes' Principle of Maximum Entropy [45]. The entropy so defined will prove useful for calculating the average number of particles in the various regions of the space.

A measure of entropy can only properly be applied to an *ensemble*, or probability distribution over states, rather than to a single configuration. However, a specific configuration of the system defines an ensemble by the set of all microstates that are consistent with our coarse-grained, or macroscopic, knowledge of that configuration. Macroscopic knowledge may be obtained by measurement and includes the conserved quantities, but may also include things like particle densities or currents in various regions of the space. The distinction between a state and the ensemble it defines will usually blurred because we do not know—and generally are not interested in knowing—all the details of the system. Hence, the entropy of a state will be taken to be the logarithm of the number of states in the corresponding ensemble.

The particular initial configuration of the particle system used in this chapter is shown in figure 3-1(b). The gas has a *uniform* density of 25%, which means that each cell has been assigned a particle with an independent probability of 1/4. Note that this implies that there is no correlation whatsoever between one cell and the next. The gas considered by itself is thus in a state of maximum entropy and is consequently at infinite temperature. On the other hand, the heat bath is initially cleared of demons, leaving it in its (unique) lowest possible energy state. This is a minimum (equal to zero in this case) entropy state, and by analogy with the third law of thermodynamics, the heat bath is considered to be at absolute zero. Therefore, the entire system is in a comparatively low entropy state, far from equilibrium, and the dynamics will serve to increase the entropy subject to the constraints.

What is the mechanism by which the entropy increases, and how is the increase manifested? Given the above constraints and a lack of fine-grained, or microscopic, knowledge about the system, we can and should assume an effectively ergodic dynamics under which the system wanders through its configuration space subject only to conservation laws. Most of the states have a higher entropy than does the initial configuration, so the entropy will most likely increase. Similarly, most of the states have more particles in the well than does the initial configuration, so the particles are biased to flow into the well. This bias is the statistical force which is derived from interaction with the potential well. The force will act to drive the system to a maximum entropy state—that is, one for which all accessible states (subject to any constraints) are equally likely.

The response of the gas under a specific dynamics satisfying the above conditions is shown in figure 3-2, while the energy released into the heat bath is shown in figure 3-3. The particular dynamics used will be described in detail in section 3.3, though the

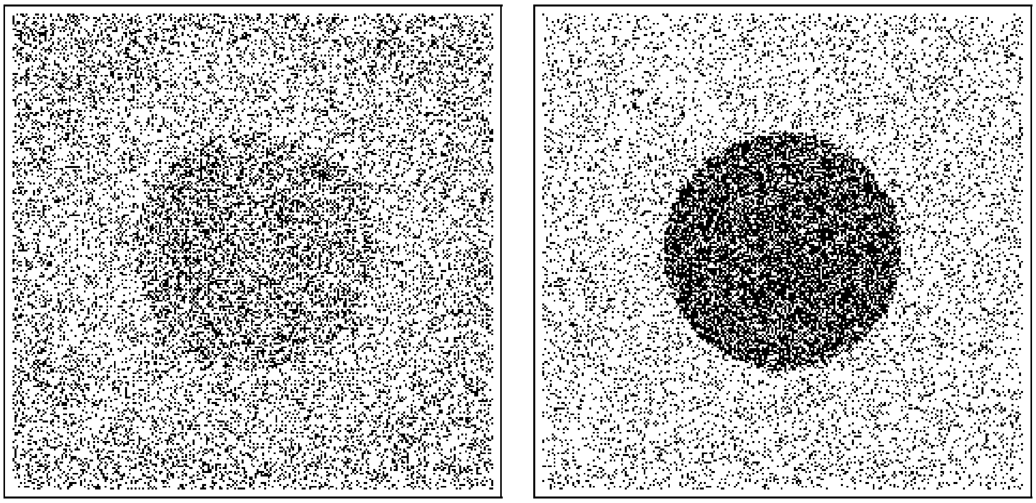
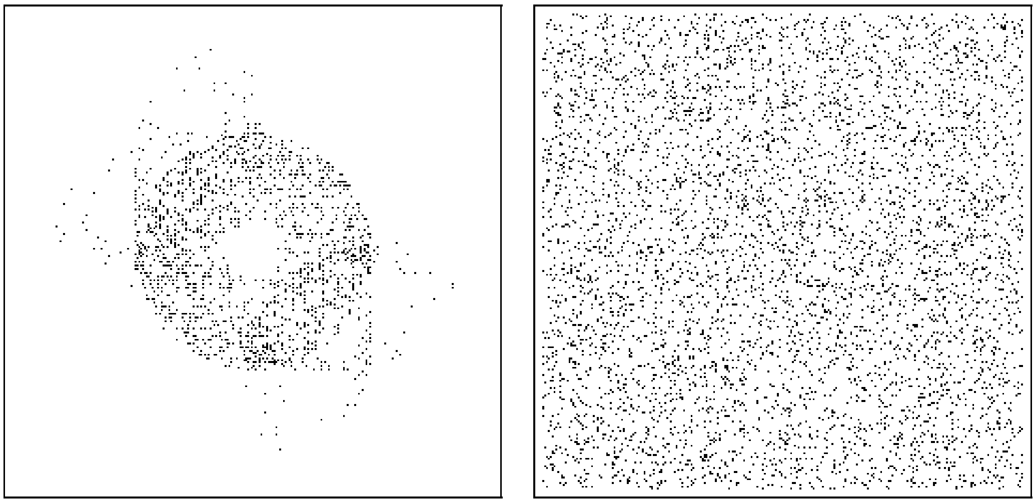

(a) (b)

Figure 3-2: Particle configurations taken from a simulation of a lattice gas in a potential well. (a) Nonequilibrium transient near the beginning of the simulation showing the particles falling into the potential well. (b) Equilibrium configuration of the lattice gas.

behavior is quite generic. After 100 steps (a), we see that the density of the gas in the well has increased while a surrounding rarefaction has been left behind. Energy tokens are being created at the boundary of the well as the particles fall in. After 10,000 steps, the system has reached an equilibrium state (b), where the gas is once again uniform in regions AB and C, and the heat bath has come to a uniform, nonzero temperature.

In order to get a better feel for the mechanism underlying this behavior, it is useful to augment the demonstration and qualitative arguments above with quantitative calculations. The calculations can be compared with the results of simulations to increase our confidence in the predictive power of CA methods and to elicit further properties of the model. A computer simulation of a discrete system makes it possible to follow the complete time dependence of all the degrees of freedom, but we are seldom interested in such detail. Rather, we are primarily interested in knowing the macroscopic quantities which result from averaging over the microscopic information. The non-equilibrium evolution depends on the particular dynamics whereas the equilibrium situation does not (to lowest order), so only the later will be analyzed.



(a) (b)

Figure 3-3: Heat bath corresponding to figure 3-2. (a) Nonequilibrium transient showing the release of heat energy in the form of demons. (b) Equilibrium configuration of the heat bath.

3.2.2 Basic Statistical Analysis

The equilibrium situation can be analyzed through combinatorial methods. We primarily seek to obtain the expected number of particles in the three regions. The analysis will illustrate in a conceptually clear context several of the most important ideas in statistical mechanics. These include entropy, temperature, Fermi statistics, and the increase of entropy in the face of microscopic reversibility. It will also bring out some of the conceptual difficulties encountered in its justification.

The basic plan for calculations such as this is to find the maximum entropy ensemble subject to the constraints and then to determine the expectation values assumed by the dynamical variables. The calculated ensemble averages can be compared to time averages taken from a computer simulation. After a sufficiently long relaxation period, the experimental samples of the system should be representative of the whole ensemble. The differences between theory and experiment will reflect a number of approximations in the calculation as well as incorrect assumptions about the ergodicity of the system and how measurements are made.

In order to extract some meaningful information out of the mass of details, it is useful to adopt a description in terms of macroscopic dynamical variables. Let

N_x and n_x denote the number of cells and the number of particles respectively in region x. Also define the density (which can also be thought of as a probability or an occupation number) $\rho_x = n_x/N_x$. The complement, $\overline{\rho}_x = 1 - \rho_x$, will also be useful. The total number of particles n_T and the total energy E_T are conserved and satisfy the constraints,

$$n_T = n_{AB} + n_C \quad \text{and} \quad E_T = n_{AB} + n_D. \tag{3.1}$$

The above description can only approximately capture the maximum entropy distribution because it replaces actual particle numbers (which are not constant) with single averages (i.e., it ignores fluctuations). Furthermore, the occupation numbers in the cells in any given region are correlated and cannot, strictly speaking, be represented by independent, identically distributed random variables as we are doing. The differences become negligible in the thermodynamic limit, but it does affect the results in finite systems as we shall see in section 3.4.

With these caveats in mind, we can proceed to work towards the maximum entropy solution. The number of accessible states with the given occupation numbers is

$$\Omega = \binom{N_{AB}}{n_{AB}}\binom{N_C}{n_C}\binom{N_D}{n_D}, \tag{3.2}$$

and the entropy is given by

$$\begin{aligned} S &= \ln \Omega \\ &\cong N_{AB}\ln N_{AB} - n_{AB}\ln n_{AB} - (N_{AB} - n_{AB})\ln(N_{AB} - n_{AB}) \\ &\quad + N_C \ln N_C - n_C \ln n_C - (N_C - n_C)\ln(N_C - n_C) \\ &\quad + N_D \ln N_D - n_D \ln n_D - (N_D - n_D)\ln(N_D - n_D) \\ &= N_{AB}(-\rho_{AB}\ln \rho_{AB} - \overline{\rho}_{AB}\ln \overline{\rho}_{AB}) + N_C(-\rho_C \ln \rho_C - \overline{\rho}_C \ln \overline{\rho}_C) \\ &\quad + N_D(-\rho_D \ln \rho_D - \overline{\rho}_D \ln \overline{\rho}_D). \end{aligned} \tag{3.3}$$

The final expression above is just the sum over the Shannon information function for each cell times the number of cells of that type [75].

The entropy of the equilibrium ensemble will assume the maximum value subject to the constraints (3.1). To find this extremum, introduce Lagrange multipliers α and β, and define the auxiliary function

$$f = S + \alpha(n_T - N_{AB}\rho_{AB} - N_C\rho_C) + \beta(E_T - N_{AB}\rho_{AB} - N_D\rho_D). \tag{3.4}$$

The Lagrange multipliers will give us measure of temperature T and chemical potential μ, where $T = 1/\beta$ and $\alpha = -\beta\mu$.

Differentiating f with respect to α, β, ρ_{AB}, ρ_C, and ρ_D and setting the results to zero returns the constraint equations (3.1) along with

$$\begin{aligned} -N_{AB}\ln\frac{\rho_{AB}}{\overline{\rho}_{AB}} - \alpha N_{AB} - \beta N_{AB} &= 0, &(3.5)\\ -N_C\ln\frac{\rho_C}{\overline{\rho}_C} - \alpha N_C &= 0, &(3.6)\\ -N_D\ln\frac{\rho_D}{\overline{\rho}_D} - \beta N_D &= 0. &(3.7)\end{aligned}$$

Solving for the densities gives

$$\begin{aligned} \rho_{AB} &= \frac{1}{1+e^{\beta(1-\mu)}}, &(3.8)\\ \rho_C &= \frac{1}{1+e^{\beta(-\mu)}}, &(3.9)\\ \rho_D &= \frac{1}{1+e^{\beta}}. &(3.10)\end{aligned}$$

These are just the occupation numbers for particles obeying Fermi statistics. Note that they turned out to be intensive quantities since we assumed the thermodynamic limit.

Equations (3.1) and (3.8)–(3.10) constitute five equations in five unknowns, but the later three can be combined to eliminate β and μ to give

$$\rho_{AB}\overline{\rho}_C\overline{\rho}_D = \rho_C\rho_D\overline{\rho}_{AB}. \tag{3.11}$$

This equation can be rewritten in terms of the numbers of particles in the three regions as follows:

$$n_{AB}(N_C - n_C)(N_D - n_D) = n_C n_D(N_{AB} - n_{AB}). \tag{3.12}$$

This equation along with (3.1) gives three equations in three unknowns which is useful for solving for the particle numbers directly. The interaction of particles in the cells on the boundary of the potential can be described by the reaction $AB \rightleftharpoons C + D$, and for the reaction to proceed in either direction, there must be a vacancy for each product species. Therefore, in equilibrium, equation (3.11) can be interpreted as saying that the probability of a particle falling into the well must equal the probability of a particle leaving the well.

The microcanonical heat bath technique illustrated in this section can be generalized and used in conjunction with any reversible CA having a locally defined, conserved energy. We just need to arrange it so that the subsystems are free to exchange energy but that the dynamics are not otherwise constrained. In this case, the number of states factors into the numbers of states in each subsystem, and the entropies add. The characteristics of the heat bath are then independent of the system to which it is coupled since it attains its own maximum entropy state subject only to the amount of energy it contains. As far as the primary system is concerned, the heat bath is characterized only by its propensity to absorb and release energy. This is reflected in the last equation above where the density of the demons is determined entirely by the inverse temperature β.

An analysis similar to the one above can thus be carried out for any heat bath considered in isolation and having a given energy (or equivalently, a given temperature). Appendix A develops the statistical mechanics and thermodynamics of one such microcanonical heat bath which holds up to four units of energy per cell. In addition to being useful for generating statistical forces, these heat baths act as thermal substrates for the primary system and can be used for measuring and setting the temperature. This can be done, for example, to maintain a constant temperature, as in a chemical reaction vessel, or to follow a particular annealing schedule in a simulated annealing algorithm. The appendix discusses these and other issues.

3.2.3 A Numerical Example

The version of this experiment that comes with the CAM-6 distribution software [59, 60] can be run under repeatable conditions by re-initializing the random number generator. The specific numerical values given in the rest of this chapter correspond to this case.

CAM-6 has a total of $N_T = N \times N = 65536$ cells with $N = 256$. The parameters $N_{AB} = 52896$, $N_C = 12640$, and $N_D = 65536$ give the total number of cells available for particles outside the well, in the center of the well, and in the heat bath respectively. The particle density is *uniformly* initialized to 25% which gives *approximately* $N_T/4 = 16384$ particles and $N_{AB}/4 = 13224$ units of energy. The actual initial values of these conserved quantities are $n_T = 16191$ and $E_T = 13063$. Since there are no demons to start with ($n_D = 0$), the initial particle counts in regions AB and C are $n_{AB} = 13063$ and $n_C = 3128$ respectively.

Solving equations (3.1) and (3.12) gives the theoretical equilibrium densities, while the experimental time averages can be found in section 3.5. The results are summarized in table 3.1. Note that the theoretical and experimental values differ by about 2.7 particles. This discrepancy is resolved by the theory of section 3.4.

It is also illuminating to look at the entropies of the components of the system as calculated from the theoretical values using equation (3.3). As expected, the entropy of the gas goes down while that of the demons goes up. However, the change in the total entropy, $S_f - S_i = 11874$, is positive as required by the second law of thermodynamics. Furthermore, the counting procedure that we used to calculate S implies that the final state is

$$e^{S_f - S_i} \approx 10^{5156} \tag{3.13}$$

times as likely as the initial state! In other words, for every state that "looks like" figure 3-1(b), there are approximately 10^{5156} states that "look like" figure 3-2(b)

x	T	AB	C	D
N_x	65536	52896	12640	65536
n_x (initial)	16191	13063	3128	0
n_x (basic theory)	16191	7795.73	8395.27	5267.27
n_x (experiment)	16191	7798.44	8392.56	5264.56
S_x (initial)	36640	36640		0
S_x (final, theory)	48514	30185		18329

Table 3.1: A table showing the number of cells, the number of particles, and the entropy in various regions of the potential-well system. The initial counts as well as theoretical and experimental results are given.

(taken together with figure 3-3(b)).[3] This numerical example should make it perfectly obvious *why* the entropy as defined here is likely to increase or stay the same in a reversible CA. Similarly, since the final state has the highest entropy consistent with the constraints, the origin of the statistical force should also be clear.

3.3 CAM-6 Implementation of the Model

In order to test hypotheses about a proposed CA mechanism, it is useful to write a rule and run it. At this point, a cellular automata machine becomes a great asset. Having the ability to watch a real-time display of the dynamical behavior of a system greatly increases one's intuition about the processes involved. Furthermore, having an efficient simulation opens the possibility of doing quantitative mathematical experiments and developing additional properties of the model. These paradigmatic aspects of doing physics with CA are well illustrated by the present example.

Thus the model described in the previous section was implemented on CAM-6, and here we cover the details of the implementation.[4] While some of these details could be considered artifacts of the limits on hardware resources with which one has to work, they could also say something about computation in other physical

[3]Recall that we are considering two states to be equivalent for counting purposes if they are in the same ensemble of states defined by the macroscopic quantities.

[4]The author would like to thank Bruce Smith for helping with this implementation.

situations. Going through the exercise of programming such a rule is also useful for thinking about ways of *describing* computation (parallel or otherwise). This interplay between hardware, software, and the design of dynamical laws is an interesting facet of this line of research.

This section is not strictly necessary for understanding the other sections and can be skipped or referred to as needed. It is included for completeness and to illustrate characteristic problems encountered and solutions devised when writing a CA rule. The techniques are analogous to the detailed constructions contained in a mathematical proof. Finally, it will serve as a primer on programming cellular automata machines for those who have not been exposed to them before.

3.3.1 CAM-6 Architecture

The CAM-6 cellular automata machine is a programmable, special-purpose computer dedicated to running CA simulations. The hardware consists of about one hundred ordinary integrated circuits and fits into a single slot of an IBM-PC compatible personal computer [83]. The machine is controlled and programmed in the FORTH language which runs on the PC [10]. For the range of computations for which CAM-6 is best suited, it is possible to obtain supercomputer performance for roughly $1/10000th$ the cost.

The circuitry of any computer (including cellular automata machines) can be divided into two categories: data and control. The data circuits "do the work" (e.g., add two numbers), while the control circuits "decide what to do." The dividing line between these two functions is sometimes fuzzy, but it is precisely the ability to control computers with data (i.e., with a program) that makes them so remarkable. While the control circuits of a computer are essential to its operation, it is the flow of the data that really concerns us. Consequently, the data paths common to all cellular automata machines are (1) a cellular state space or memory, (2) a processing unit, and (3) a feed from the memory to the processor (and back). The specifics of these units in CAM-6 are described in turn below.

The state space of CAM-6 consists of a 256×256 array of cells with periodic boundary conditions. Each cell is comprised of four bits, numbered 0–3. It is often convenient to think of this state space as four planes of 256×256 bits each. Planes 0 and 1 are collectively known as CAM-A, and similarly, planes 2 and 3 are known as CAM-B. The distinction between A and B is made because the two halves have limited interaction, and this in turn constrains how we assign our dynamical variables to the different planes.

The processor of CAM-6 replaces the data in each cell with a function of the bits in the neighboring cells. This is accomplished with two lookup tables (one for CAM-A and one for CAM-B), each with $2^{12} = 4096$ four-bit words. Each table can be programmed to generate an arbitrary two-bit functions of twelve bits.[5] How the twelve bits are selected and fed into the processor is crucial and is described in the next paragraph. The processor is shared among the cells in such a way as to effectively update all of them simultaneously. In actuality, each cell is updated as the electron beam in the TV monitor sweeps over the display of that cell.

At the heart of CAM-6 is a pipeline which feeds data from the memory to the processor in rapid sequence. Roughly speaking, the pipeline stores data from the three rows of cells above the current read/write location, and this serves to delay the update of cells which have yet to be used as inputs to other cells. This unit also contains many multiplexers which select predefined subsets (also called "neighborhoods") of bits from adjacent or special cells. Well over forty bits are available in principle, and while not all combinations are accessible through software, one can bypass the limitations on neighbors and lookup tables by using external hardware. The subsets available in software include the Moore, von Neumann, and Margolus neighborhoods. The potential-well rule uses the Margolus neighborhood, as do many other physical rules, so it is worthwhile to describe it in more detail.

The standard dynamical format of a CA rule is that each cell is replaced by a function of its neighbors. An alternative is the partitioning format, wherein the state space is partitioned (i.e., completely divided into non-overlapping groups of cells), and each partition is replaced by a function of itself. Clearly, information cannot

[5]Since the tables return four bits, they actually contain two separate tables—regular and auxiliary.

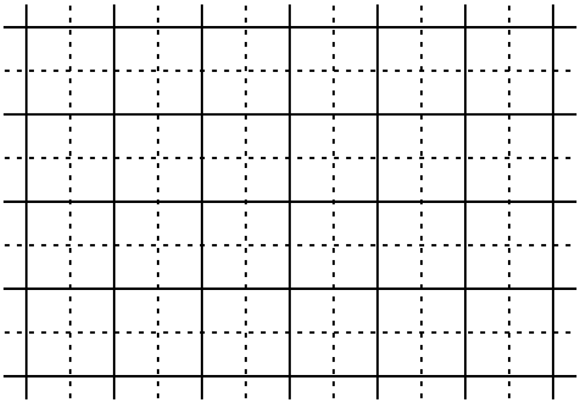


Figure 3-4: The Margolus neighborhood: the entire array is partitioned into 2×2 blocks, and the new state of each block only depends on the four cells in that block. The blocking can be changed from even to odd (solid to dotted) between time steps in order to couple all the blocks.

cross a partition boundary in a single time step, so the partitions must change from step to step in order to couple the space together. The Margolus neighborhood is a partitioning scheme where each partition is 2 cells by 2 cells as shown in figure 3-4. The overall pattern of these partitions can have one of four spatial phases on a given step. The most common way to change the phases is to shift the blocking both horizontally and vertically.

The partitioning format is especially good for many CA applications because it makes it very easy to construct reversible rules and/or rules which conserve "particles." To do so, one merely enforces the desired constraint on each partition separately. Furthermore, the 2×2 partitions in particular are very economical in terms of the number of bits they contain and are therefore highly desirable in cellular automata machines where the bits in the domain of the transition function are at a premium. The partitioning format has turned out to be so useful for realizing physical CA models that the spatial and temporal variation intrinsic to the Margolus neighborhood has been hardwired into CAM-6.

In addition to the above data processing elements, CAM-6 has a facility for doing some real-time data analysis. The most basic type of analysis one can do is to integrate a function of the state variables of the system. In a discrete world, this amounts to counting the number of cells having a neighborhood which satisfies a given criterion. Indeed, many measurements, such as finding the average number density in a given region, can be reduced to this type of local counting. Each step, the counter in CAM-6 returns a value in the range 0 to $256^2 = 65536$. Whether or not a cell is counted is determined with lookup tables in a similar way and at the same time that the cell is updated.

This concludes the description of the resources contained in CAM-6, and now I will outline a strategy for implementing a model. The design of a rule involves (1) giving an interpretation to the data in the planes corresponding to the system of interest, (2) determining the functional dependencies between cells and selecting appropriate neighborhoods, (3) mapping out a sequence of steps which will transform the data in the desired way, (4) defining the transition functions to be loaded into the lookup tables, (5) setting up the counters for any data analysis, and (6) loading an appropriate initial condition. Typically, one has to work back and forth through this list before coming up with a good solution. In the next section, this formula is applied to the potential energy model.

3.3.2 Description of the Rule

Recall that the point of this rule is to make a lattice gas *reversibly* accumulate in a region which has been designated as having a lower potential. The result will be a more ordered state, and thus it will have a lower entropy. However, it is impossible for an isolated, reversible system to exhibit spontaneous self-organization in this way.

To get around this limitation, it is necessary to introduce extra degrees of freedom which can, in effect, "remember" the details of how each particle falls into the potential well. This is accomplished by coupling the gas to a heat bath at a lower temperature. More specifically, the coupling occurs in such a way as to conserve energy and information whenever and wherever a gas particle crosses the contour of the potential. The heat bath consists of energy tokens (also known as demons) which behave as conserved particles except during interactions at the contour. In order to achieve a greater heat capacity and good thermalization, the particles comprising the heat bath are *deterministically* stirred with a second lattice gas dynamics. Introducing a heat bath in this way removes a constraint and opens up a larger configuration space into which the system can expand. Hence the entropy of the overall system can increase while that of the gas alone decreases.

The preceding discussion leads us to consider a model with three components: (1) a lattice gas, (2) a two-level potential, and (3) a heat bath. Each of these sub-systems requires a separate bit per cell, so the most natural thing to do is to devote a separate bit plane to each one. Given this, we can make a tentative assignment of data to the planes, conditioned on the availability of an appropriate neighborhood. Thus, the state of the system is represented as follows: in CAM-A, plane 0 is the lattice gas and plane 1 serves as the heat bath. In CAM-B, plane 2 contains a binary potential, and plane 3 ends up being used in conjunction with plane 2 to discern the *slope* of the potential as described below.

Now we turn to the choice of a neighborhood. Since the model incorporates two reversible lattice gases, the partitioning afforded by the Margolus neighborhood is a must. In addition, a key constraint is the weak coupling of CAM-A and CAM-B that was alluded to before: the only data available to one half of a cell from the other half of the machine are the two bits in the other half of the *same* cell. This is important because each half-cell can only change its state based on what it can "see." Given that there are three sub-systems, two must be in CAM-A and the third in CAM-B, and we have to justify how the planes were allocated above. The two lattice gases should be in the same half of the machine because the crossing of a contour involves a detailed exchange of energy and information between these two sub-systems. Furthermore, the potential is not affected in any way by the lattice gases, so it might as well be in the other half. As we shall see, it is possible for the lattice gases to obtain enough information about spatial changes in the potential in CAM-B with only the two bits available.

The dynamics of the system has four phases: two temporal phases which each take place on each of two spatial phases. The temporal phase alternates between even steps and odd steps. On even steps, the heat bath evolves and the potential

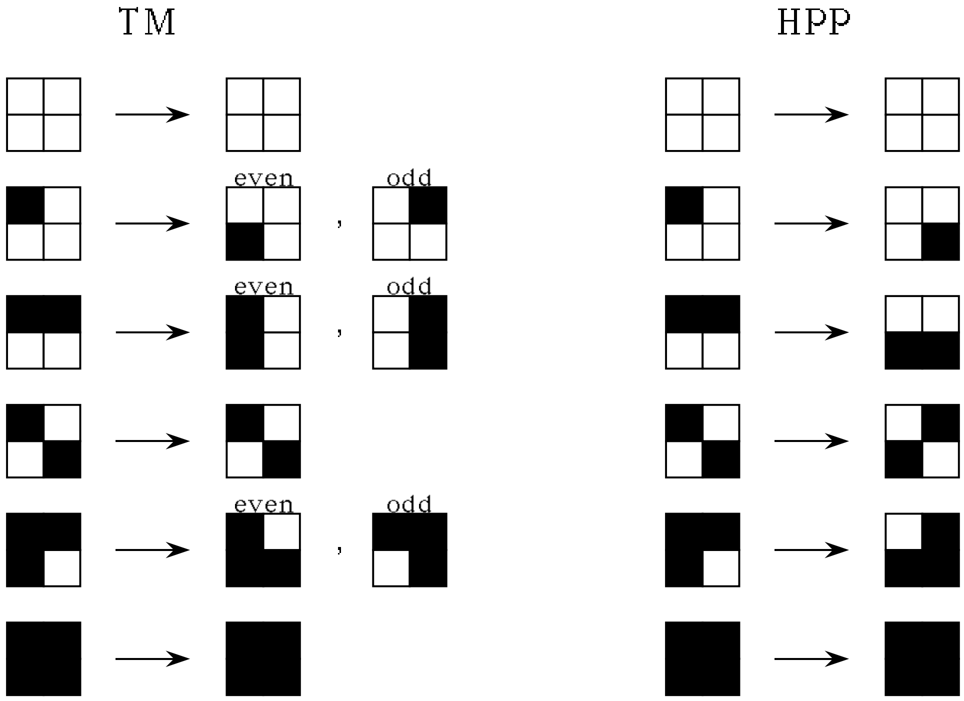


Figure 3-5: Transformation rules for the TM (left) and HPP (right) lattice gases.

is "gathered" in preparation for use by CAM-A. On odd steps, the gas evolves and, depending on the potential, interacts with the heat bath. After the odd steps, the partitioning of the space into 2×2 blocks is shifted to the opposite spatial phase (as shown in figure 3-4) in order to accommodate the next pair of steps.

The even time steps can be thought of as ancillary steps wherein bookkeeping functions are performed. During these steps, the gas particles are held stationary while the heat particles are *deterministically* and *reversibly* stirred with the "TM" lattice gas rule [89] as shown in figure 3-5 (left). In the TM rule, particles move horizontally and vertically unless they have an isolated collision with a certain impact parameter, in which case they stop and turn 90°. This constitutes a reversible, particle conserving, momentum conserving, nonlinear lattice gas dynamics. In CAM-B, the potential is gathered in plane 3, the manner and purpose of which will become clear below. For now, the gathering can be thought of as "differentiating" the potential to obtain the "force" felt by the particles in the neighborhood.

The dynamics of primary interest takes place in CAM-A on odd time steps, while

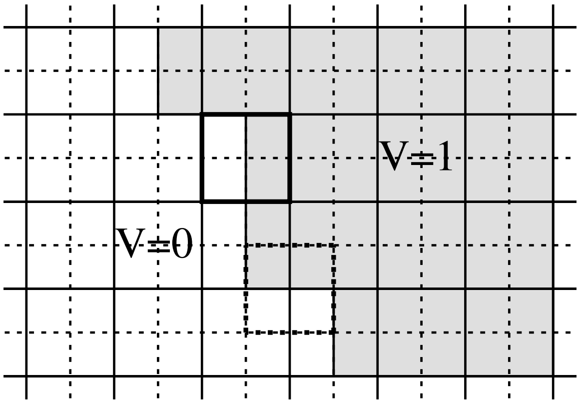


Figure 3-6: Diagram showing the relation of the even and odd Margolus neighborhoods to the edge of the potential. Using this scheme, any cell can detect a change in the potential by merely comparing its own potential with that of the opposite cell.

the data in CAM-B remains unchanged. In regions of constant potential, the gas particles follow the "HPP" lattice gas rule [89] as shown in figure 3-5 (right). A summary of this rule is that particles move diagonally unless they have an isolated, head-on collision, in which case they scatter by 90°. This rule has the property that the momentum along *every* diagonal is conserved—a fact which will be important when doing the statistical analysis of the model. The TM rule followed by the heat bath has a behavior similar to the HPP rule except that momentum along every line is *not* conserved—a fact which yields improved ergodic properties.

In addition to the dynamics described in the preceding paragraph, the key interaction with the potential takes place on odd time steps. If the neighborhood contains a change in the potential, as shown in figure 3-6, the HPP rule is overridden and each cell only interacts with its *opposite* cell (including the heat bath). By looking at the potential in the opposite cell as well as the center cell, a particle can tell whether or not it is about to enter or exit the potential well. The particle then crosses the boundary if there is not a particle directly ahead of it and the heat bath is able to exchange energy appropriately. Otherwise, the particle remains where it is, which results in it being reflected straight back. By convention, heat particles are released or absorbed in the cell on the interior of the potential.

Now it is finally clear why gathering is necessary. A limitation of the neighborhoods in CAM-6 does not allow a cell to access information in the other half of the machine unless it is in the same cell. Hence, the potential in the opposite cell must be copied into the extra bit in CAM-B before the system in CAM-A can be updated. The potential itself is never affected—only the copy in plane 3.

It is also possible to explain why the potential is aligned so that the contour always passes horizontally or vertically through the 2×2 blocks. Once the potential of the opposite cell has been gathered, the information in the CAM-B half of the cell allows each of the four cells in the neighborhood to independently determine that this is an interaction step and not an HPP step. Also, each of the cells knows whether the potential steps up or down and can evolve accordingly without having to look at the potential in any of the other cells.

The current implementation is also set up to measure several quantities of interest. These quantities are completely specified by counting (1) the number of particles inside the well, (2) the number of particles outside the well, and (3) the number of demons. These numbers can be initialized independently, but once the simulation has begun, conservation laws make it possible to find all three by only measuring one of them. The main quantities of interest that can be derived from these measurements are the temperature and the particle densities inside and outside the potential well.

The initial conditions for this particular experiment consist of a two-level potential and a gas of particles. The potential starts out as a single bit plane containing a disk 128 cells in diameter. This pattern must then be aligned with the Margolus neighborhood so that the contour always bisects the 2×2 blocks. This yields a bit plane with 52896 cells having a unit potential and a well of 12640 cells having zero potential. Plane 0 is initialized with particles to give a uniform density of 25 percent. The heat bath on plane 1 is initially empty corresponding to a temperature of absolute zero.

3.3.3 Generalization of the Rule

The implementation described above is rather limited in that it only allows binary potentials. However, it is important to note that only *differentials* in the potential

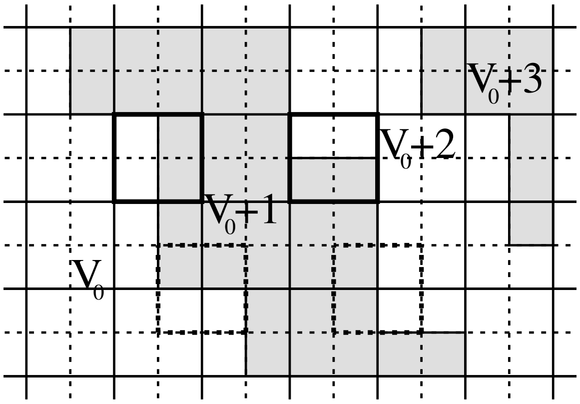


Figure 3-7: Contours of a multi-level potential showing how neighborhoods having the same spatial phase can be interpreted as positive or negative depending on whether the contour is horizontal or vertical.

are actually used to generate the force. This observation leads to an encoding of the potential in terms of its gradient. Despite having only one bit per cell, such a difference encoding makes it possible to represent any potential having a limited slope.

The specific encoding used here is illustrated in figure 3-7. An integer-valued approximation to a potential is stored, modulo 2, in a single bit plane. Without additional information, this would merely give a corrugated binary potential—the potential would just alternate between zero and one. There has to be a way of reinterpreting "negative" contours which step down when they should step up. This can be done by insisting that, with a given partitioning, the negative contours cross the partitions with the opposite orientation as the positive contours. In particular, when the spatial phase of the partitions is even, horizontal contours are inverted and when the phase is odd, vertical contours are inverted.

It remains to be shown how this scheme can be incorporated into the CAM-6 rule above. In order to update the system, the two bits in the CAM-B half of a cell should indicate the directional derivative along the particle's path, i.e., whether the potential in the opposite cell higher, lower, or the same as the center cell. Therefore, it would suffice to complement both of these bits while gathering whenever a negative contour is encountered. This way, the rule for CAM-A can remain exactly the same—only the gathering, which occurs entirely within CAM-B, has to be modified.

In the basic model, most of the resources of CAM-B are idle, but now they will be needed to compute the force from the encoded potential for use by CAM-A. The computation involves comparing the orientation of the potential in a neighborhood with the spatial phase to discern if a contour is positive or negative. Thus on even steps, the potential is gathered and, if necessary, inverted in place. On the following odd steps, the update can proceed as usual. Finally, one more administrative task has to be performed. In the binary-potential model the data in CAM-B may be left unchanged as the system is being updated. However in the extended model, the gathering of the potential must be reversed at that time in order to restore the negative contours of the potential to their uninverted state.

An example system which uses this rule is shown in figure 3-8. In (a) is the special binary encoding of a harmonic potential revealing the peculiar way in which the contours follow the spatial phase. The radial coordinate could be taken to represent position or momentum depending on the interpretation one assigns. In the former case the potential energy is given by $V = \frac{1}{2}m\omega^2 r^2$, and in the latter case the kinetic energy is given by the classical expression $T = p^2/2m$. The initial state of the gas is exactly the same as in the previous experiment (figure 3-1(b)), but the outcome is rather different. Since the gas falls into a deeper well than before, more energy is released into the heat bath, and it is quickly saturated. The gas will be able to fall deeper into the well if the system is cooled, and energy can be (irreversibly) removed by clearing the demons in plane 1. After cooling and equilibrating the system five times the result is shown in figure 3-8(b).

3.4 The Maximum Entropy State

At the end of section 3.2, a discrepancy was noted between the elementary statistical analysis and the time averages obtained from a long run of the simulation. Recall what is assumed for them to agree: (1) the ensemble of states that will be generated by the dynamics is completely defined by the conserved energy and particle number, (2) the

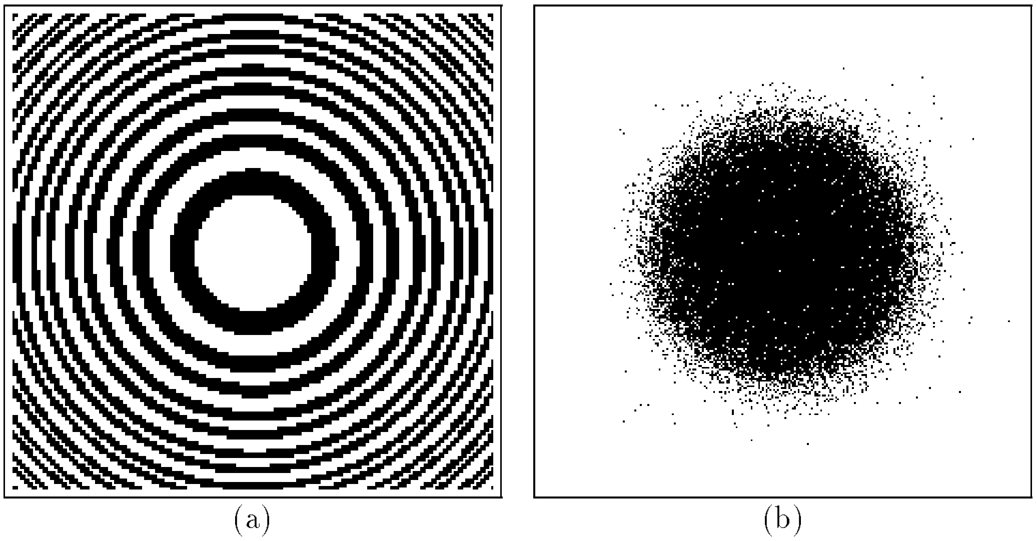


Figure 3-8: (a) Multi-level potential corresponding to a quadratic energy dependence on the radial coordinate. (b) The final configuration of the gas, yielding the characteristic Fermi sea (interpreted as a picture in momentum space).

calculated maximum entropy state is representative of the entire ensemble, and (3) the sequence of measured configurations taken from the simulation are representative of the entire ensemble. Which of these assumptions is in error? It turns out that all three may be partially responsible, but the primary culprit is point (1). In particular, any additional conserved quantities restrict the accessible region of configuration space, and one such set of conserved currents will be treated in the following section. With regards to point (2), one must be careful when redoing the calculation to take into account the finite size of the system as discussed below. The relevance of point (3) will show up again in section 3.5.

3.4.1 Broken Ergodicity

It should be clear from the preceding sections that discrepancies between measured and calculated expectation values could be a sign of previously ignored or unknown conservation laws. Each conserved quantity will constrain the accessible portion of the configuration space of the system, and the true ensemble may differ from the microcanonical ensemble. Furthermore, the change in the ensemble may (or may not) be reflected in calculated expectation values. If this is the case, one may say that ergodicity is broken because the time average is not the same as a standard ensemble average. However, the nature of the deviation provides a hint for finding new integrals of the motion [45]. Therefore, when faced with such a discrepancy, one should consider the possibility of hidden conservation laws.

The data in table 3.1 show that, in equilibrium, there are fewer particles in the well than a simple calculation would suggest, and we would like to see if this can be explained by constraints that were previously unaccounted for. The analysis given in section 3.2 made minimal assumptions about the dynamics, but additional conservation laws will, in general, depend on the details. Hopefully, we can find some reason for the discrepancy by looking back over the description of the rule in section 3.3. One possibility that was mentioned in connection with the HPP lattice gas is that there is a conserved momentum or current along every single diagonal line in the space. In the present model, the particles bounce straight back if they cannot cross the boundary of the potential, so the conserved currents only persist if the diagonals do not intersect the potential. However, with periodic boundary conditions and the particular potential well used here, there happen to be a total of $L = 74$ diagonal lines (37 sloping either way), each of length $N = 256$, that do not cross the potential and are entirely outside the well (see figure 3-9(a)). Thus, region AB naturally divides into A and B, where all the lines in region A couple to the potential while those in region B do not. In addition, region A crosses region B perpendicularly, so they are coupled to each other by collisions in this overlap region.

The current on each diagonal line in region B consists of a constant difference in the number of particles flowing in two directions: $j = n_+ - n_-$. The total number of particles on the diagonal, $n = n_+ + n_-$, is not constant, but it clearly has a minimum of $|j|$ (and a maximum of $N - |j|$). The minimum currents on all the lines can be demonstrated by repeatedly deleting all the particles (and all the demons) that fall into the well, since any excess will scatter into region A.[6] The residual configuration shown in figure 3-9(b) was obtained in this manner. This configuration has a total

[6]There is a small chance that they will scatter where region B intersects itself, but a different sequence of erasures can be tried until this does not occur.

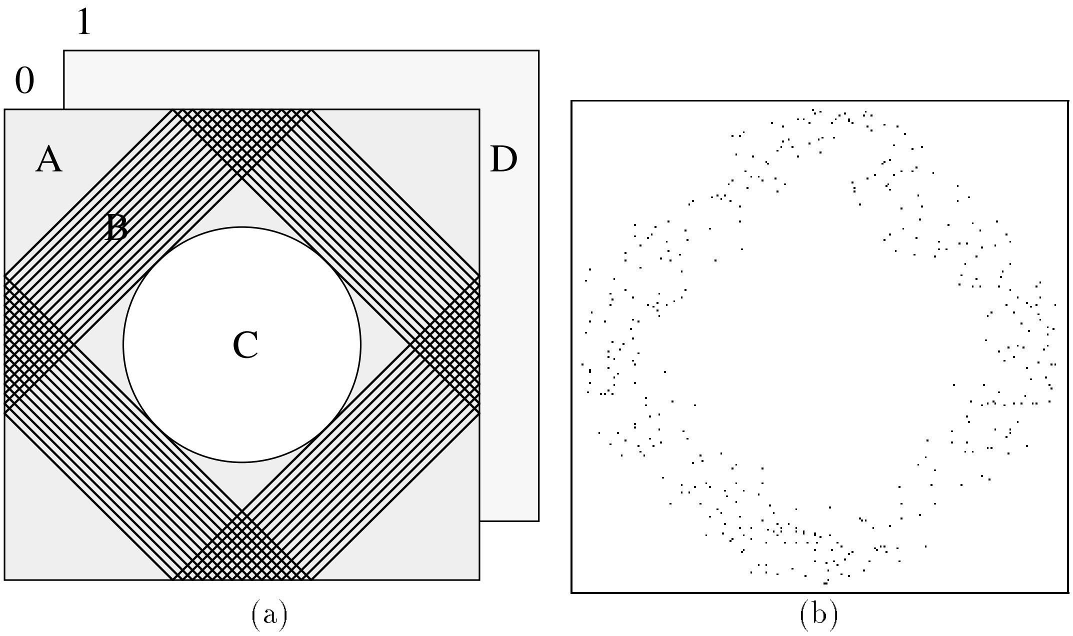


Figure 3-9: (a) The outer region is divided into two parts (A and B) in order to isolate the effects of additional conserved quantities. (b) The current remaining in region B after any extra particles on each line have scattered out due to head-on collisions.

of 394 particles which can never fall into the well, so the mean magnitude of the current on each line is $\langle |j| \rangle \cong 5.32$. The actual discrepancy in the particle counts are much less than 394 because the currents are screened by the presence of all the other particles, but the demonstration shows that the effect of these additional conservation laws will be to reduce the number of particles n_C in the well in equilibrium, provided that the density of the initial configuration is low. The numbers given above can be explained by considering the disorder contained in the initial configuration (figure 3-1(b)). Detailed calculations of all the results stated in this section can be found in appendix B.

In order to determine the effect of these currents on the entropy, one must recount the number of available states and take the logarithm. One attempt at the counting assumes that every line has the same average number of particles n and then factors each line into two subsystems contain opposing currents. The total entropy of all L lines in region B is then

$$S_B \cong L \ln \binom{N/2}{\frac{n+j}{2}} \binom{N/2}{\frac{n-j}{2}}, \tag{3.14}$$

and this can be expanded to second order in j by making the usual approximation $\ln \mathcal{N}! \cong \mathcal{N} \ln \mathcal{N} - \mathcal{N}$. When $j = 0$, this returns the same result as the basic analysis of section 3.2, but we are interested in the net change. Since the entropy is a physically measurable quantity, the separate values of j^2 for each line can be replaced with the expected value $\langle j^2 \rangle = N\rho_0\overline{\rho}_0$, where $\rho_0 = 1/4$ is the initial density of particles in region B. Finally, the net change in the entropy as a function of the final density in region B is

$$(\Delta S_B)_{nonergodic} = -\frac{L\rho_0\overline{\rho}_0}{2\rho_B\overline{\rho}_B}. \tag{3.15}$$

Note that this correction is not extensive since it is proportional to L rather than to $N_B = NL$.

The logical effect of (3.15) is to reduce the entropy as the conserved currents increase. And since $\rho_B\overline{\rho}_B$ reaches a maximum at $\rho_B = 1/2$, it tends to favor intermediate densities in the maximum entropy state. This is a direct manifestation of the difficulty of pumping particles out of region B. The effect of adding this term to the entropy is tabulated in appendix B, but the correction overshoots the experimental value and actually makes the theoretical answer even worse. Therefore, we must reexamine the analysis.

3.4.2 Finite Size Effects

The combinations above give the exact number of ways of distributing n_x particles in N_x cells, but an approximation must be made when taking the logarithm. The size of the approximation can be gauged by looking at the following asymptotic expansion [1]:

$$\ln \mathcal{N}! \sim \mathcal{N} \ln \mathcal{N} - \mathcal{N} + \tfrac{1}{2} \ln \mathcal{N} + \tfrac{1}{2} \ln 2\pi + \frac{1}{12\mathcal{N}} - \frac{1}{360\mathcal{N}^3} + \cdots. \tag{3.16}$$

For any given number of terms, this expansion is better for large $\mathcal{N}$, but for small $\mathcal{N}$, more terms are significant and should be included. In the present calculation, the logarithmic term, $\frac{1}{2} \ln \mathcal{N}$, is included for the factorial of any number smaller than $N = 256$. These small numbers show up because of the way region B must be divided up into L regions of size N in order to compute the effect of the individual currents.

Despite being one level farther down in the expansion, the logarithmic terms give contributions comparable in size to the conservation term (3.15). When terms of this order are included, the approximation of equal numbers of particles on every line must also be modified as shown in appendix B. Thus, the change in the entropy due to logarithmic terms as a function of the final density in region B is

$$(\Delta S_B)_{finite} = -\frac{L}{2} \ln \rho_B \overline{\rho}_B, \tag{3.17}$$

where constant terms have been omitted. As before, this correction is not extensive since it is proportional to L rather than to $N_B = NL$.

The physical meaning of (3.17) is not as clear as the meaning of (3.15), but it has a compensatory effect. The former is lower when $\rho_B \overline{\rho}_B$ is higher, and this tends to push the maximum entropy state away from $\rho_B = 1/2$. Together, the above corrections bring the experimental and theoretical values into excellent agreement. The effect of adding (3.17) alone to the entropy is also tabulated in the appendix, so the size of its contribution can be seen separately.

3.4.3 Revised Statistical Analysis

Adding the corrections (3.15) and (3.17) to the bulk entropies for each region gives the revised expression for the total entropy,

$$\begin{aligned} S &= \ln \Omega \\ &\cong N_A(-\rho_A \ln \rho_A - \overline{\rho}_A \ln \overline{\rho}_A) + N_B(-\rho_B \ln \rho_B - \overline{\rho}_B \ln \overline{\rho}_B) \\ &\quad + N_C(-\rho_C \ln \rho_C - \overline{\rho}_C \ln \overline{\rho}_C) + N_D(-\rho_D \ln \rho_D - \overline{\rho}_D \ln \overline{\rho}_D) \\ &\quad - \frac{L\rho_0\overline{\rho}_0}{2\rho_B\overline{\rho}_B} - \frac{L}{2} \ln \rho_B \overline{\rho}_B, \end{aligned} \tag{3.18}$$

while the revised constraints on the particle number and energy are given by

$$n_T = n_A + n_B + n_C \quad \text{and} \quad E_T = n_A + n_B + n_D. \tag{3.19}$$

The solution for the maximum entropy state under these constraints exactly parallels the derivation given before, and the resulting densities are

$$\rho_A = \frac{1}{1+e^{\beta(1-\mu)}}, \tag{3.20}$$

$$\rho_B = \frac{1}{1+e^{\beta(1-\mu)}\exp\left\{\frac{1}{2N}\frac{(1-2\rho_B)}{\rho_B\overline{\rho}_B}\left(1-\frac{\rho_0\overline{\rho}_0}{\rho_B\overline{\rho}_B}\right)\right\}}, \tag{3.21}$$

$$\rho_C = \frac{1}{1+e^{\beta(-\mu)}}, \tag{3.22}$$

$$\rho_D = \frac{1}{1+e^{\beta}}. \tag{3.23}$$

The densities in regions A, C, and D obey Fermi statistics as before since the thermodynamics limit applies to them. However, the density in region B reflects the fact that it is broken up into $L = 74$ finite-sized regions, each of which contains an additional conserved current. The analysis presented in appendix B manages to describe all of these subsystems with single average.

Since region AB has been split in two, we now have six equations in six unknowns. Equations (3.20)–(3.23) can be combined to eliminate β and μ and express the maximum entropy state directly in terms of the numbers of particles in the four regions:

$$n_A(N_C - n_C)(N_D - n_D) = n_C n_D(N_A - n_A) \tag{3.24}$$

$$n_A(N_B - n_B) = n_B(N_A - n_A)\exp\left\{\frac{1}{2N}\frac{(1-2\rho_B)}{\rho_B\overline{\rho}_B}\left(1-\frac{\rho_0\overline{\rho}_0}{\rho_B\overline{\rho}_B}\right)\right\}. \tag{3.25}$$

These equations are essentially statements of detailed balance: the first is in actuality the same as equation (3.12) and ensures equilibrium of the reaction $A \rightleftharpoons C + D$, while the second equation (squared) ensures equilibrium of the reaction $2A \rightleftharpoons 2B$. Broken ergodicity and finite size effects have the net effect of skewing the later reaction to the right hand side since $\rho_B\overline{\rho}_B < \rho_0\overline{\rho}_0$. Note that the extra factor is less important when the final density in region B happens to be close to the initial density. These equations in conjunction with (3.19) comprise four equations in the four unknown equilibrium particle numbers. Solving gives the revised results shown in table 3.2, where the numbers for regions A and B have been added together for comparison

x	T	AB	C	D
N_x	65536	52896	12640	65536
n_x (initial)	16191	13063	3128	0
n_x (revised theory)	16191	7798.42	8392.58	5264.58
n_x (experiment)	16191	7798.44	8392.56	5264.56

Table 3.2: A revised version of table 3.1. The theoretical values have been updated to take into account additional conserved quantities and a logarithmic correction to the entropy.

with table 3.1.

The agreement in table 3.2 is very good, but a certain amount of luck must be acknowledged because of the many possible sources of disagreement. For one thing, approximations were involved in the way the states were counted and in the way the disorder in the initial configuration was averaged out. The expansion of $\ln \mathcal{N}!$ involved an asymptotic series, and it is not clear how many terms to keep. Additional terms could account for more subtle finite size effects, but still others could make the calculation worse. There is also a conceptual mismatch in that theoretically, we calculate the mode of the ensemble, whereas experimentally, we measure ensemble averages. This would be a problem if the distributions of the observables were skewed. Finally, there is the possibility of further conservation laws (known or unknown). For example, the HPP gas has another conservation law in the form of the moving invariant shown in figure 2-13. While the "soliton" waves would not survive upon hitting the potential well, some version of the invariant could very well persist. Another set of invariants that the gas particles in the potential-well rule definitely have is the parity of the number of particles (0 or 1) on each of the $2N$ diagonals. These are not all independent of the conservation laws previously mentioned, but they would probably have a weak effect on some expectation values. Still another type of conservation law has to do with discrete symmetries: a symmetrical rule that acts on a symmetrical initial configuration cannot generate asymmetrical configurations.

3.5 Results of Simulation

This section describes how the numerical experiments on the potential-well system were performed on CAM-6. The experiments consist of five runs, each of which was started from the initial condition described in section 3.2. A run consists of letting the system equilibrate for some long period of time and then taking a running sum of measurements for another long period of time. The system was also allowed to relax a short period of time (20 steps) between measurement samples.[7] For each sample, the numbers of particles in regions AB, C, and D were counted separately and added to their respective running totals. Only one of the three counts is independent because of the constraints (3.1), but having the sum of all three provides sensitive cross checks for possible errors. If any errors occur, the simulation can just be run again.

The results of the simulations are shown in table 3.3. The numbers in the center columns are are just the running totals in the respective regions divided by the number of samples. The column labeled t gives the time period over which each simulation ran. The first number gives the equilibration time, and the second number is the finishing time. The number of samples taken is the difference of these times divided by 20. Thus, runs #1 and #2 contain 10,000 samples each while runs #3–5 each contain 300,000 samples each. Note that none of the sampling intervals overlap, and in practice, the earlier runs serve as equilibration time for the later ones. CAM-6 runs at 60 steps per second, but since there is some overhead associated with taking the data, the sequence of all five runs required about a week to complete.

The measurements consist of exact integer counts, so the only possible source of error on the experimental side would be a lack of sufficient statistics. One way the statistics could be misleading is if the system were not ergodic, though this could just as well be considered a theoretical problem. The particular initial condition may happen to produce a short orbit by "accident," but this could also be attributed to an ad hoc conservation law. The statistics could also suffer if the equilibration time

[7]As described in section 3.3, it takes two steps to update both the particles and the heat bath. This makes for 10 updates of the system between samples, but t will continue to be measured in steps.

x	T	AB	C	D	t
N_x	65536	52896	12640	65536	–
n_x (initial)	16191	13063	3128	0	0
n_x (expt. run #1)	16191	7801.26	8389.74	5261.74	10–210k
n_x (expt. run #2)	16191	7799.31	8391.69	5263.69	210–410k
n_x (expt. run #3)	16191	7798.43	8392.57	5264.57	2–8M
n_x (expt. run #4)	16191	7798.51	8392.49	5264.49	8–14M
n_x (expt. run #5)	16191	7798.38	8392.62	5264.62	14–20M
n_x (expt. average)	16191	7798.44	8392.56	5264.56	2–20M
n_x (figure 3-9(b))	394	0+394	0	0	–

Table 3.3: Experimental values for the time averaged number of particles in each region of the system. The data is taken from simulations using the same initial conditions but from nonoverlapping time intervals. The final average is over the last three runs.

is too short. For example, the averages for runs #1 and #2 seem to indicate that the system is approaching the final equilibrium averages gradually, but since there are only 10,000 samples in each of these runs, this could also be due to random fluctuations. Accumulation of data for run #1 starts from figure 3-2(b) which certainly *looks* like an equilibrium configuration, but perhaps it still has some macroscopic memory of the initial state. Finally, a third way the statistics could be poor is if the run is too short to sample the ensemble uniformly. Because of the small amount of time between samples, consecutive counts are by no means independent. If they were, the standard error in the measured values would be the standard deviation divided by the square root of the number of samples. The standard deviation was not measured, but the individual counts comprising the totals were observed to fluctuate within about 100–200 particles of the averages, and this provides a reasonable estimate instead. Since we have 900,000 samples in the final average, a crude estimate of the standard error is $\xi \approx 0.16$ particles. The averages in runs #3–5 are well within the resulting error bars. One should really measure the relaxation time (as is done in the polymer simulations in chapter 5) to get an idea of how good the statistics are.

3.6 Applications and Extensions

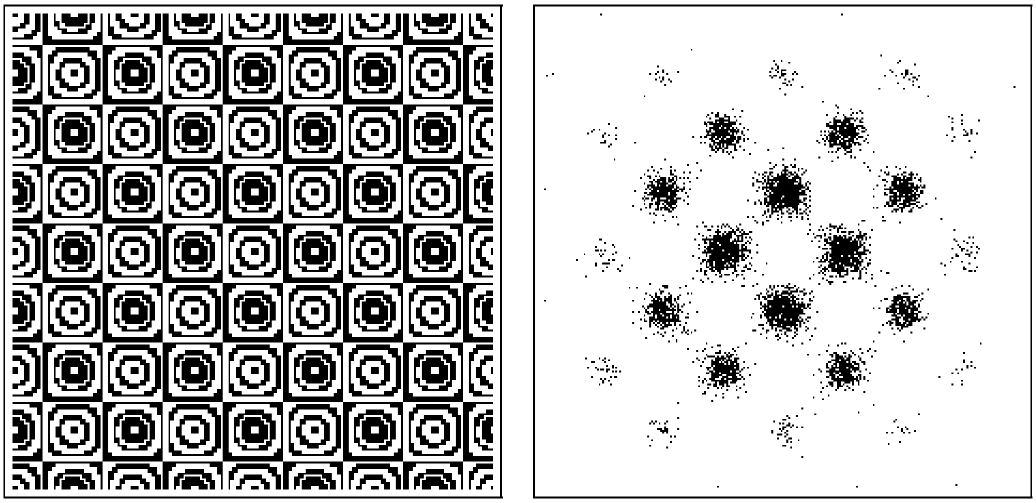
The model presented here is interesting both in terms of its applicability to further modeling with CA and for the basic scientific issues it raises. The technique introduced for modeling potentials and statistical forces can be used with little or no further development. Microcanonical heat baths will certainly find application in thermodynamic modeling and as a method for controlling dissipation. This section presents just a few ideas and speculations on how these techniques might be used or extended. Hopefully the discussion will spark further ideas for modeling and mathematical analysis.

3.6.1 Microelectronic Devices

Several interesting physical systems consist of a fermi gas in a potential well including individual atoms and degenerate stars. Another very common and important situation that involves the diffusion of particles in complicated potentials is that of electrons in integrated circuits. This topic is pertinent to information mechanics because of our interest in the physics of computation, particularly in the context of cellular computing structures. While detailed physical fidelity may be lacking in the present model, it could probably be improved by introducing some variation of stochastic mechanics [64]. Nevertheless, it is worthwhile to pursue these simple models because interesting findings often transcend specific details.

Figure 3-10(a) shows a CAM-6 configuration of a potential which is intended as a crude representation of a periodic array that might be found on or in a semiconductor. In order of decreasing size, some examples of such periodic electronic structures are dynamic memory cells, charge-coupled devices, quantum dots, and atoms in a crystal lattice. The actual potential used is a discrete approximation to $\cos x \cos y$ (in appropriate units), and the alternating peaks and valleys which result are the 32×32 squares visible in the figure.

The system was initialized with a 64×64 square of particles in the center and no energy in the heat bath. Hence, the number of particles is just sufficient to cover



(a) (b)

Figure 3-10: (a) A multi-level potential representing a periodic array on an integrated circuit. (b) Particles in the center working their way outward via thermal hopping from well to well.

exactly two peaks and two valleys, so there will be a net release of energy into the heat bath as the particles seek their level. After 10,000 steps, figure 3-10(b) shows how the particles can diffuse from well to well. Depending on the situation, the lattice gas could represent either a collection of electrons or probability densities for single electrons. One possible research problem based on this example would be to determine the rates of thermodynamic transitions between the wells as a function of temperature.

One conceptually straightforward modification to the rule would be to have time dependent potentials. By switching portions of the potential on and off in the appropriate ways, it would be possible to shuffle the clouds around in an arbitrary way. An important improvement in the model would be to have one cloud modulate the behavior of another, but finding a good way to do this is an open area of research. Another interesting challenge would be to model a bose gas in a potential well.

3.6.2 Self-organization

The real strength of CA as a modeling medium is in the simulation of the dynamical phenomena of complex systems with many degrees of freedom. An important aspect of such systems is that they are often marked by emergent inhomogeneity or *self-organization* which occurs in non-equilibrium processes. Rather than approaching a uniform state as might be expected from an increase in entropy, many macroscopic systems spontaneously form a wide variety of interesting structures and patterns [70]. Examples of such dynamical behavior can be found in disciplines ranging from geology and meteorology to chemistry and biology [97]. The concepts discussed in this chapter are relevant to the study of self-organization in a number of ways.

Self-organization in a dynamical system is characterized by having a final state which is more ordered than the initial state—i.e., it has a lower coarse-grained entropy.[8] Given what was said above about the connection between entropy and reversibility, self-organization clearly requires irreversibility or an open system which allows entropy (or information) to be removed. The actual process of self-organization involves "reactions" that occur more often in one direction than in the reverse direction, and this bias can be said to arise from forces. In the potential-well rule, the forces are entirely statistical which means that they are, in fact, *caused* by an increase in entropy. However, the increase in entropy takes place in the heat bath in such a way that the entropy of the system itself actually decreases. Conservation laws also play an important role in the dynamics of self-organization because they determine how the configuration of a system *can* evolve, while the entropy determines, in part, how it *does* evolve. Furthermore, the organized system usually consists of identifiable "particles" which are conserved.

Dissipation refers to the loss of something by spreading it out or moving it elsewhere. In the present context, it usually refers to the loss or removal of information from the system of interest. However, the term could also be applied to energy, momentum, particles, or waste products of any kind. The potential-well model suggests a general method of incorporating dissipation of information in a reversible scheme. Whenever an interaction would normally result in a many-to-one mapping, it is only allowed to proceed when the extra information that would have been lost can be writ-

[8]Others may also talk about various measures of complexity such as algorithmic complexity and logical depth, but for purposes of this discussion, we are more interested in statistical mechanics than, say, natural selection.

ten to a heat bath. In this case, the heat bath is acting very much like a heat sink. Just as heat sinks are required for the continual operation of anything short of a perpetual motion machine, the reversible dissipation of information requires something like a heat bath. More generally, a heat bath can be used to absorb whatever is to be dissipated, and if one is careful, it can be accomplished reversibly. Finally, reversibly coupling a system to a heat bath in a low entropy state is a disciplined way of adding dissipation and statistical forces to any rule. The energy or information which falls into the bath can then be deleted gradually, and this gives us better control over how the dissipation occurs. It is less harsh than using an irreversible rule directly because the degrees of freedom cannot permanently freeze.

Let us think for a moment about how the process of self-organization might work in CA in terms of abstract particles. Some features that seem to characterize self-organizing systems are (1) energy, (2) feedback, (3) dissipation, and (4) nonlinearity; their respective roles are as follows: (1) Particles are set into motion by by virtue of having an energy. (2) The placement of the particles is dictated by forces which are caused by the current state of the system. (3) When a particle comes to rest, the energy associated with the motion as well as the information about the particle's history are dissipated. (4) The above processes eventually saturate when the system reaches its organized state. All of these features except feedback can be identified in the potential-well model—the organization is imposed by an external potential rather than by the system itself.

The information about the state of a lattice gas is entirely encoded in the position of the particles since they have no other structure. During self-organization, the positions of many particles become correlated, and the positional information is lost. Similar comments apply to the self-organization of systems consisting of macroscopic particles (for example, think of patterns in sand). Also, the role of energy in moving particles around may be played by "noise" of another sort. The real point is that the dissipation of entropy (or information) and "energy" that was discussed above can equally apply to large scales, not just at scales of k or kT. In fact, a CA simulation makes no reference to the actual size of the system it is simulating. These final comments just serve to show that self-organization has more to do with the behavior of information rather than with the behavior of matter.

3.6.3 Heat Baths and Random Numbers

One drawback with the heat bath described above (and its generalizations from appendix A) is that it has a relatively small heat capacity. This is a problem because its temperature will not remain steady as it interacts with the main system. However, the potential-well model itself presents a technique for increasing the capacity as can be understood by looking back at figure 3-8. Instead of a particle holding a single unit of energy, it absorbs a unit of energy every time it jumps up a contour. By having a steep V-shaped potential, it would be possible to have 128 contours in the 256×256 space of CAM-6. The gas in the potential would therefore buffer the temperature of the heat bath by absorbing and releasing energy as needed. The heat bath would not equilibrate as quickly however, and in general, there is a tradeoff between the heat capacity of the bath and its statistical quality.

Microcanonical heat baths have been offered as a technique for adding dissipation to a deterministic CA model while maintaining strict microscopic reversibility. Such dissipation is a nonequilibrium process which can be used to generate statistical forces. However, once the system is near equilibrium, the same heat bath plays the seemingly contradictory role as a random number generator. On one hand, it is known that the heat bath is perfectly correlated with the primary system, and on the other hand, it is assumed that it is uncorrelated. How is this to be understood?

Intuitively, the dynamics are usually so complex that the correlations are spread out over the whole system, and they are unlikely to show up in a local, measurable way. The complexity is reflected, in part, by the extremely long period ($T_0 \sim 2^{\mathcal{O}(N^2)}$) of the system (after which all the correlations come back together in perfect synchrony). The period could be substantially shorter if there are conservation laws that impose a hidden order on the randomness, so any such degeneracy would say something interesting about the CA's dynamics. As in any stochastic computer simulation, one must be cognizant of the possibility that the quality of random numbers may be lacking and be prepared to deal with the problem. The mathematical problems and philosophical questions associated with the use of deterministic randomness are by no means settled.

3.6.4 Discussion

There are many topics related to the potential-well model which have yet to be explored. This section gives a list of suggestions, observations, and problems from which the research could be continued.

One project using the exact same system would be to keep statistics on the number of particles in region A and B separately to check how well the predicted numbers agree with the simulation. Another project would be to test the effect of removing the additional conserved currents. This could be done by introducing interactions which do not respect the conservation law responsible for constraining the system. For example, when the gas particles are are not interacting at the edge of the potential, they could be allowed to collide with the demons as well as each other.

The condensation of the gas in the well was caused by statistical forces arising through a coupling of the system to a heat bath in a low entropy state. However, there are other ways to generate reversible statistical forces. For example, instead of having a heat bath, one could allow the particles to have one of several different "kinetic" energy levels, provided that there are more states available to higher energy particles. The key interaction would then be to increment the energy of a particle when it falls into the well and decrement the energy when it leaves. Such a gas that starts with most of the particles in the low energy states would again preferentially fall into the well because of the resulting increase in entropy. The associated increase in thermal energy at the expense of potential energy would be similar to what happens to a real gas when it is placed into an external potential. The gas would fall into the well because it would result in a decrease in the free energy, even though the internal energy (kinetic plus potential) would remain constant. One final thing to ponder along these lines is the use of the term "energy." In fact, the entire system is made out of nothing but information in the memory of a computer. To what degree are other identifications between physics and information possible?

One of the purposes of the potential-well model was to develop a technique for incorporating forces in CA simulations. While the model is very successful in its own domain, the statistical forces which result seem to be quite unlike those encountered in elementary mechanical systems. In particular, statistical forces act dissipatively on many independent degrees of freedom in nonequilibrium situations. Mechanical forces, on the other hand, act on individual bodies whose detailed internal structure, if any, seems to not matter; furthermore, the forces are often macroscopically reversible. When thinking about how to use CA for general physical modeling, the problem of forces acting on macroscopic solid bodies comes up in many contexts (see chapter 6).

The potential-well model illustrates many aspects of statistical mechanics and thermodynamics, but one important ingredient that is missing is a treatment of work. Work being done by or on a thermodynamic system requires the existence of a variable macroscopic external parameter. The prototypical example is volume (which can be varied with a piston). Such a parameter would have a conjugate force and a corresponding equation of state. In principle, it would be possible to construct a heat engine to convert energy from a disordered to an ordered form, and the maximum efficiency would be limited by the second law of thermodynamics. All of this would be interesting to check, but it seems to be predicated on first solving the problem of creating macroscopic solid bodies.

Finally, note that the potential itself does not possess a dynamics. If the potential does not move, the force it generates cannot be thought of as an exchange of momentum since momentum is not conserved. An external potential such as this which acts but is not acted upon is somewhat contrary to the physical notion of action and reaction. A more fundamental rule would have a dynamics for all the degrees of freedom in the system. At the very least, one would like to have a way of varying the potential as a way of doing work on the system. One idea for having a dynamical potential would be to use lattice polymers to represent the contours of the potential (see chapter 5). Another interesting possibility would be to use a time dependent contour map as a potential, especially if the potential could be made to depend on the gas. CA rules that generate such contour maps have been used to model the time evolution of asynchronous computation [58].

3.7 Conclusions

Microscopic reversibility is a characteristic feature of fundamental physical laws and is an important ingredient of CA rules which are intended to model fundamental physics. In addition to providing dynamical realism, reversibility gives us the second law of thermodynamics; in particular, the disorder in the system will almost always increase or stay the same. In other words, the entropy can only increase, and yet at the same time, the dynamics can be run backwards to the exact initial state. An examination of the model presented here should remove most of the mystery typically associated with this fact.

The related concepts of force and energy are essential elements of physics because they are intimately tied up with determining dynamics. Unfortunately, the usual manifestation of forces in terms of position and acceleration of particles does not occur in CA because space and time are discrete. However, in many circumstances, forces can be derived from potentials, and it is possible to represent certain potential energy functions as part of the state of the CA. Such potentials can then be used to generate forces statistically, as in the example of a lattice gas in the vicinity of a potential well.

A statistical force is any systematic bias in the evolution of a system as it approaches equilibrium. Normally one thinks of such nonequilibrium processes as dissipative and irreversible. However, one can arrange for prespecified biases to occur in a reversible CA—while maintaining the desired conservation laws—by appropriately coupling the system to another system having a low entropy. The entropy of one subsystem may decrease as long as the entropy of the other subsystem increases by an equal or greater amount. This is the key to obtaining self-organization in reversible systems. In general, the greater the increase in entropy, the stronger the bias.

The concept of entropy usually applies to an ensemble or probability distribution, but this chapter shows how it can be successfully applied to individual states by coarse graining. The entropy of a state is therefore defined as the logarithm of the number of microscopic states having the same macroscopic description. Entropy increases when there are no conservation laws preventing it from doing so because there are *vastly* more states with higher entropy. Since the entropy increases while the total number of states is finite, the system eventually reaches the maximum entropy state defined by a set of constraints, though there will be fluctuations around this maximum. The final result of statistical forces acting in a reversible CA can be quantified by finding this equilibrium state, and it can be used to calculate expectation values of measurable macroscopic quantities. Any disagreement between theory and experiment can be used to find conservation laws.

Cellular automata machines amount to flexible laboratories for making precise numerical measurements, and they enable close and fruitful interaction between theory and experiment. In the context of the present model, they are also useful for demonstrating important ideas in statistical mechanics and provide an instructive complement to algebraic calculations. Reviewing the implementation of a CAM-6 rule in detail reveals a definite methodology for developing CA models and could help in the future development of hardware and software.

The model presented in this chapter opens up a number of lines for further research. The most direct line would be to apply the model and its generalizations to systems that consist of a gases in external potentials. Another avenue of research would entail studying the purely numerical and mathematical aspects of these models such as fluctuations, nonequilibrium processes, mathematical calculation methods, and ergodic theory. Still another related area is that of self-organization and pattern formation. Finally, there is the original, broad, underlying question of how to model forces among the constituents of a CA system.

Chapter 4

Lorentz Invariance in Cellular Automata

4.1 Introduction

4.1.1 Relativity and Physical Law

The theory of special relativity is a cornerstone of modern physics which has broad implications for other physical laws and radically alters the classical view of the very fabric of physics: space and time. The consequences of relativity show up in the form of numerous interesting paradoxes and relativistic effects. Despite its far-ranging significance, special relativity follows from two innocuous postulates. The principal postulate is the *Principle of Relativity* which states that the laws of physics must be the same in any inertial frame of reference.[1] Any proposed fundamental law of physics must obey this meta-law. The secondary postulate states that *the speed of light is a constant* which is the same for all observers. This second postulate serves to select out *Lorentzian* relativity in favor of *Galilean* relativity. The primary implications for theoretical physics are contained in the first postulate.

Cellular automata are appealing for purposes of physical modeling because they

[1]This statement applies to *flat* spacetime and must be modified to include the law of gravity which involves *curved* spacetime.

have many features in common with physics including deterministic laws which are uniform in space and time, albeit discrete. Another feature is that the rules are local in the sense that there is a maximum speed at which information can propagate between cells. This inherent limit of CA is often referred to as the *speed of light*, and it imposes a causal structure much like that found in physics. In any event, such a speed limit is suggestive, and it has led to speculation on the role of relativity in CA [58, 84]. The existence of a maximum speed in CA is completely automatic and corresponds to the secondary postulate of relativity, but the more important and more difficult task is to find ways to implement the primary postulate.

So is there some sense in which CA laws are Lorentz invariant? Or does the existence of a preferred frame of reference preclude relativity? Ideally, we would like to have a principle of relativity that applies to a broad class of CA rules or to learn what is necessary to make a CA model Lorentz invariant. Preferably, relativity would show up as dynamical invariance with respect to an overall drift, but it could also manifest itself by mimicking relativistic effects. Rather than claiming the absolute answers to the questions above, the approach taken here is to develop some properties of a particular Lorentz invariant model of diffusion. This is also significant for CA modeling at large because diffusion is such an important phenomenon in many areas of science and technology.

4.1.2 Overview

This chapter describes a Lorentz invariant process of diffusion in one dimension and formulates it both in terms of conventional differential equations and in terms of a CA model. The two formulations provide contrasting methods of description of the same phenomenon while allowing a comparison of two methodologies of mathematical physics. Most importantly, the CA model shows how one can have a Lorentz invariant dynamical law in a discrete spacetime. Finally, the model provides a starting point for further research into elucidating analogs of continuous symmetries in CA.

The remaining sections cover the following topics:

Section 4.2 discusses the meaning and implications of Lorentz invariance in the context of a one-dimensional model of relativistic diffusion. The model can be described in terms of the deterministic dynamics of a massless billiard ball or in terms of a gas of massless particles undergoing independent random walks of a special kind. The continuum limit can be described by a set of partial differential equations which can be expressed in manifestly covariant form using spacetime vectors. These equations can in turn be used to derive the telegrapher's equation which is a more common equation for a scalar field, but it contains the same solutions.

Section 4.3 discusses the general solution to the continuum model in terms of an impulse response. Plots of the separate components of the field are shown for a typical case. The diffusion of individual massless particles can be well approximated on a lattice, and therefore, the process can be implemented as a parallel CA model using a partitioning strategy. The impact of the lattice on Lorentz invariance, particle independence, reversibility, and linearity are each considered in turn. The results of a CAM-6 simulation are presented for comparison with the exact solution of the continuum model.

Section 4.4 brings up the issue of finding Lorentz invariant CA rules in higher dimensions. The fact that isotropy is an inherent part of Lorentz invariance in two or more spatial dimensions turns out to be a major consideration for a lattice model. A CA rule may be considered Lorentz invariant in situations where it is possible find a direct mapping between individual events in different frames of reference or where the correct symmetry emerges out of large numbers of underlying events. Finally, we show how analogs of relativistic effects must arise in any CA in which there is *some* principle of relativity in effect.

Appendix D presents some of the more mathematical details of this chapter including an introduction to differential geometry, a discussion of Lorentz and conformal invariance, and a derivation of the exact solution.

4.2 A Model of Relativistic Diffusion

The ordinary diffusion equation, $\partial_t \phi = D\nabla^2\phi$, is not Lorentz invariant. This is easiest to see by noting that it is first order in time, but second order in space, whereas properly relativistic laws must have the same order in each because space and time are linearly mixed by Lorentz transformations. Furthermore, we know that, unlike the solutions of the diffusion equation, the spread of a disturbance must be limited to the interior of its future light cone in order to not violate the speed of light constraint. However, it is possible to generalize the diffusion equation in a way that is consistent with relativity [49]. This section gives one way of deriving such diffusion law from a simple underlying process [81, 86, 88].

Consider a one-dimensional system of massless billiard balls with one ball marked (see figure 4-1). We are interested in following the motion of this single marked ball. Since the balls are massless, they will always move at the speed of light (taken to be unity), and it is assumed that when two balls collide, they will always bounce back by exchanging their momenta. Suppose that the balls going in each direction are distributed with Poisson statistics, possibly having different parameters for each direction. In this case, the marked ball will travel a random distance between collisions, with the distances following an exponential distribution. Hence, the motion can also be thought of as a continuous random walk with momentum or as a continuous random walk with a memory of the direction of motion. We are then interested in describing the diffusion of the probability distribution which describes the position and direction of the marked ball.

The dynamics of the probability distribution can also be described in terms of number densities of a one-dimensional gas of massless, point-like particles. The particles travel at the speed of light and independently reverse direction with probabilities proportional to the rate of head-on encounters with the flow of an external field. The external field is actually another gas in the background which also consists of massless particles and is "external" in the sense that it streams along unaffected by any "collisions." A diffusing particle of the primary system has some small, fixed probability

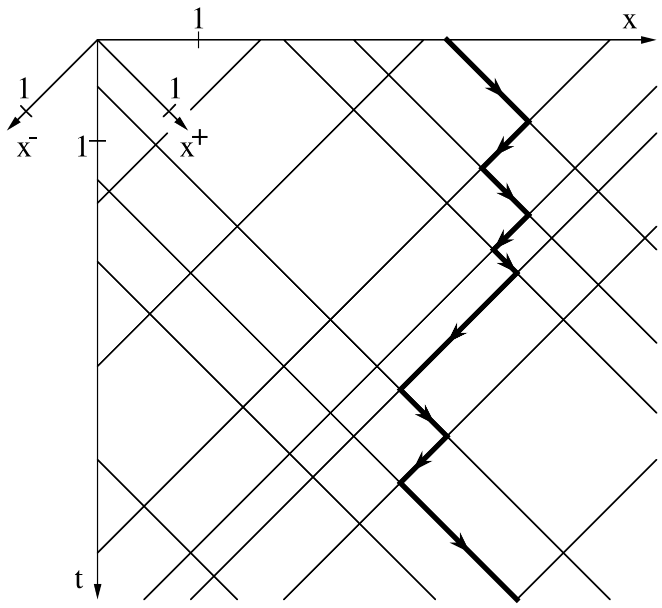


Figure 4-1: A spacetime diagram showing collisions in a one-dimensional gas of massless billiard balls. The motion of the marked ball can be described as Lorentz-invariant diffusion. The two pairs of axes show the relationship between standard spacetime coordinates and light-cone coordinates.

of reversing direction upon encountering a background particle. If we look at the diffusion on a macroscopic scale, then the gas can effectively be considered a continuous fluid [88]. This alternative description will be more closely suited to implementation in CA form.

It is apparent from figure 4-1 that the particle dynamics considered above is a Lorentz-invariant process. A Lorentz transformation (or boost) in 1+1 dimensions is easy to describe in terms of the light-cone coordinates, $x^{\pm} = \frac{1}{\sqrt{2}}(t \pm x)$: it is just a linear transformation which stretches one light-cone axis by a factor of $\gamma(1+\beta)$ and compresses the other by a factor of $\gamma(1-\beta)$, where $\gamma = 1/\sqrt{1-\beta^2}$, and $\beta = v/c$ is the normalized relative velocity of the two inertial frames. The effect of this transformation is to stretch or translate every segment of the world lines into a new position which is parallel to its old position, while leaving the connectivity of the lines unchanged. In the case of the number density description, the world lines of the background gas are transformed in a completely analogous manner. The set of the transformed world lines obviously depicts another possible solution of the original dynamics, so the process is invariant under Lorentz transformations.

Now we turn to formulating a continuum model of the process described above in terms of conventional partial differential equations which then allows a more formal demonstration of Lorentz invariance. The derivation of these transport equations is is accomplished through simple considerations of advection and collisions by the particles. The description is in terms of densities and currents which together give one way of representing two-dimensional (spacetime) *vector* fields. Section D.1.3 in the appendix elaborates on the concepts and notation used in the analysis.

The continuum model describes two probability (or number) densities, $\rho^+(x,t)$ and $\rho^-(x,t)$, for finding a particle (or particles) moving in the positive and negative directions respectively. One can also define a conserved two-current, $J^\mu = (\rho, j)$, where $\rho = \rho^+ + \rho^-$ and $j = \rho^+ - \rho^-$. The background particles give well-defined mean free paths, $\lambda^+(x^+)$ and $\lambda^-(x^-)$, for the diffusing particles to reverse direction. The transport equations for the densities can then be written down immediately:

$$\frac{\partial\rho^+}{\partial t} + \frac{\partial\rho^+}{\partial x} = -\frac{\rho^+}{\lambda^+} + \frac{\rho^-}{\lambda^-} \tag{4.1}$$

$$\frac{\partial\rho^-}{\partial t} - \frac{\partial\rho^-}{\partial x} = -\frac{\rho^-}{\lambda^-} + \frac{\rho^+}{\lambda^+}. \tag{4.2}$$

Adding these equations gives

$$\frac{\partial\rho}{\partial t} + \frac{\partial j}{\partial x} = 0, \tag{4.3}$$

and subtracting them gives

$$\frac{\partial j}{\partial t} + \frac{\partial\rho}{\partial x} = -j\left(\frac{1}{\lambda^+} + \frac{1}{\lambda^-}\right) + \rho\left(\frac{1}{\lambda^-} - \frac{1}{\lambda^+}\right). \tag{4.4}$$

The condition that the background particles stream at unit speed is

$$\frac{\partial}{\partial x^+}\left(\frac{1}{\lambda^-}\right) = 0, \quad \text{and} \quad \frac{\partial}{\partial x^-}\left(\frac{1}{\lambda^+}\right) = 0. \tag{4.5}$$

If one defines

$$\sigma^t = \frac{1}{\lambda^+} + \frac{1}{\lambda^-}, \quad \text{and} \quad \sigma^x = \frac{1}{\lambda^-} - \frac{1}{\lambda^+}, \tag{4.6}$$

then equations (4.3)–(4.5) can be rewritten in manifestly covariant form as

$$\begin{aligned} \partial_\mu J^\mu &= 0 && (4.7)\\ (\partial_\mu + \sigma_\mu)\varepsilon^{\mu\nu} J_\nu &= 0 && (4.8)\\ \partial_\mu \sigma^\mu &= 0 && (4.9)\\ \partial_\mu \varepsilon^{\mu\nu}\sigma_\nu &= 0. && (4.10) \end{aligned}$$

The parameter σ^μ is essentially the two-momentum density of the background particles and gives a proportional cross section for collisions. Writing the equations in this form proves that the model is Lorentz invariant.

Equations (4.7)–(4.10) also happen to be invariant under the conformal group; furthermore, when properly generalized to curved spacetime, they are *conformally invariant*. These concepts are expanded on in section D.2. Physicists are always looking for larger symmetry groups, and the conformal group is interesting because it is the largest possible group that still preserves the causal structure of spacetime. Clearly, it contains the Lorentz group. In 1+1 dimensions, the conformal group involves a local scaling of the light-cone axes by any amount ($x^\pm \to f^\pm(x^\pm)$). The fact that the above model is invariant under this group is easy to see from figure 4-1, since any change in the spacing of the diagonal lines yields a similar picture. Conformally invariant field theories possess no natural length scale, so they must correspond to massless particles (though the converse is not true). It therefore seems only natural that the underlying process was originally described in terms of massless particles.

In the case of a uniform background with no net drift ($\lambda^\pm = \lambda \Rightarrow \sigma^x = 0$), each component of the current satisfies the the telegrapher's equation [35]. Indeed, if σ^μ is a constant, equations (4.7)–(4.8) can be combined to give

$$\Box^2 J^\mu + \sigma^\nu \partial_\nu J^\mu = 0, \tag{4.11}$$

where $\Box^2 = g^{\mu\nu}\partial_\mu\partial_\nu$ is the d'Alembertian operator. More explicitly, equation (4.8) can be written $(\partial_{[\mu} + \sigma_\mu)J_{\nu]} = 0$. Acting on this equation by ∂^μ gives $\partial^\mu(\partial_\mu + \sigma_\mu)J_\nu -$

$\partial^\mu(\partial_\nu + \sigma_\nu)J_\mu = 0$, and the second term vanishes by equation (4.7).

4.3 Theory and Experiment

As usual when developing theoretical modeling techniques, it is desirable to have analytic solutions which can be compared to experiments (numerical or otherwise). This section gives the exact solution to the continuum model above as well as the results of a CA simulation of a comparable situation.

4.3.1 Analytic Solution

Here we discuss the general solution to equations (4.7) and (4.8). For a given a background field σ^μ, the equations are linear in J^μ, and an arbitrary linear combination of solutions is also a solution. In particular, it is possible to find a Green's function from which all other solutions can be constructed by superposition. As an aside, note that the equations are not linear in σ^μ in the sense that the sum of solutions for J^μ corresponding to different σ^μ's will not be the solution corresponding to the sum of σ^μ's. This happens because σ^μ and J^μ are multiplied in equation (4.8), and σ^μ affects J^μ but not vice versa.

Solutions are best expressed in terms of the component probability densities ρ^+ and ρ^- for finding the finding the particle moving in the $\pm$ directions ($\rho^\pm$ are proportional to the the light-cone components of the probability current $J^\pm$). For the moment, the background will be taken to be a constant which is parameterized by a single mean free path λ. We want to find the Green's function which corresponds to a delta function of probability moving in the positive direction at $t = 0$. The initial conditions are therefore $j(x,0) = \rho(x,0) = \delta(x)$, and the corresponding components will be denoted by ϱ^+ and ϱ^- respectively. This is an initial value problem which can be solved with standard methods of mathematical physics; more detail can be found in section D.3. Inside the light cone (i.e., the region $t^2 - x^2 > 0$) the solution is

$$\varrho^+(x,t) = \frac{e^{-t/\lambda}}{2\lambda}\sqrt{\frac{t+x}{t-x}}I_1\left(\frac{\sqrt{t^2-x^2}}{\lambda}\right) + e^{-t/\lambda}\delta(t-x) \tag{4.12}$$

$$\varrho^-(x,t) = \frac{e^{-t/\lambda}}{2\lambda}I_0\left(\frac{\sqrt{t^2-x^2}}{\lambda}\right), \tag{4.13}$$

where I_0 and I_1 are modified Bessel functions of the first kind. Outside the light cone, $\varrho^\pm = 0$.

Plots of $\varrho^+(x,t)$ and $\varrho^-(x,t)$ are shown for $\lambda = 32$ in figure 4-2(a) and (b) respectively. The independent variables range from $0 < t < 128$ and $-128 < x < 128$. The skewed density function for particles moving in the positive direction is marked by the exponentially decaying delta function along the x^+ axis, and it has been truncated for purposes of display (a). The density of particles moving in the negative direction follows a symmetrical distribution which ends abruptly at the light cone (b). The total density $\rho(x,t) = \varrho^+ + \varrho^-$ is plotted in figure 4-2(c).

Equations (4.12) and (4.13) can be used to find the solution for any other initial conditions. First, they can be reflected around $x = 0$ to give the solution starting from a delta function of probability moving in the negative direction. Second, the light-cone axes can be scaled (along with an appropriate scaling of $\sigma^\pm$ and $\varrho^\pm$—see section D.2.2) to change $\lambda \to \lambda^\pm(x^\pm)$ corresponding to an arbitrary background gas σ^μ satisfying equations (4.9)–(4.10). Finally, solutions starting from different values of x can be added to match any $\rho^+(x,0)$ and $\rho^-(x,0)$.

4.3.2 CAM Simulation

The process depicted in figure 4-1 can be approximated to an arbitrarily high degree by discrete steps on a fine spacetime lattice. Particles are merely restricted to collide only at the vertices of the lattice. If there are many lattice spacings per mean free path, the continuum distribution of free path lengths (a decaying exponential) is well approximated by the actual discrete geometric progression. A Lorentz transformation would change the spacings of the diagonals of the lattice, but as long as γ is

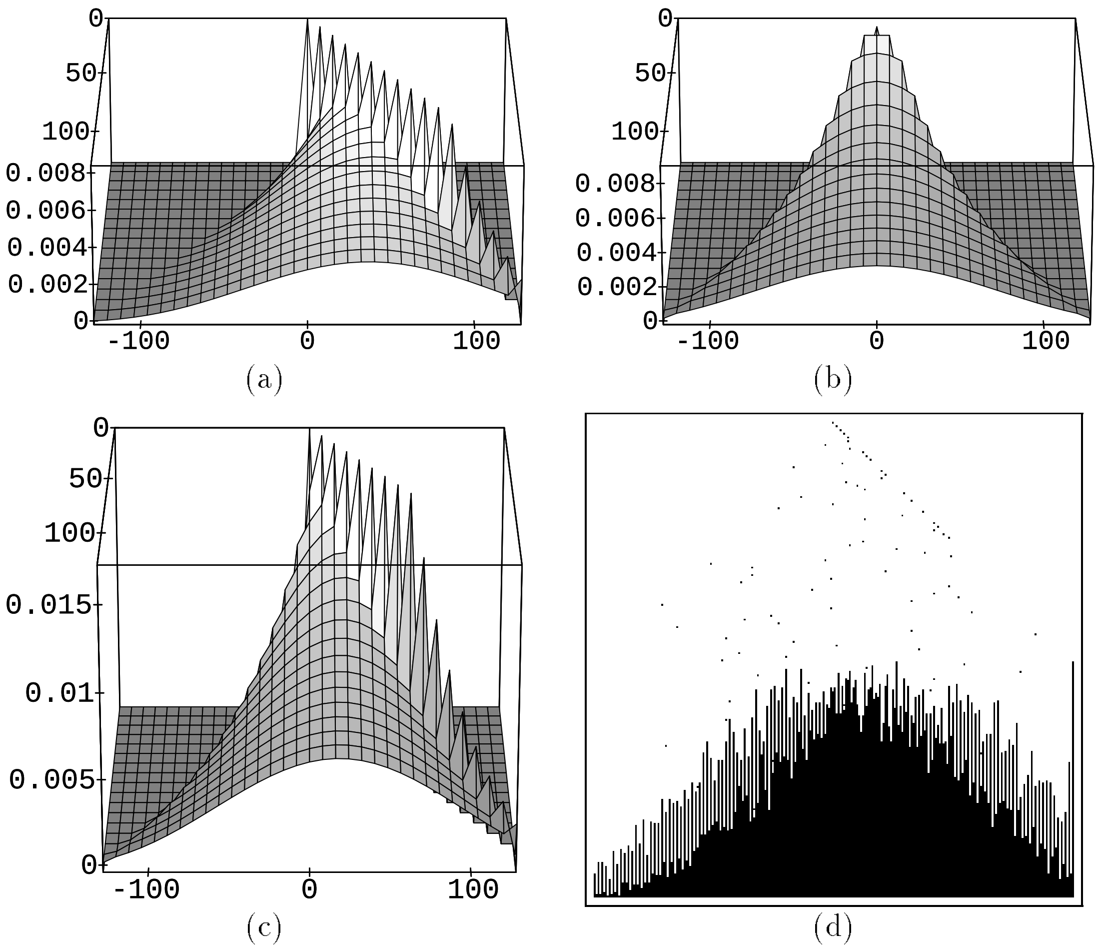


Figure 4-2: Green's functions for models of relativistic diffusion: (a) Continuous probability density for right-moving particles where t is increasing out of the page. (b) Probability density for left-moving particles. (c) The total probability density. (d) Result of a CAM-6 experiment showing particles executing random walks with memory in the top half. A histogram of the position and direction of the particles at $t = 128$ is taken in the bottom half.

not too large, the lattice approximation to the continuum is still a good one. The approximation to Lorentz invariance is also good, because it is still possible to map the new set of world lines onto the nearest diagonal of the original lattice.

The probabilistic description of the position and direction of a single particle is fine, but we really want a parallel dynamics for many particles diffusing in the same space independently. Thus, in this section, $\rho^{\pm}$ will be interpreted as a number density instead of a probability, and the number of particles must then be large relative to the scale of observation. Also, the collisions with the background gas must take place with a small probability rather than with certainty; otherwise, the motions of adjacent particles will be highly correlated (e.g., their paths would never cross). Furthermore, the density of particles must be low relative to the number of lattice points so that no two particles ever try to occupy the same state of motion. If these conditions are satisfied, the movements of the particles can be considered independent, and we can assume a superposition of many individual processes like that shown in figure 4-1. This again gives equations (4.7)–(4.10) in the continuum limit.

The CA version of this process uses the partitioning format shown in figure 2-2(b). Such a format is used because it is easy to conserve particles.[2] The diffusion rule usually swaps the contents of the two cells in every partition on every time step, but it has a small probability p of leaving the cells unchanged. Referring again to figure 2-2(b), one can see that this will cause particles to primarily stream along the lattice diagonals in spacetime. However, on any given step, the particles have a probability p of a "collision" which will cause them switch to the other diagonal instead of crossing over it. In contrast to a standard random walk, the particles have a memory of their current direction of motion, and they tend to maintain it. By tuning the probability of reversal (using the techniques for randomness presented in appendices A and C for example), it is possible to adjust the mean free path of the particles. In fact, it is easy to show that $\lambda = (1-p)/p$ in lattice units.

The process described in section 4.2 assumes that the particles diffuse indepen-

[2]Though it is not done here, partitioning also makes it possible to construct a reversible version of the diffusion law by using a reversible source of randomness, such as a third lattice gas.

dently. This is not a problem in a continuum, but on a lattice, there is a possibility that particles will interfere with each other's motion. In particular, if there are two particles in a partition and one turns, the other must also turn in order to conserve particles and maintain exclusion. In other words, there is a small probability that the motions of nominally independent particles will be correlated. This introduces a small nonlinearity to the diffusion, though it is negligible in the limit of low particle density per lattice site. On the other hand, as long as there is only one particle in a partition, it is possible to have two different probabilities for a particles to turn depending on its original direction. This corresponds to a drift in the background gas and is equivalent to the original dynamics in a different Lorentz frame of reference.

Figure 4-2(d) shows the result of a CAM-6 simulation with the randomness chosen so that $\lambda = 32$. This implementation also serves to show one of the ways CA can be used as a display tool as well as a simulation tool. The top half shows a spacetime diagram of the region $0 < t < 128$ and $-128 < x < 128$ where time has been spread out over successive rows. On every other time step of the CA, a point source at $t = 0$ emits a particle initially moving to the right, and the particle is passed to subsequent rows as it diffuses. The rows can therefore be thought of as a sequence of independent systems, each containing a single particle as in figure 4-1, though the dynamics is capable of supporting up to 256 particles per row. Over time, the rule generates an ensemble of particles which approximates the probability distributions given in figure 4-2(a)–(c). The bottom half of the simulation serves to collect a histogram of the final positions corresponding to an elapsed time $t = 128$.

Note that the odd and even columns of the histogram appear to follow different distributions. Indeed this is the case: because of the partitioning format shown in figure 2-2(b), the direction of motion of a particle is determined entirely by the parity of the column. Thus, the even columns contain $\rho^+(x, 128)$, and the odd columns contain $\rho^-(x, 128)$. The envelopes of these two components are quite apparent in the histogram and should be compared to the final time of $t = 128$ in figure 4-2(a) and (b) respectively. Finally, the rightmost bin $\rho^+(128, 128)$ is full, so the delta function it represents has been truncated. Also, some of this delta function has been deflected into the neighboring bin $\rho^+(126, 128)$ because of spurious correlations in the noise source.

4.4 Extensions and Discussion

The above model provides an example of how one can have a Lorentz invariant CA rule in one dimension. Now we would like to consider how similar results might be obtained in higher dimensions. Of course, the full continuous symmetry is is impossible to achieve in a discrete space, so we will always imagine limiting situations. Two possible approaches are (1) to find simple rules which have bijective mappings of individual spacetime events between any two frames of reference, and (2) to find complex rules from which Lorentz invariant laws emerge out of a large number of underlying events. The first approach works for the present example, while the second approach would be more in the spirit of lattice gas methods for partial differential equations [24, 25].

The case of diffusion of massless particles in one dimension is special because there is a direct mapping between the world lines in one frame and the world lines in another. However, this mapping does not work for higher dimensional CA because invariance under arbitrary Lorentz boosts implies invariance under arbitrary rotations (as shown below). Since straight world lines cannot match the lattice directions in every frame of reference, any system for which the most strongly correlated events preferentially lie along the lines of the lattice cannot easily be interpreted as Lorentz invariant. This would be important, for example, in situations where particles stream for macroscopic distances without collisions.

The fact that isotropy is an intrinsic part of a relativistically correct theory can be seen by constructing rotations out of pure boosts. This will be shown by examining the structure of the Lorentz group near the identity element. A convenient way to represent such continuous matrix groups (Lie groups) is as a limit of a product of infinitesimal transformations using exponentiation.[3] Thus, an arbitrary Lorentz

[3]Since $\lim_{n\to\infty}(1+\frac{x}{n})^n = e^x$.

transformation matrix can be written [43]

$$A = e^{-\boldsymbol{\theta}\cdot\boldsymbol{S}-\boldsymbol{\zeta}\cdot\boldsymbol{K}} \tag{4.14}$$

where $\boldsymbol{\theta}$ and $\boldsymbol{\zeta}$ are 3-vectors parameterizing rotations and boosts respectively. The corresponding generators of the infinitesimal rotations and boosts are given by

$$S_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad S_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix}, \quad S_3 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$K_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad K_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad K_3 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}. \tag{4.15}$$

These generators satisfy the following commutation relations (Lie brackets) as can be verified by direct substitution:

$$\begin{aligned} [S_i, S_j] &= \varepsilon_{ijk} S_k \\ [S_i, K_j] &= \varepsilon_{ijk} K_k \\ [K_i, K_j] &= -\varepsilon_{ijk} S_k. \end{aligned} \tag{4.16}$$

The reason for expanding the Lorentz transformation matrices in this way will become apparent next.

Consider a sequence of four boosts making a "square" of size ε in "rapidity space." To lowest nonvanishing order in ε,

$$\begin{aligned} A &= (1 + \varepsilon K_i + \tfrac{1}{2}\varepsilon^2 K_i^2)(1 + \varepsilon K_j + \tfrac{1}{2}\varepsilon^2 K_j^2) \\ &\quad \cdot (1 - \varepsilon K_i + \tfrac{1}{2}\varepsilon^2 K_i^2)(1 - \varepsilon K_j + \tfrac{1}{2}\varepsilon^2 K_j^2) \end{aligned}$$

$$
\begin{aligned}
&= 1 + \varepsilon^2 [K_i, K_j] + \mathcal{O}(\varepsilon^3) \\
&= 1 - \varepsilon^2 \varepsilon_{ijk} S_k + \mathcal{O}(\varepsilon^3)
\end{aligned}
\tag{4.17}
$$

If (i, j, k) is a cyclic permutation of $(1, 2, 3)$, then A is not a pure boost but a rotation around axis k through an angle ε^2. Therefore, appropriate sequences of boosts can result in rotations, and similar sequences can be repeated many times to construct an arbitrary rotation. The implication for generalizing the relativistic model of diffusion of massless particles to more than one spatial dimension is that particles must be able to stream with few collisions in any direction, not just along the axes of the lattice. At present, we do not know any reasonable way of doing this. Another way to see the problem is that if a particle is moving along a lattice direction which is perpendicular to the direction of a boost in one frame, it will suffer aberration and will not, in general, be moving along a lattice direction in the other frame.

What about the second approach where we don't demand that all spacetime events in any two frames are in one-to-one correspondence? The intuition here is that if a CA is dense with complex activity, then cells in all directions will be affected, and anisotropy will not be as much of a problem. Now, any CA which gives a Lorentz invariant law in some limit can reasonably be said to be Lorentz invariant. For example, lattice gases in which all particles move at a single speed (the speed of light), exhibit sound waves which obey the wave equation with an isotropic speed of sound of $1/\sqrt{d}$ in d spatial dimensions [39] (see figure 2-14). But the wave equation *is* Lorentz invariant if the wave speed instead of the particle speed is interpreted as the speed of light. This is somewhat unsatisfying since some information can travel at greater than the speed of light, but it may still be used to obtain some measure of Lorentz invariance in higher dimensions. In general, the lack of isotropy of the underlying lattice means that the speed of light of the CA cannot coincide with the speed of light of the emergent model. Furthermore, in such cases, there is no natural interpretation of all possible Lorentz transformations on the lattice as there was in the 1+1 dimensional case. Finally, we mention in passing the possibility of having an isotropic model by using a random lattice since it would clearly have no preferred directions.

A final point about special relativity in the context of CA concerns the connection between Lorentz invariance and relativistic effects. *If* it were the case that a given CA rule could simulate certain physical phenomena (such as particle collisions) in any frame of reference *then* it would be the case that the evolution in those frames that are moving faster with respect to the preferred lattice frame would have to evolve slower. The reason for this is that the number of cells in the causal past of a cell would be restricted. In the extreme case, a CA configuration moving at the maximum speed of light could not evolve at all because each cell would only depend on one other cell in its past light cone; consequently, no interaction between cells could take place. In information mechanical terms, this "time dilation" would happen because some of the computational resources of the CA would be used for communication and fewer would be available for computation. In this manner, one obtains phenomena reminiscent of relativistic effects. However, the hard problem is to find *any* nontrivial rules which obey the primary postulate of relativity. The above model of relativistic diffusion in 1+1 dimensions represents only a limited class of such rules. Special relativity is not so much a theory of how physical effects change when high speeds are involved as it is a theory of how the physical laws responsible for those effects are *unchanged* between frames in relative motion. Relativistic effects such as time dilation and the like come about as mere consequences of how observable quantities transform between frames, not because of motion with respect to some kind of "ether."

4.5 Conclusions

Symmetry is an important aspect of physical laws, and it is therefore desirable to identify analogous symmetry in CA rules. Furthermore, the most important symmetry groups in physics are the Lorentz group and its relatives. While there is a substantial difference between the manifest existence of a preferred frame in CA and the lack of a preferred frame demanded by special relativity, there are still some interesting connections. In particular, CA have a well-defined speed of light which imposes a causal structure on their evolution, much as a Minkowski metric imposes a causal structure on spacetime. To the extent that these structures can be made to coincide between the CA and continuum cases, it makes sense to look for Lorentz invariant CA.

The diffusion of massless particles in one spatial dimension provides a good example of a Lorentz invariant process that can be expressed in alternative mathematical forms. A corresponding set of linear partial differential equations can be derived with a simple transport argument and then shown to be Lorentz invariant. A CA formulation of the process is also Lorentz invariant in the limit of low particle density and small lattice spacing. The equations can be solved with standard techniques, and the analytic solution provides a check on the results of the simulation. Generalization to higher dimensions seems to be difficult because of anisotropy of CA lattices, though it is still plausible that symmetry may emerge in complex, high-density systems. The model and analyses presented here can be used as a benchmark for further studies of symmetry in physical laws using CA.

Chapter 5

Modeling Polymers with Cellular Automata

5.1 Introduction

5.1.1 Atoms, the Behavior of Matter, and Cellular Automata

Perhaps the single most important fact in science is that the world is made of atoms. Virtually every aspect of physics that we encounter in our daily lives can be accounted for by the electromagnetic interactions of atoms and their electrons.[1] It is difficult to directly incorporate all of the detailed microscopic physics of a complex system in a calculation or computer simulation, but it is often more illuminating to find approximations at higher levels of organization anyway. For example, one may introduce phenomenological parameters and stylized variables which describe the configurations, energies, motions, or reaction rates of atoms or collections of atoms. Under favorable circumstances and with a suitable dynamics, the correct behavior will emerge in the limit of large systems.

When is it possible to successfully construct CA models of physical phenomena in terms of abstract atoms in this way? This is a hard question that doesn't have a

[1] Of course this claim is drastically oversimplified in terms of the quantum-mechanical details involved, but the general idea is sound. The primary exception to the rule is due to gravity.

final answer, but a few pertinent observations can be made. First, the phenomena in question should be characteristic of many atoms rather than of the structure of relatively simple individual molecules. In other words, the systems should fall into the domain of statistical mechanics. It may be conceptually possible to decompose the underlying physics into a statistical substructure through quantum Monte Carlo or stochastic mechanics [64], but for practical purposes, it is better to develop models at or above the scale of real atoms. Second, the dynamics should rely on local or even point-like interactions as opposed to long range forces. It is possible for CA to exhibit large coherence lengths, but for efficiency's sake, it is better to avoid models having complicated rules or widely disparate time scales. The continuing development of models and supporting analysis for a wide variety of physical situations will bring us closer and closer to a resolution of this question.

Even within the above classical framework, CA have a great deal of potential for studies in condensed matter because they have the basic structure of physics and a large number of programmable degrees of freedom. Promising areas of application include fluid dynamics, chemistry, solid state physics, materials science, and perhaps even plasma physics. Besides conventional homogeneous substances it is also possible to model more exotic heterogeneous materials which are not usually thought of as matter in the physicists' sense of the word: currencies, alluvia, populations, and even "grey matter." These possibilities lead to the notion of "programmable matter" [91] which is a powerful metaphor for describing the relationship between CA and the essential information-bearing degrees of freedom of nature. Furthermore, programmability means that the atoms in CA are not limited by the rules of physical atoms, and one can explore the behavior of exotic forms of matter that may not even exist in nature. Such investigations are interesting and important because they increase the utility of parallel computers for physical simulation while extending the boundaries of mathematical physics.

5.1.2 Monte Carlo Methods, Polymer Physics, and Scaling

An important tool in computer modeling and statistical mechanics is the so-called Monte Carlo method [5, 6]. It is a technique for doing numerical integration in high dimensional spaces and is useful for finding expectation values in complex systems. The procedure is to select points from a sample space according to some probability measure and then take the sum of contributions from all these points. The samples are usually generated by imposing an artificial dynamics on the configuration of the system in question, and the dynamics can sometimes be used as an alternative to molecular dynamics. In the case of large, dense systems, CA are well suited to execute Monte Carlo simulations of abstract molecular dynamics.

An active area of research where the behavior of collections of atoms is important is that of polymer physics. The interest in polymers stems from their ubiquity—plastics, proteins, nucleic acids, and cellulose are conspicuous examples—and their importance in biology and industrial applications. Polymers form complex biological structures such as enzymes, organelles, microtubules, and viruses, while they also play important practical roles in petroleum products and advanced composite materials. In order to understand and control the physical behavior of polymers, one must study their structure, dynamics, chemistry, rheology, and thermodynamic properties. However, the phenomena are often so complex that computational methods are becoming increasingly necessary to the research.

From a more theoretical and mathematical point of view, polymers provide interesting phases of matter and a rich set of phenomena which are often well described by scaling laws [21]. A scaling law is a relation of the form $y \sim x^\alpha$. It says that over a certain range, multiplying (or *scaling*) x by a constant g has the effect of multiplying y by a related constant g^α. However, fundamental theoretical questions remain about the scaling behavior and dynamical origin of certain quantities, most notably, viscosity. There are some questions as to whether or not the pictures on which scaling arguments are based are even correct (for example, what is the physical meaning of "entanglement length"?), and computer simulations are invaluable for educing the answers. In addition, such simulations can be used to explore theoretical novelties such as polymer solutions in the semi-dilute regime or the existence of irreversibly knotted materials [71].

5.1.3 Overview

This chapter successfully demonstrates the atomistic approach advocated above for physical modeling with CA by developing techniques for simulating polymers on a lattice using only local interactions. Consequently, it opens up new ways of exploiting the power of parallel computers for an important class of problems in computational physics. The essential features of abstract polymers that are needed to give the correct basic scaling behavior are that the strands not break and that they not overlap. Furthermore, the resulting models show that it is possible to construct mobile macroscopic objects having absolute structural integrity by using only local CA rules. Finally, the numerous possible embellishments of the basic polymer model make it a paradigm for the entire discipline of modeling interactions among particles using CA.

The following is a breakdown of the respective sections:

Section 5.2 discusses the general problem of simulating polymers in parallel and describes the rationale behind using abstract polymers and Monte Carlo dynamics. Starting from the ideal case of a very high molecular weight polymer in a continuum, a sequence of polymer models are introduced which illustrate some of the problems associated with creating lattice polymers having local dynamics. The discussion leads to the double space algorithm which solves the problems in an elegant fashion. Finally, this algorithm is compared to another popular lattice polymer algorithm: the "bond fluctuation" method.

Section 5.3 covers some of the quantitative aspects of polymer physics which give the subject a firm experimental and theoretical foundation. Samples taken from simulations can be used to measure expectation values of quantities having to do with the static structure of the polymers. Time series of sampled quantities can be used to measure dynamic properties such as time autocorrelation functions and the associated relaxation times. Simulation results from the double space algorithm are presented which verify the theoretical scaling laws and thereby serve as a test of the algorithm.

The double space algorithm along with the availability of powerful parallel computers and CA machines opens up a variety of prospects for additional research in computational polymer physics. Section 5.4 discusses extensions of the basic model as well as a number of possible applications. In particular, the specific example of a simple model for pulsed field gel electrophoresis is given.

5.2 CA Models of Abstract Polymers

The great computational demands of polymer simulation make it necessary to find new techniques for working on the problem, and parallel computers are an important part of the search. The most straightforward approach to parallelizing the simulation is to assign a processor to each monomer and have all of them compute the motion much as they would on a single processor. In principle, this leads to a large amount of communication between the processors during the calculation of the interactions because any two monomers in the system may come into contact [8]. If this is not enough of a problem, it also happens that communication and synchronization between processors is usually a time-consuming operation [67]. Because of these considerations, it has generally been considered difficult to perform efficient parallel processing simulations of polymers. However, one can get around these problems by recognizing that the interactions are local in space and allowing the natural locality and synchronous parallelism of CA to guide the design of a model. The resulting algorithms will be useful on all kinds of parallel computers, especially massively-parallel computers.

Once it has been decided to arrange the processors in a local, spatially parallel fashion, it is still necessary to describe the polymer model and specify the dynamics. It is tempting to make the model as direct and realistic as possible using molecular dynamics [61], i.e., essentially integrating Newton's laws under artificial force laws and small time steps. However, Monte Carlo dynamics in which the steps are motivated by considerations of real polymer motion (much like Brownian motion) are sufficient to model many aspects of polymer behavior. Furthermore, the steps are larger and

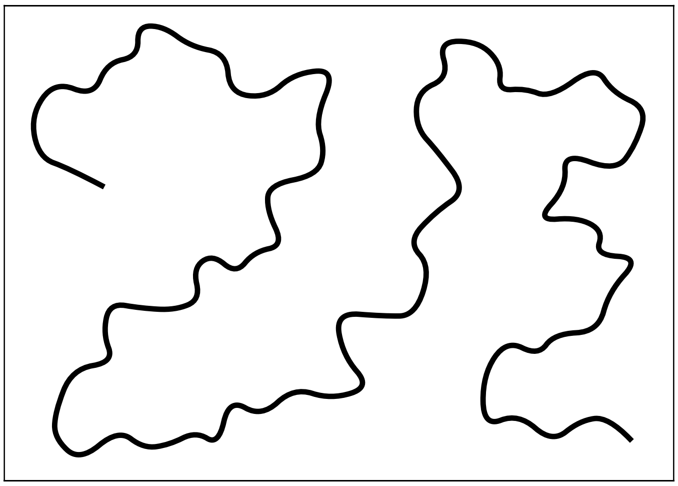


Figure 5-1: A typical configuration of a random, self-avoiding strand which represents a polymer with a very high degree of polymerization N.

there is less computation involved in each monomer update, and this enables much more extensive simulations. Using a simplified dynamics is also in accordance with our desire to find abstract models which capture the most essential features of physics.

Now consider an isolated polymer in a good solvent with short-range, repulsive interactions between monomers. In the limit of high polymerization, the details of the local polymer structure don't effect the large-scale structure [21], and the polymer effectively becomes a continuous, self-avoiding strand as shown in figure 5-1. Thus it is possible to approximate this situation with an abstract polymer model consisting of a chain of nonoverlapping monomers, where all configurations have equal energies. The essential constraints on the dynamics of an abstract polymer are

1. Connectivity: consecutive monomers remain bonded, and

2. Excluded volume: no two monomers can overlap.

The basic Monte Carlo dynamics works by picking a single monomer at random and making a trial move in a random direction. The move is then accepted if the constraints are satisfied. However, if two or more monomers were to be moved simultaneously and independently, it could lead to violations of these constraints.

The purpose of this section is to develop CA representations of polymers along with *parallel* Monte Carlo dynamics. A series of models is presented in order to illustrate some of the issues involved. The polymer configurations shown below are approximations to the polymer strand shown in figure 5-1, and each one uses $N \cong 60$ monomers. Each model requires a different lattice pitch and gives a different quality of approximation and degree of flexibility. Modifications beyond the simple algorithms given here will be discussed in section 5.4.

5.2.1 General Algorithms

A general procedure for turning the basic Monte Carlo algorithm given above into a parallel algorithm is to use the CA notion of spatial *partitioning* [56]. Partitioning refers to breaking up space into isolated regions with several processors per region. The dynamics can then act on these partitions in parallel without interference. Other spatial domain decomposition schemes can be used for a wide range of simulations [30].

To see how partitioning can be used, consider the abstract polymer shown in figure 5-2. Each monomer is represented by an excluded volume disk of radius r, and the bond between monomers are indicated by line segments. Each square represents a region of space assigned to a processor. The monomers are allowed to have any position subject to some minimum and maximum limits on bond lengths. Two monomers can simultaneously take a Monte Carlo step of size $\leq l$ if they are sufficiently far apart. Instead of randomly picking which monomers to update, those whose centers lie in the black squares are chosen. To prevent interference on any given step, the marked squares should have a space of at least $2(r+l)$ between them. In this case, the partitions consist of the 3×3 blocks of squares surrounding the black squares, and only the processors within a partition need to cooperate to maintain the constraints. Finally, the overall offset of the sublattice of black squares should be chosen at random before each parallel update.

Further simplification of Monte Carlo polymer models is obtained by restricting the positions of the monomers to the vertices of a lattice, or equivalently, to the cells of a CA. This substantially simplifies the moves and the checking of constraints. Removing a monomer from one cell must be coupled to adding a monomer to a nearby cell, and this is where partitioning CA neighborhoods are essential. In the

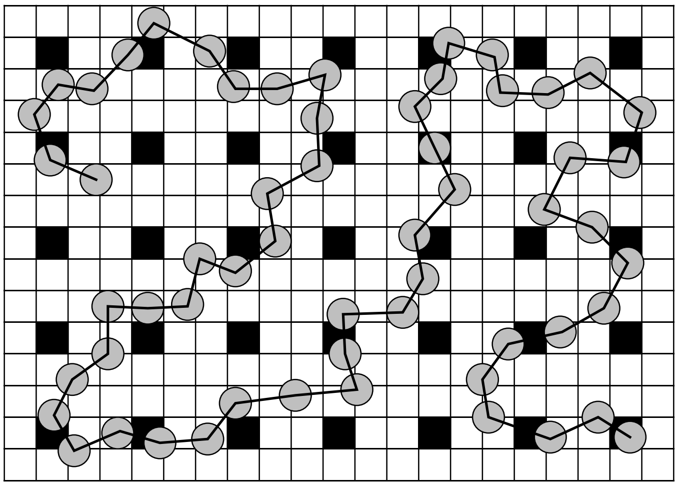


Figure 5-2: An abstract version of the continuous polymer strand with $N = 58$. Monomers whose centers lie in one of the black squares can take a Monte Carlo step in parallel without interference.

figures below, the partitions are indicated by dots in the centers, and arrows indicate all possible moves within the current partitions. Connectivity of the CA polymers is determined entirely by adjacency—the bonds do not have to be represented explicitly. The excluded volume constraint is then equivalent to not creating new bonds.

Perhaps the simplest CA polymer model one can imagine is illustrated by figure 5-3. Monomers are connected if and only if they are adjacent horizontally or vertically. The only moves allowed in this model consist of folding over corners, and as in the previous model, moves are attempted in parallel for monomers in the marked cells. Note that the approximation to the continuous example is rather crude because the lattice is fairly coarse. The polymer is not very flexible in that out of five marked monomers, only one move is possible. An even more serious problem with flexibility is that there are sometimes very few, if any, paths in configuration space between certain pairs of configurations. For example, consider a polymer shaped like a shepherd's staff. In order for the crook of the polymer to bend from one side to the other, it must pass through a unique vertical configuration. Situations such as this form bottlenecks in configuration space which greatly hamper the dynamics.

A somewhat improved CA polymer model is illustrated by figure 5-4. In this case, a monomer is connected to another if it is adjacent in any of the eight surrounding

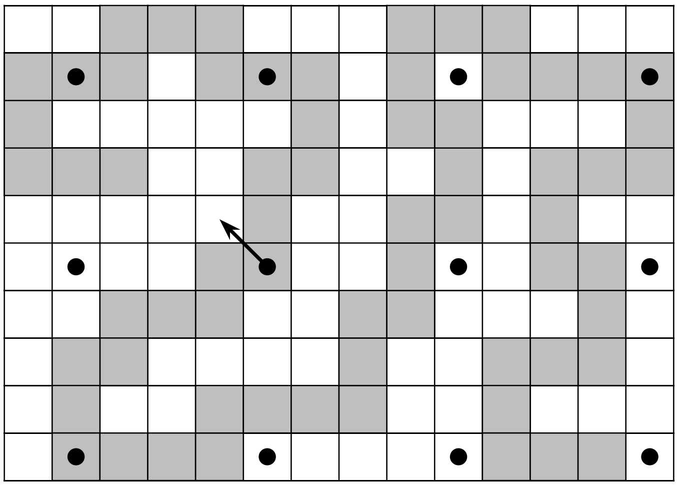


Figure 5-3: A simple CA realization of the polymer strand using a lattice model having only nearest-neighbor connections. Here, $N = 65$. Dots mark the cells from which monomers may move in parallel. The arrow indicates the only possible move.

cells. Trial moves are chosen at random from the four compass directions. With this scheme, the lattice can be finer, and one obtains a closer approximation to the shape of the original polymer strand. Furthermore, more monomer moves are typically possible on a given time step than in the previous model. However, the model also suffers from severe bottlenecks in configuration space because the end of a polymer must pass through a diagonal configuration in order to make a transition between horizontal and vertical.

The models presented in this section prove the general concept of parallel CA polymer dynamics, but they are lacking in two respects. First, very few monomers can be moved in parallel because only one out of sixteen cells is marked for movement on a given step. Nature moves all atoms in parallel, and we would like as far as possible to approach this level of parallelism with our abstract atoms. Perhaps the more serious difficulty is that, unlike real polymers which can move in any direction, the lattice polymers described above cannot move along their own contours. This is a root cause of the inflexibility encountered in the polymers above. These problems are solved with the algorithm presented in the next section.

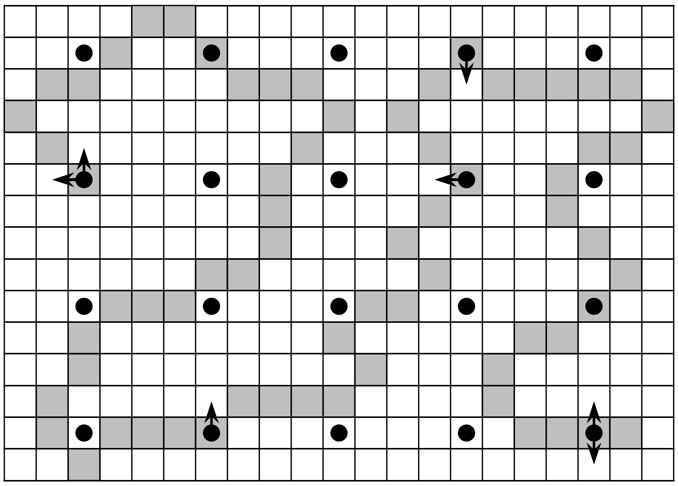


Figure 5-4: Another CA representation of the polymer having either nearest or next nearest neighbor connections. The lattice is finer than in the previous model while $N = 68$ is comparable. Dots indicate active cells, and the arrows show all possible moves.

5.2.2 The Double Space Algorithm

The inherent rigidity of the CA polymer models given above makes it difficult for a bend in a polymer strand to flip to the other side, especially in two dimensions. The rigidity of the polymers stems from the incompressibility of the internal bonds and the inability of monomers to move parallel to the polymer chain. Basically, the monomers cannot move in the direction they need to because adjacent monomers get in each other's way. One way to get around this problem would be to allow connected monomers to occasionally overlap somehow. While this is not physically realistic, it will not matter to the correct scaling behavior as long as some measure of excluded volume is still maintained. Others have recognized various forms of rigidity in lattice polymers and have proposed several algorithms to get around the problems [11, 26].

In order for adjacent monomers in a CA simulation to overlap, there must be extra space in a cell for a second monomer to occupy. The simplest way to do this is to invoke a double space [2, 79, 80] and have odd and even monomers alternate between spaces along the chain as shown in figure 5-5. Connectivity is again determined solely by adjacency in any direction, but *only* between monomers in opposite spaces. Excluded volume arises by demanding that no new connections are formed during the

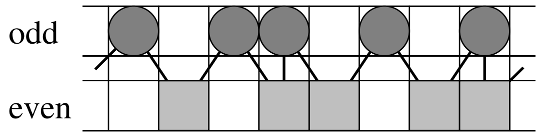


Figure 5-5: Cross section through the CA space along a polymer chain showing the double space. Odd and even monomers are only connected to monomers in the opposite space of neighboring cells. The bonds show the connections explicitly, but they are redundant and are not actually used.

dynamics. This has the remarkable effect that the constraints are maintained solely with respect to monomers in the opposite space. Therefore, by checking constraints against monomers held fixed in one space, the Monte Carlo dynamics can update all of the monomers in the other space simultaneously without interference. This amounts to a partitioning of the polymer chain rather than a partitioning of space.

A polymer represented using the double space scheme is shown in figure 5-6 where circles denote odd monomers and squares denote even monomers. The algorithm proceeds by alternately updating one space and then the other, and a pair of these updates is considered to be a single step of the CA rule.[2] The rule for updating each monomer in one space is to pick one of the four compass directions at random and accept a move in that direction if the constraints allow (see figure 5-7). The resulting lattice polymers are very flexible.

The double space algorithm just described clearly satisfies the connectivity constraint. However, one must still check that distinct monomers can never coalesce into a single monomer. This is trivially true for monomers in opposite spaces. It can be proved for any two monomers in the same space on different chains by assuming the converse. Since the dynamics was constructed to preserve the connectivity of the chains, the set of monomers to which any particular monomer is bonded remains the same. However, if two monomers were to move to the same site, they would have to have been connected to the same monomers in the first place, and that contradicts

[2] In order to achieve strict detailed balance in this model, one would have to pick which space to update at random as part of the Monte Carlo step. This is because an update of one space cannot undo the prior update of the other, and inverse transitions do not have the same probability as the forward transitions. However, it is still the case that all configurations become equally likely in equilibrium.

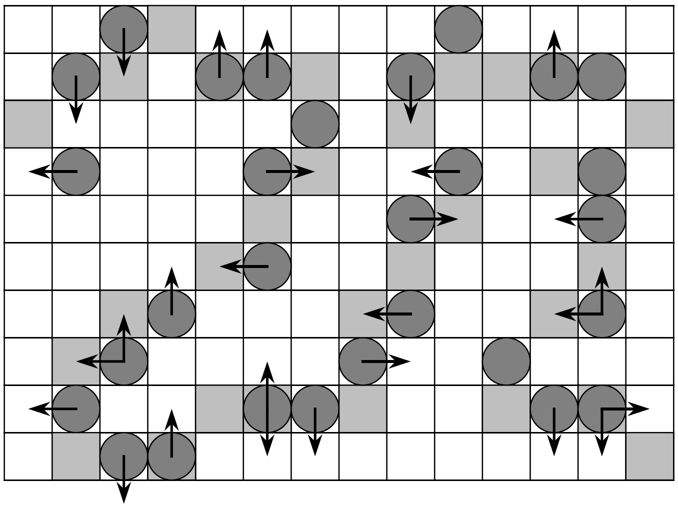


Figure 5-6: Abstract CA representation of the continuous polymer strand using the double space algorithm with $N = 58$. The odd monomers, indicated by circles, are currently selected to move, and the arrows show all possible moves. The comparatively large number of available moves makes the polymer very flexible.

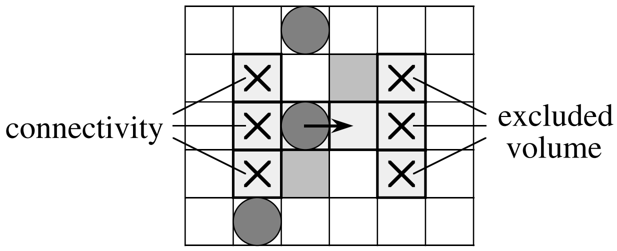
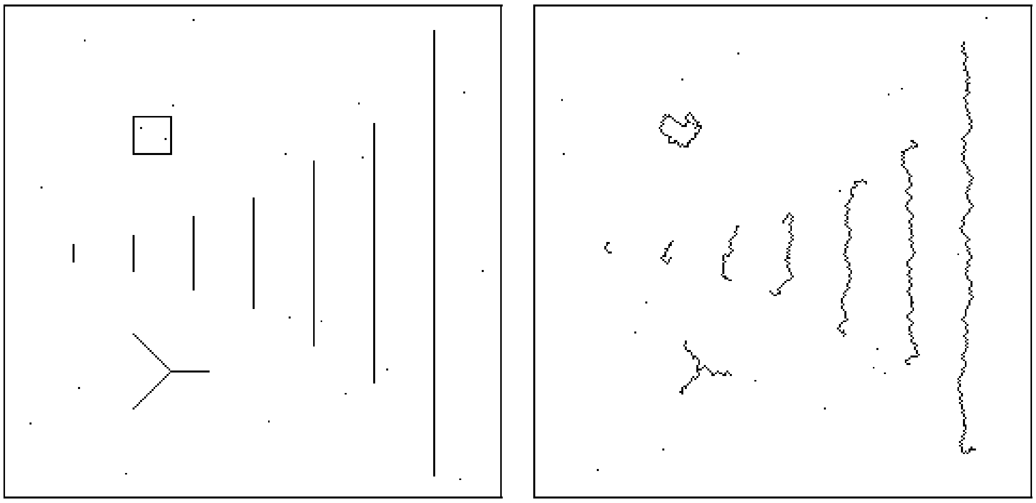


Figure 5-7: Checking constraints for a trial move in the double space algorithm. The odd monomer can make the indicated move if there are no even monomers in the cells marked with x's. Checking the marked cells to the left and right respectively serves to prevent the polymer from breaking and to give excluded volume to the chains.



(a) (b)

Figure 5-8: Left: Initial CAM-6 polymer configuration showing chains with $N =$ 10, 20, 40, 60, 100, 140, and 240. The implementation also supports rings and branched polymers as well as individual monomers. Right: After 1000 time steps, the contraction of the chains is apparent.

the assumption that the monomers are on different chains. Note that this proof fails for polymers with $N = 1$ and $N = 3$ because distinct monomers in the same space *can* have the same set of bonded monomers.

Figure 5-8 shows configurations taken from a CAM-6 simulation using the double space algorithm. The initial configuration shows a variety of polymer types including several fully stretched linear polymers ranging in length from $N = 10$ to $N = 240$. The dynamics generates a statistical tension since there are vastly more contracted configurations than stretched ones; thus, the chains start to contract. After 1000 time steps, the shortest two chains have undergone several relaxation times, while the third ($N = 40$) has undergone almost one. Due to limited computational resources, the implementation of the polymer rule on CAM-6 chooses the directions of movement deterministically and whether to accept the move is made randomly rather than the other way around (which makes two trial moves necessary for the equivalent of one step). Furthermore, the trial moves always occur between pairs of cells in such a way that an accepted move merely swaps the contents of the cells. This variation admits polymers with $N = 1$ and $N = 3$ and makes it possible to construct a reversible version of the rule.

5.2.3 Comparison with the Bond Fluctuation Method

The double space algorithm was motivated as a parallel algorithm ideally suited for CA, but it is also an excellent lattice polymer Monte Carlo algorithm in general. The advantages of the double space algorithm can be gauged by comparing it with the bond fluctuation method, a state-of-the-art Monte-Carlo algorithm [11]. There are basically four ways in which the double space algorithm excels: (1) step speed (less computer time per time step), (2) relaxation rate (fewer steps per relaxation time), (3) inherent parallelism, and (4) simplicity. Each of these point along with possible drawbacks will be considered briefly below.

The bond fluctuation method is illustrated by figure 5-9. The monomers fill 2×2 blocks of cells which can touch but not overlap. This is the conventional, direct way of implementing the excluded volume constraint. All bonds between a minimum length of 2 (which can be derived from the excluded volume) and a maximum length of $\sqrt{13}$ are allowed, giving a total of 36 possible bond vectors (compared to 9 for the double space algorithm). The bond fluctuation method was not originally thought of as a CA model, and unlike the CA models given above, it requires explicit representation of the bonds since nonbonded monomers may be closer than bonded ones (this can also be done with a local CA rule, but it would be fairly complicated). As in two of the above lattice polymer models, the Monte Carlo update of a monomer involves picking one of the four compass directions at random and accepting the move if the constraints allow. Which monomer to move should be chosen at random, but a random partition can also be used for a parallel update. The fine lattice gives a good approximation to the continuous polymer strand, but the step size is correspondingly small. Finally, the algorithm does not suffer from severe bottlenecks in configuration space because the variable bond lengths allow movement parallel to the chain.

The relative speeds of the double space algorithm and the bond fluctuation method were tested on a serial computer using very similar programs on single polymers with $N = 100$. The details of the comparison are given in [80]. The upshot of the discussion is that the extrinsic speed of the double space algorithm in terms of computer time is about $2\frac{1}{2}$ times greater than the bond fluctuation method. This is partly because

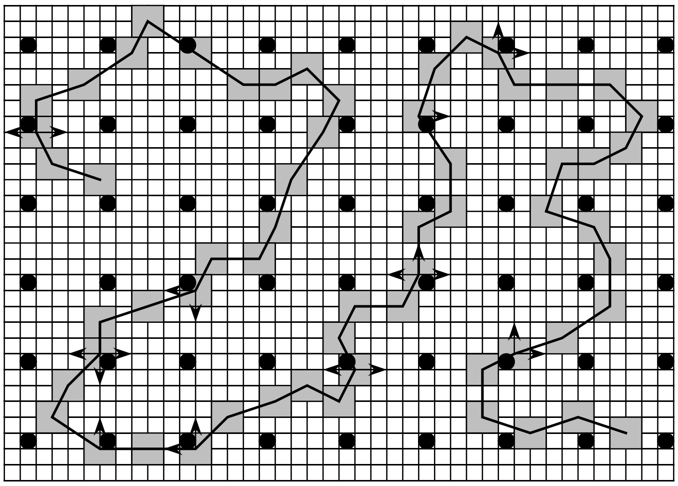


Figure 5-9: Realization of the polymer strand in the bond fluctuation method with $N = 60$. All monomers containing a dot can move in parallel without interference, and the arrows show all moves allowed by the constraints.

the double space algorithm is simpler, but it is primarily due to the fact that, in the bond fluctuation method, which monomer to update must be decided at random in order to satisfy detailed balance. This requires an extra call to the random number generator followed by finding the remainder of division by the length of the polymer.

The intrinsic speed of the double space algorithm in terms of relaxation time in units of Monte Carlo steps is also about $2\frac{1}{2}$ times faster than the bond fluctuation method because the polymers are more flexible. This value is the ratio of the prefactors in the fit of the scaling law for the relaxation time to measurements (to be described in the next section) taken from simulations using the two algorithms. The reason for this dramatic difference stems from the fact that the relative change in the bond vectors on a given step is much larger in the double space algorithm, allowing the polymer to relax faster.

Perhaps the most significant advantage of the double-space algorithm from a theoretical point of view is the fact that it is inherently parallel—a full half of the polymers can be updated simultaneously. Whereas great effort is required to vectorize traditional Monte-Carlo polymer simulations [98], formulation of the problem in terms of cellular automata insures that the algorithm is parallelizable from the outset. Hence, the technique marks a conceptual milestone in our thinking about computational polymer physics. With the advent of massively parallel computers, cellular automata machines, and the like, inherent parallelism is becoming an increasingly practical advantage as well.

The final advantage of the double-space algorithm is a practical consideration as well as an aesthetic one: simplicity. Simplicity often makes it easier to work with and reason about models. The initial incentive for developing the simplest possible algorithm stems from the desire to implement polymer simulations on cellular automata machines, where compact representations are important. Similarly, with conventional computers and programming languages, a simple algorithm leads to small, fast programs, and small programs are easier to write, debug, optimize, execute, maintain and modify. The conceptual simplicity can be appreciated when extending the algorithm to three dimensions. In the bond fluctuation method, the extension requires careful consideration of the problem of polymer crossing, and the resulting solutions are unnatural. In contrast, it is easy to see that the short bonds in the double-space algorithm, combined with exclusion of non-neighbors, do not allow crossing, and no ad-hoc restrictions have to be imposed to make the model work.

A potential disadvantage of the double space algorithm relative to the bond fluctuation method is that the bonds are short compared to the excluded volume radius of a monomer. This makes more monomers necessary for a given level of approximation to a true polymer configuration (compare figures 5-6 and 5-9). Thus, more monomers may be required to show the effects of entangled or knotted polymers. Finally, the polymer partitioning of the double space algorithm may be hard to generalize to more realistic polymer simulations. In this case, spatial partitioning can still be used to achieve parallelism.

5.3 Results of Test Simulations

Now that we have successfully developed CA algorithms for abstract polymers, we want to put the techniques on a more mathematical footing and gain a quantitative understanding of polymer behavior. This is accomplished through measurements taken from simulations and through theoretical arguments and calculations. Of primary interest here is the characteristic radius of a polymer along with its associated relaxation time and how these quantities scale with N. The measurements given below together with the derivation of scaling laws given in appendix E provide a satisfying closure of theory and experiment. This in turn extends the applicability of CA methods and paves the way for more simulations and theoretical development.

This section presents an example of the type of measurements and analysis that can be done on lattice polymers while making quantitative tests of the double space algorithm. There are numerous static quantities one might sample, but the radius of gyration is one of the most basic and gives us some essential information. Checking the measurement of the radius against the Flory scaling law serves as a verification of the concepts behind abstract models as well as as a test of the correctness of the algorithm. The measurement of the relaxation time of the radius of gyration shows that the algorithm is consistent with Rouse dynamics [23] and also serves to verify the efficiency of the algorithm.

The graphs below are based on data taken from a series of 30 simulations of isolated polymers ranging from $N = 2$ to $N = 240$. These are low density systems where serial computers are adequate and are used for convenience.[3] Each run consists of 100,000 samples, and for economy of measurement, the system is allowed to relax δt steps between samples. The value of δt ranges from 1–3000 in rough proportion to the expected scaling law for the relaxation time, $\delta t \cong \tau \sim N^{5/2}$. This gives approximately 40 samples per relaxation time which is sufficient to resolve the slower fluctuations. Each system was allowed to equilibrate by skipping the first 100 samples (which is over two relaxation times), and the statistics are based on the remaining "good" data.

The radius of gyration of a polymer is a second moment of the monomer distribution relative to the center of mass and is given at a time t by

$$R_g(t) = \sqrt{\overline{\mathbf{r}^2} - \overline{\mathbf{r}}^2}, \tag{5.1}$$

[3]For $N = 3$, the serial implementation allows the monomers at the ends to overlap without fusing. Hence, for $N = 2$ and 3, the polymers reduce to ideal random walks with nearest and next nearest neighbor bonds.

where $\mathbf{r}$ is the current position of a monomer, $\overline{\mathbf{r}}$ is the center of mass, and the bar denotes an average over the N monomers. Initially the polymers are fully stretched in the y direction, giving a radius of gyration of

$$R_g(t=0) = \sqrt{\frac{N^2-1}{12}}. \tag{5.2}$$

It should be noted in passing that the moments of a distribution can be calculated using the resources of a cellular automata machine. Built-in counters (e.g., see section 3.3.1) make it possible to rapidly count features of interest in different regions of the space, and the counts can be combined in an arbitrary way on the host computer. For example, by summing the nth power of x (the column number) weighted by the count of monomers on that column, one obtains the moment

$$\overline{x^n} = \frac{1}{N}\sum_{i=1}^{N} x_i^n, \tag{5.3}$$

where x_i is the x coordinate of the ith monomer. By adding another bit plane running a CA rule which sweeps a diagonal line across the space, it is possible to count monomers on lines of constant $(x+y)$. This can in turn be used to find expectations of powers of $(x+y)$ which enables one to extract the expectation of xy and so on. By diagonalizing the matrix of second moments, one can find the overall orientation of a polymer. Determining additional ways of measuring characteristics of polymeric systems using CA directly without reading out an entire configuration is an interesting area of research. Appendix F deals with some mathematical issues related to the general problem of measurement by counting over patches of space.

Figure 5-10 shows a typical time series for the radius of gyration. It consists of the first 1000 samples (1% of the data) from the simulation of the polymer with $N = 140$ and $\delta t = 800$. The line segment above the plot shows one relaxation time as derived from the time series using the method described below. Starting from a stretched state, the polymer contracts due to statistical forces, and the radius fluctuates around its mean value with a characteristic standard deviation σ. The relaxation time gives a

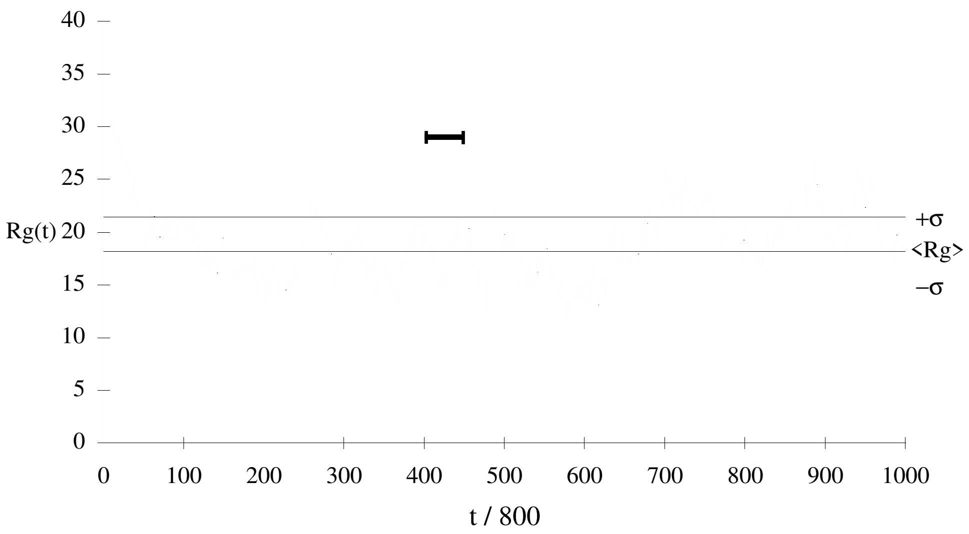


Figure 5-10: The radius of gyration as a function of sample number for a lattice polymer with $N = 140$ in the double space model. At $t = 0$, the polymer starts out fully stretched in the vertical direction. The horizontal lines show the mean and standard deviation. One relaxation time is indicated above the plot.

typical time scale for the initial contraction and for fluctuations away from the mean.

The mean radius of gyration for a given value of N is

$$\langle R_g \rangle \equiv \langle R_g(t) \rangle, \tag{5.4}$$

where $\langle \cdots \rangle$ denotes a time average over the good data. The variance of the radius of gyration is then

$$\sigma^2 \equiv \langle (R_g(t) - \langle R_g \rangle)^2 \rangle. \tag{5.5}$$

Figure 5-11 shows a log-log plot of the mean radius of gyration vs. the number of bonds $(N - 1)$. The number of bonds was used instead of the number of monomers N because it gives a straighter line, and it may be a better measure of the length of a polymer besides. In either case, the radius follows the Flory scaling law in the limit of large N:

$$\langle R_g \rangle \sim N^\nu \quad \text{where} \quad \nu = \frac{3}{d+2}. \tag{5.6}$$

The dotted line shows a least-squares fit through the last thirteen points and is given

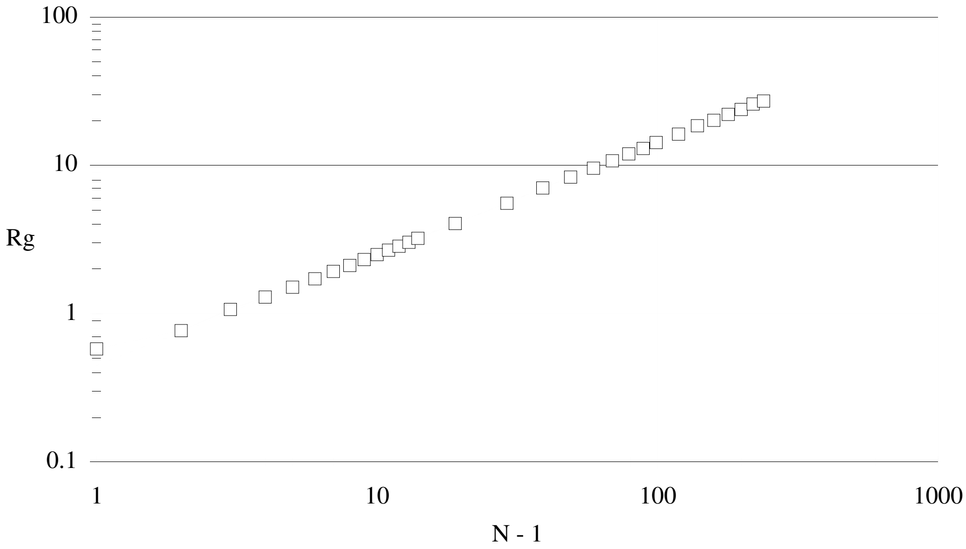


Figure 5-11: The mean radius of gyration of an isolated polymer in the double space model as a function of the number of bonds. The dotted line shows a least-squares fit to the asymptotic region of the graph.

by

$$\langle R_g \rangle \cong (0.448)(N-1)^{0.751}. \tag{5.7}$$

The exponent is close to the exact value of $\nu = 3/4$ for two dimensions, and the line gives an excellent fit to the data even down to $N = 3$.

The time autocorrelation function of the radius of gyration for a given time separation Δt is

$$\rho(\Delta t) = \frac{\langle (R_g(t+\Delta t) - \langle R_g \rangle)(R_g(t) - \langle R_g \rangle) \rangle}{\langle (R_g(t) - \langle R_g \rangle)^2 \rangle}, \tag{5.8}$$

where the time average is again limited to the good data. Figure 5-12 shows a semilogarithmic plot of the time autocorrelation function vs. the sample separation. This curve shows how much the polymer "remembers" about its former radius of gyration after a time Δt has passed. Such correlations are often assumed to fall off exponentially, $\rho(\Delta t) = \exp(-\Delta t/\tau)$, and the dotted line shows the best-fit exponential as explained below. However, it has been my experience that these curves always appear stretched, and a stretched exponential $\rho(\Delta t) \cong \exp(-(\Delta t/\tau)^{\alpha})$ with $\alpha \cong 0.8$ would give a much better fit.

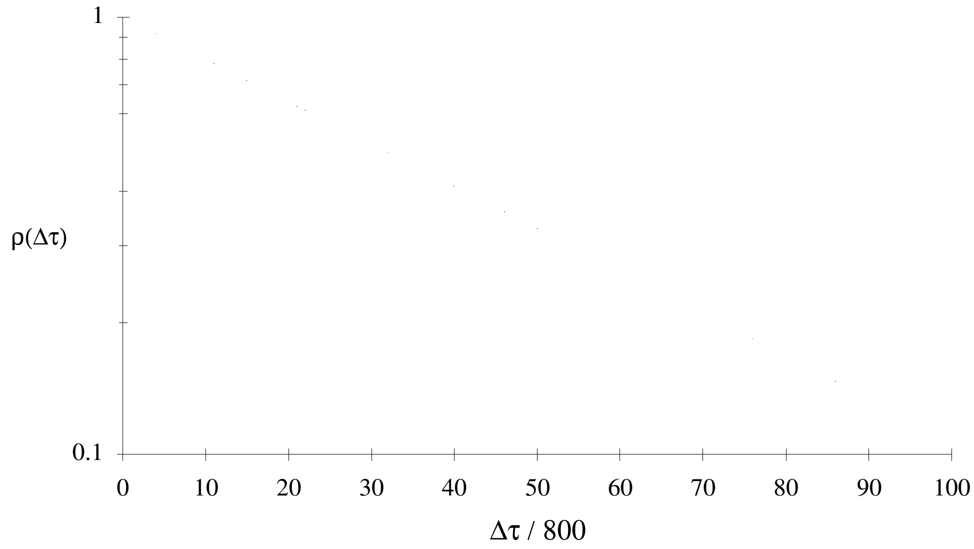


Figure 5-12: A semilogarithmic plot of the time autocorrelation function of the radius of gyration for the polymer with $N = 140$. The dotted line is chosen to have the same area under the curve, and its slope gives a measure of the relaxation time.

Assuming an exponential form for the time autocorrelation function, the relaxation time τ is given by the area under the curve. In order to avoid the noise in the tail of the curve, the area can be approximated by (as in [63])

$$\tau = \frac{\int_0^{t_e} \rho(t)dt}{1 - \rho(t_e)}, \tag{5.9}$$

where t_e is the first point after the curve drops below $1/e$. The value of τ can be interpreted as a characteristic time over which the polymer forgets its previous size. Figure 5-13 shows a log-log plot of the relaxation time vs. the number of bonds. The relaxation time is expected to obey the following scaling law in the limit of large N:

$$\tau \sim N^{2\nu+1} \quad \text{where} \quad 2\nu + 1 = \frac{d+8}{d+2}. \tag{5.10}$$

The dotted line shows a least-squares fit through the last thirteen points and is given by

$$\tau \cong (0.123)(N-1)^{2.55}. \tag{5.11}$$

The exponent is close to the exact value of $5/2$ for two dimensions, and the fit is very

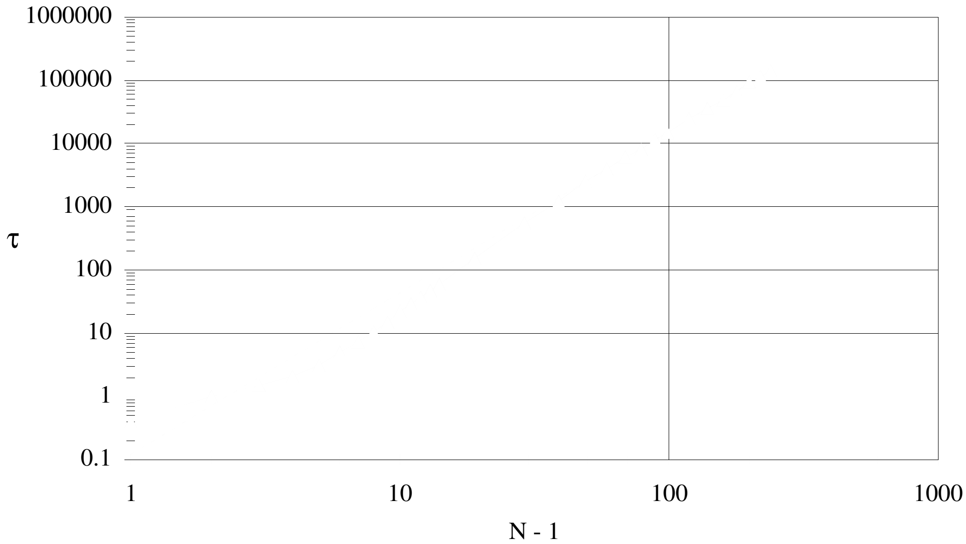


Figure 5-13: The Rouse relaxation time as a function of the number of bonds as derived from the radius of gyration. The dotted line shows the least-squares fit to the asymptotic region of the graph.

good for $N \geq 50$. The fact that the prefactor is much less than one time step says that the intrinsic relaxation rate in the double space model is high. Recall that the prefactor was 2.5 times greater for the bond fluctuation method. The relaxation is even faster than expected from about $N = 4$ to $N = 40$, but the origin of the dip is not entirely clear.

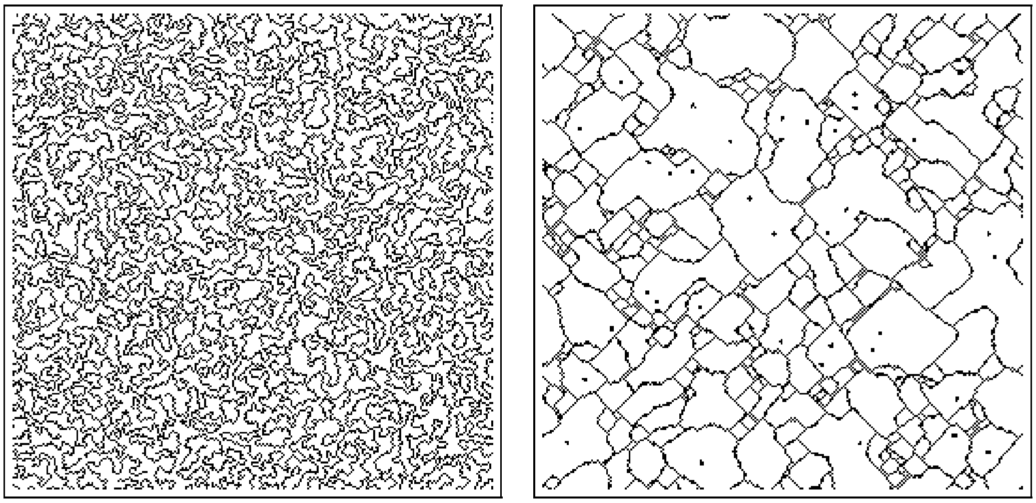
5.4 Applications

The models introduced above for doing polymer simulations with CA open up numerous possibilities for applications and further development. Some investigations, including those involving three dimensional systems, are fairly straightforward, while others will require creative breakthroughs in CA modeling techniques. To begin with, it is always possible to do more detailed measurements and analysis, and the results may verify or demand theoretical explanations. Of course eventually, one will want to reconcile the newfound understanding with actual experiments. In addition to more analysis, much of what can be done involves varying the system geometry or the initial conditions, e.g., studying different polymer topologies [37, 51, 63]. Most interesting from the point of view of developing CA methods is the potential for introducing new rules and interactions. Examples of each of these types of extensions is evident in what follows.

More realistic polymer models would have to exhibit chemical reactions arising from changes in internal energy between different configurations, and ways to do this will now be described. Different types of monomers and solvent molecules (which can be denoted by extra bits in a CA) may attract each other to varying degrees, and this can result in gelation, folding, or further polymerization. Equilibrium ensembles of polymer mixtures having well-defined energies can be sampled using the Metropolis algorithm [62] as follows. Trial moves are generated as before, but instead of checking for violation of absolute constraints, the moves are accepted or rejected probabilistically. The move is accepted unless the new state has a higher energy, in which case it is only accepted with a probability of $\exp(-\Delta E/kT)$. One must be careful to define the energy in such a way that a parallel update only changes the energy by the sum of the energies of the individual moves. The resulting Monte Carlo dynamics will generate a Boltzmann weight of $\exp(-E/kT)$ for configurations with energy E. As an alternative to having an energy-driven dynamics, one can construct ad hoc rules for stochastically forming or breaking bonds and let the dynamics emerge statistically. This would be more in line with our desire to find the simplest mechanism for causing a given behavior. Either way, random numbers are clearly an important part of CA rules for abstract molecular simulation, and useful techniques for incorporating randomness in CA are discussed in appendices A and C.

5.4.1 Polymer Melts, Solutions, and Gels

This section and the next describe some of the variations on the double space algorithm that have just begun to be explored (see [21] for ideas). The models are still athermal, meaning that there is no temperature parameter which can be adjusted, but they show how much can already be done with such trivial energetics. The simulations discussed below have been run on a variety of computer platforms, but cellular automata machines and other massively parallel computers are better for high den-



(a) (b)

Figure 5-14: CAM-6 simulations showing complex polymer configurations. Left: A medium density polymer melt consisting of 128 chains each with $N = 140$. Right: A gel spontaneously formed from a random configuration of monomers by ignoring the excluded volume constraint.

sity systems [67]. However, even for low density systems, an advantage in doing them on cellular automata machines is that real-time visualization is immediately possible. These examples bring up several problems for future research.

An important application of polymer simulation is to polymer melts [12, 51]. An ongoing project is to study two-dimensional abstract polymer melts such as the one shown in figure 5-14. Preliminary measurements consisting of histograms, averages, and relaxation times have been made on over a dozen quantities including radii, bond lengths, and Monte Carlo acceptance ratios. Also measured are structure functions, mass distribution functions, and the diffusion rates of individual polymers. One novel part of this project is to quantify the shape of the region occupied by individual polymers as a measure of polymer interpenetration. This is done in order to determine how interpenetration scales with increasing polymer concentration c (taken with respect to the concentration c^* where separate polymers start to overlap) [72]. The necessary techniques for measuring the geometry of separate domains in a cellular space are presented in appendix F.

The presence of a solvent is an important factor in the behavior of real polymers. If the solvent it good, an isolated polymer will obey the Flory scaling law for the radius as indicated in the previous section, but if the solvent is poor, the polymer will tend to collapse. This naturally leads to the question of *how* a polymer collapses when it goes from good to poor solvent conditions. A crude way to mimic a poor solvent without introducing bonding energies is to simply remove the check of the excluded volume constraint shown in figure 5-7. This allows new bonds to form, and many monomers in the same space can occupy a single cell. If these monomers are taken together as a single monomer, it is possible to take into account the higher mass by having the monomer move less frequently. Such simulations on single polymers have been done on serial computers with the effect of mass on diffusion included in a phenomenological way [68]. These simulations suggest that polymers preferentially collapse from the ends instead of uniformly along their length in a hierarchical fashion as is the case for chains near the so-called Θ point (where steric repulsion and van der Waals attraction between monomers cancel).

The collapse mechanism can also be performed even more simply by allowing monomers to fuse while *not* changing their mass. This is the easiest thing to do on a cellular automata machine because there is no room for more than one monomer per cell in one space. While nonconservation of monomers is unphysical, it leads to interesting behavior. Starting from a random configuration, monomers quickly polymerize to create a network of polymer strands. Figure 5-14 (right) shows a foam that results from running for 500 steps from an initial monomer density of 28% in each space. CA dynamics along these lines could form the basis of models of gel formation, and it would be possible to investigate, for example, the correlation length or distribution of pore sizes as a function of the initial monomer concentration.

Possibly the most interesting area for application of polymer simulation in general is that of protein folding. This is a rich area for future research in CA simulations of abstract polymer models as well. In fact, investigations of polymer collapse were originally motivated with a view towards the protein folding problem. While real proteins are very complex objects and local structure counts for almost everything, there are still some interesting things that can be done with simulations on lattice proteins [13]. Furthermore, it is still possible to model highly complex interactions with

CA (including solvents) by introducing intermolecular potentials and using spatial partitioning.

One important aspect of polymer physics that is missing from all of the above lattice Monte Carlo models are hydrodynamic interactions; that is, the transport of a conserved momentum between polymer strands and the intervening fluid. The examples here assume that diffusion is the dominant mode of transport, but hydrodynamics is often necessary for a proper description of dynamical behavior. Having a CA model of such behavior would enable interesting studies of, for example, the effect of polymers on fluid flow. It is easy to have momentum conservation in a gas of point particles, but it is difficult to obtain for systems of polymers. The question of how to obtain momentum conservation for extended objects is brought up again in chapter 6.

5.4.2 Pulsed Field Gel Electrophoresis

Electrophoresis of polymers through a gel can be used to separate DNA fragments by length, and this constitutes an important technique in advancing genetic research in general and the the Human Genome project in particular. Separation using a constant electric field is only effective on relatively short chains because longer chains become aligned with the field and subsequently exhibit mobilities that are independent of chain length. However, periodically changing the direction of the applied field by 90° (referred to as pulsing) causes entanglement and disentanglement of the DNA with the gel and leads to effective separation for longer chains [74]. Computer simulations of polymers and gels are useful for understanding and improving electrophoretic separation techniques [76]. Therefore we would like to modify the double space algorithm in order to simulate electrophoresis, with and without pulsed fields. This section discusses procedures and results from this project [77, 78].

Two straightforward tasks are necessary to turn a lattice Monte Carlo dynamics for polymers into a model of DNA electrophoresis: (1) modify the rule, and (2) create a suitable initial condition. The rule should be modified to give the polymers a net drift velocity and to support a background of obstacles which represents the gel. The bias in the direction of diffusion can be effected most simply by assigning unequal probabilities of performing steps in different directions. Alternatively, one can, at regular intervals, chose a definite direction of movement (say, to the right) for all the monomers. The initial condition will consist of relaxed polymers in a background gel matrix, and the simplest model of such a matrix is a fixed set of small obstacles. In two dimensions, these obstacles can be thought of as polymer strands of the gel which intersect the space perpendicular to the plane. The ability to vary the parameters in the rule, as well as the size of the polymers relative to the typical pore size of the gel, opens up a range of possibilities for experimentation with the model. The paragraphs below describe the implementation on CAM-6.

The electrophoresis dynamics on CAM-6 proceeds in cycles which contain the equivalent of four steps of the ordinary double space algorithm: essentially one for each of the four possible directions of movement. For this reason, time in the simulation described below will be measured in terms of cycles. Once per cycle, only moves to the right are accepted, which causes an overall drift to the right. Because of the way monomers are swapped in the CAM-6 implementation, 2×2 blocks of even monomers superimposed on a similar blocks of odd monomers (eight monomers total) forms a tightly bound polymer that cannot move. Therefore, the immovable obstructions can be made out of monomers arranged in these blocks, and no further modifications to the rule are necessary. Moreover, there is a one-cell excluded volume around each block, so that no other monomers can touch it. Thus, each block maintains an excluded volume of 4×4 which reduces the pore size accordingly.

Initial configurations of the system are prepared on a serial computer in three stages and then transferred to CAM-6. First, a given number of 2×2 blocks comprising the matrix are placed at random, one at a time. Placements within a prespecified distance from any other block (or the edges of the space) are rejected and another placement is found for the block. This causes an anticlustering of the obstructions which makes them more evenly distributed than they would be without the distance constraint. Second, the polymers are distributed in a similar fashion to the blocks by starting them out in straight, vertical configurations. In this case, the polymers must be kept at least one cell away from the blocks and other polymers. Finally, the polymers are allowed to relax into an equilibrium state by using a fast, nonlocal Monte Carlo dynamics. This so-called reptation dynamics randomly moves monomers from either end of a polymer to the other end, subject to the constraints of the model. This algorithm produces the correct equilibrium ensemble and is used here only to establish a starting configuration for the modified double space algorithm dynamics.

Configurations from the CAM-6 simulation of DNA gel electrophoresis are shown in figure 5-15. The initial condition (a) contains 1000 of the immovable blocks and 128 polymers, each with $N = 30$. The blocks were scattered with a minimum allowed separation of 5 in lattice units, giving a typical separation of 8 (for a typical pore size of 4). The density of polymers was chosen to be artificially high in order to show many kinds of interactions in a single frame. The characteristic radius of the polymers is larger than the pore size in this case, so the chains must orient themselves with the field in order to squeeze though. Figure 5-15(b) shows the system after running for 100 cycles, and virtually all of the polymers have started to be caught on the gel. Furthermore, many of the polymers are draped over more than one obstacle, while the few that are oriented along the field can continue to move. After 100,000 cycles (figure 5-15(c)), the polymers have diffused around the space several times and have become entangled with the matrix to form extended mats. Note that this means the polymers must be able to repeatedly catch and free themselves from the blocks.

The final part of this simulation shows the effect of pulsed fields (figure 5-15(d)). The switching was accomplished by periodically turning the configuration by $\pm 90°$ while keeping the rule (and therefore, the direction of the bias) the same. Starting from figure 5-15(b), the field direction was switched five times followed by running 20 cycles of the rule after each switch (for a total of 100 additional cycles). The result is a net drift at 45°, meaning that the polymers do not have time to reorient themselves in 20 cycles. At very low frequencies, one would expect the polymers to have time to respond to the change and alternately drift at right angles. At some intermediate "resonant" frequency, the polymers should have a minimum mobility due to being constantly obstructed by the gel.

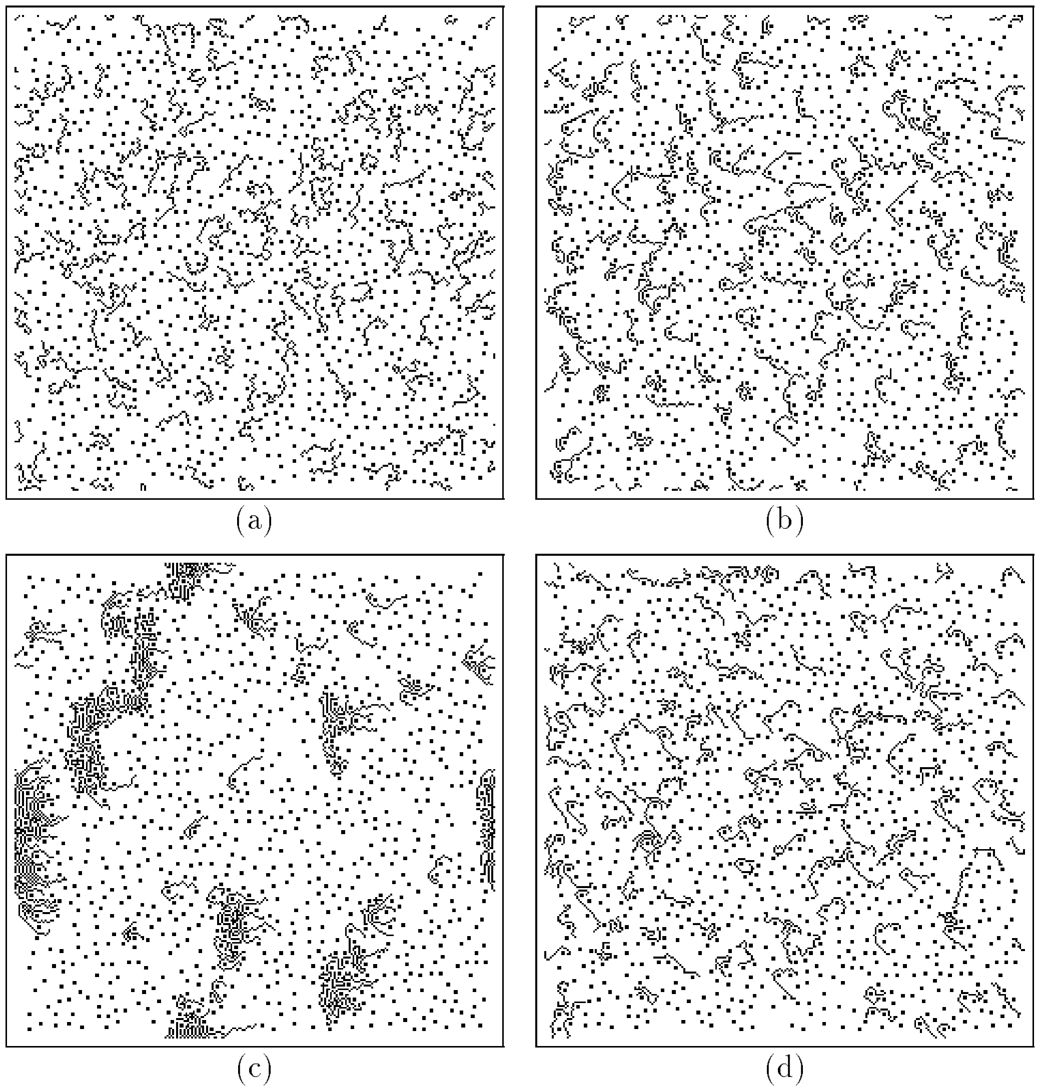


Figure 5-15: Electrophoresis demonstration on CAM-6. (a) Initial equilibrium configuration consisting of 128 polymers with $N = 30$ in a background of 1000 immovable 2×2 blocks. (b) After 100 cycles of the rule, most of the polymers have become caught on an obstruction. (c) After 100,000 cycles of the rule, the polymers have drifted around the space until most are caught in one of a few clumps. (d) A configuration showing the effect of switching the direction of the field.

In addition to visual examination of more configurations, this research could be continued in several ways. For example, by adding labels to the polymers one could trace the position of individual polymers and measure polymer mobility as a function of polymer size, gel density, and switching frequency. Other possibilities include modifying the polymer rule, changing the matrix, and going to three dimensions.

One problem with this rule as it stands (and in lattice models in general) is that tension is not transmitted along the chain as readily as it would be in a real polymer. This is related to the problem of momentum conservation mentioned previously. Furthermore, there are configurations in which polymers cannot slide along their own contours (e.g., see figure 5-15(c)), and it takes an inordinately long time for them to break free from the obstructions. This would be less of a problem with a weaker driving force, though simulation time would also increase.

5.5 Conclusions

Cellular automata are capable of modeling complex physical situations by moving abstract atoms according to physically motivated rules. An important technique in many of these simulations is the Monte Carlo method, and polymer physics is one area where CA can be successfully employed. Polymers are especially interesting from the point of view of abstract modeling because they can often be described by scaling laws which arise from averaging out the relatively unimportant details.

The availability of cellular automata machines and other massively parallel computers make it desirable to find CA rules for modeling polymers. The essential features of lattice polymer dynamics that must be retained in a parallel implementation are that the chains maintain connectivity and excluded volume. The technique of spatial partitioning can be used to construct parallel polymer dynamics quite generally, but the resulting lattice polymers are inflexible and the dynamics are slow. These problems are elegantly solved by the double space algorithm. This algorithm is well suited to CA and is superior in most respects to the alternative bond fluctuation method.

Physical modeling with CA is a quantitative field which lends itself to a range of interesting experiments. In many cases, it is possible to use additional CA rules as tools for measurement and analysis. Extensive simulations of the double space algorithm show that it obeys the correct scaling relations for the characteristic radius and its relaxation time. Hence, it forms a suitable basis for further modeling.

Many applications are immediately made possible by the techniques given here including extensive simulations and measurements of polymers in melts and complex geometries. Even more applications would be made possible by adding chemical reactions, and this can be done with the Metropolis algorithm or by ad hoc means. Simulations of electrophoresis of polymers in a gel show how they can get caught on obstructions. While momentum conservation is automatic in molecular dynamics simulations, it is still an outstanding problem to find a sensible way to add momentum conservation to CA simulations of polymers.

Chapter 6

Future Prospects for Physical Modeling with Cellular Automata

6.1 Introduction

Each of the models presented in the previous chapters brings up a number of problems for additional research. Besides these follow-up problems, several topics for research in new directions are indicated below, and some trial models are given as seeds of investigation. The models are lacking in some respects, but they can already be studied as they stand; this will undoubtedly lead to further ideas in the course of the work. Undoubtedly by now, the reader also has many ideas for modeling with CA.

6.2 Network Modeling

A class of complex dynamical systems that is becoming increasingly important in the modern age is that of information networks. The most familiar example of such a network is the telephone system. Other notable examples are the Internet and the local area networks that link computers within a single building. Depending on the amount of bandwidth available, they can carry textual, numerical, audio, or even video data. The technology can be put to numerous uses, and there is always more demand to increase the throughput [40].

Networks can be anywhere from single chip to global in scale. They consist of a number of nodes (which are often just computers) connected by communications channels. Information flows from a source node to a destination node, possibly passing though several nodes in between. The nodes are responsible for traffic control, and a central problem is to determine how to route the signals. A *circuit-switched* network is an older type of network, originally suited to carrying analog signals, and it provides a continuous connection from source to destination. The advent of digital information processing has made possible the more sophisticated *packet-switched* network in which digital signals are chopped into packets of data which travel separately, only to be reassembled at the destination. Each packet of data contains an address telling it where to go and a time stamp telling its order. The advantage of packet switching over circuit switching is that it is much more flexible in how the resources of the network are used: the packets can travel whenever and wherever a channel is open without having to have a single unbroken connection established for the duration of the transmission.

The dynamics of a packet-switched network emerge from the switching of the nodes in response to the traffic. As the traffic in the network increases, conflicts start to arise when two or more packets want to use the same channel, and one must devise a communication protocol which works correctly in such cases. Some possibilities are to temporarily store the packet, drop the packet and try the transmission later, or deflect the packet along another channel that is open and hope that it eventually finds a way to its destination. These techniques may be used in combination, but each of them obviously has drawbacks. Deflecting the packets is a good way to keep the data moving, but it may also lead to further conflicts. Under certain conditions, the conflicts can feed back and develop into what is known as a "packet storm," wherein the information throughput suddenly drops. It is not entirely understood how packet storms originate, how to prevent them, or even what they *look* like, and this is where CA come in. Cellular automata have the potential of simulating the behavior of complex networks, and hopefully they can provide a visual understanding

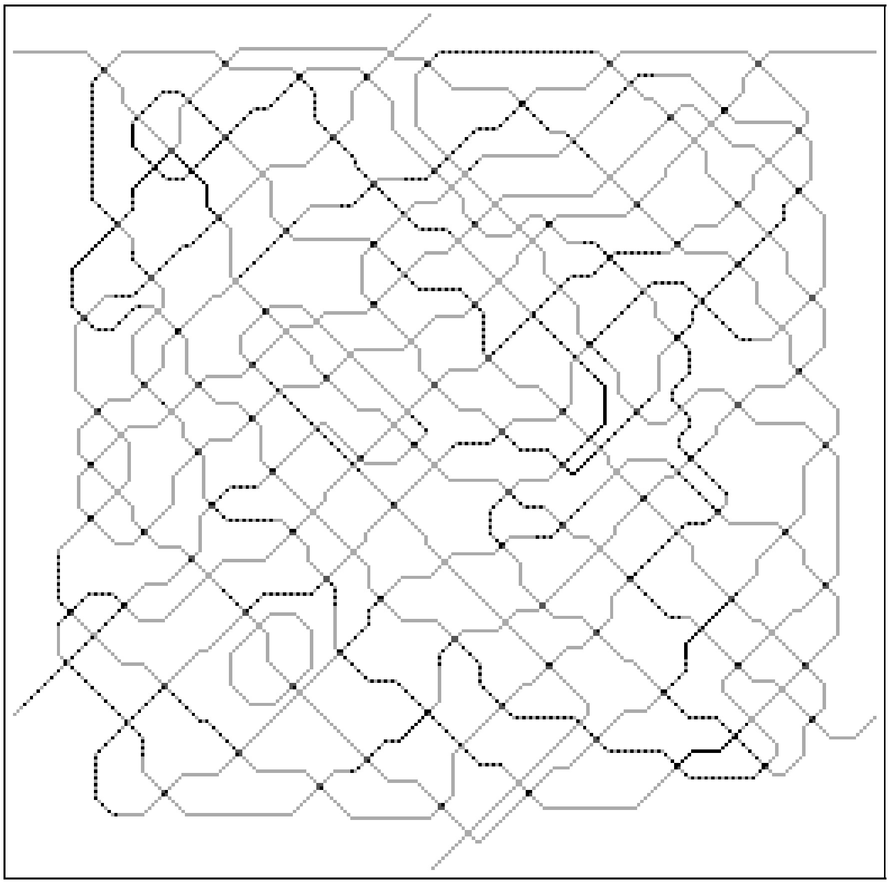


Figure 6-1: Example of a network with data packets of various lengths traveling along bidirectional wires. The nodes have four ports and switch signals coming in on one port to one of the other three ports.

of phenomena such as packet storms.[1]

The basic idea for using CA to simulate networks is illustrated by the CAM-6 configuration shown in figure 6-1 (compare this to the digital circuits shown in figure 2-20). The lines represent wires, and the sequences of bits on the wires represent indivisible packets of data. There are a total of 38 data packets ranging in length from 1 to 51 bits. The dark squares where the wires meet are the nodes, whereas the light intersections are just crossovers. This network has 100 nodes, each of which is connected to four others (with a total of 200 wires). The wires are actually bidirectional since they transmit the bits by swapping the values in adjacent cells.

The network operates as follows. The nodes receive packets as they come in

[1]The author would like to thank Randy Hoebelheinrich for suggesting this application.

through one of the four ports and then immediately send them out on *different* ports. Furthermore, the packets are always sent out on a wire that is not currently being used, so the packets will never merge into each other. This is already describes the essential functionality of the network, but there are some additional details. Since each node has four inputs and four outputs, it can have up to four packets going through it at any given time. Each node has nine possible internal settings corresponding to the nine derangements (i.e., permutations that change the position of every object) of the four ports, so that each derangement specifies which input is connected to which output. The internal state of each node merely cycles through all nine settings unless there are packets going through in which case it cycles through only those settings that will not break the packets.

This network dynamics as described has a number of special properties. First, the packets are just strings of 1's with no sources and no destinations, so they are just being switched around in a pseudorandom fashion. Second, the packets keep their identity and never merge. Third, the wires carry bits in a perfectly reversible way. Finally, the nodes cycle through their internal states in a definite order, so the overall dynamics is reversible. Reversibility gives the remarkable property that the information in the network can never be lost, and there can never be a conflict. Unfortunately, it also means that the packets cannot have destinations to which they home. If they did, it would mean that there is ambiguity as to where a given packet has come from, and this incompatible with a one-to-one map. In order for them to be able to retrace their paths, they would have to have additional degrees of freedom in which to record their history. All of this is interesting from a physical point of view because it means that the network has a nondecreasing entropy, and could be described in thermodynamic terms. A more realistic network simulation would inject packets containing forwarding addresses and would remove them once they reach their destinations. Note, however, that there may not be room to inject a packet and the operation of removing them would be irreversible.

6.3 The Problem of Forces and Gravitation

One of the most basic physical notions is that of a force. Intuitively, forces are what "cause things to happen." There are several ways in which they are physically manifested: as a push or pull between the pieces of a composite object, as a contribution to the Hamiltonian (or energy) of a system, and even as "fictitious" effects due to a choice of a coordinate system. In one way or another, forces underlie almost every aspect of physics. Hence, a desirable feature of any paradigm for physical modeling is the ability to include forces. Forces can be of many types, and eventually we would like to be able to simulate them all.

We would like to be able to create forces within CA in order to simulate a variety of phenomena of physical interest, and in particular, gravitational attraction. In chapter 3 one approach to generating an attractive interaction was to invoke statistical forces. A statistical force is created by exploiting the tendency for entropy to increase during the evolution of any complex reversible system. One merely has to devise a scheme whereby the statistically most likely (and therefore typical) response of the system is to follow the desired path.

The evolution of any nontrivial dynamical system can be attributed to forces, and CA are no exception. However, the usual view of CA as field variables means that the forces act to change the *states* of the cells, as opposed to changing the *locations* of the cells. If we want to model general motion through space, we must invent reversible interactions that lead to acceleration in the spatial dimensions of the CA. Now in a discrete spacetime, one seems to be limited to a finite set of velocities and positions, so forces will be impulsive and take on a finite set of values.[2] Therefore, it will not be possible to have a solitary particle following a nontrivial smooth trajectory as one would like.

Attractive forces are arguably more important than repulsive forces because they serve to hold matter together into functional entities. This is especially true in re-

[2]This conclusion is difficult to escape if the state of the system is to be interpreted locally and the amount of information is locally finite. What other interpretations might be useful to get around this problem?

versible CA since the natural tendency for systems of particles is to diffuse. Furthermore, it is harder to simulate long-range forces since interactions in CA are, by definition, short-ranged. Therefore, the particular problem we want to address in this section is the construction of a reversible CA for simulating attractive, long-range forces such as gravity.

In classical field theory, forces are gauge fields which are coupled to matter fields. In quantum field theory, interactions are often pictured as exchanges of gauge particles between matter particles. One approach to simulating forces in CA is to build a classical analogy based on these pictures. The idea is to introduce exchange particles which carry momentum between distant gas particles. The resulting models differ from real physics in several significant respects. First, the model is based on classical, deterministic collisions rather than on quantum mechanical probability amplitudes. Second, the exchange particles are real instead of virtual. Third, the exchange particles have a momentum which is opposite to their velocity. Finally, to make the model reversible, the exchange particles are conserved instead of being created and destroyed.

The model given here couples the TM and HPP gases which are defined by figure 3-5, and each gas obeys its own dynamics when it is not interacting with the other. The particles of each gas carry a momentum proportional to their velocity, but the key to obtaining attractive forces is that the exchange particles are considered to have a *negative* mass. The gases interact in a 2×2 partition whenever they can exchange momentum conservatively. The basic coupling is shown in figure 6-2, and others can be obtained by rotation and reflection of the diagram. Note that this interaction is reversible. One may think of the TM particles as the "matter" gas and the HPP particles as the "force" field. Instead of having virtual exchange particles with imaginary momenta, the exchange particles are considered to be real with negative mass which means that their momenta are directed opposite to their velocity.

What is the effect of this coupling between the gases? Since the rule is reversible, gravitational collapse cannot occur starting from a uniform initial condition because the entropy must increase, and there are no low entropy degrees of freedom available

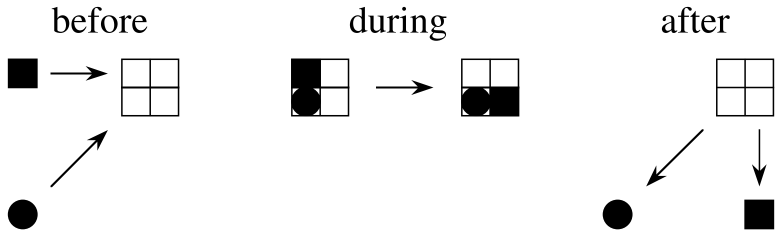
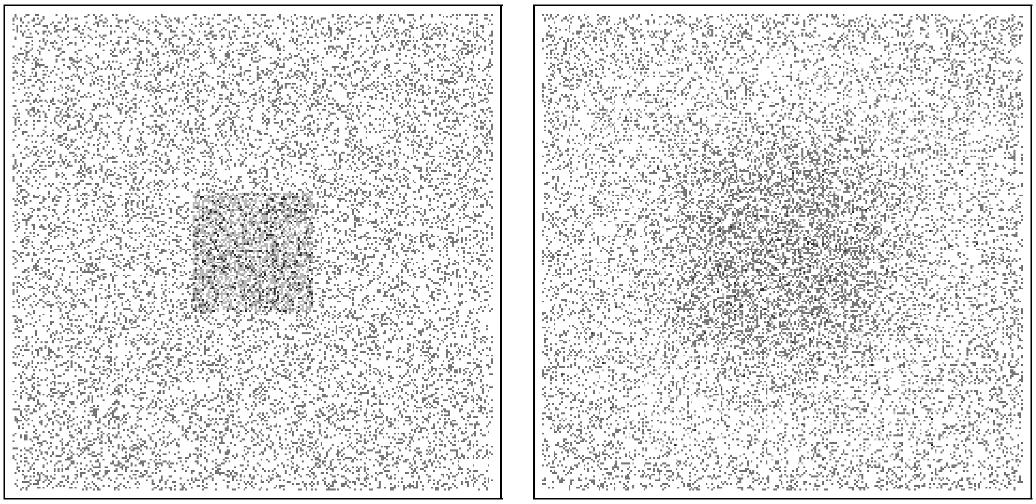


Figure 6-2: A collision rule for generating an attractive coupling between the two lattice gases. In order to conserve momentum, the mass of the HPP particle (circle) is taken to be negative one-half of the mass of the TM particle (square).



(a) (b)

Figure 6-3: (a) A uniform gas of matter particles, containing a block of force mediating particles. (b) Attraction of the matter particles by the expanding cloud of force particles.

to use as a heat sink. However, it is possible to obtain an attraction by starting from a nonuniform initial condition as shown in figure 6-3. Initially (6-3(a)), there is a random 64×64 block of HPP (force) particles in a 20% background of TM (matter) particles. As the field moves outward, it attracts the gas and reverses direction as described by figure 6-2. After a few hundred steps, the block has swollen somewhat, but a rarefaction has developed in the gas while its density in the center has increased (6-3(b)).

Unfortunately, the contraction is only a transient effect because the force particles continue to diffuse through the matter particles while the entropy continues to rise. Eventually, the system becomes completely uniform, and there is no further attraction. One problem with having real exchange particles instead of virtual ones is that they are not obligated to be reabsorbed in an attractive collision. Rather, as is the case here, they often miss the matter particles they were meant to contain, and then they can no longer contribute to the gravitational field. However, the question still remains, is there some way to get around these problems and make a reversible CA model of gravitation?[3]

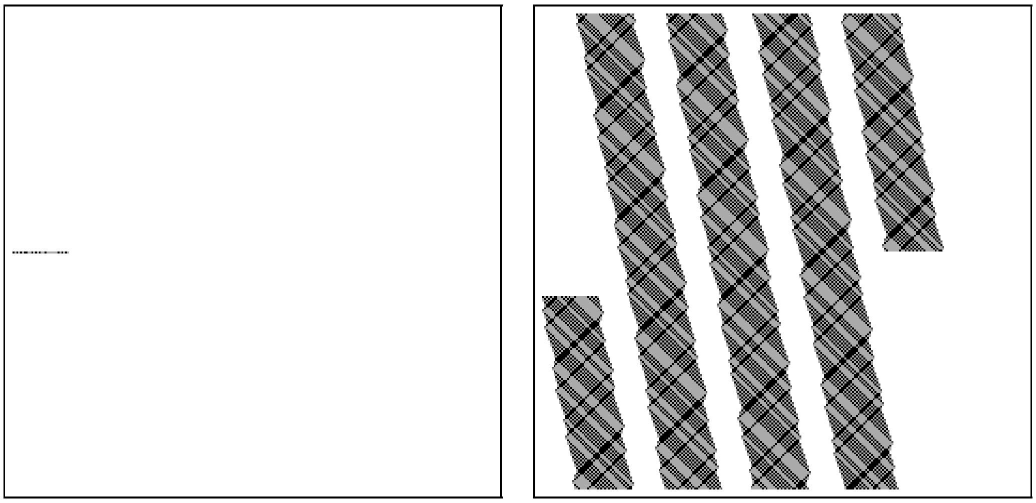
6.4 The Dynamics of Solids

Another area of physics in which forces play an important role is in the dynamics of "rigid" bodies. Examples such as billiard balls are common in classical mechanics, and the ability to simulate them would open a host of applications. A successful simulation would probably solve the more general problem of simulating intermolecular forces as well.

The question is basically this: How does one program a CA to support the motion of macroscopic solid bodies? This question has been dubbed the "solid body motion problem" and was first asked by Margolus [57], but a definitive statement of the problem has never really been elucidated. The question also illustrates the problem with coming up with fundamental dynamics of physical interest. Several members of the Information Mechanics Group have worked on this problem with limited success [15, 14, 42]. The one-dimensional case has been satisfactorily solved by Hrgovčić and Chopard independently by using somewhat different modeling strategies. Unfortunately, one-dimensional problems often have special properties which make them uniquely tractable [54].

An implementation of the solution proposed by Hrgovčić for the one-dimensional case is shown in figure 6-4. Because of the way the object moves, it has been dubbed "plasmodium" after the amoeboid life form. The object consists of a "bag" which is under tension and which contains a lattice gas (figure 6-4(a)). The gas particles

[3]It *is* possible to simulate condensation and phase separation in a lattice gas by having a nonlocal, irreversible rule that is reminiscent of the one given here [102].



(a) (b)

Figure 6-4: (a) The initial condition consisting of a line segment 30 cells long and with the internal particles moving predominantly to the right. (b) As time evolves, the momentum of the internal lattice gas carries the object to the right.

travel back and forth through the bag at the speed of light, and they extend the bag by one cell by being absorbed when they hit the end. Conversely, the bag is retracted by one cell when a particle leaves. Figure 6-4(b) shows a spacetime diagram of 1000 steps of the evolution of this object. The excess of internal gas particles moving to the right is apparent.

6.4.1 Statement of the Problem

To make the problem precise, we need to establish some definitions. The key features of CA that we want are local finiteness, local interactions, homogeneity, and determinism. What we mean by a solid body is a region of space that has an approximately constant shape and is distinguishable from a background "vacuum." Such regions should be larger than a CA neighborhood and have variable velocities. Roughly in order of importance, the solid bodies should have the following:

- A reversible dynamics
- Arbitrary velocities with a conserved momentum
- Realistic collisions

- Arbitrary angular velocities with a conserved angular momentum
- Arbitrary size
- Arbitrary shape
- A conserved mass and energy or mass-energy
- An internal solid state physics

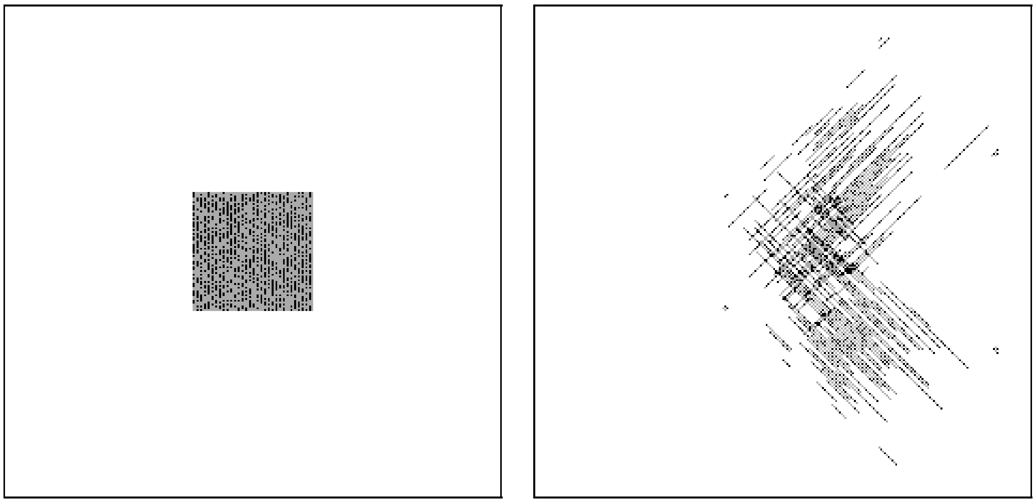
As mentioned before, reversibility is necessary to make our physical models as realistic as possible. The finite speed of propagation of information limits the velocities, so we can't demand too much in this regard. However, the dynamics of the objects should obey some principle of relativity. Similarly, conservation laws should be local. These requirements may imply an internal physics, but we would like to rule out "unnatural" solutions (such as navigation under program control).

This might not even be a precise enough statement of the problem since there may be solutions to the above that miss some characteristic that we really meant to capture. Deleting conditions will make the problem easier and less interesting. Alternatively, we may be ruling out some creative solutions by imposing overly strict demands.

6.4.2 Discussion

The problem as stated in its full generality above may very well be unsolvable. While no formal proof exists, there are indications to that effect. First and foremost is the constraint imposed by the second law of thermodynamics. In any closed, reversible system, the amount of disorder or entropy, must increase or stay the same. This is because a reversible system cannot be attracted (i.e., mapped many-to-one) to the relatively few more ordered states, so it is likely to wander to the vastly more numerous generic (random looking) states.[4] The wandering is constrained by conservation laws,

[4]This argument holds for even a *single* state in a deterministic system—it is not necessary to have probabilistic ensembles. The entropy can be defined as the logarithm of the number of states that have the same macroscopic conserved quantities.



(a) (b)

Figure 6-5: (a) A "solid" block of plasmodium which contains an HPP lattice gas moving to the right. (b) The object moves to the right by fragmentation along the directions of motion of the lattice gas particles.

but any interesting, nonlinear rule will not be so highly constrained that the system cannot move at all. This is especially true for systems whose constituent particles (as in a solid) are constantly jumping from cell to cell. This is clearly not a problem in classical mechanics where the constituent particles can smoothly move from one position to another.

Figure 6-5 shows an attempt to extend the plasmodium model to two dimensions. The idea is to run a lattice gas (HPP in this case) inside of a bag which grows and shrinks depending on the flux of particles hitting the inside edges of the bag. The resulting rule is reversible and almost momentum conserving. A more complex rule could be devised which would conserve momentum exactly, but the basic phenomenon would be the same. Figure 6-5(a) shows the initial state consisting of a square bag containing a lattice gas moving to the right. Figure 6-5(b) show the results of 500 steps of the rule. However, since the dynamics is reversible, the entropy of the system tends to increase. Adding surface tension to maintain the integrity of the object would make the rule irreversible. Such considerations illustrate the difficulty of making a reversible dynamics stable [3, 92].

The second problem is the lower limit on the size of a disturbance in a discrete system. For example, in a discrete space, one can't even imagine the smooth, continuous acceleration usually associated with $F = ma$. Another argument goes as follows. Any continuum mechanics of solids is likely to support sound waves of some description. Such a dynamics tends to distribute disturbances evenly, yet in more than one dimension, this implies a thinning of the wave. But it seems impossible to have a disturbance which shows up as less than a single bit in a cell. Perhaps correlated information can be less than a bit, but this seems too abstract to comprise a real solid object. It may be the case that any CA which has material transport always amounts to a lattice gas.

The final problem is particular to Chopard's "strings" model [14, 15], but it may be a reflection of deeper underlying problems. This model is quite simple and elegant, and it has several of the desired features. One reason that it works as well as it does is that the string is effectively one-dimensional, and as we have seen, this simplifies things greatly. Despite being so simple, it is not even described as a CA field rule, but rather as a rule on (integer) coordinates of particles. This would be fine, except that it is not clear that there is a good way to extend the rule to arbitrary CA configurations. One must instead place global restrictions on the initial conditions. This is a particular problem with regards to collisions, which cannot be predicted in advance. The collision rules considered lead to strings which overlap or stick.

Examining these points will give us clues to what restrictions need to be relaxed in order to solve the problem. For example, if we drop the demand for reversibility, we might be able to make a "liquid drop" model by adding a dissipative surface tension which would keep the drops from evaporating. Such considerations probe the power and limitations of CA in general.

6.4.3 Implications and Applications

The solid body motion problem is central because it embodies many of the difficulties encountered when trying to construct a nontrivial dynamics that gradually and reversibly transforms the state of any digital system. It addresses some hurdles encountered when attempting to apply CA to physics including continuity, relativity, and ultimately, quantum mechanics. Solving the more general problem of finding realistic dynamical fields would obviously have many applications. If this cannot be done, then it would eliminate many approaches to building worlds with CA. For this reason, I think the solid body motion problem is important.

This problem could lead to the development of CA methods for modeling stress and strain in solids that are complementary to lattice gas methods. One would like to be able to set up an experiment by loading a design of a truss, for example, into the cellular array. Given other methods for incorporating forces in these models, say, gravity, one could also put an external load on the structure. Running the automaton would generate a strain in the truss with stresses distributing themselves at the speed of sound until the system breaks or relaxes into an equilibrium configuration. The stresses and strains could be monitored for further analysis.

Finally, this problem has implications for the field of parallel computation as a whole. The basic question is how best to program parallel machines to get a speedup over serial machines which increases with the number of processors. And, of course, CA machines are the ultimate extreme in parallel computers. One ambitious, but difficult, approach is to make intelligent compilers that will automatically figure out how to parallelize our serial programs for us. Perhaps a more practical approach is to drive parallel algorithm design with specific applications. The solid body motion problem is a concrete example, and solving it involves the inherent coordination of the efforts of adjacent processors.

Chapter 7

Conclusions

Cellular automata constitute a versatile mathematical framework for describing physical phenomena, and as modeling tools they offer a number of conceptual and practical advantages. In particular, they incorporate essential physical features such as causality, locality, determinism, and homogeneity in space and time. Furthermore, CA are amenable to efficient implementation in parallel hardware as cellular automata machines which provide important computational resources. The study of physics using CA combines physics, mathematics, and computer science into a unified discipline.

This thesis advances CA methods in mathematical physics by considering fundamental physical principles, formulating appropriate dynamical laws, and developing the associated mathematics. A major underlying theme has been the importance of reversibility and its connection with the second law of thermodynamics. The primary contributions of this thesis are that it:

- Establishes the field of cellular automata methods as a distinct and fruitful area of research in mathematical physics.

- Shows how to represent external potential energy functions in CA and how to use them to generate forces statistically.

- Identifies and illustrates how to design dissipation into discrete dynamical systems while maintaining strict microscopic reversibility.

- Gives an efficient algorithm for doing dynamical simulations of lattice polymers and related systems on massively parallel computers.
- Elaborates some properties of a Lorentz invariant model of diffusion and discusses the relationship between CA and relativity.
- Presents and analyzes several techniques for exploiting random numbers on cellular automata machines.
- Sets out the groundwork for creating a calculus of differential forms for CA.
- Identifies promising areas of application of CA in a number of fields.

These contributions have been supported by simulations and mathematical analysis as appropriate.

The examples given in this thesis help to establish a catalog of CA modeling techniques. By combining these techniques, it is becoming possible for physicists to design CA and program CA machines—much as they now write down differential equations—that contain the essential phenomenological features for a wide range of physical systems. We are now in a position to embark on cooperative projects with scientists in many disciplines.

Appendix A

A Microcanocial Heat Bath

This appendix presents the derivation of some statistical properties of a microcanonical heat bath while illustrating some standard thermodynamic relationships in the context of a simple, self-contained dynamical model. It also discusses some of the uses of such a bath including that of a random number generator in Monte Carlo simulations. The heat bath given here is somewhat more general than the one illustrated in chapter 3. In particular, the bath consists of two bit planes instead of one, and they form a pair of coupled lattice gases which can hold from zero to three units of energy per cell. This makes for a greater heat capacity as well as a richer set of statistics. Similar baths of demons have been used in a number of variants of dynamical Ising models [20, 73], and further generalization is straightforward.

A.1 Probabilities and Statistics

For purposes of analysis, it is convenient to view the heat bath as completely isolated and having a fixed energy while its dynamics redistributes the energy among its internal degrees of freedom. Almost any randomly chosen reversible, energy-conserving dynamics will suffice. However, in an actual simulation, the heat bath will be free to interchange energy (or information if you prefer)—and nothing else—with the primary system.[1] When the systems come into thermal equilibrium, the zeroth law of thermodynamics tells us that the heat bath will be characterized by a temperature parameter which must be the same as that of the primary system. If the heat bath relaxes quickly compared to the system and they are not too far out of equilibrium, the assumption of a uniform temperature and independent random energy in each cell is approximately valid.

A.1.1 Derivation of Probabilities

Consider a collection of N cells of a heat bath that share a total energy E between them ($N = 65536$ on CAM-6). In equilibrium, there will be approximately N_i cells with energy $\varepsilon_i = i \in \{0, 1, 2, 3\}$ respectively, but if the cells are viewed as independent random variables, each will have an energy ε_i with a probability $p_i = N_i/N$. The number of equally likely (*a priori*) ways of assigning the N_i's to the N cells is

$$\Omega = \binom{N}{N_0\ N_1\ N_2\ N_3} = \frac{N!}{N_0!N_1!N_2!N_3!} = e^S. \tag{A.1}$$

The distribution of N_i's that will be seen (with overwhelming probability) is the same as the most likely one; i.e., it maximizes $S = \ln \Omega$ subject to the constraints:

$$N = \sum_i N_i \quad \text{and} \quad E = \sum_i N_i \varepsilon_i. \tag{A.2}$$

This problem can be solved by finding the extremum of the auxiliary function

$$\begin{aligned} f(\{N_i\}, \alpha, \beta) &= \ln \Omega + \alpha(N - \sum_i N_i) + \beta(E - \sum_i N_i \varepsilon_i) \\ &= N \ln N - N - \sum_i N_i \ln N_i + \sum_i N_i + \alpha(\cdots) + \beta(\cdots), \end{aligned} \tag{A.3}$$

where α and β are Lagrange multipliers which will be determined by satisfying the

[1] As in the potential well model, the energy transfers take place as part of updating the system. The heat bath can be stirred in between steps with its own separate dynamics. These phases may occur simultaneously as long as no additional constraints are inadvertently introduced and reversibility is maintained.

constraints (A.2):

$$\frac{\partial f}{\partial \alpha} = 0 \Rightarrow \text{number constraint}, \quad \frac{\partial f}{\partial \beta} = 0 \Rightarrow \text{energy constraint}. \tag{A.4}$$

The condition for maximum entropy then becomes

$$\begin{aligned} \frac{\partial f}{\partial N_i} &= -\ln N_i - 1 - \alpha - \beta\varepsilon_i = 0 \\ &\Rightarrow N_i \propto e^{-\beta\varepsilon_i}, \end{aligned} \tag{A.5}$$

which is the usual Boltzmann distribution for energy where the temperature is defined as $T \equiv 1/\beta$. Therefore, the probabilities are given by

$$p_i = \frac{e^{-\beta\varepsilon_i}}{z} \quad \text{where} \quad z \equiv \sum_{i=0}^{3} e^{-\beta i} = \frac{1 - e^{-4\beta}}{1 - e^{-\beta}}. \tag{A.6}$$

The energy of each cell can be represented in binary by introducing a description in terms of energy demons—tokens of energy of a given denomination. In this case, we need two demons, each carrying a bit of energy: bit $j \in \{1, 2\}$ denoting energy $\epsilon_j = j$. The two kinds of demons can be thought of as a pair of lattice gases which may be coupled with an arbitrary, reversible, energy-conserving dynamics in order to stir the heat bath. The possible energies of a cell are then $\varepsilon_0 = 0$, $\varepsilon_1 = \epsilon_1$, $\varepsilon_2 = \epsilon_2$, and $\varepsilon_3 = \epsilon_1 + \epsilon_2$ as required. Let $R_j = N\rho_j$ be the total number of j's bits, where ρ_j is the density (read probability) of j's bits. Then the probabilities are related by

$$\begin{aligned} \rho_1 &= p_1 + p_3 = (e^{-\beta} + e^{-3\beta}) \left(\frac{1 - e^{-\beta}}{1 - e^{-4\beta}} \right) \\ &= \frac{1}{1 + e^{\beta}} = \frac{1}{1 + e^{\beta\epsilon_1}}, \end{aligned} \tag{A.7}$$

and

$$\begin{aligned} \rho_2 &= p_2 + p_3 = (e^{-2\beta} + e^{-3\beta}) \left(\frac{1 - e^{-\beta}}{1 - e^{-4\beta}} \right) \\ &= \frac{1}{1 + e^{2\beta}} = \frac{1}{1 + e^{\beta\epsilon_2}}, \end{aligned} \tag{A.8}$$

which are just Fermi distributions for the occupation number of the respective energy levels. Also note that $0 \leq \rho_j < 1/2$ for $0 \leq T < \infty$.

A.1.2 Alternate Derivation

The counting of equilibrium states can also be done directly in terms of the numbers of demons in order to illustrate the effect of particle conservation on occupation numbers. Let R_j ($j \in \{1, 2\}$) be the number of demons of type j, and introduce the constraints

$$R = \sum_j R_j \quad \text{and} \quad E = \sum_j R_j \epsilon_j. \tag{A.9}$$

Now we want to maximize the number of ways, W, of putting R_j indistinguishable particles in the N j's bits subject to the constraints, where

$$W = w_1 w_2 \quad \text{and} \quad w_j = \binom{N}{R_j}. \tag{A.10}$$

As before, this can be accomplished by extremizing an auxiliary function

$$\begin{aligned} g(\{R_j\}, \alpha, \beta) &= \ln W + \alpha(R - \sum_j R_j) + \beta(E - \sum_j R_j \epsilon_j) \\ &= 2N \ln N - 2N - \sum_j R_j \ln R_j + \sum_j R_j + \sum_j (N - R_j) \\ &\quad - \sum_j (N - R_j) \ln(N - R_j) + \alpha(\cdots) + \beta(\cdots), \end{aligned} \tag{A.11}$$

which incorporates the constraints (A.9) and leads to

$$\begin{aligned} \frac{\partial g}{\partial R_j} &= -\ln R_j - 1 + \ln(N - R_j) + 1 - \alpha - \beta \epsilon_j = 0 \\ &\Rightarrow \frac{N - R_j}{R_j} = e^{\beta(\epsilon_j - \mu)}, \end{aligned} \tag{A.12}$$

where $\mu \equiv -\alpha/\beta$. Thus,

$$\rho_j = \frac{R_j}{N} = \frac{1}{1 + e^{\beta(\epsilon_j - \mu)}}, \tag{A.13}$$

and the chemical potential, $\mu = -T\frac{\partial g}{\partial R}$, gives a measure of how sensitive the maximum entropy configuration is to changes in the constraint on the total number of demons, R. In the actual heat bath, the number of demons is not fixed, so in fact, $\mu = 0$ as before. Similarly, if E were unconstrained, there would result $\beta = 0$ and $T = \infty$. Since the number of states of the system factors as in equation (A.10), the entropies of the two subsystems add and are maximized individually (i.e., there are no correlations between them in equilibrium). Therefore, the probabilities, ρ_j, of the two bits in a cell are independent, and we obtain (as before)

$$\begin{aligned}
p_0 &= (1-\rho_1)(1-\rho_2) = \left(1 - \frac{1}{1+e^{\beta\epsilon_1}}\right)\left(1 - \frac{1}{1+e^{\beta\epsilon_2}}\right) \\
&= \frac{e^{\beta(\epsilon_1+\epsilon_2)}}{1+e^{\beta\epsilon_1}+e^{\beta\epsilon_2}+e^{\beta(\epsilon_1+\epsilon_2)}} = \frac{e^{-\beta\varepsilon_0}}{1+e^{-\beta\varepsilon_1}+e^{-\beta\varepsilon_2}+e^{-\beta\varepsilon_3}} = \frac{1}{z}, \qquad (A.14) \\
p_1 &= \rho_1(1-\rho_2) = \frac{e^{-\beta\varepsilon_1}}{z}, \qquad (A.15) \\
p_2 &= \rho_2(1-\rho_1) = \frac{e^{-\beta\varepsilon_2}}{z}, \qquad (A.16) \\
p_3 &= \rho_1\rho_2 = \frac{e^{-\beta\varepsilon_3}}{z}. \qquad (A.17)
\end{aligned}$$

The heat bath was originally motivated here as a way of adding dissipation to Monte Carlo simulations within a reversible microcanonical framework. In other applications, the extra degrees of freedom help speed the evolution of system by making the dynamics more flexible [20, 89]. In both of these cases, the subsystems interact and affect each other in a cooperative fashion so as to remember the past of the coupled system.

However, such a heat bath can also be used as a purely external random number generator which is *not* affected by the dynamics of the primary system. In this case, causality only runs in the direction of the heat bath towards the main system, although under certain circumstances, the overall system can still be reversible. The energy distribution of a cell will settle down to yield the Boltzmann weights (A.14–A.17) given above, with all of the other cells acting as its heat bath. Any of the four probabilities of attaining a particular energy can then be used as an acceptance ratio in a more conventional Monte Carlo algorithm [62]. The cells will be independent from each other at any given time if they were in maximum entropy configuration initially (since any correlations would reduce the entropy while it must increase or stay the same [87]). Of course the random variables in the cells will not be totally independent from one time step to the next, but they will be adequate for many applications.

A.2 Measuring and Setting the Temperature

In addition to being useful for implementing dynamical effects such as dissipation, a microcanonical heat bath provides a good way to measure and set the temperature of a CA system with which it is in contact. This is because the properties of a heat bath are easy to describe as a function of temperature, while those of the actual system of interest are not. Measuring the temperature requires fitting the experimentally derived probabilities to the formulas given above, and setting the temperature involves randomly distributing demons to give frequencies that match the probabilities.

CA machines have a built-in mechanism for counting the number of cells having a given characteristic, so it is makes sense to work from formulas for the total counts (or averages of counts) of cells of each energy:

$$N_i = N \left(\frac{1 - e^{-\beta}}{1 - e^{-4\beta}} \right) e^{-\beta i} = N p_i \quad \text{for } i = 0,\, 1,\, 2,\, 3, \tag{A.18}$$

and the total counts of demons:

$$R_j = \frac{N}{1 + e^{\beta j}} = N \rho_j \quad \text{for } j = 1,\, 2. \tag{A.19}$$

Numerous formulas for the temperature ($T = 1/\beta$) can be derived from (A.18) and (A.19), and several of the simplest ones are given below. All of them would give the same answer in the thermodynamic limit, but since the system is finite, fluctuations show up in the experimental counts. Thus, each derived formula will, in general, give a slightly different answer, but only three can be independent due to the constraining relationships among the N_i's and R_j's. Taking the ratios of the counts in (A.18) leads to:

$$T_{ij} = \frac{j-i}{\ln N_i - \ln N_j}, \tag{A.20}$$

and solving directly for T in (A.19) gives:

$$T_j = \frac{j}{\ln\left(\frac{N}{R_j} - 1\right)}. \tag{A.21}$$

Finally, taking the ratios of the counts in (A.19) gives:

$$r \equiv \frac{\rho_1}{\rho_2} = \frac{R_1}{R_2} = \frac{1+e^{2\beta}}{1+e^{\beta}},$$

which becomes

$$\begin{aligned} \left(e^{\beta}\right)^2 - r\left(e^{\beta}\right) + 1 - r &= 0 \\ \Rightarrow e^{\beta} &= \frac{1}{2}(r \pm \sqrt{r^2+4r-4}), \end{aligned}$$

and for positive temperatures,

$$T_r = \frac{1}{\ln\left[\frac{1}{2}(r+\sqrt{r^2+4r-4})\right]}. \tag{A.22}$$

The measured temperature can be estimated by any kind of weighted mean of the nine quantities above. For example, discarding the high and low three and averaging the middle three would give the smoothness of an average while guarding against contributions from large fluctuations. T_1 would probably give the best single measure because it only depends on R_1 which is large and will have the smallest relative error ($\sim 1/\sqrt{R_1}$). Alternative methods for extracting the temperature are suggested elsewhere [19, 20, 73]. Note in the formulas above that a population inversion in the N_i,s or the R_j's will lead to a negative value for T. This can happen in any system which has an upper bound on the amount of energy it can hold. A heat bath with a negative temperature is sometimes described as being "hotter that infinity" because it will spontaneously give up energy to any other system having an arbitrarily high (positive) temperature.

The temperature of the heat bath can be set by randomly initializing demons 1 and 2 with probabilities ρ_1 and ρ_2 as given by equations (A.7) and (A.8) respectively. Note that since μ is always zero in equilibrium, not all values of ρ_1 and ρ_2 are allowed. However, another way to set the temperature would be to initialize the heat bath with an amount of energy equal to the expected value corresponding to a given temperature and then allow the bath to equilibrate. By alternately measuring and setting the temperature, one can effectively implement a thermostat which can follow any given temperature schedule. This would be useful for annealing a system, varying reaction rates, or mapping out a phase diagram.

A.3 Additional Thermodynamics

This section demonstrates some important relationships between thermodynamic functions and brings up the issue of macroscopic work in discrete systems. Let $Z = \sum_s e^{-\beta E_s}$ where the sum is over *all* states of the *entire* heat bath:

$$\begin{aligned} Z(\beta, N) &= \sum_E \Omega(E) e^{-\beta E} \quad \text{where} \quad E = E(\{N_i\}) \\ &= \sum_{\{N_i\}} \Omega(\{N_i\}) e^{-\beta \sum_i N_i \varepsilon_i} = \sum_{\{N_i\}} \binom{N}{N_0 N_1 N_2 N_3} \prod_i \left(e^{-\beta \varepsilon_i}\right)^{N_i} \\ &= \left(\sum_i e^{-\beta \varepsilon_i} \right)^N = z^N. \end{aligned} \tag{A.23}$$

Note that there is no $1/N!$ in the final expression because all the cells are distinct. Now

$$\begin{aligned} S(\langle E \rangle, N) &= \ln \Omega = N \ln N - N - \sum_i (N_i \ln N_i - N_i) \\ &\quad \left(= -N \sum_i p_i \ln p_i \right) \\ &= N \ln N - \sum_i \left(\frac{N e^{-\beta \varepsilon_i}}{z} \ln \frac{N e^{-\beta \varepsilon_i}}{z} \right) \end{aligned}$$

$$
\begin{aligned}
&= N \ln N - \left(\frac{N}{z} \ln \frac{N}{z}\right)\left(\sum_i e^{-\beta\varepsilon_i}\right) + \beta \sum_i \frac{N e^{-\beta\varepsilon_i}}{z}\varepsilon_i \\
&= \ln z^N + \beta\langle E\rangle = \ln Z + \beta\langle E\rangle \qquad \text{(A.24)}
\end{aligned}
$$

where β is implicitly defined in terms of $\langle E\rangle$. Then we can see that $\beta = \frac{\partial S}{\partial\langle E\rangle}$. Also define

$$F(T, N) = \langle E\rangle - TS = -T(S - \beta\langle E\rangle) = -T \ln Z \qquad \text{(A.25)}$$

which is the (Helmholtz) free energy. The decrease in the free energy is a measure of the maximum amount of *work* that can be extracted from a system held at constant T by varying an external parameter such as N (read V, the volume). Work done by a thermodynamic system serves to convert the disordered internal energy into an ordered (i.e., low entropy) form. Examples of ordered energy from classical physics include kinetic and potential energy of macroscopic objects as well as the energy stored in static electric and magnetic fields. However in the case of CA, it is not clear how one would go about changing the volume or even what it would mean to convert the random internal energy into an ordered form. This, in my view, represents a fundamental obstacle in the application of CA to physical modeling.

The final relationship given here establishes the connection between the physical and information-theoretic aspects of the heat bath, namely $I = S/\ln 2$. This quantity is the amount of information (i.e., the number of bits) needed to specify a particular configuration out of the equilibrium ensemble of the heat bath (given $\langle E\rangle$ and N). Sometimes information is taken to be how much knowledge one has of the particular configuration of a system in which case it defined to be the negative of the quantity above. In other words, the higher the entropy of a system, the less detail one knows about it.

Appendix B

Broken Ergodicity and Finite Size Effects

This appendix gives some additional details behind the calculation of the entropy of the potential well system described in chapter 3. The primary correction to the elementary analysis given in section 3.2 comes from the presence of a family of conserved currents which limits the number of particles that can fall into the well. These currents are nonzero to begin with due to fluctuations in the random initial conditions. Another correction reflects the finite diameter of the system and comes from the way the the problem is broken up to calculate the effect of the conserved currents. These corrections are of comparable magnitude even though they have somewhat different origins, and both would disappear in the thermodynamic limit. The existence of additional conservation laws must always serve to lower the total entropy of the system because the number of accessible states is reduced. However, additive constants can be ignored because we are ultimately interested in how the derivatives of the extra terms shift the equilibrium densities.

B.1 Fluctuations and Initial Conserved Currents

The first step is to understand the statistics of the currents on the diagonal lines shown in figure 3-9. Half of the cells on each line are devoted to particles moving in either direction, and the currents arise because the two halves will not generally be equally populated. Let $N = 256$ be the number of cells on each line, and let $L = 74$ be the total number of lines. The total number of cells in region B is then $N_B = NL = 18944$. Call the number of particles moving with and against the current, n_+ and n_- respectively—each will follow a binomial distribution where p is the probability of any given cell being occupied:

$$p(n_\pm) = \binom{N/2}{n_\pm} p^{n_\pm}(1-p)^{N/2-n_\pm}. \tag{B.1}$$

The mean and variance of the number of particles flowing in either direction are then

$$\overline{n}_\pm = \tfrac{1}{2}Np \quad \text{and} \quad \sigma_\pm^2 = \tfrac{1}{2}Np(1-p). \tag{B.2}$$

The current on any given line is a random variable valued function of these two random variables, $j = n_+ - n_-$. Since n_+ and n_- are independent and identical,

$$\begin{aligned} \langle j^2 \rangle &= \langle (n_+ - n_-)^2 \rangle = \langle n_+^2 - 2n_+n_- + n_-^2 \rangle \\ &= \sigma_+^2 + \overline{n}_+^2 - 2\overline{n}_+\overline{n}_- + \sigma_-^2 + \overline{n}_-^2 = 2\sigma_\pm^2 = Np(1-p) \\ &= N\rho_0\overline{\rho}_0, \end{aligned} \tag{B.3}$$

where ρ_0 is the initial density of particles used in the simulation. For the particular initial condition used here, $\rho_0 = 1/4$ which gives $\langle j^2 \rangle = 48$. For comparison, explicit counting of the number of particles on each line shown in figure 3-9 gives a true mean value of $\langle j^2 \rangle = 3448/74 \cong 46.59$.

The expected absolute value of the current is difficult to calculate in closed form, but a numerical summation gives $\langle |j| \rangle = 5.5188\ldots$, while a Gaussian approximation gives $\langle |j| \rangle \cong 2\sigma_\pm/\sqrt{\pi} = 5.5279\ldots$. The true value of $\langle |j| \rangle$ in this experiment is $394/74 \cong 5.3243\ldots$. Note that the low values of $\langle j^2 \rangle$ and $\langle |j| \rangle$ are consistent with the fact that n_T and E_T are also less than expected, although the overall fluctuation is within the typical range:

$$\frac{N_T}{4} - n_T = 16384 - 16191 = 193 = 0 \pm 256 = 0 \pm 2\sqrt{\frac{N_T}{4}}. \tag{B.4}$$

B.2 Corrections to the Entropy

This section recalculates the entropy of the potential well system taking into account the the presence of the conserved currents while being careful to include all corrections of a comparable size. The approach taken here is to build the extra conservation laws into the counting of the states while leaving the constraints of particle and energy conservation out until the entropy maximization step. This is done to maintain a close similarity with the basic analysis, but it also turns out to be necessary because the $L = 74$ subsystems which contain the currents are so small that their entropies do not simply add. Significant fluctuations between these subsystems results in entropy from shared information about the total number of particles as will be explained next.

Each of the four regions A–D is large enough that the number of particles it contains can be treated as a constant, and the total number of states factors into the product of the number of states in each subsystem separately (or to put it another way, their entropies add):

$$\Omega = \binom{N_A}{n_A}\widetilde{\binom{N_B}{n_B}}\binom{N_C}{n_C}\binom{N_D}{n_D}. \tag{B.5}$$

The tilde over the number of states in region B indicates that the expression is only approximate, and that the combination is just being used as a symbolic reminder of what we are really trying to count.

Now let us turn to counting the actual number of states in region B assuming that there are $n_B = nL$ particles and $N_B = NL$ cells. To do this to the required accuracy, we will keep one more factor in Stirling's approximation than is customary for small numbers [1]:

$$\ln \mathcal{N}! \cong \begin{cases} \mathcal{N}\ln\mathcal{N} - \mathcal{N} + \frac{1}{2}\ln\mathcal{N}, & \mathcal{N} \leq N \\ \mathcal{N}\ln\mathcal{N} - \mathcal{N}, & \mathcal{N} \gg N. \end{cases} \tag{B.6}$$

Suppose for a moment that we can ignore the constraints imposed by the conservation laws. Then the entropy of region B would be exactly

$$S_B = \ln\binom{N_B}{n_B} = \ln\binom{NL}{nL} \cong L\left[N\ln N - n\ln n - (N-n)\ln(N-n)\right]. \tag{B.7}$$

However, if attempted to factor the number of states into $L \gg 1$ identical pieces before computing the entropy, we would obtain

$$\ln\binom{N}{n}^L \cong L\left[N\ln N - n\ln n - (N-n)\ln(N-n) + \frac{1}{2}\ln\frac{N}{n(N-n)}\right]. \tag{B.8}$$

The fact that these two expressions are not the same shows that the entropy is not quite extensive, though the difference becomes negligible in the thermodynamic limit. In order to factor the number of states for finite N we can write

$$\binom{N_B}{n_B} = \binom{NL}{nL} \cong \binom{N}{n}^L \exp\left\{\frac{L}{2}\ln\frac{n(N-n)}{N}\right\}. \tag{B.9}$$

The extra factor accounts for the fact that the particles can fluctuate between the lines. Therefore, even if the conserved currents are present, we make the approximation

$$\widetilde{\binom{N_B}{n_B}} = \widetilde{\binom{NL}{nL}} \cong \widetilde{\binom{N}{n}}^L \exp\left\{\frac{L}{2}\ln\frac{n(N-n)}{N}\right\}. \tag{B.10}$$

The extra factor contains the shared information, and it could be ignored in the thermodynamic limit.

Now we want to count the number of states on a line containing n particles and a current j. The particles in each line are divided among the two directions so that $n = n_+ + n_-$, and

$$n_+ = \frac{n+j}{2} \quad\text{and}\quad n_- = \frac{n-j}{2}. \tag{B.11}$$

The total number of states on a line is therefore

$$\binom{\widetilde{N}}{n} = \binom{N/2}{n_+}\binom{N/2}{n_-} = \binom{N/2}{\frac{n+j}{2}}\binom{N/2}{\frac{n-j}{2}}, \tag{B.12}$$

and the entropy due to a single line is

$$\begin{aligned}
\ln\binom{\widetilde{N}}{n} &= \ln\binom{N/2}{\frac{n+j}{2}}\binom{N/2}{\frac{n-j}{2}} \cong N\ln N \\
&\quad -\left(\frac{n+j}{2}\right)\ln n\left(1+\frac{j}{n}\right) - \left(\frac{N-n-j}{2}\right)\ln(N-n)\left(1-\frac{j}{N-n}\right) \\
&\quad -\left(\frac{n-j}{2}\right)\ln n\left(1-\frac{j}{n}\right) - \left(\frac{N-n+j}{2}\right)\ln(N-n)\left(1+\frac{j}{N-n}\right) \\
&\quad +\frac{1}{2}\ln\frac{2N}{(n+j)(N-n-j)} + \frac{1}{2}\ln\frac{2N}{(n-j)(N-n+j)} \\
&\cong N\ln N - n\ln n - (N-n)\ln(N-n) \\
&\quad -\left(\frac{n+j}{2}\right)\left[\frac{j}{n}-\frac{j^2}{2n^2}\right] - \left(\frac{N-n-j}{2}\right)\left[-\frac{j}{N-n}-\frac{j^2}{2(N-n)^2}\right] \\
&\quad -\left(\frac{n-j}{2}\right)\left[-\frac{j}{n}-\frac{j^2}{2n^2}\right] - \left(\frac{N-n+j}{2}\right)\left[\frac{j}{N-n}-\frac{j^2}{2(N-n)^2}\right] \\
&\quad -\ln\frac{n(N-n)}{2N} - \frac{1}{2}\ln\left[1-\frac{j^2}{n^2}-\frac{j^2}{(N-n)^2}+\frac{j^4}{n^2(N-n)^2}\right] \\
&= N(-\rho\ln\rho - \overline{\rho}\ln\overline{\rho}) - \frac{j^2}{2N}\left(\frac{1}{\rho}+\frac{1}{\overline{\rho}}\right) - \ln\rho\overline{\rho} - \ln\frac{N}{2} + \mathcal{O}\left(\frac{j^2}{N^2}\right),
\end{aligned} \tag{B.13}$$

where $\rho = n/N$.

The above expression can be used to obtain the contribution to the entropy for all of region B by averaging over the currents:

$$\begin{aligned}
S_B &= \ln\binom{\widetilde{N_B}}{n_B} \cong L\ln\binom{\widetilde{N}}{n} + \frac{L}{2}\ln\frac{n(N-n)}{N} \\
&\cong LN(-\rho_B\ln\rho_B - \overline{\rho}_B\ln\overline{\rho}_B) - L\frac{\langle j^2\rangle}{2N}\left(\frac{1}{\rho_B}+\frac{1}{\overline{\rho}_B}\right) \\
&\quad - L\ln\rho_B\overline{\rho}_B - \ln\frac{N}{2} + \frac{L}{2}\ln\frac{n(N-n)}{N}.
\end{aligned} \tag{B.14}$$

Using $\langle j^2 \rangle = N\rho_0\overline{\rho}_0$ and dropping constants gives

$$S_B \cong N_B(-\rho_B \ln \rho_B - \overline{\rho}_B \ln \overline{\rho}_B) - \frac{L\rho_0\overline{\rho}_0}{2\rho_B\overline{\rho}_B} - \frac{L}{2} \ln \rho_B\overline{\rho}_B. \tag{B.15}$$

This expression consists of a bulk entropy term, a correction for the extra conservation laws, and a correction for finite size.

B.3 Statistics of the Coupled System

The counting of regions A, C, and D requires no special treatment, so the entropy for the whole system is

$$\begin{aligned} S &= \ln \Omega \\ &\cong N_A(-\rho_A \ln \rho_A - \overline{\rho}_A \ln \overline{\rho}_A) + N_B(-\rho_B \ln \rho_B - \overline{\rho}_B \ln \overline{\rho}_B) \\ &\quad + N_C(-\rho_C \ln \rho_C - \overline{\rho}_C \ln \overline{\rho}_C) + N_D(-\rho_D \ln \rho_D - \overline{\rho}_D \ln \overline{\rho}_D) \\ &\quad - \frac{L\rho_0\overline{\rho}_0}{2\rho_B\overline{\rho}_B} - \frac{L}{2} \ln \rho_B\overline{\rho}_B, \end{aligned} \tag{B.16}$$

while the revised constraints on the particle number and energy are

$$n_T = n_A + n_B + n_C \quad \text{and} \quad E_T = n_A + n_B + n_D. \tag{B.17}$$

The entropy of the equilibrium ensemble will assume the maximum value subject to the constraints (B.17). To find this extremum, introduce Lagrange multipliers α and β, and define the auxiliary function

$$f = S + \alpha(n_T - N_A\rho_A - N_B\rho_B - N_C\rho_C) + \beta(E_T - N_A\rho_A - N_B\rho_B - N_D\rho_D). \tag{B.18}$$

Extremizing f with respect to α, β, ρ_A, ρ_B, ρ_C, and ρ_D returns the constraint equations (B.17) along with

$$-N_A \ln \frac{\rho_A}{\overline{\rho}_A} - \alpha N_A - \beta N_A = 0, \tag{B.19}$$

x	T	A	B	C	D
N_x	65536	33952	18944	12640	65536
n_x (initial)	16191	13063		3128	0
n_x (basic theory)	16191	5003.79	2791.94	8395.27	5267.27
n_x (nonergodic)	16191	4984.44	2819.39	8387.17	5259.17
n_x (finite size)	16191	5017.09	2773.09	8400.82	5272.82
n_x (revised theory)	16191	4997.37	2801.05	8392.58	5264.58

Table B.1: Theoretical values for the expected number of particles in each region of the system. The last three lines show the effects of broken ergodicity and finite size, separately and together, as calculated by including their respective correction terms in the entropy.

$$-N_B \ln \frac{\rho_B}{\overline{\rho}_B} + \frac{L\rho_0\overline{\rho}_0}{2} \frac{1-2\rho_B}{(\rho_B\overline{\rho}_B)^2} - \frac{L}{2}\frac{(1-2\rho_B)}{\rho_B\overline{\rho}_B} - \alpha N_B - \beta N_B = 0, \quad \text{(B.20)}$$

$$-N_C \ln \frac{\rho_C}{\overline{\rho}_C} - \alpha N_C = 0, \quad \text{(B.21)}$$

$$-N_D \ln \frac{\rho_D}{\overline{\rho}_D} - \beta N_D = 0. \quad \text{(B.22)}$$

Solving for the densities gives

$$\rho_A = \frac{1}{1+e^{\beta(1-\mu)}}, \quad \text{(B.23)}$$

$$\rho_B = \frac{1}{1+e^{\beta(1-\mu)}\exp\left\{\frac{1}{2N}\frac{(1-2\rho_B)}{\rho_B\overline{\rho}_B}\left(1-\frac{\rho_0\overline{\rho}_0}{\rho_B\overline{\rho}_B}\right)\right\}}, \quad \text{(B.24)}$$

$$\rho_C = \frac{1}{1+e^{\beta(-\mu)}}, \quad \text{(B.25)}$$

$$\rho_D = \frac{1}{1+e^{\beta}}. \quad \text{(B.26)}$$

The above equations can be solved numerically, and it is illuminating to do so with and without each correction term in the entropy. Table B.1 gives the results. Note that the conservation term drives the number of particles in region B up, while the finite size term drives the number down.

Appendix C

Canonical Stochastic Weights

This appendix describes a frugal technique for utilizing random bits in Monte Carlo simulations. In particular, it defines a canonical set of binary random variables along with a recipe for combining them to create new binary random variables. By using just a few well-chosen random bits in the right combination, it is possible to fine tune the probability of any random binary decision. The technique is especially useful in simulations which require a large number of differentially biased random bits, while only a limited number of fixed sources of randomness are available (as in a cellular automata machine).

C.1 Overview

Random bits are often used as digital coins to decide which of two branches a computation is to take. However, in many physical situations, one may want one branch to be more probable than the other. Furthermore, it may be necessary to choose the probability depending on the current state of the system. For example, in the Metropolis algorithm for sampling a canonical ensemble [62], one accepts a trial move with a probability of $\min(1, e^{-\Delta E/T})$, where ΔE is the change in energy and T is the temperature in energy units. This acceptance probability can be anywhere from 0 to 1 depending on the current state, the chosen trial move, and the temperature. Hence, it is desirable to be able to switch coins on the fly.

The simplest way to make a biased decision is to flip a suitably unfair coin a single time. This is effectively what is done in CA simulations when using a lattice gas as a random number generator where the density of the gas is equal to the desired probability. However, this only gives us one level of randomness at a time, and changing it is slow because it requires initializing the gas to a different density.

An obvious way to synthesize a biased coin flip out of N fair coins flips is to combine them into an N-bit binary number and compare the number to a preset threshold. This method can discriminate between $2^N + 1$ probabilities ranging uniformly from 0 to 1. Furthermore, by combining bits in this way, it is possible to tune the probabilities without changing the source of randomness.

However, it is possible to do far better than either of the above two strategies. By combining and extending the strategies, it is possible to obtain 2^{2^N} probabilities ranging uniformly from 0 to 1. The technique can be subsequently generalized to multiple branches.

C.2 Functions of Boolean Random Variables

In what follows, the term, "the probability of b," will be used to mean the probability p that the binary random variable b takes the value 1 (the other possibility being 0). Thus, the expectation (or weight) of b is simply p. The vector $\mathbf{b}$ will denote a single N-bit binary number whose components are its binary digits: $\mathbf{b} = \sum_{n=0}^{N-1} b_n 2^n$. Let $\overline{p} = 1 - p$ be the universal complement of the probability p, $\overline{b} = \neg b$ the logical complement of the bit b, and $\overline{\mathbf{b}} = 2^N - 1 - \mathbf{b}$ the one's complement of $\mathbf{b}$.

C.2.1 The Case of Arbitrary Weights

There are 2^{2^N} Boolean functions $f(\mathbf{b})$ of N Boolean (or binary) variables $\mathbf{b}$.[1] Any of these functions can be written in the following sum of products format:

$$\begin{aligned} f(\mathbf{b}) &= \bigvee_{\mathbf{a}=0}^{2^N-1} f(\mathbf{a})\delta_{\mathbf{ab}} \\ &= \bigvee_{\mathbf{a}=0}^{2^N-1} f(\mathbf{a}) \bigwedge_{n=0}^{N-1} b_n^{a_n}\overline{b}_n^{\overline{a}_n} \\ &= f(0)\overline{b}_{N-1}\overline{b}_{N-2}\cdots\overline{b}_1\overline{b}_0 \\ &\quad + f(1)\overline{b}_{N-1}\overline{b}_{N-2}\cdots\overline{b}_1 b_0 \\ &\quad \vdots \\ &\quad + f(2^N-1)b_{N-1}b_{N-2}\cdots b_1 b_0, \end{aligned} \tag{C.1}$$

where I have adopted the convention that $0^0 = 1$. For each $\mathbf{a}$, $\delta_{\mathbf{ab}}$ consists of a product of N b's which singles out a specific configuration of the inputs and thereby corresponds to one row in a lookup table for f. Hence, the above column of $f(\mathbf{b})$'s ($0 \le \mathbf{b} \le 2^N - 1$) consists of the entries of the lookup table for this function. The lookup table considered as a single binary number will be denoted by $\mathbf{f} = \sum_{\mathbf{b}=0}^{2^N-1} f(\mathbf{b})2^{\overline{\mathbf{b}}}$ (m.s.b.$=f(0)$, l.s.b.$=f(2^N-1)$). This number can be used as an index into the set of all Boolean functions of N bits:

$$\mathcal{F}_N = \{f_{\mathbf{f}}(\mathbf{b})\}_{\mathbf{f}=0}^{2^{2^N}-1}. \tag{C.2}$$

A given probability distribution on the N arguments will induce a probability on each of these functions. Suppose the inputs are independent, and let the weight of bit n be $p(b_n) = p_n$. Then the probability of a particular $\mathbf{b}$ is

$$P(\mathbf{b}) = \prod_{n=0}^{N-1} p_n^{b_n}\overline{p}_n^{\overline{b}_n}. \tag{C.3}$$

[1]The argument list $\mathbf{b}$ will often be referred to as the "inputs," since they will ultimately be fed into a lookup table (which happens to be a memory chip in a CA machine).

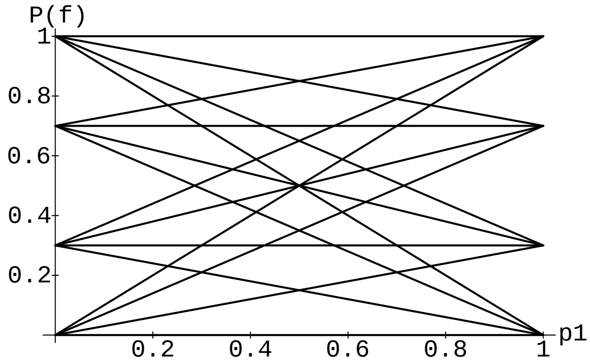


Figure C-1: Probabilities of the 16 functions for $N = 2$ showing their dependence on p_1 when $p_0 = .3$ is fixed. When $p_0 = \frac{1}{3}$ and $p_1 = \frac{1}{5}$, the $\mathcal{P}(f)$'s become equally spaced.

Occurrences of different values of $\mathbf{b}$ are disjoint events; therefore, the probability of a given f can be written as the following sum:

$$\begin{aligned}
\mathcal{P}(f) &= \sum_{\mathbf{b}=0}^{2^N-1} f(\mathbf{b})P(\mathbf{b}), \\
&= \sum_{\mathbf{b}=0}^{2^N-1} f(\mathbf{b}) \prod_{n=0}^{N-1} p_n^{b_n} \overline{p}_n^{\overline{b}_n} \\
&= f(0)\overline{p}_{N-1}\overline{p}_{N-2}\cdots\overline{p}_1\overline{p}_0 \\
&\quad + f(1)\overline{p}_{N-1}\overline{p}_{N-2}\cdots\overline{p}_1 p_0 \\
&\quad \vdots \\
&\quad + f(2^N-1)p_{N-1}p_{N-2}\cdots p_1 p_0.
\end{aligned} \tag{C.4}$$

For $N = 2$, there are $2^{2^N} = 16$ distinct f's. Figure C-1 shows the complex way in which the probabilities of the f's vary as a function of p_1 alone. This "cat's cradle" shows that it is hard to get an intuitive grasp of how the set $\{\mathcal{P}(f)\}$ as a whole behaves as a function of the input probabilities. The numerous crossings even make the order of the f's unclear.

C.2.2 The Canonical Weights

We have a set of N free parameters $\{p_n\}$ and would like to be able to specify any distribution of probabilities for the 2^{2^N} f's. Perhaps the most useful distribution one could obtain would be a uniform one. Though we are doubly exponentially far from being able to specify an arbitrary distribution, we can, in fact, obtain this most favorable case! For example, for $N = 2$, one can generate the set of probabilities

$$\mathcal{P}_2 = \left\{0, \frac{1}{15}, \frac{2}{15}, \frac{3}{15}, \frac{4}{15}, \frac{5}{15}, \frac{6}{15}, \frac{7}{15}, \frac{8}{15}, \frac{9}{15}, \frac{10}{15}, \frac{11}{15}, \frac{12}{15}, \frac{13}{15}, \frac{14}{15}, 1\right\} \tag{C.5}$$

by taking all possible Boolean functions of two Boolean random variables $\{b_0, b_1\}$ having the weights $\mathcal{W}_2 = \{\frac{1}{3}, \frac{1}{5}\}$ (cf. figure C-1). The functions which give $\mathcal{P}_2$ are respectively

$$\mathcal{F}_2 = \left\{0, \wedge, <, b_1, >, b_0, \not\equiv, \vee, \not\vee, \equiv, \overline{b}_0, \leq, \overline{b}_1, \geq, \not\wedge, 1\right\}, \tag{C.6}$$

where the binary operators are understood to mean b_0 op b_1.

More generally, for N bits, we want to choose the set of weights $\{p_n\}$ so as to generate the set

$$\{\mathcal{P}(f_{\mathbf{f}}(\mathbf{b}))\}_{\mathbf{f}=0}^{2^{2^N}-1} = \mathcal{P}_N \equiv \left\{0, \frac{1}{2^{2^N}-1}, \frac{2}{2^{2^N}-1}, \ldots, \frac{2^{2^N}-2}{2^{2^N}-1}, 1\right\}. \tag{C.7}$$

This can be done by choosing the weights of the inputs to be reciprocals of the first N Fermat numbers, $F_n = 2^{2^n} + 1$:

$$\{p_n\}_{n=0}^{N-1} = \mathcal{W}_N \equiv \left\{\frac{1}{F_n}\right\}_{n=0}^{N-1} = \left\{\frac{1}{3}, \frac{1}{5}, \frac{1}{17}, \ldots, \frac{1}{2^{2^{N-1}}+1}\right\}. \tag{C.8}$$

That this choice is unique up to a relabeling of the probabilities and their complements is proved in the next section.

The above choice of probabilities also gives the answer to the following question. Suppose you want to make two rectangular cuts through a square of unit mass to make four separate weights. There are $2^4 = 16$ ways of combining the resulting weights to make composite masses ranging from 0 to 1. The question is then, how do you make the cuts in order to obtain the largest uniform set of composite masses? Clearly, the four weights should be

$$\left\{\frac{1}{15}, \frac{2}{15}, \frac{4}{15}, \frac{8}{15}\right\}. \tag{C.9}$$

This can be accomplished by cutting the square one third of the way across in one direction and one fifth of the way across in the other direction. The fractions of the square on either side of the cuts correspond to the probabilities and their complements, but in the latter case, the weights are stochastic. For cubes and hypercubes, additional cuts are made according to subsequent Fermat numbers.[2]

Substituting the weights $\mathcal{W}_N$ in equation (C.3) gives

$$P(\mathbf{b}) = \prod_{n=0}^{N-1} \left(\frac{1}{F_n}\right)^{b_n} \left(\frac{F_n - 1}{F_n}\right)^{\overline{b}_n}. \tag{C.10}$$

Now either b_n or $\overline{b}_n$ is 1, and the other must then be 0; therefore,

$$P(\mathbf{b}) = \left(\prod_{n=0}^{N-1} \frac{1}{F_n}\right) \left(\prod_{n=0}^{N-1} (F_n - 1)^{\overline{b}_n}\right). \tag{C.11}$$

It is easy to show that

$$\prod_{n=0}^{N-1} F_n = 2^{2^N} - 1, \tag{C.12}$$

so we obtain

$$\begin{aligned} P(\mathbf{b}) &= \frac{1}{2^{2^N} - 1} \prod_{n=0}^{N-1} 2^{\overline{b}_n 2^n} \\ &= \frac{1}{2^{2^N} - 1} 2^{\sum_{n=0}^{N-1} \overline{b}_n 2^n} \\ &= \frac{2^{\overline{\mathbf{b}}}}{2^{2^N} - 1}. \end{aligned} \tag{C.13}$$

In this case, the expression for $\mathcal{P}(f)$ takes a simple form:

$$\mathcal{P}(f) = \frac{1}{2^{2^N} - 1} \sum_{\mathbf{b}=0}^{2^N - 1} f(\mathbf{b}) 2^{\overline{\mathbf{b}}}$$

[2] If putting masses on either side of the scale is allowed, it is even better to cut the unit mass according to $3^{2^n} + 1$.

$\mathbf{b}$	b_1	b_0	$f(\mathbf{b})$	$P(\mathbf{b})$
0	0	0	1	$\overline{p}_1\overline{p}_0 = 8/15$
1	0	1	0	$\overline{p}_1 p_0 = 4/15$
2	1	0	1	$p_1\overline{p}_0 = 2/15$
3	1	1	1	$p_1 p_0 = 1/15$

Table C.1: Construction of an $f(\mathbf{b})$ with $N = 2$ inputs, b_0 and b_1, which gives a net probability of $\frac{11}{15}$.

$$\begin{aligned} &= \frac{\mathbf{f}}{2^{2^N} - 1} \\ &= \frac{f(0)f(1)\cdots f(2^N-1)_2}{2^{2^N}-1}, \end{aligned} \tag{C.14}$$

where the numerator of the last expression is not a product but a binary number, with each $f(\mathbf{b})$ $(0 \le \mathbf{b} \le 2^N - 1)$ determining one of the bits.[3]

In order to choose the appropriate f with the desired $\mathcal{P}(f)$, approximate $\mathcal{P}$ as a fraction, where the denominator is $2^{2^N} - 1$, and the numerator is a 2^N-bit binary number. Then simply fill in the lookup table for f with the bits of the numerator. For example, suppose we want an $N = 2$ function f having a probability near $\frac{11}{15}$, i.e.,

$$\mathcal{P}(f) = \frac{\mathbf{f}}{2^{2^2} - 1} = \frac{11}{15} = \frac{1011_2}{1111_2}. \tag{C.15}$$

This will be given by $f_{\mathbf{f}} = f_{11}$, and the lookup table and component probabilities for this function are listed in table C.1. This function can also be put into more conventional terms:

$$f_{11}(\mathbf{b}) = \overline{b}_1\overline{b}_0 + b_1\overline{b}_0 + b_1 b_0 = (b_0 \le b_1) = \neg b_0 \vee b_1, \tag{C.16}$$

and the total probability comes from

$$\mathcal{P}(f_{11}) = \overline{p}_1\overline{p}_0 + p_1\overline{p}_0 + p_1 p_0. \tag{C.17}$$

[3] N.B. The bits in the lookup table are reversed relative to what one might expect, viz., $f(0)$ is the most significant bit of $\mathbf{f}$ and $f(2^N - 1)$ is the least significant bit. Alternatively, the input bits or their probabilities could have been complemented.

C.2.3 Proof of Uniqueness

In the last section, we derived $\mathcal{P}_N$ from one particular choice of weights $\mathcal{W}_N = \{p(b_n)\}$. Now we will work backwards from $\mathcal{P}_N$ to construct the set $\mathcal{W}_N$, thus proving that the choice is unique.

In order for $\{P(\mathbf{b})\}$ to span $\mathcal{P}_N$ using only binary coefficients as in equation (C.4), we clearly must have

$$\{P(\mathbf{b})\}_{\mathbf{b}=0}^{2^N-1} = \left\{ \frac{1}{2^{2^N}-1}, \frac{2}{2^{2^N}-1}, \frac{4}{2^{2^N}-1}, \ldots, \frac{2^{2^N-1}}{2^{2^N}-1} \right\}. \tag{C.18}$$

Therefore, each $P(\mathbf{b})$ must be proportional to a distinct one of the first 2^N powers of two, with a constant of proportionality of $(2^{2^N}-1)^{-1}$.

Recall that each $P(\mathbf{b})$, as given in equation (C.3), is a product consisting of either p_n or $\overline{p}_n$ for every n. In order to get 2^N different products, none of the p_n's can be equal, nor can they be zero, $\frac{1}{2}$, or one. So without loss of generality, we can relabel the p_n's so that

$$0 < p_{N-1} < p_{N-2} < \cdots < p_1 < p_0 < \tfrac{1}{2}, \tag{C.19}$$

$$1 > \overline{p}_{N-1} > \overline{p}_{N-2} > \cdots > \overline{p}_1 > \overline{p}_0 > \tfrac{1}{2}. \tag{C.20}$$

Since the p's are all less than the $\overline{p}$'s, the smallest product is $P\left(\overline{0}\right)$, and it must be proportional to 1:

$$P\left(\overline{0}\right) = p_{N-1}p_{N-2}\cdots p_1 p_0 \propto 2^0. \tag{C.21}$$

The order of the p's also lets us conclude that the second smallest is $P\left(\overline{1}\right)$ which must be proportional to 2:

$$P\left(\overline{1}\right) = p_{N-1}p_{N-2}\cdots p_1 \overline{p}_0 \propto 2^1. \tag{C.22}$$

The ratio of these proportionalities is

$$\frac{P\left(\overline{1}\right)}{P\left(\overline{0}\right)} = \frac{\overline{p}_0}{p_0} = \frac{2^1}{2^0} = 2, \tag{C.23}$$

and solving gives $p_0 = \frac{1}{3}$. Similarly, the third smallest is $P\left(\overline{2}\right)$ which must be proportional to 4:

$$P\left(\overline{2}\right) = p_{N-1}p_{N-2}\cdots\overline{p}_1 p_0 \propto 2^2. \tag{C.24}$$

Then

$$\frac{P\left(\overline{2}\right)}{P\left(\overline{0}\right)} = \frac{\overline{p}_1}{p_1} = \frac{2^2}{2^0} = 4, \tag{C.25}$$

giving $p_1 = \frac{1}{5}$. This in turn generates

$$P\left(\overline{3}\right) = p_{N-1}p_{N-2}\cdots\overline{p}_1\overline{p}_0 \propto 2^3, \tag{C.26}$$

which is evidently the fourth smallest.

Once the lowest 2^n products are obtained, the next lowest $P(\mathbf{b})$ will introduce $\overline{p}_n$. This in turn generates the next 2^n products, thereby doubling the set of lowest products. By induction,

$$P\left(\overline{2^n}\right) = p_{N-1}\cdots p_{n+1}\overline{p}_n p_{n-1}\cdots p_1 p_0 \propto 2^{2^n}. \tag{C.27}$$

Dividing this by $P\left(\overline{0}\right)$ gives

$$\frac{P\left(\overline{2^n}\right)}{P\left(\overline{0}\right)} = \frac{\overline{p}_n}{p_n} = \frac{2^{2^n}}{2^0} = 2^{2^n}, \tag{C.28}$$

and finally,

$$p_n = \frac{1}{2^{2^n}+1}. \quad \square \tag{C.29}$$

C.3 Application in CAM-8

The technique described above is particularly well-suited to use in CAM-8, which is the most recent cellular automata machine to be developed by the Information Mechanics Group (cf. the description of CAM-6 in section 3.3.1).[4] At the risk of oversimplifying the situation, it is useful for present purposes to think of the state

[4]For a more detailed description of CAM-8, see [24], pp. 219–249.

space in CAM-8 as consisting of 16 parallel "planes" of bits with periodic boundary conditions. Each cell contains one bit from each of these planes, and data is shifted between cells by sliding the planes an arbitrary distance relative to the others. Each cell is updated between shifts by replacing the 16 bits of data in each cell with an arbitrary 16-bit function of the original data. For a stochastic rule, this typically means using a few (e.g., $N = 3$) of the 16 planes as a random number generator and the rest for the system itself.

Once and for all, each of the N bit-planes devoted to randomness is filled with its own density $p_n = 1/F_n$ of 1's (or "particles"). These planes should be separately "kicked" (i.e., shifted) by random amounts every time step to give each cell a new random sample. Thus, for each cell, the probability of finding a particle on one of the random planes will be given by its corresponding p_n. Since the same set of particles will be used over and over, one should set exactly $\left[\frac{4M}{F_n}\right]$ bits to one (assuming $4M$-bit planes), where $[\cdots]$ denotes the nearest integer. This is the only way to guarantee satisfaction of the law of large numbers.

The update of the system on the other $16 - N$ planes can now depend on any one of 2^{2^N} different coins. For $N = 3$, this translates into a choice of 256 uniformly distributed probabilities. For many applications, this tunability of $\mathcal{P}(f)$ is effectively continuous from 0 to 1. Furthermore, the choice of a coin in each cell can depend on the rest of the data in the cell—the densities in the random planes need never be changed. This would be useful, for example, in a physical simulation in which various reactions take place with a large range of branching ratios depending on which particle species are present.

C.4 Discussion and Extensions

This section briefly discusses some fine points which are characteristic of those which inevitably come up when dealing with computer-generated random numbers. It also suggests some problems for further research. Many of the problems concern the quality of the randomness obtained, while still others address practical considerations in the application of the technique. Finally, an algorithm for extending the idea to multiple branches is presented.

Computer-generated randomness will always contain some residual correlations which could conceivably throw off the results of Monte Carlo simulations. An important extension of this work, therefore, would be to investigate the correlations in the coin tosses and their effects on the reliability of CA experiments. In CAM-8 for example, the same spatial distribution of random bits will occur over and over again as each plane of randomness is merely shifted around. Could this cause a spurious "resonance" with the dynamics under investigation?

A mathematical problem along these lines would be to find a good "random" sequence of kicks which minimizes the degree of repetition in the spatial distribution of coin flips. Truly random kicks would give good results, but a deterministic, anti-correlated sequence could possibly be better. The optimal solution may resemble a "Knights Tour" or "8-Queens" configuration on a chess board.

One way to enhance the randomness would be to use extra bit planes to stir or modulate the particles. Stirring would require an "ergodic," particle conserving rule with short relaxation times—perhaps a lattice gas would do. Modulation could take the form of logical combinations of planes (such as 'and' or 'exclusive or'). On the other hand, one could save storage of randomness in the cell states by using an external random number generator as inputs to the processor (the lookup table). The problem would be to build a fast, hardwired source of probability $1/F_n$ random bits.

A practical problem is to find a good way to throw down, at random, exactly a given number of particles. A "physical" way to do this would be to put down the desired number of particles in a non-random way and then use the machine to stir the particles. Having a fixed number of particles in a plane introduces mild correlations, but they become negligible as the size of the system is increased.

Note that there are severe correlations between the various coins in a single cell. So while coins are virtually independent from cell to cell, one cannot, in general, use more than one at a time from a single cell and expect good results. By looking at several of these coins, one can never get more than I_N bits of randomness per cell, where

$$\begin{aligned}
I_N &= -\sum_{n=0}^{N-1}(p_n \log_2 p_n + \overline{p}_n \log_2 \overline{p}_n) \\
&= \log_2 \frac{\prod_{n=0}^{N-1}(2^{2^n}+1)}{\prod_{n=0}^{N-1}(2^{2^n})} + \sum_{n=0}^{N-1} \frac{2^n}{2^{2^n}+1} \\
&= \log_2 \frac{(2^{2^N}-1)}{2^{(2^N-1)}} + \frac{2^{2^N}-2^N-1}{2^{2^N}-1} \\
&= 2 - \frac{2^N}{2^{2^N}-1} - \log_2\left(\frac{2^{2^N}}{2^{2^N}-1}\right).
\end{aligned} \tag{C.30}$$

I_N approaches 2 bits in the limit $N \to \infty$. While this seems small, it would require exponentially more perfectly random bits to tune the derived probabilities to the same extent as given here.

The main drawback of the technique as presented so far is that it is limited to 2-way branches. However, I have extended the technique to an arbitrary number of branches, and the probability of each outcome is drawn from the largest possible set of probabilities ranging uniformly from 0 to 1. The resulting lookup tables take in random binary inputs whose weights are all reciprocals of integers and combine them to create "loaded dice" with any number of sides. The common denominators of the output probabilities grow more slowly as the number of branches increases, but eventually, they too grow as a double exponential. As before, this gives very fine control over the exact probabilities of the different branches.

The extended technique does not yield simple formulas for the lookup table and the input probabilities as in the binary case, so the solution is given in algorithmic form instead—see the subroutine listing at the end of this section. The subroutine can be used immediately to generate a suitable lookup table whenever appropriate constraints are satisfied. It will also work in the binary case, even if the denominators of the p_n's grow more slowly than F_n. The theory behind the algorithm will be the subject of a future paper.

A final problem along these lines is to understand how to fine tune the output probabilities for a particular application by modifying the input weights. The difficulty is that the output probabilities do not form a uniform set if the reciprocals of the input weights are not integers or if the integers grow too fast. While very good results can be obtained by using only rational probabilities, it may be worthwhile in certain circumstances to relax this constraint. Further software will probably be necessary to optimize the probabilities for each application.

```
/* This subroutine recursively loads a portion of a lookup table in order to
generate a single random variable as a function of a given set of binary random
variables.  The function takes on one of F values, f = 0, 1,..., F-2, F-1, and
the probability of each outcome can be varied by swapping lookup tables without
changing the given set of binary inputs.  Such tables yield an efficient
technique for implementing Markov processes in situations where the number of
sources of random bits is limited, while their fairness is adjustable (such as
in Cellular Automata Machines).  The special characteristics of the random
variables involved are described below.

The bits which form the input address to the lookup table are numbered n = 0,
1,..., N-2, N-1 (in order of increasing significance) and can be viewed as being
drawn from a set of N "bit planes."  The bits so taken are considered to be
independent binary random variables with probabilities p_n = 1/d_n, where each
d_n is an integer >= 2.  An array of these integer plane factors is given as
input to the subroutine in order to specify the input probabilities that will be
used.  The N low order bits in the lookup table index passed into the subroutine
must be set to zero.  The high order bits of the lutindex determine which
subtable will be loaded, and can also be used to represent the current state of
a Markov chain (i.e., the table can vary from state to state).

The F possible outcomes may be interpreted differently depending on the current
state of the Markov process, and the probability of the next state in turn
depends on certain branching ratios.  These branching ratios are proportional to
nonnegative integer branching factors, S(f), which are specified by an array
given as input to the subroutine.  The probability of a particular output of the
table is given by the ratio of the corresponding branch factor to the product of
all of the plane factors.  Since the branching ratios must add up to one, the
branch factors must add up to the product of the plane factors, and the actual
number of degrees of freedom is therefore one less than the number of branches.

Interpreted geometrically, the branching ratios are rational numbers which lie
on a fine lattice in a delta-dimensional simplex, where delta = F-1.  The
subroutine works by recursively decomposing the vector of numerators into a sum
of major and minor components lying on coarser and coarser sublattices
(actually, a common factor of d_n-1 has been cancelled from the list of major
components).  The denominator at each stage is D_n, which is the product of all
the plane factors up to and including the current plane (that is, for all planes
numbered <= n).  If at any stage, the branch factors do not add up to the
correct denominator, an error message is printed.

In order for an arbitrary set of branch factors to be allowed, each plane
factor, d_n, must satisfy d_n <= 2 + D_(n-1)/delta.  If the plane factors grow
too fast, and thus do not satisfy this constraint, some branch factors will
cause an overflow error message because the minor components will not fall in
the required simplex.  In this case, the overall probability space will have
unreachable lattice points, giving a reachable set which is reminiscent of a
Serpinski gasket.  For other branch factors, no error will be produced, and the
lookup table will work as desired.  Finally, the lookup table which results from
the decomposition is not, in general, unique, because subsimplexes must overlap
to fill the larger simplexes. */
```

```
/*
Source: /im/smith/programs/randlut/randlut.c
Author: Mark Andrew Smith
Last Modified: June 3, 1993
*/

randlut(int branches,           /* Number of possible outcomes = F. */
        int branch_factors[],   /* Numerators, S(f), of branching probabilities. */
        int planes,             /* Number of planes of random bits = N. */
        int plane_factors[],    /* Inverse probabilities, d_n, for each plane. */
        int lutindex,           /* Specifies which subtable is being filled. */
        int lut[])              /* Lookup table to be filled. */
{
    int plane, branch, denominator;
    int quotients[branches], remainders[branches];
    int quotient_total, remainder_total;
    int quotient_excess, quotient_decrement;

    if(planes == 0) /* Base case. */
    {
        for(branch = 0; branch < branches; branch++)
        {
            if(branch_factors[branch] == 1) /* Only remaining outcome. */
            {
                lut[lutindex] = branch;
                break;
            }
        }
    }
    else /* Recursive case. */
    {
        planes--; /* Continue with the next most significant bit plane. */

        /* Find the required total for the constituent branching factors. */
        for(denominator = 1, plane = 0; plane < planes; plane++)
            denominator *= plane_factors[plane];

        /* Compute the major components of the branching factors and their sum. */
        for(quotient_total = 0, branch = 0; branch < branches; branch++)
            quotient_total += quotients[branch] = branch_factors[branch] /
                                                  (plane_factors[planes]-1);

        /* Cut the quotient total down until it is equal to denominator. */
        quotient_excess = quotient_total - denominator;
        for(branch = 0; quotient_excess > 0; branch++)
        {
            quotient_decrement = quotients[branch] < quotient_excess ?
                                 quotients[branch] : quotient_excess;
            quotients[branch] -= quotient_decrement;
            quotient_excess -= quotient_decrement;
        }
```

```
        /* Compute the minor components of the branching factors and their sum. */
        for(remainder_total = 0, branch = 0; branch < branches; branch++)
            remainder_total += remainders[branch] = branch_factors[branch] -
                                quotients[branch]*(plane_factors[planes]-1);

        /* Check to see if the remainder total is what it should be. */
        if(remainder_total > denominator)
            fprintf(stderr, "Overflow: I=%d d=%d D=%d r=%d\n", lutindex,
                    plane_factors[planes], denominator, remainder_total);
        if(remainder_total < denominator)
            fprintf(stderr, "Underflow: I=%d d=%d D=%d r=%d\n", lutindex,
                    plane_factors[planes], denominator, remainder_total);

        /* Recursively decompose the quotient and remainder vectors. */
        randlut(branches, quotients, planes, plane_factors, lutindex, lut);
        lutindex |= 1 << planes; /* Switch to the other half of the table. */
        randlut(branches, remainders, planes, plane_factors, lutindex, lut);
    }
}
```

Appendix D

Differential Analysis of a Relativistic Diffusion Law

This appendix expands on some mathematical details of the model of relativistic diffusion presented in chapter 4. The symmetries of the problem suggest underlying connections between a physical CA lattice model and the geometrical nature of the corresponding set of differential equations. Solving the equations rounds out the analysis and serves to make contact with some standard techniques in mathematical physics.

The Lorentz invariance of the model can be explained and generalized by adopting a description in terms of differential geometry. In order to establish a common understanding of this branch of mathematics as it is used in physics, a brief digression is made to lay out some of the fundamental concepts and terminology. Further terminology will be introduced as needed. Using this language, the invariance of the system under conformal rescalings and the conformal group is demonstrated.

The general solution to the equations in the case of a uniform background can be written as a convolution of the initial data with a diffusive kernel, and a simple derivation of the kernel is given. Just as symmetry and conservation laws are powerful tools for problem solving, the invariance of the equations under the conformal group can be used in conjunction with this basic solution to solve the equations in the most general case.

D.1 Elementary Differential Geometry, Notation, and Conventions

The study of differential equations involving general fields and arbitrary spacetimes requires the use of differential geometry. Consequently, the sections below serve as a more or less self-contained introduction to the mathematics needed to elucidate the invariances of the relativistic diffusion law. The presentation is meant to be intuitive and readily absorbed, so no great attempt is made to be rigorous. Instead, the terms are used rather loosely, and many logical shortcuts are taken. The reader who requires more detail can consult any number of references [7, 16, 96]. The first section covers those aspects of geometry that do not require any notion of distance—in particular, manifolds and tensor fields. In section D.1.2, further geometric structure is added by introducing a metric tensor. Finally, the special-case definitions used in chapter 4 are given in section D.1.3.

D.1.1 Scalar, Vector, and Tensor Fields

Differential geometry starts with the concept of a *manifold* which is the mathematical name and generalization of the physical notion of a space or spacetime, and these terms will be used interchangeably. For purposes of this section, a manifold is smooth, has a definite dimensionality, and a fixed global topology, but is otherwise unconstrained. In what follows, a point on an arbitrary manifold will be denoted by x, and the dimension will be denoted by n; however, the discussion will often be specialized to an ordinary, two-dimensional spacetime.

After manifolds comes the analysis of smooth functions on manifolds (a.k.a. fields in physics). Specifically, the concepts and operations of differential calculus can be extended to non-Cartesian spaces and multicomponent objects. Consideration of the differential structure of a manifold leads to the notion of vectors and vector fields. The algebra of vectors leads to higher rank vectors (i.e., elements of vector spaces) called tensors, and the algebra of tensors is the ultimate result of this section. The reason for constructing all of this machinery it that tensor-valued functions of spacetime constitute the classical fields which are found in physics.[1]

The simplest kind of field is a *scalar* (rank 0 tensor) which merely assigns a number to each point in the space, e.g., $f(x)$. An important class of scalar functions are the *coordinate* functions. Each set of n coordinate functions is called a *coordinate system* and will be indexed by Greek letters, e.g., x^μ, where $\mu \in \{0, 1, 2, \ldots, n-1\}$. In general, specifying a value for each of the n functions in a coordinate system uniquely specifies a point on the manifold which in turn specifies the values of the coordinate functions in any other system. Hence there is a functional relationship between one set of coordinates and any other, and any one of them can be taken as a domain of independent coordinate variables on the manifold.

The relationships between two coordinate systems (unprimed and primed) will now be used to develop the differential properties of the manifold. Two complementary types of vector (rank 1 tensor) fields on the manifold can in turn be expressed in terms of directional derivatives and coordinate differentials respectively. This approach to representing physical quantities illustrates the sense in which the fields of physics are rooted in the geometry of spacetime. However, it should be stressed at this point that the fields that ultimately make their way into the fundamental laws of physics should not depend on a particular coordinate system in order to comply with the principle of relativity.

A directional derivative is defined by its action on scalar functions, but it can be written as an operator without explicitly specifying a function. Thus the chain rule for partial derivatives with respect to coordinate variables (with the others held constant) can be written

$$\partial_{\mu'} = \frac{\partial x^\mu}{\partial x^{\mu'}} \partial_\mu \quad \text{and} \quad \partial_\mu = \frac{\partial x^{\mu'}}{\partial x^\mu} \partial_{\mu'}, \tag{D.1}$$

where by the *summation convention*, there is an implicit sum over any index that occurs twice in the same term (unless otherwise noted). An index that occurs once

[1]The exception, spinor fields, will not be covered here.

in every term in a formula is a *free* index, one that occurs twice and is summed over in a term is a *dummy* index (because any other unused letter could take its place), and one that occurs more than twice in a term is probably a mistake.

Similarly, the total differentials of a coordinate in one set with respect to differentials of coordinates in the other set can be written

$$dx^{\mu'} = \frac{\partial x^{\mu'}}{\partial x^{\mu}} dx^{\mu} \quad \text{and} \quad dx^{\mu} = \frac{\partial x^{\mu}}{\partial x^{\mu'}} dx^{\mu'}. \tag{D.2}$$

The above formulas must be consistent under the reverse transformation, so $\frac{\partial x^{\mu'}}{\partial x^{\mu}}$ and $\frac{\partial x^{\mu}}{\partial x^{\mu'}}$ must be inverses of each other:

$$\frac{\partial x^{\mu'}}{\partial x^{\mu}} \frac{\partial x^{\nu}}{\partial x^{\mu'}} = \delta^{\nu}_{\mu}, \quad \text{and} \quad \frac{\partial x^{\mu}}{\partial x^{\mu'}} \frac{\partial x^{\nu'}}{\partial x^{\mu}} = \delta^{\nu'}_{\mu'}, \tag{D.3}$$

where δ^{ν}_{μ} is 1 if $\mu = \nu$ and 0 otherwise. This so-called *Kronecker delta* is an example of a rank 2 tensor and is sometimes called the substitution tensor because its presence in a term has the effect of replacing a dummy index for the other index. Also note that the summation convention implies $\delta^{\mu}_{\mu} = \delta^{\mu'}_{\mu'} = n$. The δ's can be represented as an identity matrix in any coordinate system in any number of dimensions. For example,

$$[\delta^{\nu}_{\mu}] \equiv [\delta^{\nu'}_{\mu'}] \equiv \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{D.4}$$

in two dimensions, where "$\equiv$" denotes a numerical equivalence.

The above relationships can be used to define *vector* quantities on the manifold that have an existence which is *independent* of any coordinate system. For example, the essence of "a quantity with magnitude and direction" can be rigorously captured with directional derivatives. An arbitrary field of directional derivatives can be written as a linear combination of coordinate derivatives, $\boldsymbol{v} = v^{\mu}\partial_{\mu}$, where the v^{μ}'s can be any set of n scalar functions on the manifold. In order for $\boldsymbol{v}$ to be a quantity which doesn't refer to a particular coordinate system, it must also be the case that $\boldsymbol{v}$ can be written as $\boldsymbol{v} = v^{\mu'}\partial_{\mu'}$ for some other set of functions, $v^{\mu'}$. However, from equation (D.1) we have

$$v^{\mu}\partial_{\mu} = v^{\mu}\frac{\partial x^{\mu'}}{\partial x^{\mu}}\partial_{\mu'} \quad \text{and} \quad v^{\mu'}\partial_{\mu'} = v^{\mu'}\frac{\partial x^{\mu}}{\partial x^{\mu'}}\partial_{\mu}. \tag{D.5}$$

Therefore, the sets of component functions of $\boldsymbol{v}$ must transform according to

$$v^{\mu'} = \frac{\partial x^{\mu'}}{\partial x^{\mu}}v^{\mu}, \quad \text{and} \quad v^{\mu} = \frac{\partial x^{\mu}}{\partial x^{\mu'}}v^{\mu'}. \tag{D.6}$$

A quantity that obeys this transformation law is called a *contravariant vector*.

The other type of vector whose existence is independent of any coordinate system is called a *covariant vector* (also called a covector or a 1-form). Such a vector can be thought of as "a quantity with slope and orientation" and its essence can be rigorously captured with differentials. An arbitrary field of 1-forms can be written as a linear combination of coordinate differentials, $\boldsymbol{\omega} = \omega_{\mu}dx^{\mu}$, where the ω_{μ}'s can be any set of n scalar functions on the manifold. In order for $\boldsymbol{\omega}$ to be a quantity which doesn't refer to a particular coordinate system, it must also be the case that $\boldsymbol{\omega}$ can be written as $\boldsymbol{\omega} = \omega_{\mu'}dx^{\mu'}$ for some other set of functions, $\omega_{\mu'}$. However, from equation (D.2) we have

$$\omega_{\mu}dx^{\mu} = \omega_{\mu}\frac{\partial x^{\mu}}{\partial x^{\mu'}}dx^{\mu'}, \quad \text{and} \quad \omega_{\mu'}dx^{\mu'} = \omega_{\mu'}\frac{\partial x^{\mu'}}{\partial x^{\mu}}dx^{\mu}. \tag{D.7}$$

Therefore, the sets of component functions of $\boldsymbol{\omega}$ must transform according to

$$\omega_{\mu'} = \frac{\partial x^{\mu}}{\partial x^{\mu'}}\omega_{\mu}, \quad \text{and} \quad \omega_{\mu} = \frac{\partial x^{\mu'}}{\partial x^{\mu}}\omega_{\mu'}. \tag{D.8}$$

Note the differences and similarities between the transformation laws for ∂_{μ}, dx^{μ}, v^{μ}, and ω_{μ} given by equations (D.1), (D.2), (D.6), and (D.8) respectively. In particular, the transformation law is entirely determined by the name and position of the index: an upper index is always summed against a lower index and vice versa. The name of the free index determines the index on the new component, and it is in the same position as on the original component (contravariant components transform to contravariant components, etc.). Furthermore, the primes are written on the indices because it determines in which coordinate system the components are taken while the actual vector being represented is always the same, independent of coordinate system.

These transformation laws can be extended to fields of higher rank tensors—that is, multicomponent vectors which "have more indices." The number of upper and lower indices indicates the *type* of the tensor, sometimes written (n_u, n_l). For example, the components of a type $(1, 2)$ tensor $\boldsymbol{T}$ transform according to

$$T^{\rho'}_{\mu'\nu'} = \frac{\partial x^{\mu}}{\partial x^{\mu'}} \frac{\partial x^{\nu}}{\partial x^{\nu'}} \frac{\partial x^{\rho'}}{\partial x^{\rho}} T^{\rho}_{\mu\nu} \tag{D.9}$$

The fact that a multicomponent object transforms according to this law is what makes it a tensor, not merely that it has a certain number of components.

The components of a tensor are taken with respect to a basis made of a *tensor product* of basis vectors. For example, $T^{\rho}_{\mu\nu}$ is the component of the tensor $\boldsymbol{T}$ in the coordinate basis, $\partial_\rho \otimes dx^\mu \otimes dx^\nu$. The tensor product is associative, distributive over addition, but not commutative, so the indices must be understood to have some particular order which must be maintained. Summing over all basis tensors gives $\boldsymbol{T} = T^{\rho}_{\mu\nu}\, \partial_\rho \otimes dx^\mu \otimes dx^\nu$. This basis decomposition, along with the transformation laws above, shows that the tensor is a geometrical object with an existence independent of any coordinate system. However, in physics, one usually only deals with the components directly without reference to the basis tensors and their underlying origin in the differential structure of a manifold. Once an order for the indices is understood, one can build up higher rank tensors componentwise by taking ordinary products of components of lower rank tensors. For example, it can be verified that $T^{\rho}_{\mu\nu} v^\sigma \omega_\tau$ are components of a type $(2, 3)$ tensor.

In addition to the tensor or *outer* product of tensors it is possible to define an *inner* product of a contravariant and a covariant vector through the relation $\langle \partial_\mu, dx^\nu \rangle \equiv \delta^\nu_\mu$. It follows from linearity of the inner product that $\langle \boldsymbol{v}, \boldsymbol{\omega} \rangle = v^\mu \omega_\mu$. This product is independent of coordinates as is shown by the following calculation:

$$v^\mu \omega_\mu = v^\mu \delta^\nu_\mu \omega_\nu = v^\mu \frac{\partial x^{\mu'}}{\partial x^{\mu}} \frac{\partial x^{\nu}}{\partial x^{\mu'}} \omega_\nu = v^{\mu'} \omega_{\mu'}. \tag{D.10}$$

The operation of equating an upper and lower index in a term and summing is known as *contraction*, and it maps a type (n_u, n_l) tensor to a type $(n_u - 1, n_l - 1)$ tensor. It is often useful to think of tensors as multilinear mappings on vectors to lower rank tensors via the operation of contraction with the vector components. As shown above, contraction is a coordinate invariant operation because the contravariant and covariant components transform inversely to each other. Note, however, that we don't yet have a coordinate invariant way of taking an inner product of two vectors of the *same* type.

As a final topic of basic differential geometry, we introduce the notion of integration of differential forms. A type $(0, p)$ tensor $(p \leq n)$ that is totally antisymmetric in all its indices is called an p-form (also called a differential form or simply a form). An example in the usual coordinate basis for $p = n = 2$ can be represented as a matrix:

$$[\epsilon_{\mu\nu}] \equiv \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}. \tag{D.11}$$

Differential forms are important because they can be used to define a coordinate independent notion of integration on p-dimensional (sub)manifolds. In the current example,

$$\int f\boldsymbol{\epsilon} = \int f(x)\, \epsilon_{\mu\nu} dx^\mu \otimes dx^\nu \equiv \int f(x)\, dx^0 dx^1, \tag{D.12}$$

where $f(x)$ is any scalar function. The antisymmetry along with the transformation law (D.2) guarantees that, under a coordinate transformation, one recovers the standard Jacobian formula for changing variables in integration. Therefore, the integral of a form is independent of coordinate system, just as the form itself.

Throughout this discussion, nothing has been said about the "shape" of the manifold. Roughly speaking, the manifold can be smoothly deformed and nothing that has been presented so far is affected. This will not be the case in the next section where the manifold will be made "rigid" with the specification of distance between the points of the manifold.

D.1.2 Metric Geometry

Up to this point, our space has had no notion of distance, angles, volume, or any of the other things that one usually associates with geometry. All of these things are obtained by specifying a preferred *metric*, which is an arbitrary nondegenerate, symmetric, type $(0,2)$ tensor field, denoted by $g_{\mu\nu}$.

The primary effect of a metric is to determine the squared interval between "nearby" points on the manifold:

$$ds^2 = g_{\mu\nu} dx^\mu dx^\nu. \tag{D.13}$$

The interval can be thought of as an infinitesimal distance or proper time, depending on the application. The net interval between arbitrary points depends on the path taken between them and is the integral of the infinitesimal intervals. Thus, the total interval along a parameterized curve, $x^\mu(\lambda)$, is given by

$$s = \int ds = \int \sqrt{g_{\mu\nu} \frac{dx^\mu}{d\lambda} \frac{dx^\nu}{d\lambda}}\, d\lambda. \tag{D.14}$$

An equation for the shortest distance (or maximum proper time) between any two points in spacetime can be found by taking the variation of s with respect to $x^\mu(\lambda)$ and setting it to zero:

$$\frac{\delta s}{\delta x^\mu} = 0 \;\Rightarrow\; \frac{d^2 x^\rho}{d\lambda^2} + \Gamma^\rho_{\mu\nu} \frac{dx^\mu}{d\lambda} \frac{dx^\nu}{d\lambda} = 0. \tag{D.15}$$

Solutions to this *geodesic equation* are called *geodesics*, and they generalize the concept of straight lines to arbitrary manifolds. The $\Gamma^\rho_{\mu\nu}$ are called the *Christoffel symbols*, and they are given by the general formula

$$\Gamma^\rho_{\mu\nu} = \tfrac{1}{2} g^{\rho\sigma} (g_{\sigma\nu,\mu} + g_{\mu\sigma,\nu} - g_{\mu\nu,\sigma}), \tag{D.16}$$

where an index following a comma denotes the corresponding partial derivative, (e.g., $f_{,\mu} = \partial_\mu f$), and $g^{\mu\nu}$ is defined as the unique matrix inverse, $g_{\mu\nu} g^{\nu\rho} = \delta^\rho_\mu$. However, it is often easier to compute the components of the Christoffel symbols directly by variation of the argument of the square root in equation (D.14).

Despite appearances, the Christoffel symbols are are not the components of a tensor because they don't obey the tensor transformation law exemplified by equation (D.9). Nor is the ordinary derivative of a vector a tensor because the derivative does not commute with the transformation matrix:

$$\partial_{\mu'} v^{\nu'} = \frac{\partial x^{\mu}}{\partial x^{\mu'}} \partial_{\mu} \frac{\partial x^{\nu'}}{\partial x^{\nu}} v^{\nu} = \frac{\partial x^{\mu}}{\partial x^{\mu'}} \frac{\partial x^{\nu'}}{\partial x^{\nu}} \partial_{\mu} v^{\nu} + \frac{\partial x^{\mu}}{\partial x^{\mu'}} \frac{\partial^2 x^{\nu'}}{\partial x^{\mu} \partial x^{\nu}} v^{\nu}. \tag{D.17}$$

However, these elements can be used together to define a *covariant derivative* which does yield a tensor. The definition of a derivative requires a taking a difference between fields at different points, and $\Gamma^{\rho}_{\mu\nu}$ serves as a *connection*, which tells how tensors at different points are to be transported to the same point where they can be subtracted. In any event, the covariant derivative of a contravariant vector is defined by

$$(\nabla \boldsymbol{v})^{\nu}_{\mu} \equiv \nabla_{\mu} v^{\nu} \equiv \partial_{\mu} v^{\nu} + \Gamma^{\nu}_{\mu\rho} v^{\rho}. \tag{D.18}$$

Similarly, the covariant derivative of a covariant vector defined by

$$(\nabla \boldsymbol{\omega})_{\mu\nu} \equiv \nabla_{\mu} \omega_{\nu} \equiv \partial_{\mu} \omega_{\nu} - \Gamma^{\rho}_{\mu\nu} \omega_{\rho}. \tag{D.19}$$

The covariant derivative can be extended to higher rank tensors by adding (or subtracting) analogous terms for each index. By direct substitution, a calculation such as equation (D.17) shows that the nontensorial terms cancel under a coordinate transformation.

The metric can also be used to define an inner product on vectors of the same type. For contravariant vectors, we have $(\boldsymbol{u}, \boldsymbol{v}) \equiv g_{\mu\nu} u^{\mu} v^{\nu}$, and for covariant vectors, $(\boldsymbol{\alpha}, \boldsymbol{\omega}) \equiv g^{\mu\nu} \alpha_{\mu} \omega_{\nu}$. In direct analogy with the ordinary dot product, this inner product can be used to define the magnitude of a vector as well as the angle between two vectors: $(\boldsymbol{v}, \boldsymbol{v}) = |\boldsymbol{v}|^2$ and $(\boldsymbol{u}, \boldsymbol{v}) = |\boldsymbol{u}||\boldsymbol{v}| \cos\theta$, etc.

The metric also gives a natural isomorphism between vectors and forms; i.e., a linear bijection that doesn't depend on the choice of a basis, but rather on the invariant structure imposed by the metric. In particular, in order for the two kinds of inner product defined above (contraction and dot) to agree, it must be the case that $v_\mu = g_{\mu\nu} v^\nu$ and $\omega^\mu = g^{\mu\nu}\omega_\nu$. The operation of contraction with the metric to convert from covariant to contravariant components and vice versa by is referred to as *raising* and *lowering* indices respectively. In an entirely analogous manner, one can also use the metric to raise and lower indices on higher rank tensors, but care must be exercised to maintain a consistent ordering of the indices.

Finally, since we now have definite notions of distance and angles, it seems only natural that there should be a unique notion of volume, and indeed this is the case. Specifically, the determinant of the metric (when viewed as a matrix), $g \equiv \det(g_{\mu\nu}) = 1/\det(g^{\mu\nu})$, gives a definite scale to differential forms. By considering an orthonormal set of 1-forms, it can be shown that the magnitude of each component of the preferred *volume form* in the coordinate basis must be $\sqrt{|g|}$.[2] For example, in two dimensions the volume form, $\varepsilon_{\mu\nu}$, is given by

$$[\varepsilon_{\mu\nu}] \equiv \sqrt{|g|}\,[\epsilon_{\mu\nu}] \equiv \sqrt{|g|}\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}. \tag{D.20}$$

By raising the indices, one obtains the contravariant version:

$$[\varepsilon^{\mu\nu}] \equiv [g^{\mu\rho} g^{\nu\sigma} \varepsilon_{\rho\sigma}] \equiv \frac{\mathrm{sgn}(g)}{\sqrt{|g|}}\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \tag{D.21}$$

where $\mathrm{sgn}(g)$ is the sign of the determinant. Also note that $\varepsilon^{\mu\nu}\varepsilon_{\mu\nu} = 2!\,\mathrm{sgn}(g)$.

D.1.3 Minkowski Space

This section specializes the discussion above to the case of ordinary two-dimensional Minkowski space. It contains the basic definitions needed to understand the manifest covariance of the relativistic diffusion law under Lorentz transformations. The

[2]The choice of sign is arbitrary, but a positive sign will denote a *right-handed* coordinate system.

notation will also be used in deriving the solution in section D.3.

Two coordinate systems are particularly useful for the description and analysis of this model, and the formulas developed above allow us to transform back and forth. The unprimed coordinates are the usual space and time coordinates, $x^\mu = (x^0, x^1) \equiv (x, t)$, and the primed coordinates are the so-called *light cone-coordinates*, $x^{\mu'} = (x^{0'}, x^{1'}) \equiv (x^+, x^-)$, where $x^\pm \equiv \frac{1}{\sqrt{2}}(t \pm x)$. The inverse coordinate transformation looks similar: $t = \frac{1}{\sqrt{2}}(x^+ + x^-)$, and $x = \frac{1}{\sqrt{2}}(x^+ - x^-)$.

In cases such as this where the coordinate transformation is linear, the component transformation matrix is constant throughout spacetime:

$$\left[\frac{\partial x^{\mu'}}{\partial x^\mu}\right] \equiv \left[\frac{\partial x^\mu}{\partial x^{\mu'}}\right] \equiv \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \tag{D.22}$$

Thus, $\partial_\pm = \frac{1}{\sqrt{2}}(\partial_t \pm \partial_x)$, $\partial_t = \frac{1}{\sqrt{2}}(\partial_+ + \partial_-)$, and $\partial_x = \frac{1}{\sqrt{2}}(\partial_+ - \partial_-)$. Furthermore, the old notion of a displacement being a vector is also meaningful, and the coordinate functions are related by the vector transformation law:

$$x^{\mu'} = \frac{\partial x^{\mu'}}{\partial x^\mu} x^\mu, \quad \text{and} \quad x^\mu = \frac{\partial x^\mu}{\partial x^{\mu'}} x^{\mu'}. \tag{D.23}$$

In the unprimed coordinate system, the metric can be written as

$$[g_{\mu\nu}] \equiv [g^{\mu\nu}] \equiv \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \tag{D.24}$$

Using this metric to raise and lower indices of a vector, $\boldsymbol{v}$, gives $v_0 = v^0$, and $v_1 = -v^1$. Using the above transformation matrix, the metric in the light-cone coordinates can be written

$$[g_{\mu'\nu'}] \equiv [g^{\mu'\nu'}] = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \tag{D.25}$$

The rule for raising and lowering indices in the new system is then: $v_\pm = v^\mp$. Of course, this is consistent with raising and lowering indices before transforming coordinates.

The metric is important for the relativistic diffusion law because it enters into the expressions for the covariant derivative and the preferred volume form. Since the metric is constant in both coordinate systems, $\Gamma^{\rho}_{\mu\nu} \equiv \Gamma^{\rho'}_{\mu'\nu'} \equiv 0$, and the covariant derivative reduces to an ordinary derivative: $\nabla_\mu = \partial_\mu$ and $\nabla_{\mu'} = \partial_{\mu'}$. Since $g = -1$, the preferred volume forms can be written

$$[\varepsilon_{\mu\nu}] \equiv [-\varepsilon_{\mu'\nu'}] \equiv \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}. \tag{D.26}$$

The volume form in the primed system differs by a minus sign because the light-cone coordinate system has the opposite orientation than the original coordinate system (i.e., the new coordinate axes have been interchanged, not just rotated—see fig. (4-1)). By raising the indices we find that

$$[\varepsilon^{\mu\nu}] \equiv [-\varepsilon^{\mu'\nu'}] \equiv \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \tag{D.27}$$

where the sign difference from the last equation reflects the sign of the determinant of the metric.

The subject of this appendix is the relativistic diffusion law, so here we reproduce the central equations (4.7)–(4.10) for convenience:

$$\begin{aligned} \partial_\mu J^\mu &= 0 && \text{(D.28)} \\ (\partial_\mu + \sigma_\mu)\varepsilon^{\mu\nu} J_\nu &= 0 && \text{(D.29)} \\ \partial_\mu \sigma^\mu &= 0 && \text{(D.30)} \\ \partial_\mu \varepsilon^{\mu\nu} \sigma_\nu &= 0. && \text{(D.31)} \end{aligned}$$

This form of the equations is valid in any coordinate system that is linearly related to the standard spacetime coordinates because the transformation matrix is a constant. In particular, the coordinate systems which correspond to physically meaningful frames of reference are related by Lorentz transformations. Hence the equations written in this form are *manifestly covariant* under Lorentz transformations, and the diffusion law obeys the principle of relativity.

D.2 Conformal Invariance

Equations (D.28)–(D.31) can be rewritten in terms of completely general tensors by replacing ordinary partial derivatives with covariant derivatives so that they are valid in *any* coordinate system:

$$\nabla_\mu J^\mu = 0 \tag{D.32}$$

$$(\nabla_\mu + \sigma_\mu)\varepsilon^{\mu\nu} J_\nu = 0 \tag{D.33}$$

$$\nabla_\mu \sigma^\mu = 0 \tag{D.34}$$

$$\nabla_\mu \varepsilon^{\mu\nu}\sigma_\nu = 0. \tag{D.35}$$

This form of the equations is also the proper generalization for curved spacetimes and is appropriate for studying their properties under general coordinate and metric transformations.

D.2.1 Conformal Transformations

A *conformal transformation* is a change of metric resulting from a rescaling of the original: $\hat{g}_{\mu\nu}(x) = \Omega^2(x) g_{\mu\nu}(x)$. Such a transformation may or may not be derived from a mapping of the space to itself (as in the next section), but for now, $\Omega^2(x)$ is just an arbitrary, positive scalar field. This section shows that equations (D.32)–(D.35) are *conformally invariant*, which means that under a conformal transformation and a suitable corresponding change in the tensor fields, the equations take on the same form as before. Conformal invariance is of substantial interest in field theory and implies that the field is composed of massless particles (though the converse not true). This is reflected in the CA model since the particles always move at the speed of light. While the existence of a maximum speed of propagation in all CA is suggestive, the full mathematical consequences of this fact are unknown.

A metric rescaling changes the covariant derivative, the volume form, and possibly the field variables, so it is necessary to evaluate these with respect to the new metric:

$$\begin{aligned}
\hat{\Gamma}^\rho_{\mu\nu} &= \tfrac{1}{2}\hat{g}^{\rho\sigma}(\hat{g}_{\mu\sigma,\nu} + \hat{g}_{\sigma\nu,\mu} - \hat{g}_{\mu\nu,\sigma}) \\
&= \tfrac{1}{2}\Omega^{-2}g^{\rho\sigma}(\Omega^2 g_{\mu\sigma,\nu} + \Omega^2 g_{\sigma\nu,\mu} - \Omega^2 g_{\mu\nu,\sigma} \\
&\quad + 2\Omega g_{\mu\sigma}\Omega_{,\nu} + 2\Omega g_{\sigma\nu}\Omega_{,\mu} - 2\Omega g_{\mu\nu}\Omega_{,\sigma}) \\
&= \Gamma^\rho_{\mu\nu} + \Omega^{-1}(\delta^\rho_\mu\Omega_{,\nu} + \delta^\rho_\nu\Omega_{,\mu} - g^{\rho\sigma}g_{\mu\nu}\Omega_{,\sigma}). \qquad \text{(D.36)}
\end{aligned}$$

Contraction gives

$$\begin{aligned}
\hat{\Gamma}^\mu_{\mu\nu} &= \Gamma^\mu_{\mu\nu} + \Omega^{-1}(n\Omega_{,\nu} + \Omega_{,\nu} - \Omega_{,\nu}) \\
&= \Gamma^\mu_{\mu\nu} + n\Omega^{-1}\Omega_{,\nu}. \qquad \text{(D.37)}
\end{aligned}$$

One must allow for the possibility that the fields have a nonzero conformal weight, s, so the scaled field is defined as $\hat{J}^\mu = \Omega^s J^\mu$, and the left hand side of equation (D.32) becomes

$$\begin{aligned}
\hat{\nabla}_\mu \hat{J}^\mu &= \hat{\nabla}_\mu \Omega^s J^\mu \\
&= \Omega^s\partial_\mu \hat{J}^\mu + \Omega^s\hat{\Gamma}^\mu_{\mu\nu}J^\nu + s\Omega^{s-1}\Omega_{,\mu}J^\mu \\
&= \Omega^s\nabla_\mu J^\mu + n\Omega^{s-1}\Omega_{,\nu}J^\nu + s\Omega^{s-1}\Omega_{,\mu}J^\mu \\
&= \Omega^s\nabla_\mu J^\mu \ \text{ if } s = -n \ (= -2 \text{ in 2d}) \\
&= 0. \qquad \text{(D.38)}
\end{aligned}$$

Therefore, $\nabla_\mu J^\mu = 0$ is conformally invariant if $\hat{J}^\mu = \Omega^{-2}J^\mu$. Similarly, $\nabla_\mu \sigma^\mu = 0$ is conformally invariant if $\hat{\sigma}^\mu = \Omega^{-2}\sigma^\mu$. In other words, the conformal weight of J^μ and σ^μ must be $s = -2$.

Equations (D.32) and (D.35) involve the covariant components of the fields, and the index must be lowered with the new metric:

$$\hat{\sigma}_\mu = \hat{g}_{\mu\nu}\hat{\sigma}^\nu = \Omega^2 g_{\mu\nu}\Omega^{-2}\sigma^\nu = \sigma_\nu. \qquad \text{(D.39)}$$

Therefore, σ_μ (and likewise, J_μ) has a conformal weight of zero. In general, lowering an index increases the conformal weight by two and vice-versa.

Equations (D.32) and (D.35) also involve the volume form, $\varepsilon_{\mu\nu}$, which may have some nonzero conformal weight, s: $\hat{\varepsilon}_{\mu\nu} = \Omega^s \varepsilon_{\mu\nu}$, and by raising the indices, $\hat{\varepsilon}^{\mu\nu} = \Omega^{s-4}\varepsilon^{\mu\nu}$. Since $\hat{\varepsilon}^{\mu\nu}\hat{\varepsilon}_{\mu\nu} = 2!\,\mathrm{sgn}(g)$ is the same constant for any metric, one obtains

$$\hat{\varepsilon}^{\mu\nu}\hat{\varepsilon}_{\mu\nu} = \Omega^{s-4}\varepsilon^{\mu\nu}\Omega^s\varepsilon_{\mu\nu} = \Omega^{2s-4}\varepsilon^{\mu\nu}\varepsilon_{\mu\nu} = \varepsilon^{\mu\nu}\varepsilon_{\mu\nu} \;\Rightarrow\; s = 2. \tag{D.40}$$

Hence, the volume changes as $\hat{\varepsilon}_{\mu\nu} = \Omega^2\varepsilon_{\mu\nu}$ and $\hat{\varepsilon}^{\mu\nu} = \Omega^{-2}\varepsilon^{\mu\nu}$, which is intuitively clear since the effect of the metric rescaling is to increase all lengths by a factor of Ω. The volume form also appears inside a derivative; however,

$$\begin{aligned} 0 &= \nabla_\rho \varepsilon^{\mu\nu}\varepsilon_{\mu\nu} = 2\varepsilon^{\mu\nu}\nabla_\rho\varepsilon_{\mu\nu} \\ &\Rightarrow \nabla_\rho\varepsilon_{\mu\nu} = 0, \end{aligned} \tag{D.41}$$

because all the relevant terms in $\varepsilon^{\mu\nu}\nabla_\rho\varepsilon_{\mu\nu}$ are the same and $\varepsilon^{\mu\nu}$ is nonzero. Raising the indices gives $\nabla_\rho\varepsilon^{\mu\nu} = 0$. Therefore, the volume form can be moved through the covariant derivative.

The right hand side of equation (D.35) then becomes

$$\begin{aligned} \hat{\nabla}_\mu\hat{\varepsilon}^{\mu\nu}\hat{\sigma}_\nu &= \hat{\nabla}_\mu\hat{\varepsilon}^{\mu\nu}\sigma_\nu = \hat{\varepsilon}^{\mu\nu}\hat{\nabla}_\mu\sigma_\nu \\ &= \hat{\varepsilon}^{\mu\nu}(\partial_\mu\sigma_\nu + \hat{\Gamma}^\rho_{\mu\nu}\sigma_\nu) = \hat{\varepsilon}^{\mu\nu}\partial_\mu\sigma_\nu \\ &= \hat{\varepsilon}^{\mu\nu}(\partial_\mu\sigma_\nu + \Gamma^\rho_{\mu\nu}\sigma_\nu) = \hat{\varepsilon}^{\mu\nu}\nabla_\mu\sigma_\nu \\ &= \Omega^{-2}\varepsilon^{\mu\nu}\nabla_\mu\sigma_\nu = \Omega^{-2}\nabla_\mu\varepsilon^{\mu\nu}\sigma_\nu \\ &= 0, \end{aligned} \tag{D.42}$$

where the contraction of the volume form with the connection vanishes by antisymmetry. Therefore, $\nabla_\mu\varepsilon^{\mu\nu}\sigma_\nu = 0$ is conformally invariant. Similarly,

$$(\hat{\nabla}_\mu + \hat{\sigma}_\mu)\hat{\varepsilon}^{\mu\nu}\hat{J}_\nu = (\hat{\nabla}_\mu + \sigma_\mu)\hat{\varepsilon}^{\mu\nu}J_\nu = \Omega^{-2}(\nabla_\mu + \sigma_\mu)\varepsilon^{\mu\nu}J_\nu = 0, \tag{D.43}$$

so $(\nabla_\mu + \sigma_\mu)\varepsilon^{\mu\nu}J_\nu = 0$ is also conformally invariant. Thus, it has been shown that the entire system of equations, (D.32)–(D.35), is conformally invariant.

The system of equations can also be written with a compact, index-free (and therefore coordinate-invariant) notation[3] which sheds some light on the issue of conformal invariance. The covariant components of the fields, J_μ and σ_μ, are the components of 1-forms, $\boldsymbol{J}$ and $\boldsymbol{\sigma}$, so equations (D.32) and (D.35) can be expressed purely in terms of exterior derivatives and products:

$$\varepsilon^{\mu\nu}\nabla_\mu\sigma_\nu = 0 \;\Leftrightarrow\; \nabla_{[\mu}\sigma_{\nu]} = 0 \;\Leftrightarrow\; (\boldsymbol{d\sigma})_{\mu\nu} = 0 \;\Leftrightarrow\; \boldsymbol{d\sigma} = 0. \tag{D.44}$$

Also,

$$\varepsilon^{\mu\nu}(\nabla_\mu + \sigma_\mu)J_\nu = 0 \;\Leftrightarrow\; \nabla_{[\mu}J_{\nu]} + \sigma_{[\mu}J_{\nu]} = 0 \;\Leftrightarrow\; \boldsymbol{dJ} + \boldsymbol{\sigma}\wedge\boldsymbol{J} = 0. \tag{D.45}$$

Since the exterior derivative, $\boldsymbol{d}$, and the exterior product, $\wedge$, don't depend on the metric, these equations are automatically conformally invariant.

On the other hand, equations (D.32) and (D.34) can be written as

$$\nabla_\mu J^\mu = 0 \quad\Leftrightarrow\quad \boldsymbol{\delta J} = 0 \tag{D.46}$$

$$\nabla_\mu \sigma^\mu = 0 \quad\Leftrightarrow\quad \boldsymbol{\delta\sigma} = 0. \tag{D.47}$$

The $\boldsymbol{\delta}$ operator *does* depend on the metric, so these equations require a conformal rescaling of the contravariant components of the fields in order to be conformally invariant.

D.2.2 The Conformal Group

The set of smooth mappings of a spacetime to itself that result in a rescaling of the metric is called the *conformal group*, and it also happens to be the group that preserves the causal structure defined by the light cones (i.e., it maps null vectors to

[3]See [16] for definitions of the operators $\boldsymbol{d}$, $\wedge$, and $\boldsymbol{\delta}$.

null vectors). The picture of mapping the space to itself is the *active* point of view, but the conformal group can also be viewed as a change of coordinates, which is the *passive* point of view. The later viewpoint, which is adopted here, is perhaps more physical because the space remains the same: only the description of the points in the space changes.

The purpose of this section is to show that the *original* equations (D.28)–(D.31) are invariant under the conformal group. In addition to a change of coordinates, the fields may need to be rescaled as before. In $n \geq 2$ spacetime dimensions, the conformal group has $(n+1)(n+2)/2$ dimensions, but for $n = 2$, it has an infinite number of dimensions. The later can be seen by imagining an arbitrary local scaling between two sets of light-cone coordinates, $x^\mu = (x^+, x^-)$ and $x^{\mu'} = (x^{+'}, x^{-'})$, which can be written as $x^{\pm'} = f^\pm(x^\pm)$ for any strictly monotone functions $f^\pm$. This enormous freedom in two dimensions is related to the fact that the light cone consists of two disconnected branches.

The transformation matrix in this case consists of two elements which will be abbreviated as

$$Df^\pm \equiv \frac{df^\pm}{dx^\pm}(x^\pm) = \frac{\partial x^{\pm'}}{\partial x^\pm}. \tag{D.48}$$

Then $\partial_\pm = (Df^\pm)\partial_{\pm'}$, and the metric transformation is simply

$$[g_{\mu'\nu'}] = \frac{1}{(Df^+)(Df^-)}[g_{\mu\nu}] = \frac{1}{(Df^+)(Df^-)}\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \tag{D.49}$$

The metric in the new coordinate system can be rescaled, $\hat{g}_{\mu'\nu'} = \Omega^2 g_{\mu'\nu'}$ where $\Omega^2 \equiv (Df^+)(Df^-)$, to give the original metric:

$$[\hat{g}_{\mu'\nu'}] = (Df^+)(Df^-)[g_{\mu'\nu'}] = [g_{\mu\nu}] = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \tag{D.50}$$

The fields scale as in the previous section: $\hat{J}^{\mu'} = \Omega^{-2}J^{\mu'}$, $\hat{\sigma}^{\mu'} = \Omega^{-2}\sigma^{\mu'}$, and $\hat{\sigma}_{\mu'} = \sigma_{\mu'}$.

To see how all of this helps, consider the following sequence of transformations:

$$\begin{aligned}
\partial_\mu J^\mu &= 0 \text{ (given)} && \text{(D.51)}\\
\Rightarrow\ \nabla_\mu J^\mu &= 0 \text{ (constant } g_{\mu\nu} \Rightarrow \Gamma^\rho_{\mu\nu} = 0) && \text{(D.52)}\\
\Rightarrow \nabla_{\mu'} J^{\mu'} &= 0 \text{ (by tensor transformation)} && \text{(D.53)}\\
\Rightarrow \hat{\nabla}_{\mu'} \hat{J}^{\mu'} &= 0 \text{ (by conformal invariance)} && \text{(D.54)}\\
\Rightarrow\ \partial_{\mu'} \hat{J}^{\mu'} &= 0 \text{ (constant } \hat{g}_{\mu'\nu'} \Rightarrow \hat{\Gamma}^{\rho'}_{\mu'\nu'} = 0). && \text{(D.55)}
\end{aligned}$$

Therefore, by rescaling the fields, one can offset the effect of a transformation of the conformal group.

This can be shown more explicitly in light-cone coordinates as follows:

$$J^{\mu'} = \frac{\partial x^{\mu'}}{\partial x^\mu} J^\mu \ \Rightarrow\ J^\pm = \frac{J^{\pm'}}{(Df^\pm)} = \frac{\Omega^2 \hat{J}^{\pm'}}{(Df^\pm)} = (Df^\mp)\hat{J}^{\pm'}. \qquad \text{(D.56)}$$

Similarly, $\sigma^\pm = (Df^\mp)\hat{\sigma}^{\pm'}$ and $\sigma_\pm = (Df^\pm)\sigma_{\pm'} = (Df^\pm)\hat{\sigma}_{\pm'}$. Equations (D.28)–(D.31) (i.e., without covariant derivatives) also apply in the original light-cone coordinates as defined in section D.1.3 because the metric is constant and the connection vanishes. Using the fact that $\varepsilon^{\mu\nu} J_\nu \equiv (-J^+, J^-)$ gives:

$$\begin{aligned}
\partial_+ J^+ + \partial_- J^- &= 0 && \text{(D.57)}\\
\partial_+ J^+ - \partial_- J^- + \sigma_+ J^+ - \sigma_- J^- &= 0 && \text{(D.58)}\\
\partial_+ \sigma^+ + \partial_- \sigma^- &= 0 && \text{(D.59)}\\
\partial_+ \sigma^+ - \partial_- \sigma^- &= 0. && \text{(D.60)}
\end{aligned}$$

Substitution of the scaled and transformed fields gives

$$\begin{aligned}
(Df^+)\partial_{+'}(Df^-)\hat{J}^{+'} + (Df^-)\partial_{-'}(Df^+)\hat{J}^{-'} &= 0 && \text{(D.61)}\\
(Df^+)\partial_{+'}(Df^-)\hat{J}^{+'} - (Df^-)\partial_{-'}(Df^+)\hat{J}^{-'} & && \text{(D.62)}\\
+ (Df^+)\hat{\sigma}_{+'}(Df^-)\hat{J}^{+'} - (Df^-)\hat{\sigma}_{-'}(Df^+)\hat{J}^{-'} &= 0 && \text{(D.63)}\\
(Df^+)\partial_{+'}(Df^-)\hat{\sigma}^{+'} + (Df^-)\partial_{-'}(Df^+)\hat{\sigma}^{-'} &= 0 && \text{(D.64)}
\end{aligned}$$

$$(Df^+)\partial_{+'}(Df^-)\hat{\sigma}^{+'} - (Df^-)\partial_{-'}(Df^+)\hat{\sigma}^{-'} \;=\; 0, \tag{D.65}$$

which become (since $Df^{\pm}$ doesn't depend on $x^{\mp'}$)

$$\begin{aligned}
\Omega^2(\partial_{+'}\hat{J}^{+'} + \partial_{-'}\hat{J}^{-'}) &= 0 && \text{(D.66)}\\
\Omega^2(\partial_{+'}\hat{J}^{+'} - \partial_{-'}\hat{J}^{-'} + \hat{\sigma}_{+'}\hat{J}^{+'} - \hat{\sigma}_{-'}\hat{J}^{-'}) &= 0 && \text{(D.67)}\\
\Omega^2(\partial_{+'}\hat{\sigma}^{+'} + \partial_{-'}\hat{\sigma}^{-'}) &= 0 && \text{(D.68)}\\
\Omega^2(\partial_{+'}\hat{\sigma}^{+'} - \partial_{-'}\hat{\sigma}^{-'}) &= 0. && \text{(D.69)}
\end{aligned}$$

Adding and subtracting the last two equations gives $\partial_{+'}\hat{\sigma}^{+'} = 0$ and $\partial_{+'}\sigma^{+} = 0$ which can be integrated immediately to give arbitrary functions $\hat{\sigma}^{+'} = \hat{\sigma}^{+'}(x^{-'})$ and $\hat{\sigma}^{-'} = \hat{\sigma}^{-'}(x^{+'})$. The background field, $\hat{\sigma}_{\pm'}$, can be transformed to a constant by choosing coordinates so that $\sigma_{\pm} = (Df^{\pm})\sigma_0$. This makes $\hat{\sigma}_{\pm'} = \sigma_0$ (as well as $\hat{\sigma}^{\pm'} = \sigma_0$). The equation for the transformed and scaled currents then become

$$\begin{aligned}
\partial_{+'}\hat{J}^{+'} + \partial_{-'}\hat{J}^{-'} &= 0 && \text{(D.70)}\\
\partial_{+'}\hat{J}^{+'} - \partial_{-'}\hat{J}^{-'} + \sigma_0(\hat{J}^{+'} - \hat{J}^{-'}) &= 0. && \text{(D.71)}
\end{aligned}$$

As a final remark, note that it is completely clear from figure 4-1 that the *process* described by the original CA is invariant under arbitrary rescaling of the light-cone coordinates. In other words, the visual representation afforded by CA allows one to bypass all of the above formal development in order to draw significant conclusions about the properties of the model. This illustrates the latent power of CA as a mathematical framework for describing the essential aspects of physical phenomena.

D.3 Analytic Solution

This section shows how to derive the complete solution to equations (D.28)–(D.31). Given the symmetries outlined above, it makes sense to do the analysis in light-cone coordinates which results in the equivalent equations (D.57)–(D.60). It is clear from the above discussion that (D.59) and (D.60) can be solved completely to give $\sigma^+ = \sigma^+(x^-)$ and $\sigma^- = \sigma^-(x^+)$; furthermore, without loss of generality, they can be rescaled to a constant, σ_0.

Adding and subtracting equations (D.57) and (D.58) with $\sigma_\pm = \sigma_0$ gives

$$2\partial_+ J^+ + \sigma_0(J^+ - J^-) = 0 \quad \text{(D.72)}$$

$$2\partial_- J^- + \sigma_0(J^- - J^+) = 0. \quad \text{(D.73)}$$

Since σ_0 is related to the mean free path by $\sigma_0 = \sqrt{2}/\lambda$, it is convenient to rescale the problem by introducing dimensionless coordinates $z^\pm = x^\pm/\sqrt{2}\lambda$. This gives $\lambda\sqrt{2}\,\partial_\pm = \partial_{z^\pm}$. Using the fact that $J^\pm$ is proportional to $\rho^\pm$ (in fact, $J^\pm = \sqrt{2}\rho^\pm$), the above equations can be expressed in terms of the densities as

$$\partial_{z^+}\rho^+ + \rho^+ - \rho^- = 0 \quad \text{(D.74)}$$

$$\partial_{z^-}\rho^- + \rho^- - \rho^+ = 0. \quad \text{(D.75)}$$

These also could have been written down immediately from the original transport equations, (4.1) and (4.2).

The general solution can be written in terms of Green's functions as a superposition of solutions, each one of which starts from an isolated pulse moving right or left from a given position. Since the problem is invariant under translation and parity, it suffices to find the solution starting from a single pulse at the origin moving to the right. The Green's function for the right and left moving particles will be denoted by $\varrho^\pm(x,t)$. The initial conditions are therefore

$$\varrho^+(x, t=0) = \delta(x) = \delta(t-x) \quad \text{(D.76)}$$

$$\varrho^-(x, t=0) = 0. \quad \text{(D.77)}$$

From physical considerations, it is clear that the fields vanish outside the light cone. The boundaries can therefore be rotated to light-cone coordinates to give the condi-

$\tilde{f}(s)$	$f(z)$
$s\tilde{f}(s) - f(0)$	$\frac{d}{dz} f(z)$
$\tilde{f}(s+\alpha)$	$e^{-\alpha z} f(z)$
1	$\delta(z)$
$\frac{1}{s+\alpha}$	$e^{-\alpha z}$
$\frac{e^{\alpha/s}}{s}$	$I_0(2\sqrt{\alpha z})$
$e^{\alpha/s} - 1$	$\sqrt{\frac{\alpha}{z}} I_1(2\sqrt{\alpha z})$

Table D.1: A table of Laplace transforms used in the solution of the relativistic diffusion equation.

tions

$$\varrho^+(z^+ = 0, z^-) = \delta(\sqrt{2}x^-) = \delta(2\lambda z^-) = \frac{1}{2\lambda}\delta(z^-) \tag{D.78}$$

$$\varrho^-(z^+, z^- = 0) = 0. \tag{D.79}$$

Since the equations are linear and have constant coefficients, they can be treated with integral transforms. Furthermore, by using the Laplace transform, the initial boundary values can be absorbed into the transformed equations. Accordingly, the Laplace transform is defined by

$$\tilde{f}(s) \equiv \mathcal{L}\{f(z)\} \equiv \int_0^\infty f(z) e^{-sz} dz, \tag{D.80}$$

where table D.1 gives a complete list of the transforms that are needed in this section [17].

The equations are transformed term by term, and each term can be transformed one variable at a time. For example, applying the transform parallel to the z^- axis gives

$$\tilde{\varrho}^\pm(z^+, s_-) = \int_0^\infty \varrho^\pm(z^+, z^-) e^{-s_- z^-} dz^-, \tag{D.81}$$

and transforming this intermediate function parallel to the z^+ axis gives

$$\begin{aligned}\tilde{\tilde{\varrho}}^{\pm}(s_+,s_-) &= \int_0^\infty \tilde{\varrho}^{\pm}(z^+,s_-)e^{-s_+z^+}\,dz^+ \\ &= \int_0^\infty\int_0^\infty \varrho^{\pm}(z^+,z^-)\,e^{-s_+z^+-s_-z^-}\,dz^+dz^-. \end{aligned}\tag{D.82}$$

The nontrivial boundary condition on ϱ^+ enters through the ∂_{z^+} derivative term which transforms as

$$\begin{aligned}\mathcal{L}_-\mathcal{L}_+\{\partial_{z^+}\varrho^+\} &= \mathcal{L}_-\{s_-\tilde{\varrho}^+(s_+,z^-) - \varrho^+(z^+=0,z^-)\} \\ &= s_-\tilde{\tilde{\varrho}}^+(s_+,s_-) - \int_0^\infty \frac{1}{2\lambda}\delta(z^-)\,e^{-s_-z^-}\,dz^- \\ &= s_-\tilde{\tilde{\varrho}}^+(s_+,s_-) - \frac{1}{2\lambda}.\end{aligned}\tag{D.83}$$

Thus, the fully transformed equations are

$$s_+\tilde{\tilde{\varrho}}^+ + \tilde{\tilde{\varrho}}^+ - \tilde{\tilde{\varrho}}^- = \frac{1}{2\lambda}\tag{D.84}$$

$$s_-\tilde{\tilde{\varrho}}^- + \tilde{\tilde{\varrho}}^- - \tilde{\tilde{\varrho}}^+ = 0.\tag{D.85}$$

Solving these algebraic equations for $\tilde{\tilde{\varrho}}^+$ and $\tilde{\tilde{\varrho}}^-$ gives

$$\tilde{\tilde{\varrho}}^+ = \frac{1}{2\lambda}\cdot\frac{1}{s_+ + \frac{s_-}{s_-+1}}\tag{D.86}$$

$$\tilde{\tilde{\varrho}}^- = \frac{1}{2\lambda}\cdot\frac{1}{s_-+1}\cdot\frac{1}{s_+ + \frac{s_-}{s_-+1}}.\tag{D.87}$$

The inverse transforms are also done one variable at a time. Inverting with respect to s_+ gives

$$\tilde{\varrho}^+ = \frac{1}{2\lambda}\,e^{-\frac{s_-z^+}{s_-+1}} = \frac{e^{-z^+}}{2\lambda}\cdot\left[e^{\frac{z^+}{s_-+1}} - 1 + 1\right]\tag{D.88}$$

$$\tilde{\varrho}^- = \frac{1}{2\lambda}\cdot\frac{1}{s_-+1}\,e^{-\frac{s_-z^+}{s_-+1}} = \frac{e^{-z^+}}{2\lambda}\cdot\frac{e^{\frac{z^+}{s_-+1}}}{s_-+1}.\tag{D.89}$$

Inverting with respect to s_- (with an extra factor of e^{-z^-} from the shift theorem applied to $s_- + 1$) gives

$$
\begin{aligned}
\varrho^+ &= \frac{e^{-z^+ - z^-}}{2\lambda}\left[\sqrt{\frac{z^+}{z^-}}\, I_1(2\sqrt{z^+ z^-})\right] + \frac{e^{-z^+}}{2\lambda}\,\delta(z^-) \\
&= \frac{e^{-(x^+ + x^-)/\sqrt{2}\lambda}}{2\lambda}\left[\sqrt{\frac{x^+}{x^-}}\, I_1\left(2\sqrt{\frac{x^+ x^-}{2\lambda^2}}\right)\right] + e^{-x^+/\sqrt{2}\lambda}\,\delta(\sqrt{2}\,x^-) \qquad \text{(D.90)} \\
\varrho^- &= \frac{e^{-z^+ - z^-}}{2\lambda}\, I_0(2\sqrt{z^+ z^-}) \\
&= \frac{e^{-(x^+ + x^-)/\sqrt{2}\lambda}}{2\lambda}\, I_0\left(2\sqrt{\frac{x^+ x^-}{2\lambda^2}}\right). \qquad \text{(D.91)}
\end{aligned}
$$

Using the fact that $x^+ x^- = (t^2 - x^2)/2$ as well as $x^+ = \sqrt{2}\,t$ when $t = x$ gives the fundamental result:

$$
\begin{aligned}
\varrho^+(x,t) &= \frac{e^{-t/\lambda}}{2\lambda}\sqrt{\frac{t+x}{t-x}}\, I_1\left(\frac{\sqrt{t^2 - x^2}}{\lambda}\right) + e^{-t/\lambda}\delta(t-x) \qquad \text{(D.92)} \\
\varrho^-(x,t) &= \frac{e^{-t/\lambda}}{2\lambda} I_0\left(\frac{\sqrt{t^2 - x^2}}{\lambda}\right). \qquad \text{(D.93)}
\end{aligned}
$$

These can be added and subtracted to give $\rho(x,t)$ and $j(x,t)$ respectively.

The kernel for a pulse initially moving to the left can be found by interchanging $x \leftrightarrow -x$ (including $\varrho^+ \leftrightarrow \varrho^-$) in the solutions above. The solution for arbitrary initial conditions can then be written as a convolution:

$$
\begin{aligned}
\rho^+(x,t) &= \int \Big[\rho^+(x',0)\varrho^+(x - x', t) + \rho^-(x',0)\varrho^-(x' - x, t)\Big]\, dx' \qquad \text{(D.94)} \\
\rho^-(x,t) &= \int \Big[\rho^+(x',0)\varrho^-(x - x', t) + \rho^-(x',0)\varrho^+(x' - x, t)\Big]\, dx'. \qquad \text{(D.95)}
\end{aligned}
$$

If necessary, the rescaling given in the last section can be applied to give the solution for arbitrary $\lambda^+(x^+)$ and $\lambda^-(x^-)$. In this case, the complete solution will no longer be a simple convolution; rather, the Green's functions in the above integral should instead be written as $\varrho^\pm(x,t;x')$ since they will not be translation invariant but will vary from point to point.

Appendix E

Basic Polymer Scaling Laws

This appendix gives a simple, intuitive derivation of the basic polymer scaling laws seen in chapter 5. It follows in the spirit of the highly physical arguments given by deGennes [21]. While there has been a great deal of sophisticated work done on polymer statistics including field-theoretic and renormalization group calculations, no great effort is made here to be physically or mathematically rigorous. Instead, it is included for completeness and because it reveals a considerable amount about the physics of complex systems for such a small calculation. It is also notable for purposes of justification of physical modeling with CA, in that such universal phenomena can arise out of an abstract model in discrete space and time.

The goal is to obtain a measure of the characteristic radius R of an isolated polymer and the associated relaxation time τ. We are primarily interested in their dependence on the degree of polymerization N and the dimensionality d, although they also depend on the bond length a (or lattice spacing) and on the effective excluded volume of a monomer v. In particular, all macroscopic quantities grow as a power of N (which depends on d) because the system displays a self-similarity with increasing polymer length. In the present case, this scaling requires that the polymer is in a good solvent and feels only short-ranged interactions, i.e., it doesn't stick to itself and is not charged.

In what follows, the symbol "$\cong$" will be used for approximations where numerical factors may be omitted. Furthermore, the symbol "$\sim$" will be used when dimensional factors have also been dropped in order to simplify the relationships and to emphasize the scaling dependence for large N. Finally, the expected values of all macroscopic lengths (end-to-end distance, radius of gyration, Flory radius, etc.) are proportional, so any of them can be denoted simply by R.

E.1 Radius of Gyration

The scaling law for the characteristic radius of a polymer in a dilute solution with a good solvent was originally given by Flory [29]. The nontrivial behavior results from a competition between entropy and repulsion among the monomers. On one hand, the polymer would like to contract for entropic reasons to become an ideal random walk (also called a Gaussian chain). On the other hand, the chain cannot collapse that far because the monomers would start to overlap. The strategy of the calculation then, is to minimize the total free energy consisting of a sum of contributions from statistical and repulsive forces.

The elastic energy of an ideal polymer is determined by the number of possible random walks between its end points. The probability of a random walk with steps of length a ending at a position $\mathbf{R}$ relative to its beginning is a Gaussian distribution with a standard deviation given by $\sigma \cong \sqrt{N}\,a$. The number of random walks out to a radius R is therefore proportional to

$$\Omega \cong R^{d-1} e^{-R^2/Na^2}. \tag{E.1}$$

With the temperature T given in energy units, this leads to a free energy of

$$F_{\mathrm{el}} = E - TS = -T \ln \Omega \cong \frac{TR^2}{Na^2} - (d-1) T \ln R \sim \frac{R^2}{N} - \ln R, \tag{E.2}$$

where all polymer configurations have zero energy.

The energy due to repulsion of the monomers can be estimated by counting the number of monomer overlaps in a volume of R^d using a mean field approximation to the local concentration, $c \cong N/R^d$. The number of overlaps is therefore approximately

$cvN/2$, where v is an effective excluded volume parameter which may depend on the temperature, and the factor of two prevents double counting. The corresponding free energy is then

$$F_{\text{rep}} \cong \frac{1}{2} T v(T) \frac{N^2}{R^d} \sim \frac{N^2}{R^d}. \tag{E.3}$$

A solvent is considered to be good when $v > 0$.

Neglecting the logarithmic contribution to the free energy for now gives the total dependence on N and R:

$$F \sim \frac{R^2}{N} + \frac{N^2}{R^d}. \tag{E.4}$$

Minimizing this with respect to R at constant N gives

$$\frac{\partial F}{\partial R} \sim \frac{R}{N} - \frac{N^2}{R^{d+1}} \sim 0, \tag{E.5}$$

or

$$R \sim N^{\frac{3}{d+2}} \sim N^{\nu}. \tag{E.6}$$

Thus, the Flory exponent is given by

$$\nu \sim \frac{3}{d+2}. \tag{E.7}$$

This result is valid for $1 \leq d \leq 4$ and is well verified by simulations.

Discussion

Note that the exponent drops below the ideal value of $1/2$ for $d > 4$. Clearly this is unphysical, and we can see the origin of the problem if we include the logarithmic term in the free energy:

$$F \sim \frac{R^2}{N} - \ln R + \frac{N^2}{R^d}. \tag{E.8}$$

and

$$\frac{\partial F}{\partial R} \sim \frac{R}{N} - \frac{1}{R} - \frac{N^2}{R^{d+1}} \sim 0. \tag{E.9}$$

As N and R scale, the middle term becomes more significant than the last one for $d > 4$, and

$$R \sim N^{1/2}. \tag{E.10}$$

In other words, the overlap energy becomes negligible and the chains are ideal.

Another interesting point is that the overlap energy can also be viewed as entirely entropic. Consider the reduction in the volume of configuration space which results from the excluded volume of the monomers. In the original calculation, the monomers are allowed to reside anywhere within a volume of R^d. However, if the monomers are added to the volume one by one, the volume available to the n^{th} monomer is reduced to $R^d - nv$. Thus, the volume of configuration space is reduced by the factor

$$\begin{aligned}
\frac{\Omega'}{\Omega} &\cong \frac{R^d(R^d - v)(R^d - 2v)\cdots(R^d - (N-1)v)}{(R^d)^N} && \text{(E.11)} \\
&= 1 \cdot \left(1 - \frac{v}{R^d}\right) \cdots \left(1 - \frac{(N-1)v}{R^d}\right) && \text{(E.12)} \\
&\cong \prod_{n=0}^{N-1} \exp\left(-\frac{nv}{R^d}\right) = \exp\left(-v \sum_{n=0}^{N-1} \frac{n}{R^d}\right) \cong \exp\left(-v\frac{N^2}{2R^d}\right). && \text{(E.13)}
\end{aligned}$$

The total free energy is then again

$$F = -T \ln \Omega' = F_{\text{el}} - T \ln \frac{\Omega'}{\Omega} \cong T\frac{R^2}{Na^2} + Tv\frac{N^2}{2R^d}. \tag{E.14}$$

In the case of our lattice models, the monomers can never really overlap, and the internal energy is constant. The free energy comes entirely from the entropy, and the temperature doesn't matter: the model is "athermal."

Finally, note that nothing in the above argument assumes anything about the connected topology of the polymers. In other words, the polymers could just as well be "phantom chains" which are allowed to pass through each other while still maintaining an excluded volume, and the scaling law would be the same. I have verified this with a simulation using a variation of the bond-fluctuation algorithm that uses bonds long enough to allow the polymers to pass through each other. Furthermore, if the check for excluded volume is eliminated, ideal chain behavior results.

E.2 Rouse Relaxation Time

In general, dynamic scaling laws are not as universal and are not as well understood as static scaling laws, but for our individual lattice polymers, the situation is greatly simplified by the absence of hydrodynamic effects, overlapping chains, shear, and the like. We are interested in finding the characteristic time τ that it takes for the radius of gyration to change, sometimes called the Rouse relaxation time.

The relaxation of the polymer chain proceeds by the relative diffusion of one part of the chain with respect to another, e.g., the two halves of the chain. Fluctuations in R are themselves proportional to R, so the relevant time scale will be proportional to the time needed for the entire chain to diffuse across its own width. Since a diffusion length x is related to a diffusion time t by $x^2 \sim Dt$, where D is the diffusion constant, the relaxation time we wish to calculate is given by $\tau \sim R^2/D$.

The diffusion constant for the polymer scales like $D \sim 1/N$ which can be understood as follows. In equilibrium, each monomer will move in a random direction some average distance δ per time step. These random displacements will add up as in a random walk to give a total moment of $\Delta \cong \sqrt{N}\,\delta$. Hence, the center of mass of the entire polymer moves an average distance $\Delta/N \cong \delta/\sqrt{N}$ in one time step. After t such steps, the center of mass moves $\sqrt{t}$ times this amount. On the other hand, in t time steps, the center of mass will diffuse a distance proportional to $\sqrt{Dt}$. Comparing these results gives $D \sim 1/N$.

Combining the above scaling laws gives the relaxation time

$$\tau \sim \frac{R^2}{D} \sim \frac{(N^\nu)^2}{1/N} \sim N^{2\nu+1} \tag{E.15}$$

where

$$2\nu + 1 = \frac{d+8}{d+2}. \tag{E.16}$$

This result also compares favorably with the scaling of the relaxation times found in the simulations.

Appendix F

Differential Forms for Cellular Automata

This appendix discusses how some of the machinery presented in appendix D might be extended to CA. Eventually it should be possible to develop rigorous definitions of the concepts introduced here and to recapitulate the field of analysis in these terms. A lot of work is still left to be done as this is just a preliminary attempt at bringing together a few mathematical ideas for capturing physical systems in CA.

F.1 Cellular Automata Representations of Physical Fields

The fundamental laws of physics do not depend on a particular coordinate system nor on the choice of a basis. Similarly, the properties of a CA model of a physical field should not depend on the way the model is coordinatized [81]. Thus, we would like our CA fields to follow a true geometry and not have them be a mere, discretized approximation of components in some special basis. The ultimate goal in this direction is to pattern the analysis after a geometrical approach to physics [16].

Topology is also an important consideration in representing physical spaces and fields. The main properties from topology that we are concerned with are connectedness and continuity. Roughly speaking, connectedness tells us how our spaces (both spacetime lattices and sets of cell values) are woven together, and continuity tells us that mappings between spaces respect this connectivity. One way to discretize any manifold is to embed it in a high dimensional space which has been finely tiled with cells: the discretized manifold is then comprised of the cells which contain the original manifold. For example, a sphere can be approximated by the 26 states given by the 1×1 surface cubes of a 3×3 cube. Connectivity is then defined by adjacency in the original tiling. A configuration of cell states is then defined to be *continuous* if the values in adjacent cells differ by at most one with respect to a cyclic ordering of the states.

Many of the CA models of physical systems that one sees are composed of distinctly identifiable particles as in a lattice gas. A possible alternative to this particle picture is a field picture where it may not possible to unambiguously define a particle. Both lattice gases and fields have many degrees of freedom, but they differ in their kinematic descriptions and in degree of continuity they convey. In the first case position is a function of a particle index, and in the later case, amplitude is a function of a position index. It would be interesting to trace the similarities and differences between the particle and field pictures in CA and quantum field theory.

F.2 Scalars and One-Forms

The simplest field, and probably the easiest one to model, is a scalar, i.e., a real or complex function that doesn't change under transformations of spacetime. This does not involve any differential properties of space, so it is acceptable to use the values in the cells directly. These values can be smeared for interpretation ("measured") by convolution with a test function (averaged on a per cell basis). One might also want to take the limit of a sequence of simulations by letting the lattice spacing and the time step go to zero. Such limiting procedures, while not strictly necessary for the CA rule per se, are useful for us to get a continuum approximation of the behavior (if one exists).

Another closely related way to do this is to use a *unary* representation as is done for density in lattice gases. This means that one counts the number of "particles" in a region and takes the mean (averaged on a per area basis). Toffoli [85] gives a general discussion of how to do this along with other ways in which CA might be viewed as an alternative mathematical structure for physics. One thing to be stressed is that averaging representations in this way automatically gives a kind of continuity—this is good in general, but might be a disadvantage in certain situations.

A scalar field can also be represented by its *contours*. Contours are unbroken lines (or hypersurfaces in higher dimensions) that do not cross and never end. They can be represented in any way which has these properties and consistently specifies which side of each contour is higher. The magnitude of the field is inversely proportional to the "distance" between contours. The implementation used in fig. 3-8 encodes the potential very efficiently by digitizing it and then only storing the lowest bit. In this case, it is a harmonic potential, $\frac{1}{2}m\omega^2 r^2$, in real space or alternatively, the classical kinetic energy, $p^2/2m$, in momentum space. The entire potential can be recovered (up to a constant) by integration. Note that the overall slope of the potential is only discernible by looking at a relatively large region. This illustrates the multiple-cell character of this representation.

One-forms are geometrical quantities very much like vectors. Like vectors, they are rigorously described in terms of infinitesimals of space. Unlike vectors, they cannot always be pictured as arrows. A better picture of a one-form is that of a separate contour map at each point. If the maps fit together to make a complete contour map, then the one-form field is said to be exact. In this case, it can be written as a *gradient*, dV. The scalar field that it is the gradient of is a *potential*, V. Potentials are important in physics because they create forces; in this case, $F_i dx^i = -dV$.

The contour map method for one-forms could also be modified in several ways. First, they don't have to be made exact: the contours could be joined in such a way that is globally incompatible with an underlying function, e.g., figure F-1. Second, they could, by fiat, be weighted by one of the above representations of scalar functions. Finally, they could be "added" by cycling between two or more configurations. These

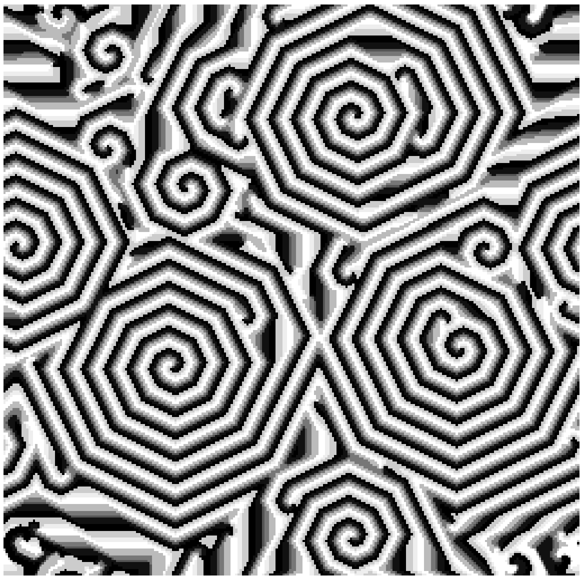


Figure F-1: Configuration from a CA model of an oscillatory chemical reaction. The cells assume eight different values, and the values change gradually between neighboring cells. The centers of the spirals are topological defects.

modifications taken together might form a suitable scheme for representing general one-forms.

F.3 Integration

Differential forms constitute the basis of integration on manifolds (see section D.1.1), while this section discusses the topic of integration in CA. The "windows" over which the counts are taken play the role of differential forms. Similar ideas occur in the field of image processing under the name *count measures* [95].

Consider a tiling of the Cartesian plane with simply connected domains. Each square cell in the space is labeled with the number of the domain to which it belongs or with zero if it is in the no-man's land between domains. The problem is to assign to each domain an area and a circumference which closely approximate those of the continuum limit as the size of the domains grows without bound. Furthermore, we would like to make these assignments by integrating a local function of the cells.

F.3.1 Area

The area of a numbered domain will simply be the count of the cells having that number. As will be explained below, it sometimes makes sense to subtract 1/2 from this value.

F.3.2 Winding Number

Winding number refers to any topological quantity which is given by the number of complete turns made by mappings between a circle and a plane. An archetype to keep in mind is provided by the function of a complex variable, $w = f(z) = z^n$. This maps the circle, $z = re^{i\theta}, 0 \leq \theta < 2\pi$, around the point $w = 0$ a total of n times. The terminology comes by analogy with the principle of the argument in complex analysis, which gives the number of zeros minus the number of poles (of a meromorphic function) surrounded by a contour. This number is the number of times the image of the contour winds around the origin.

A winding number can be associated with a set of domains in two-dimensions (see fig. F-2). It is the net number of turns made by following all the boundaries (taken with the same orientation) of all the domains. The orientation of a boundary is determined by whether the interior of a domain is to the right or to the left of the boundary as it is traversed. This number is the number of domains minus the number of holes in the domains and can be shown to be the same as the Euler number of the set. See [36] for a more complete mathematical treatment of these topics.

In the discrete case (as for CA), the plane is tiled with cells. A connected group of occupied cells (or tiles) constitutes a domain. If the sides of the tiles are straight, the boundaries of the domains only turn where three or more cells meet at a point (the set of all such points form the vertex set of the lattice *dual* to the original lattice). The net number of turns is simply the sum of these angles ($\times$cells/360°) taken over all the cells of the dual lattice. Hence it is possible to define a density whose integral gives the winding number. This connection between a topological number and a differential quantity is surprising and important.

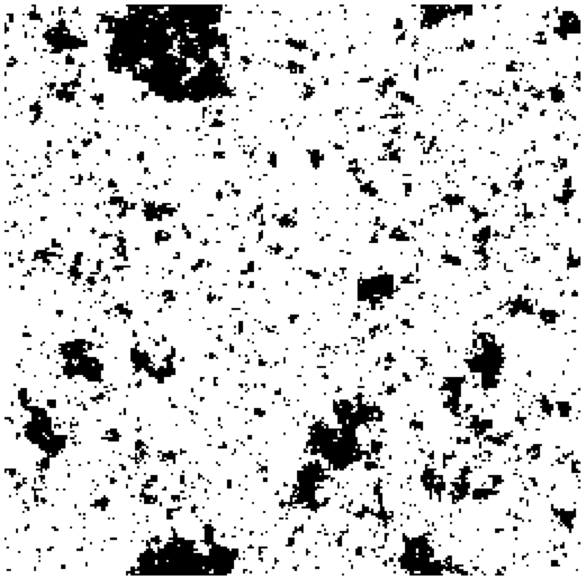


Figure F-2: Magnetic domains (in black) taken from an Ising model with 64K spins. Each black cell can be thought of as a particle. The winding number is 1357, which is somewhat less than the actual number of domains. The average number of particles per domain is approximately 7.9.

The *winding number density* on the dual lattice is defined to be zero unless there is a boundary passing though the vertex. In this case, it is 180° minus the angle subtended by the cells constituting the interior of the cluster. This is just the angle that the boundary bends through. On a square lattice, the winding number density is assigned as shown in fig. F-3. Note that this quantity depends on correlations in a group of cells, and that the groups overlap.

The winding number density can be taken as a many-body potential in a deterministic Ising model. The potential can thus be used to bias the curvature of the phase boundary in models of droplet growth. The winding number then provides an estimate of the number of droplets. In the low-density limit, the domains become simply connected (no holes), and the correspondence between winding number and the number of domains becomes exact. In the case of droplet growth, it can be used to find an estimate of the average droplet size as a function of time.

In order to find the average domain size, one needs to count the number of occurrences of cases (b), (c), and (e) in fig. F-3 as well as the total number of occupied

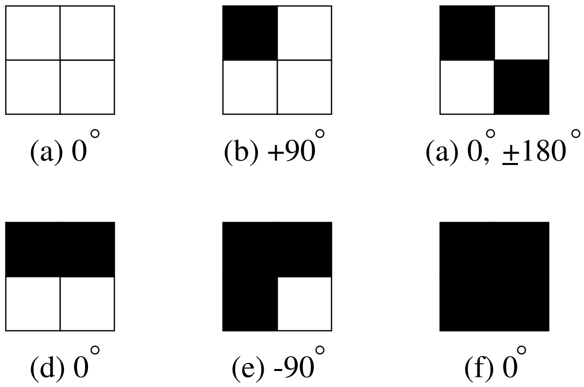


Figure F-3: Contributions to the winding number from various boundary configurations.

cells. Fig. F-2 has 7380 90° angles, 2310 −90° angles, 179 180° angles, and a total of 10,749 particles. The built-in counter in CAM-6 makes doing these counts over the 64K cells very fast. Finding and tracing out domains one by one requires a relatively slow, complicated procedure, whereas finding the average domain size on CAM-6 only takes four steps or $\frac{1}{15}$ of a second; hence, this is a practical method to use while doing real-time simulations.

Another form of winding number is illustrated by fig. F-1. The winding number of a closed loop of cells is the net number of times the cell states go through the cycle as the loop is traversed in a counterclockwise direction. Any loop with a nonzero winding number is said to contain a defect. The defects have a topological character because the winding number of a loop cannot be changed by modifying the state of one cell at a time while still maintaining continuity. The centers of the spirals are the topological defects.

Analogous topological objects are important in several disciplines. The mathematical study of these objects falls under field of algebraic topology. Their physical relevance stems from the fact that, depending on the dynamical laws and the overall topology of spacetime, most fields can form "knots"—continuity and energy conservation forbid such configurations of fields from being untied. Two physical instances of topological charges which have been postulated to exist (but not definitely observed) are magnetic monopoles and cosmic strings.

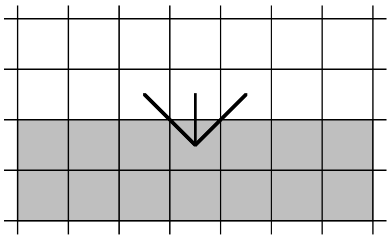
F.3.3 Perimeter

The circumference presents more of a problem since the boundary has jagged edges whose lengths persist in the continuum limit. The coarsest approximation would be to just count the number of cells which have any nearest-neighbor boundary. However, in the case of a diagonal ("staircase") boundary, this procedure would give an answer which is too low by a factor of $\sqrt{2}$. A more detailed approximation would be to count the number of cell boundaries themselves, but again, in the case of a diagonal boundary, this would give an answer which is too high by a factor of $\sqrt{2}$.

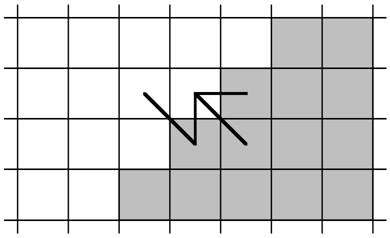
Clearly, we would like to have something in between–in particular, the length of a smooth curve which encloses a similar area and which is best approximated by the jagged boundary. The approach taken here is a third approximation which takes into account the number of next-nearest-neighbor (diagonal) boundaries as well. To obtain the length as a linear function of the counts, they will have to be weighted appropriately as shown below.

Assume we have counted the number of nearest- and next-nearest-neighbor boundaries for a given domain. Call these counts A and B respectively, and call the associated weights a and b. Thus, $l = aA + bB$. The weights will be determined by looking at two representative cases: horizontal (or vertical) boundaries, and diagonal boundaries.

A single cell in a horizontal boundary has one nearest-neighbor boundary and two next-nearest-neighbor boundaries. The neighbor relation can be indicated by line segments (or "bonds") which connect pairs of cells, one inside and one outside the domain:

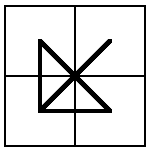


In this case, we want the contribution to the length to be unity, i.e., $1 = a + 2b$. Similarly, the basic unit along a diagonal has two nearest-neighbor boundaries and two next-nearest-neighbor boundaries:



In this case, we want the contribution to the length to be $\sqrt{2}$, i.e., $\sqrt{2} = 2a + 2b$. Solving the two simultaneous equations gives $a = \sqrt{2} - 1$ and $b = 1 - 1/\sqrt{2}$.

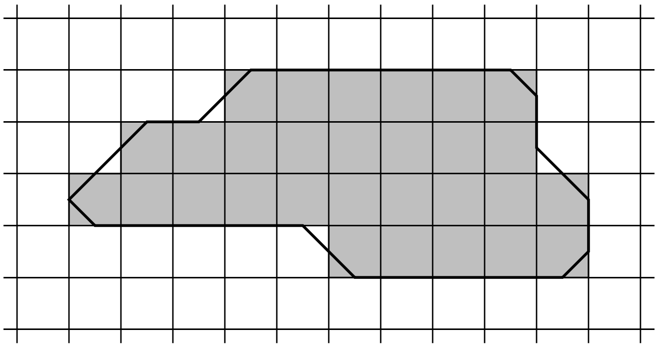
The boundary counts are local in the sense that they can be determined by looking at 2x2 "windows" of cells. Consider a block of four such cells, where some of the cells may be inside a given domain, and some may be outside. Each pair of cells has the potential to be a boundary, either horizontal and vertical (nearest-neighbor boundaries) or diagonal (next-nearest-neighbor boundaries). If the numbers in a pair of cells *differ*, then each of the respective numbered domains has a corresponding boundary. By looking at all possible 2x2 blocks, we will encompass all pairs of cells. In order to avoid double counting of bonds, we only look at four of the six possible pairs within a block as shown:



Summing over all possible 2x2 blocks gives the total number of bonds of each type. Furthermore, the counts for each domain can by tallied at the same time by using the numbers in the cells as counter indices: each bond contributes to the two domains defined by the two numbers in the pair of cells.

With the above weighting scheme, a single number, or measure, representing length can be assigned to each possible boundary configuration of a block. This assignment of a number to each block can be thought of as a differential form for arc length, and the integral of this form gives the total length of the boundary.

This is all well and good if all we have is one of the eight types of straight boundaries, but what about turning corners? We will see that the above procedure still gives very reasonable results. For example, the following shape



has 28 nearest-neighbor bonds and 44 next-nearest-neighbor bonds, and the formula gives $16+6\sqrt{2}$. This is exactly the same as the perimeter of its smoothed counterpart defined by the surrounding solid line. The smoothing is done by cutting corners defined by the midpoints of the sides of the cells which jut out or angle in.

In general, given any simply connected domain of cells, one can construct a new domain by cutting off convex corners and filling in concave corners. The formula applied to the original shape gives identically the perimeter of the new shape. This can be proven inductively by building up the domains one square at a time along nearest-neighbor boundaries, while maintaining simple connectivity.

This new shape will have a slightly smaller area than the original domain. This is a topological correction due to the fact that the boundary forms a closed curve. Since the boundary of the domain turns through a full 360°, there will be four more corners cut off than filled in. The area of the four corners is exactly one half of a cell. Hence the rule for finding area given above could be modified to subtract 1/2 from the count of cells in the domain to make a more consistent scheme overall.

The current scheme does not give the right length for boundaries at arbitrary angles. In fact, it will be proved below that no purely local cellular integration scheme can yield the continuum limit. However, better and better approximations can be developed by looking at larger and larger windows. The idea is that a better fit to a smooth curve can be made with more information. The situation is analogous to that in finite-difference methods for integration (and differentiation), where better approximations are given by looking at larger and larger templates. The topological correction mentioned above is similar to the end point corrections of numerical integration.

In light of this limitation, we can make an interesting observation in regard to the proper generalization of a circle. Note that all the sides of the new shape are always at one of eight angles, and in the continuum limit, we can talk about all polygons constructed with sides at these angles. Now, what is the largest area of such a polygon one can have with a given perimeter? First, it must be convex, since the area can be increased by folding out any concavities. Second, it must be an octagon, since after eight (convex) 45 degree turns, the boundary must close on itself. Finally, all the sides must be the same length because the perimeter could be reduced by shaving area from a shorter side and putting it on a longer side. Thus, the shape with the maximum area for a given perimeter (or equivalently, the minimum perimeter for a given area) is a regular octagon. A dimensionless way to quantify deviations in the boundary of a shape is to calculate the ratio of the perimeter squared to the area. For a regular octagon, this ratio is $32(\sqrt{2}-1) \cong 13.25$, while in comparison, for a circle, it is $4\pi \cong 12.57$. These are optimal values in the Cartesian and continuum cases respectively.

Now we will prove a basic limitation of lattice approximations to continuum mathematics:

Theorem: No cellular integration scheme which looks at a fixed window of cells can determine the length of smooth contours in all cases.

Proof: Let the integration window have a diameter d, and assume that we can recover the continuum limit as the subdivision of cells becomes finer and finer. Consider a straight line with a small slope, $0 < s < 1/d$, extending over $D \gg d$ cells in the x direction. The length of this curve must approach $D\sqrt{1+s^2}$ in the continuum limit. The cellular approximation to this curve will be a staircase with Ds unit steps separated by at least d cells. For the flat segments in between steps, the only plausible value of the integral is the length of that segment. Now, the integration window will encompass only one step at a time, and by linearity, each step will contribute some constant, α, independent of s, above and beyond the horizontal length of the step.

Hence the value of the integral must be $D(1 + \alpha s)$. As $D \to \infty$, this will have the right limit if and only if $(1 + \alpha s) = \sqrt{1 + s^2}$. However this is not identically true for all s which contradicts our hypothesis.

About the best we can do is to make $\alpha = (\sqrt{2} - 1)/d$, so that the maximum relative error is approximately $(3\sqrt{2} - 2)/(2d^2)$. This error is small, and of course, it could be made arbitrarily small by increasing the size of the *window* without bound. However, the above specification of a cellular differential form does not provide an unambiguous recipe for extending the procedure to larger windows.

Bibliography

[1] Milton Abramowitz and Irene A. Stegun, editors. *Handbook of Mathematical Functions*. Dover, New York, December 1972.

[2] Y. Bar-Yam, Y. Rabin, and M. A. Smith. Cellular automaton approach to polymer simulations. *Macromolecules Reports*, 25:2985–2986, 1992.

[3] Charles H. Bennett and G. Grinstein. Role of irreversibility in stabilizing complex and nonergodic behavior in locally interacting discrete systems. *Physical Review Letters*, 55(7):657–660, August 1985.

[4] Elwyn R. Berlekamp, John H. Conway, and Richard K. Guy. *Winning Ways for Your Mathematical Plays*, volume 2, chapter 25. Academic Press, New York, 1982.

[5] Kurt Binder, editor. *Monte Carlo Methods in Statistical Physics*, volume 7 of *Topics in current physics*. Springer-Verlag, Berlin, second edition, 1986.

[6] Kurt Binder, editor. *Applications of Monte Carlo Methods in Statistical Physics*, volume 36 of *Topics in current physics*. Springer-Verlag, Berlin, second edition, 1987.

[7] Richard L. Bishop and Samuel I. Goldberg. *Tensor Analysis on Manifolds*. Dover, New York, 1980.

[8] Bruce M. Boghosian. Computational physics on the connection machine. *Computers in Physics*, 4(1):14–33, Jan/Feb 1990.

[9] Authur Walter Burks, editor. *Essays on Cellular Automata.* University of Illinois Press, Urbana, Illinois, 1970.

[10] Andrea Califano, Norman Margolus, and Tommaso Toffoli. CAM-6: A High-Performance Cellular Automata Machine *User's Guide*.

[11] I. Carmesin and Kurt Kremer. The bond fluctuation method: A new effective algorithm for the dynamics of polymers in all spatial dimensions. *Macromolecules*, 21(9):2819–2823, 1988.

[12] I. Carmesin and Kurt Kremer. Static and dynamic properties of two-dimensional polymer melts. *Journal de Physique*, 51(10):915–932, Mai 1990.

[13] Hue Sun Chan and Ken A. Dill. The protein folding problem. *Physics Today*, 46(2):24–32, February 1993.

[14] Bastien Chopard. A cellular automata model of large scale moving objects. *Journal of Physics A*, 23:1671–1687, 1990.

[15] Bastien Chopard. Strings: A cellular automata model of moving objects. In Pierre Manneville, Nino Boccara, Gérard Y. Vichniac, and Roger Bidaux, editors, *Cellular Automata and Modeling of Complex Physical Systems*, volume 46 of *Springer proceedings in physics*, pages 246–256, Berlin, 1990. Centre de physique des Houches, Springer-Verlag. Proceedings of the winter school, held in February, 1989.

[16] Yvonne Choquet-Bruhat, Cécile DeWitt-Morette, and Margaret Dillard-Bleick. *Analysis, Manifolds and Physics.* North-Holland, Amsterdam, 1982.

[17] J. Cossar and A. Erdélyi. *Dictionary of Laplace Transforms.* The Admiralty, London, 1944. Ref. No. SRE/ACS.53.

[18] J. Theodore Cox and David Griffeath. Recent results for the stepping stone model. In Harry Kesten, editor, *Percolation Theory and Ergodic Theory of Infinite Particle Systems*, volume 8 of *The IMA Volumes in Mathematics and its Applications*, pages 73–83. Springer-Verlag, Berlin, 1987.

[19] Michael Creutz. Microcanonical Monte Carlo simulation. *Physical Review Letters*, 50(19):1411–1414, 1983.

[20] Michael Creutz. Deterministic ising dynamics. *Annals of Physics*, 167:62–72, 1986.

[21] Pierre-Gilles de Gennes. *Scaling Concepts in Polymer Physics*. Cornell University Press, Ithaca, NY, 1979.

[22] Richard Earl Dickerson and Irving Geis. *The Structure and Action of Proteins*. Harper & Row, New York, 1969.

[23] M. Doi and S. F. Edwards. *The Theory of Polymer Dynamics*, volume 73 of *International series of monographs on physics*. Clarendon Press, Oxford, 1986.

[24] Gary D. Doolen, editor. *Lattice Gas Methods for Partial Differential Equations*, volume 4 of *Santa Fe Institute Studies in the Sciences of Complexity*. Addison-Wesley, Redwood City, California, 1990. Workshop on Large Nonlinear Systems held at Los Alamos National Laboratory, Los Alamos, New Mexico, August 1987.

[25] Gary D. Doolen, editor. *Lattice Gas Methods for PDE's: Theory, Applications and Hardware*. MIT Press, Cambridge, Massachusetts, 1991. NATO Advanced Research Workshop held at the Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, New Mexico, September 6–8, 1989.

[26] J. M. Duetsch. Dynamic Monte Carlo simulation of an entangled many-polymer system. *Physical Review Letters*, 49(13):926–929, 1982.

[27] Doyne Farmer, Tommaso Toffoli, and Stephen Wolfram, editors. *Cellular Automata*. North-Holland, Amsterdam, 1984. Proceedings of an Interdisciplinary Workshop held at Los Alamos National Laboratory, Los Alamos, New Mexico, March 7–11, 1983.

[28] I. E. Farquhar. *Ergodic Theory in Statistical Mechanics*, volume 7 of *Monographs in Statistical Physics and Thermodynamics*. Interscience Publishers, London, 1964.

[29] Paul J. Flory. *Principles of Polymer Chemistry*. The George Fisher Baker non-resident lectureship in chemistry at Cornell University. Cornell University Press, Ithaca, NY, 1953.

[30] Geoffrey C. Fox and Steve W. Otto. Algorithms for concurrent processors. *Physics Today*, pages 50–59, May 1984.

[31] Hans Frauenfelder and Peter G. Wolynes. Biomolecules: Where the physics of complexity and simplicity meet. *Physics Today*, 47(2):58–64, February 1994.

[32] Uriel Frisch, Brosl Hasslacher, and Yves Pomeau. Lattice-gas automata for the navier-stokes equation. *Physical Review Letters*, 56(16):1505–1508, April 1986.

[33] Martin Gardner. The fantastic combinations of john conway's new solitaire game "life". *Scientific American*, 223(4):120–123, October 1970.

[34] Martin Gardner. On cellular automata, self-reproduction, the garden of eden and the game of "life". *Scientific American*, 224(2):112–115, February 1971.

[35] S. Goldstein. On diffusion by discontinuous movements, and on the telegraph equation. *Quarterly Journal of Mechanics and Applied Mathematics*, IV(2):129–156, 1951.

[36] Stephen B. Gray. Local properties of binary images in two dimensions. *IEEE Transactions on Computers*, C-20(5):551–561, May 1971.

[37] Gary S. Grest, Kurt Kremer, S. T. Milner, and T. A. Witten. Relaxation of self-entangled many-arm star polymers. *Macromolecules*, 22(4):1904–1910, 1989.

[38] Howard Gutowitz, editor. *Cellular Automata: Theory and Experiment.* MIT Press, Cambridge, Massachusetts, 1991. Proceeding of the Conference on Cellular Automata held at the Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, New Mexico, September 9–12, 1989.

[39] Brosl Hasslacher. Discrete fluids. *Los Alamos Science*, Special Issue (15):175–217, 1987.

[40] Michael G. Hluchyj, editor. *IEEE Infocom: The Conference on Computer Communications*, Los Alamitos, California, March 1993. Institute of Electrical and Electronics Engineers, IEEE Computer Society Press. Twelfth Annual Joint Conference of the IEEE Computer and Communications Societies, Networking: Foundation for the Future.

[41] John H. Holland. *Adaptation in Natural and Artifical Systems: an Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence.* Complex Adaptive Systems. MIT Press, Cambridge, Massachusetts, first edition, 1992.

[42] Hrvoje Hrgovčić. personal communication, 1988.

[43] John David Jackson. *Classical Electrodynamics.* Wiley, New York, second edition, 1975.

[44] Edwin T. Jaynes. Information theory and statistical mechanics. In Kenneth W. Ford, editor, *Statistical Physics*, volume 3 of *Brandeis Lectures in Theoretical Physics*, pages 181–218. W. A. Benjamin, New York, 1963. Lectures of the 1962 Summer Institute.

[45] Edwin T. Jaynes. *E. T. Jaynes: Papers on Probability, Statistics, and Statistical Physics.* D. Reidel, Dordrecht, Holland, 1983.

[46] Leo P. Kadanoff and Jack Swift. Transport coefficients near the critical point: A master-equation approach. *Physical Review*, 165(1):310–322, 1968.

[47] Amnon Katz. *Principles of Statistical Mechanics: The Information Theory Approach.* W. H. Freeman, San Francisco, 1967.

[48] Stuart A. Kaufmann. *The Origins of Order: Self-Organization and Selection in Evolution.* Oxford University Press, Oxford, 1993.

[49] Don C. Kelly. Diffusion: A relativistic appraisal. *American Journal of Physics*, 36(7):585–591, July 1968.

[50] Motoo Kimura and George H. Weiss. The stepping stone model of population structure and the decrease of genetic correlation with distance. *Genetics*, 49:561–576, April 1964.

[51] Kurt Kremer and Gary S. Grest. Dynamics of entangled linear polymer melts: A molecular dynamics simulation. *Journal of Chemical Physics*, 92(8):5057–5086, April 1990.

[52] Christopher G. Langton, editor. *Artificial Life*, volume 6 of *Santa Fe Institute Studies in the Sciences of Complexity.* Addison-Wesley, Reading, Massachusetts, 1989. Proceedings of the interdisciplinary workshop on the synthesis and simulation of living systems, held September, 1987.

[53] Christopher G. Langton, Charles Taylor, J. Doyne Farmer, and Steen Rasmussen, editors. *Artificial Life II*, volume 10 of *Santa Fe Institute Studies in the Sciences of Complexity.* Addison-Wesley, Reading, Massachusetts, 1991. Proceedings of the 2nd interdisciplinary workshop on the synthesis and simulation of living systems, held February, 1990.

[54] Elliott H. Lieb and Daniel C. Mattis, editors. *Mathematical Physics in One Dimension: Exactly Soluble Models of Interacting Particles.* Perspectives in physics. Academic Press, New York, 1966.

[55] Norman Margolus, Tommaso Toffoli, and Gérard Vichniac. Cellular-automata supercomputers for fluid-dynamics modeling. *Physical Review Letters*, 56(16):1694–1696, April 1986.

[56] Norman H. Margolus. Physics-like models of computation. *Physica D*, 10:81–95, 1984.

[57] Norman H. Margolus. personal communication, 1986.

[58] Norman H. Margolus. *Physics and Computation.* PhD thesis, Massachusetts Institute of Technology, May 1987.

[59] Norman H. Margolus et al. CAM-6 software. Systems Concepts, San Francisco, CA, 1987. ©Massachusetts Institute of Technology.

[60] Norman H. Margolus et al. CAM-PC software. Automatrix, Rexford, NY, 1991. ©Massachusetts Institute of Technology.

[61] J. Andrew McCammon and Stephen C. Harvey. *Dynamics of Proteins and Nucleic Acids.* Cambridge University Press, Cambridge, 1987.

[62] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculations by fast computing machines. *Journal of Chemical Physics*, 21(6):1087–1092, June 1953.

[63] Michael Murat and Gary S. Grest. Structure of a grafted polymer brush: A molecular dynamics simulation. *Macromolecules*, 22(10):4054–4059, 1989.

[64] Edward Nelson. Derivation of the schrödinger equation from newtonian mechanics. *Physical Review*, 150(4):1079–1085, October 1966.

[65] G. Nicolis and Ilya Prigogine. *Self-Organization in Nonequilibrium Systems: From Dissipative Structures to Order through Fluctuations.* Wiley, New York, 1977.

[66] Gregoire Nicolis and Ilya Prigogine. *Exploring Complexity: An Introduction.* W. H. Freeman, New York, 1989.

[67] B. Ostrovsky, M. A. Smith, M. Biafore, Y. Bar-Yam, Y. Rabin, and and T. Toffoli N. Margolus. Massively parallel architectures and polymer simulation. In

Richard F. Sincovec, David E. Keyes, Michael R. Leuze, Linda R. Petzold, and Daniel A. Reed, editors, *Proceedings of the Sixth SIAM Conference on Parallel Processing for Scientific Computing, vol. 1*, pages 193–202, Philadelphia, 1993. Society for Industrial and Applied Mathematics. Held in Norfolk, VA, March 22–24, 1993.

[68] Boris Ostrovsky and Yaneer Bar-Yam. Irreversible polymer collapse by Monte Carlo simulations. *Computational Polymer Science*, 3(1&2):9–13, 1993.

[69] Ilya Prigogine. *Introduction to Thermodynamics of Irreversible Processes.* Interscience Publishers, New York, third edition, 1967.

[70] Ilya Prigogine. *From Being to Becoming.* W. H. Freeman and Company, San Francisco, 1980.

[71] Yitzhak Rabin. personal communication.

[72] Yitzhak Rabin, Boris Ostrovsky, Mark A. Smith, and Yaneer Bar-Yam. Interpenetration in polymer solutions in two dimensions. In preparation, 1993.

[73] Alastair Michael Rucklidge. Phase transitions in cellular automata mixtures. Master's thesis, Massachusetts Institute of Technology, February 1988.

[74] David C. Schwartz and Charles R. Cantor. Separation of yeast chromosome-sized dnas by pulsed field gradient gel electrophoresis. *Cell*, 37:67–75, May 1984.

[75] Claude E. Shannon and Warren Weaver. *The Mathematical Theory of Information.* University of Illinois Press, Urbana, Illinois, 1949.

[76] Cassandra L. Smith. Separation and analysis of DNA by electrophoresis. *Current Opinion in Biotechnology*, 2:86–91, 1991.

[77] M. A. Smith and Y. Bar-Yam. Pulsed field gel electrophoresis simulations in the diffusive regime. In *Proceedings of the 2nd International Conference on*

Bioinformatics, Supercomputing and Complex Genome Analysis, 1992. Held in St. Petersburg, FL, June 4–7, 1992.

[78] M. A. Smith and Y. Bar-Yam. Cellular automaton simulation of pulsed field gel electrophoresis. *Electrophoresis*, 14:337–343, 1993.

[79] M. A. Smith, Y. Bar-Yam, Y. Rabin, N. Margolus, T. Toffoli, and C.H. Bennett. Cellular automaton simulation of polymers. In Eric B. Sirota, David Weitz, Tom Witten, and Jacob Israelachvili, editors, *Complex Fluids*, volume 248, pages 483–488, Pittsburgh, Pennsylvania, 1992. Materials Research Society, Materials Research Society symposium proceedings. Held in Boston, MA, December 2–6, 1991.

[80] M. A. Smith, Y. Bar-Yam, Y. Rabin, B. Ostrovsky, C. H. Bennett, N. Margolus, and T. Toffoli. Parallel-processing simulation of polymers. *Computational Polymer Science*, 2(4):165–171, 1992.

[81] Mark A. Smith. Representations of geometrical and topological quantities in cellular automata. *Physica D*, 45:271–277, 1990.

[82] Sun and Dill. Protein folding. *Physics Today*, 46(2):24–32, March 1994.

[83] Systems Concepts, San Francisco, California. CAM-6: A High-Performance Cellular Automata Machine *Hardware Manual*, 1987.

[84] Tommaso Toffoli. *Cellular Automata Mechanics*. PhD thesis, University of Michigan, November 1977. Technical Report No. 208.

[85] Tommaso Toffoli. Cellular automata as an alternative to (rather than an approximation of) differential equations in modeling physics. *Physica D*, 10:117–127, 1984.

[86] Tommaso Toffoli. Four topics in lattice gases: Ergodicity; relativity; information flow; and rule compression for parallel lattice-gas machines. In *Discrete Kinetic Theory, Lattice Gas Dynamics and Foundations of Hydrodynamics*, Turin, Italy, September 1988. Institute for Scientific Interchange.

[87] Tommaso Toffoli. Information transport obeying the continuity equation. *IBM Journal of Research and Development*, 32(1):29–36, January 1988.

[88] Tommaso Toffoli. How cheap can mechanics' first principles be? In Wojciech H. Zurek, editor, *Complexity, Entropy, and the Physics of Information*, volume 8 of *Santa Fe Institute Studies in the Sciences of Complexity*, pages 301–318, Redwood City, California, 1990. Santa Fe Institute, Addison-Wesley, The Advanced Book Program. Workshop on Complexity, Entropy, and the Physics of Information, held May-June 1989.

[89] Tommaso Toffoli and Norman Margolus. *Cellular Automata Machines: A New Environment for Modeling.* MIT Press Series in Scientific Computation. MIT Press, Cambridge, Massachusetts, 1987.

[90] Tommaso Toffoli and Norman H. Margolus. Invertible cellular automata: A review. *Physica D*, 45:229–253, 1990.

[91] Tommaso Toffoli and Norman H. Margolus. Programmable matter. *Physica D*, 47:263–272, 1991.

[92] Gérard Y. Vichniac. Instability in discrete algorithms and exact reversibility. *SIAM Journal on Algebraic and Discrete Methods*, 5(4):596–602, December 1984.

[93] Gérard Y. Vichniac. Simulating physics with cellular automata. *Physica D*, 10:96–116, 1984.

[94] John von Neumann. *Theory of Self-Reproducing Automata.* University of Illinois Press, Urbana, Illinois, 1966. Edited and completed by Authur W. Burks.

[95] Klaus Voss. *Discrete Images, Objects, and Functions in* Z^n. Number 11 in Algorithms and Combinatorics. Springer-Verlag, Berlin, 1993.

[96] Robert M. Wald. *General Relativity.* University of Chicago Press, Chicago, 1984.

[97] M. Mitchel Waldrop. *Complexity: The Emerging Science at the Edge of Order and Chaos*. Simon and Schuster, New York, 1992.

[98] Hans-Peter Wittmann and Kurt Kremer. Vectorized version of the bond fluctuation method for lattice polymers. *Computer Physics Communications*, 61:309–330, 1990.

[99] Stephen Wolfram, editor. *Theory and Applications of Cellular Automata*, volume 1 of *World Scientific Advanced Series on Complex Systems*. World Scientific, Philadelphia, 1986.

[100] Steven Wolfram. Statistical mechanics of cellular automata. *Reviews of Modern Physics*, 55:601–644, 1983.

[101] Steven Wolfram. Universality and complexity in cellular automata. *Physica D*, 10:1–35, 1984.

[102] Jeffery Yepez. A lattice-gas with long-range interactions coupled to a heat bath. In A. Lawniczak and R. Kapral, editors, *Pattern Formation and Lattice-gas Automata*. American Mathematical Society, 1994.

[103] Dmitrii Nikolaevich Zubarev. *Nonequilibrium Statistical Thermodynamics*. Studies in Soviet Science. Consultants Bureau, New York, 1974. Translated from Russian by P. J. Shepherd. Edited by P. J. Shepherd and P. Gray.