Rob J. de Boer, Jan D. van der Laan, and Pauline Hogeweg
Bioinformatica RUU, Padualaan 8, 3584 CH Utrecht, The Netherlands

Randomness and Pattern Scale in the Immune Network: A Cellular Automata Approach

INTRODUCTION

The immune system is a beautiful example of a complex information processing system. The complexity of the immune system is comparable to that of the nervous system. Both systems are comprised of a large number of different cell types communicating via the production of stimulatory or inhibitory molecules. Several of these cell types form connected networks of billions of different nodes. In the neural network the nodes are fixed and communicate via electrical signals; in the immune system the nodes recirculate and communicate via molecules (e.g., antibody) or cell-to-cell contacts. An important property of both systems is "learning." During its early life the immune system learns to discriminate between self and nonself. Additionally, the immune system has a form mei iory which is known as "immunity": secondary immune responses are usually different from primary responses.

Immune network theory[21] postulates that the functioning of the immune system is based upon network interactions. The assumption is that the lymphocyte clones comprising the immune system are capable of activating and/or inhibiting each other. Interactions between lymphocytes are based upon the binding of their receptors. Lymphocytes are recirculating blood cells, which are formed in the bone

Thinking About Biology, Eds. W. D. Stein and F. J. Varela, SFI Studies in the Sciences of Complexity, Lect. Note Vol. III, Addison-Wesley, 1993

marrow and which die in the periphery. The specificity of each lymphocyte clone is determined by its receptor molecule. Each clone is unique because each receptor has a unique variable region. Typical estimates for the number of different lymphocyte clones in mice and men are 10^6 and 10^9 different specificities. The collection of different specificities is usually called the immune repertoire.

A major difference between the immune system and the nervous system is the turnover of the nodes in the network. In a neural network the neurons typically live for years while they respond on a time scale of seconds. In the immune network, the lifetime and the response time of clones both have a time scale of days. The high turnover of network nodes in the immune system is a consequence of a huge production of novel clones in the bone marrow. The bone marrow production suffices to replace the entire immune system in just a few days.[11]

Recent views of the immune network posit that only part of the immune system is involved in network interactions.[1,4,19,34,35] These authors argue that the immune network is little involved with immune reactions to foreign antigens. Rather the network is autonomously active and/or responds to the environment of self-antigens. Further, these authors argue that the network is most pronounced during early life. This early immune network seems to play a role in the selection of the immune repertoire.[22] One possible mechanism of selection is that the activation of lymphocytes by network interactions precludes the death of the lymphocytes thus allowing the clone to be maintained in the immune repertoire. In this paper we study this hypothesis and show that such an immune repertoire self-organizes into clusters of activated and suppressed specificities. The size of these clusters turns out to depend on the bone marrow production. Thus, the main interest of the present paper is the *scale* of the patterns in the immune repertoire.

CELLULAR AUTOMATA

The immune network model that we develop below is a cellular automaton (CA). CAs were introduced in the late forties[32] as a general formalism for the study of complex systems (see Bak[2]). A CA consists of a collection of cells or automata that are ordered in a regular pattern, which is usually a rectangular grid (see Wolfram[38] or Toffoli and Margolus[31] for reviews). The neighborhood in a CA is usually formed by the four or eight cells surrounding each cell on the grid. Each automaton has a finite set of states and a next-state function that provides the next state as a function of the current state and the neighborhood. All cells behave according to the same rule and usually change to their next state synchronously. Because CAs are comprised of local automata in a spatial embedding, they have been viewed as "computing matter." They have been used to define a variety of enjoyable "artificial worlds."

A classic example of an artificial world CA is the game of "Life" by J. H. Conway.[2,12,13] Life has become famous because of its simple rules and complex behavior. In the game of Life, automata are on (i.e., "alive") or off (i.e., "dead"). At each time-step every automaton responds to the state of its local environment. Automata become alive whenever they have exactly three live neighbors. Automata remain alive when they have two or three live neighbors. Automata die when there are fewer or more neighbors alive. Thus, in this artificial world, being "overcrowded" or being too "lonely" leads to death. The most important result of the game of Life is that extremely simple rules may allow for an enormous complexity. In general, CAs have been classified into different categories on the basis of predictability[38] and complexity.[24] CAs like Life belong to the unpredictable classes and are capable of universal computation.[38]

In biology, CA models have been used to study spatial processes in general. Examples of this are reaction-diffusion-type systems,[3,14] ecological systems,[17] and immune systems.[18] The immune network CA that we develop below is based upon a rule similar to the game of Life: clones are maintained (i.e., are alive) when their stimulation is sufficient but not too high.

BITPLANE APPROACH

In our modeling approach we make use of the parallelism of the CA. The states of a binary CA, i.e., an automaton with black/white, on/off, or one/zero states, can be stored in bitplanes. This speeds up the simulations enormously because most computers have fast algorithms for manipulating bitplanes (bitplanes are used for writing the screen and for windows). A more detailed description of the bitplane approach can be found in Appendix I.

VOTING RULES

As an example illustrating the main principles of CAs, we discuss here majority voting. In a voting rule CA, every cell in the grid is a binary state automaton. For a majority vote, the next state of a cell is the state in which the majority of its neighbors and itself are in. Thus, the next state of each automaton is determined by a vector $\mathbf{I}$ of nine bits. If the sum of this vector is larger than four, i.e., if $\sum \mathbf{I} \geq 5$, the majority is black, and the next state will be black. Conversely, if $\sum \mathbf{I} < 5$, the majority is white, and the next state will be white. Starting with a random initial configuration, in which 50% of the cells is black and 50% is white (see Figure 1(a)), a voting rule CA yields a stable pattern of small white and black regions (see Figure 1(b)). This takes only a few generations (here 29). The most stable attractors of a voting rule CA are the two global patterns in which all cells have the same state (i.e., all white or all black). These attractors are never attained because the system gets stuck in smaller scale patterns of black and white regions. Locally, the boundaries between these areas are stable.

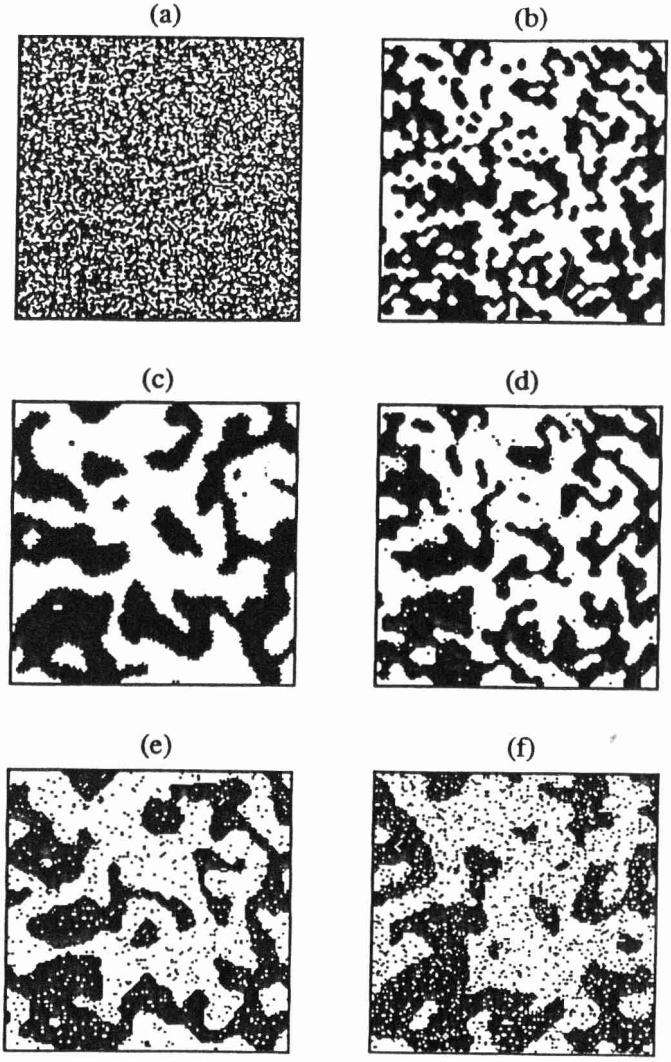


FIGURE 1 The effect of annealing or randomness on the scale of patterns in a CA with majority voting rules. This CA consists of 140 × 140 bits and is simulated using periodic boundary conditions. Thus it takes the shape of a torus where the upper and lower, and the left and right, boundaries are connected. (a) initial random distribution, (b) stable configuration attained after 29 majority voting steps, (c) snapshot after 50 voting steps with annealing, and (d–f) snapshots after 50 voting steps with 1, 5, and 10% random "errors."

The scale of these patterns can be increased by simulated annealing and/or by randomness. Simulated annealing is a general technique for avoiding local optima.[23] In the context of the CA, annealing introduces "errors" at the critical point of the next-state function.[36] Thus, if there is a profound majority, say, $\sum \mathbf{I} \geq 6$ or $\sum \mathbf{I} \leq 3$, the majority is chosen. However, if there is a majority of just one vote (i.e., $\sum \mathbf{I} = 4$ or $\sum \mathbf{I} = 5$), then the next state will be the minority. The increase in the scale of the pattern as a result of this type of annealing is shown in Figure 1(c): the black and white areas become much larger. In the long end, the system usually attains one of the two global attractors, i.e., only white or only black cells. Similar results can be obtained by introducing random "errors" in the transition rule. Figure 1(d–f) show an increase of the scale of the pattern if the minority is chosen with a low probability. Additionally, an increase in randomness increases the scale of the pattern: in Figure 1(d–f), the probability to follow the minority is 1, 2, and 4%, respectively.

The process of pattern formation by simulated annealing in voting rules is further illustrated in Figure 2. Small-scale structures can been seen to emerge from an initial, random distribution of black and white. As time proceeds the scale of these patterns increases. The pattern in Figure 2(i) largely consists of one big black island in a white sea (be reminded of the periodic boundary conditions). Because of its concave boundaries, this island will eventually disappear.

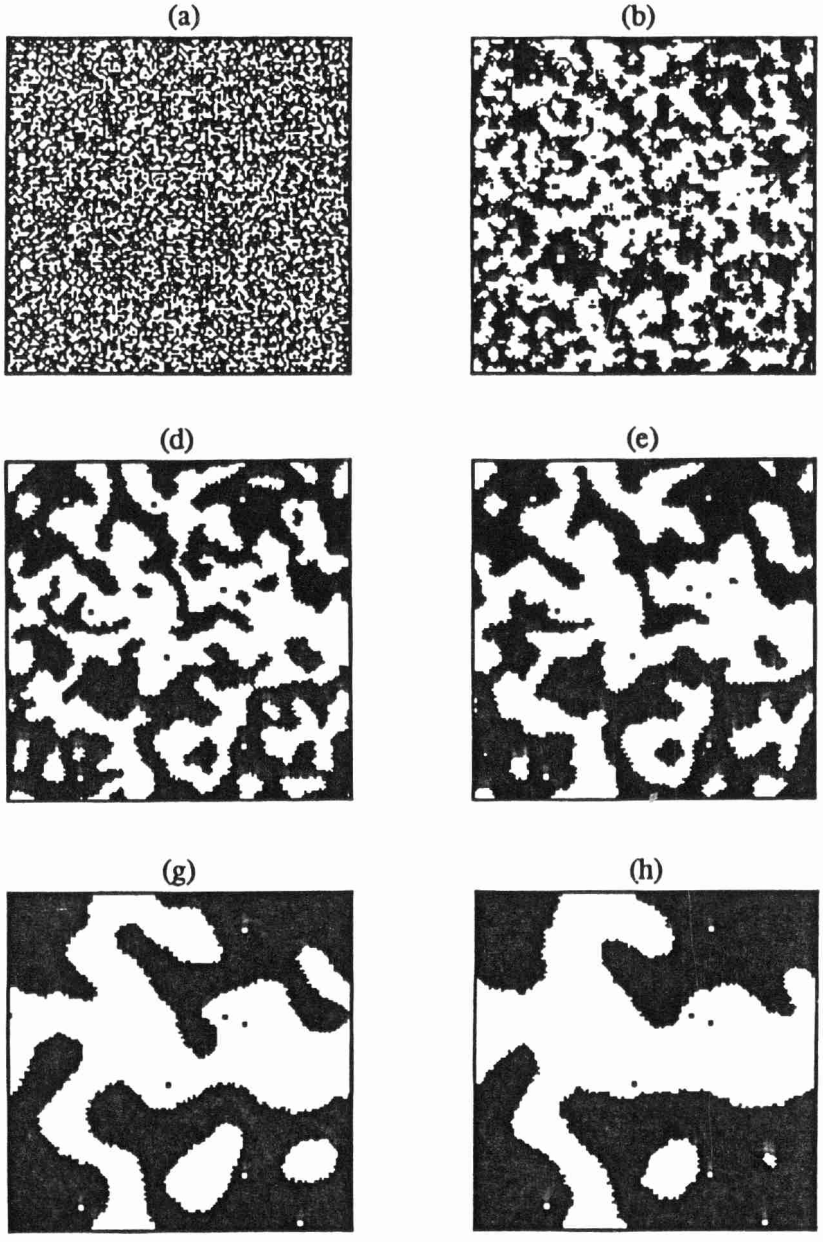
Annealing affects the boundaries between the black and white regions. Due to the annealing, curved boundaries become unstable and the system straightens its boundaries. This results in the erosion of "capes" and the filling of "bays." The increase of scale by introducing randomness in the voting rule CA can also be explained this way. Random errors will also make curved boundaries unstable.

Summarizing, the scale of the pattern increases both with the degree of randomness and with time.

IMMUNE NETWORKS

The enormous recruitment of novel clones from the bone marrow is one of the most salient characteristics of the immune system.[8,33] We have argued above that in the immune network, the decay and the growth of established clones, and the recruitment of novel clones, occur on the same time scale. In terms of modeling, this means that one cannot make an equilibrium assumption for either of the two processes, i.e., recruitment or growth, and that both have to be studied in combination.[8]

Thus, we have previously developed an asynchronous CA in which both processes were accounted for and could be scheduled at any time scale.[8] In this asynchronous CA, the scale of the pattern increases with the rate of random recruitment. Because of the similarity between this result and the effect of randomness


(a)
(b)
(d)
(e)
(g)
(h)

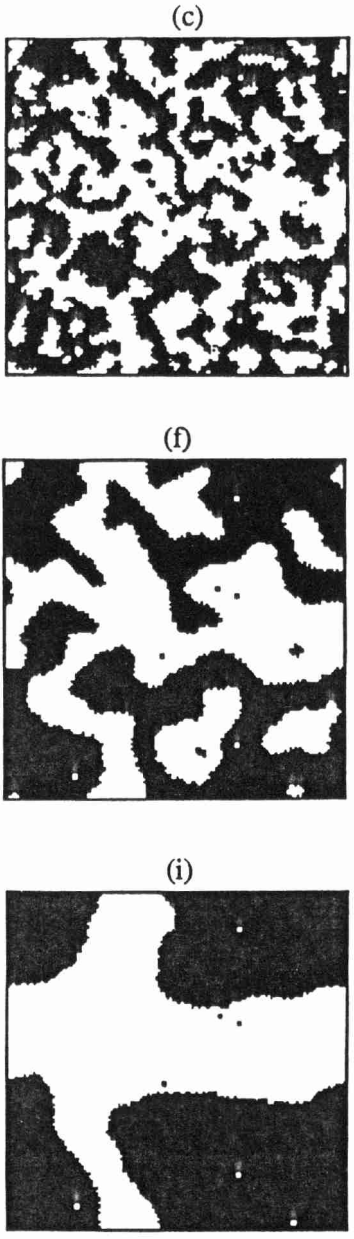


FIGURE 2 The evolution of voting rule patterns by annealing. Periodic boundary conditions were used. Starting with an initial random distribution (a), we perform majority voting with annealing, and make snapshots at the following time steps: $t = 5$ (b), $t = 10$ (c), $t = 20$ (d), $t = 40$ (e), $t = 80$ (f), $t = 160$ (g), $t = 320$ (h), and $t = 640$ (i).

in voting rule systems (see Figure 1), we here compare our immune network CA with voting rule systems. We simplify our previous CA by ignoring clonal growth and by making it parallel. Thus, for each clone, we only record its presence in the repertoire.

SHAPE SPACE

An elegant concept for the modeling of the immune network is the "shape space."[26,28,29] Because each lymphocyte carries a unique receptor molecule, each clone can be characterized by a unique shape. The degree of complementarity of the receptors determines the affinity of the binding interaction. Thus, in a shape space model, a vector $\mathbf{x}$ describes the shape of each receptor. In principle, $\mathbf{x}$ may account for several molecular characteristics such as the shape of protuberances, the hydrophobicity, and/or the charge. For reasons of simplification, one usually reduces $\mathbf{x}$ to an abstract low-dimensional shape variable. The first shape space model[28,29] was one-dimensional; later models[9] were two-dimensional.

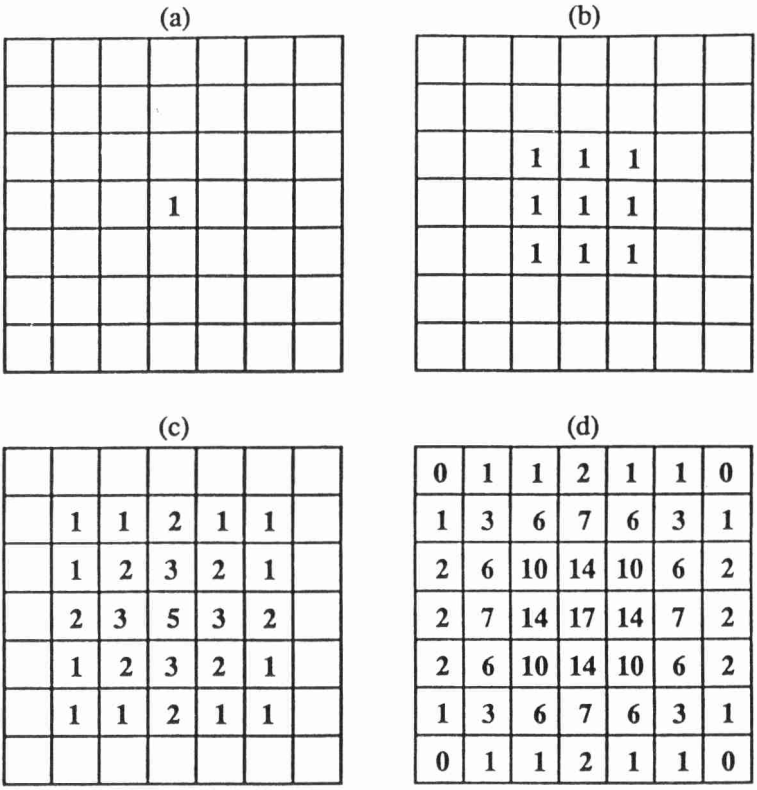
THE IMMUNE NETWORK CA

Here we study a two-dimensional shape space by means of a two-dimensional CA. We implement complementarity by considering two planes of complementary shapes, i.e., the "plus" plane with shapes $\mathbf{x}_+$ and the "minus" plane with shapes $\mathbf{x}_-$. A possible interpretation of the two planes would be to think of the plus planes as protuberances and the minus planes as clefts. The shape vector $\mathbf{x}_+$ could represent the height and the diameter of a protuberance whereas $\mathbf{x}_-$ could represent the depth and the diameter of a cleft. An alternative interpretation of the plus and the minus planes is as molecules with a positive or negative charge, respectively.

In the CA formalism this means that the neighborhood of a shape $\mathbf{x}_+$ corresponds to the local neighborhood of $\mathbf{x}_-$ in the minus plane. In our CA we use a weighted 7×7 neighborhood. The Figure 3 shows the neighborhood that a single cell evokes at the complementary plane. The perfect match, $\mathbf{x}_+ = \mathbf{x}_-$, is counted 17 times (see Appendix II). This neighborhood weighting is roughly Gaussian and is generated by just 15 parallel shifts in our bitplane approach (see Appendix II). Note that nine shifts are required for the standard 3×3 neighborhood (see Appendix I). The weighted sum of the neighborhood, which we call the "field," or h, is the total stimulation of each shape.

Figure 3 shows that $0 \leq h \leq 225$. The field h determines whether or not a clone will be maintained according to the window automaton proposed by Neumann and Weisbuch[25]

$$w(h) = \begin{cases} 1, & \text{if } \theta_1 \leq h \leq \theta_2; \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$



			1			

		1	1	1		
		1	1	1		
		1	1	1		

	1	1	2	1	1	
	1	2	3	2	1	
	2	3	5	3	2	
	1	2	3	2	1	
	1	1	2	1	1	

0	1	1	2	1	1	0
1	3	6	7	6	3	1
2	6	10	14	10	6	2
2	7	14	17	14	7	2
2	6	10	14	10	6	2
1	3	6	7	6	3	1
0	1	1	2	1	1	0

FIGURE 3 The weighted neighborhood of the immune network CA. We show how the neighborhood (d) is obtained from a single bit (a). First, the bit evokes a standard 3×3 Moore neighborhood (this requires eight parallel shifts). Second, the Moore neighborhood is shifted in the four diagonal directions. This gives a weighted 5×5 neighborhood (c). Third, the weighted neighborhood is shifted in the two horizontal and two vertical directions. This results in the final weighted 7×7 neighborhood (d).

The reason for choosing a window automaton is that experimental data and mathematical models of B lymphocyte activation suggest that the degree of activation is a log bell-shaped function of the stimulus.[27] Most recent immune network models are based upon a continuous log bell-shaped function. This class of models is the topic of several recent reviews.[6,10,34]

Finally we implement bone marrow recruitment. Recruitment is a random process which allows a fixed percentage P_R of all shapes to become present. The rate of recruitment, P_R, is varied throughout the analysis. The limits of the window $w(h)$ were fixed to $\theta_1 = 1$ and $\theta_2 = 80$. We have studied the CA for other values of the

thresholds and have always found similar patterns. The reason for choosing a low value for θ_1 is to allow for easy network activation. The large value of θ_2 reflects our intuition that clones can only be suppressed by the combined effect of several other clones.

The bitplane implementation of our immune network CA is explained in more detail in Appendix II. The CA consists of two shape planes of 140×140 bits. We here use fixed boundary conditions. This implies that there is no contribution to the field from outside the shape space.

Here, we summarize the steps as follows:

(1) Initialization: make 10% of the shape planes present.
(2) Neighborhood: calculate h, the sum of the weighted neighborhood.
(3) Recruitment: allow a percentage P_R of all shapes to become present.
(4) Selection: eliminate all shapes for which $w(h) = 0$.
(5) Time step: go to step two.

Since the recruitment makes shapes present only after the fields have been determined, this algorithm assumes that only those clones that have been maintained for at least one time step can contribute to the field. This is reasonable because it is unlikely that the small clones emerging from the bone marrow are immediately able to activate the other clones in the network.

RESULTS

Visualization of results is a notorious problem in network models. Conversely, in bitplane CAs this is straightforward because each plane of the CA can be presented by displaying the bitplane as a black and white pixel. Thus, two bitplanes of 140×140 pixels provide a complete description of the state of our CA. Further, by color coding, different bitplanes can be combined into one color display. We here use a color table of four colors. Red means present in the Shape_+ bitplane, green means present in the Shape_- bitplane, black means absent from both planes, white means present in both planes. Now, one 140×140 picture completely describes the state of the CA.

An example of one particular network after 400 time steps is shown in Plate 1(a) (see color insert for plates). The pattern consists of irregular red and green areas. The borderlines of the areas are densely populated. The red and green areas fit almost exactly into each other. A tiny empty region separates borderlines of different color. The red and green areas are almost exclusive: in panel (a) we find hardly any white areas corresponding to presence in both planes. Within each area the pattern is noisy having scattered red and green pixels. Apart from the noise, some small green clusters occur in the large red areas. Each of the green clusters is surrounded by a heavy red borderline. (The same is, of course, true for the green areas.) Because of their circular appearance we call these borderlines around the

small areas the "atolls." The pattern that we find, i.e., large areas dominated by one color and containing atolls of the same color, is of a much larger scale than the 7×7 pixel neighborhoods of our CA. Thus, the pattern that is formed cannot directly be derived from the local next-state function of the CA. Instead, the pattern emerges by self-organization.

The immunological interpretation of these patterns is that large areas in shape space are predisposed toward one class of antibodies (e.g., toward a cluster of positively charged molecules). Within such a cluster the other class of molecules is scattered at a low frequency.

The ability of a region to mount an immune response (to either self or nonself antigens) is defined by the field in that region. The absence of clones in a region may have two reasons: too little or too much stimulation. Thus, in Plate 2(a) we plot the field, h_-, of the Shape$_-$ plane. This corresponds to the field of the green pattern as it is generated by the red pattern. In Plate 2, red means that the field is suppressive, i.e., $h_- > \theta_2$, black means a lack of stimulation, i.e., $h_- < \theta_1$, and green stands for a stimulatory field, i.e., $w(h_-) = 1$. The suppressive (red) areas look like a blurred version of the red borderlines of Plate 1(a). Thus, the heavy borderlines evoke a suppressive field. The field is stimulatory (i.e., green in Plate 2(a)) in the majority of large green area of Plate 1(a). Thus, although the repertoire is noisy within each area, the field is stimulatory almost everywhere. Conversely, in the large areas that are mainly occupied by red, the field tends to be suppressive. This is only a tendency because several small spots of stimulatory fields exist in the red areas (compare Plate 1(a) with Plate 2(a)). The clones located in these small spots sustain the simulation of the noisy parts of the large areas. Areas with a lack of stimulation (i.e., black in Plate 2(a)) are rare. This implies that the network roughly covers the shape space.[7,9]

The distribution of the field strengths in Plate 2 allows us to investigate the hypothesis that only part of the immune system is involved in the network.[4,19,34,35] Thus, we consider the introduction of a foreign antigen at a random location in shape space. This area in shape space may or may not be under the influence of the network. Stimulation with any antigen corresponds to a local increase of the field. In this model an effective immune response to such an increase of the field requires that this area in shape space becomes more densely populated. Such a response is only possible if that area is not yet suppressed by the network. Thus, in Plate 2, an immune response can only come from a black or green area. A response from a black area, where $h_- < \theta_1$, would correspond to a classical immune response of that part of the system that is not functionally connected to the network. A response from a green area, where $\theta_1 \leq h_- \leq \theta_2$, would correspond to a reaction from the network. Because the black areas are rare, immune reactions from clones functionally connected to the network are expected to be most abundant in this model.

The stability of the immune repertoire, i.e., of the pattern shown in Plate 1(a), is studied by making time slices. These are shown in Plate 3(a). Each horizontal line in Plate 3(a) corresponds to a horizontal slice through the center of the shape

space of Plate 1(a). At every time step we obtain one line of pixels of Plate 3(a). In Plate 3 time increases from top to bottom. The panel is square because we only show the last 140 time steps of the simulation (i.e., the panel is 140×140 pixels). In such a time plot, any moving pattern, e.g., a so-called traveling wave, shows up as a diagonal line. Instead, steady patterns show up as vertical lines, and noise shows up as noise.

Thus, inspection of Plate 3(a) reveals that the pattern is stable, i.e., most lines are vertical. Only part of the pattern is noisy, most of the noise being confined to small regions within the large areas. The borderlines between the red and green areas are all stable.

RECRUITMENT

The effect of the rate of recruitment, i.e., of P_R, on the immune repertoire is shown in Plate 1. In each panel we show the repertoires as they are attained after 400 time steps. From panel (a) to (i), the rate of recruitment varies from $P_R = 1$ to 90. Visual inspection of the panels reveals the impact of the random recruitment on the scale of the pattern. As P_R increases, the areas grow in size and merge together to form connected patterns that span the entire shape space. For instance, in Plate 1(g) the shape space is largely one red cluster. Further, the patterns become smoother because the borderlines loose their irregular appearance. Within the areas we see more structure arising because the areas become covered with atolls.

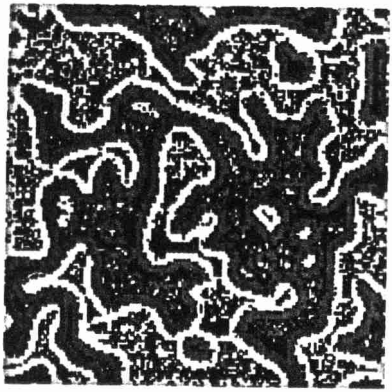
The rate of recruitment P_R not only influences the spatial scale of the pattern, it also influences its temporal scale. On Plate 3 the dynamic behavior of the shape space is shown as a function of P_R. When the rate of recruitment is small (e.g., panels (a)–(c), $P_R = 1, 2, 4$) vertical lines dominate. This indicates that the boundaries between the areas are stable. Within the areas the pattern is noisy. As P_R increases, i.e., $P_R \geq 8$, some boundaries become unstable and fewer vertical lines are observed. Note that the main reason for having fewer boundaries, either stable or unstable, is the increase in the scale of the pattern when P_R increases.

In the time series, the atolls appear as triangular or drop-shaped patterns with the top pointing downwards (e.g., panels (f) and (g)). This indicates that the atolls appear suddenly but are unstable, and decrease in size and disappear. The typical lifetime of an atoll is 15 to 30 time steps. High rates of recruitment, i.e., $P_R = 60 - 90$ (panels (g)–(i)), give rise to periodic behavior. In the time plots one observes alternating black and white lines. Thus, presence in both planes (i.e., white) oscillates with absence from both planes (i.e., black). The fact that clones from both planes are recruited in these spots means that the field is stimulatory for both planes. Such an overlap in the stimulatory fields is to be expected because we have observed above that, in the areas predisposed toward one of the planes, small stimulatory spots exist for the other plane.

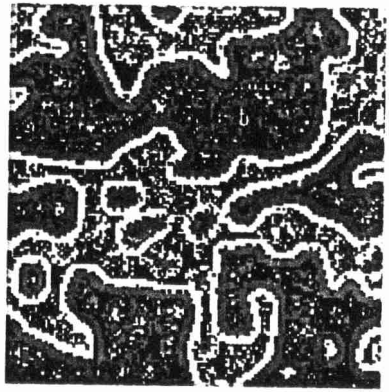
Because of the high rate of recruitment, clones from both planes invade such a stimulatory area at a high rate. Because novel (i.e., unstimulated) clones do

Color Plates

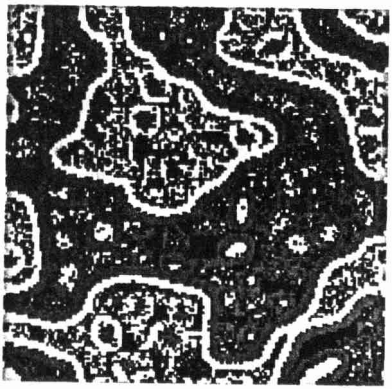
(a)



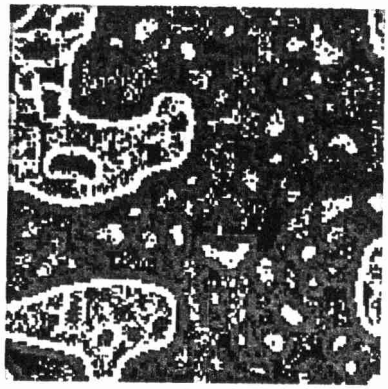
(b)



(d)



(e)

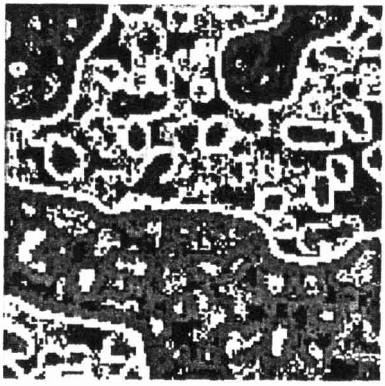


(g)

(h)

(c)

(f)

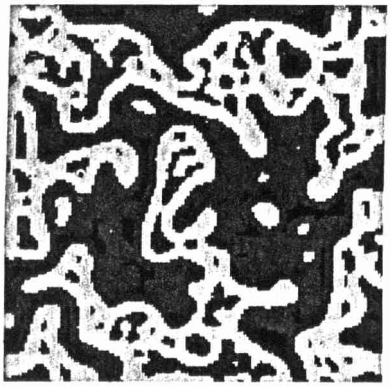


(i)

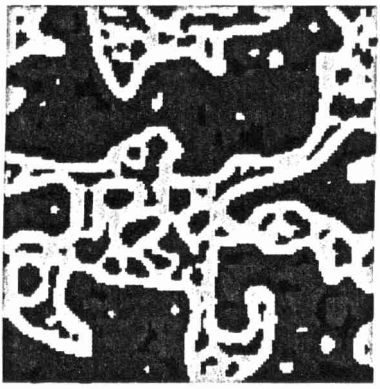
PLATE 1

The effect of random recruitment on the repertoire. The nine panels show the presence of clones in the plus and minus planes after 400 time steps. The clones present in the plus plane are colored red; those present in the minus plane are colored green. White pixels denote that a site is occupied in both planes, whereas black indicates absence from both planes. The nine panels show the repertoire for $\theta_1 = 1$, $\theta_2 = 80$, $P_R = 1, 2, 4, 8, 20, 40, 60, 80, 90$. Fixed boundary conditions are used throughout in the immune CA.

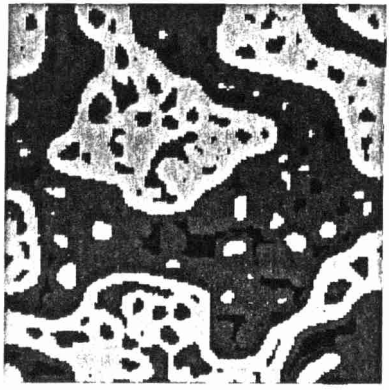
(a)



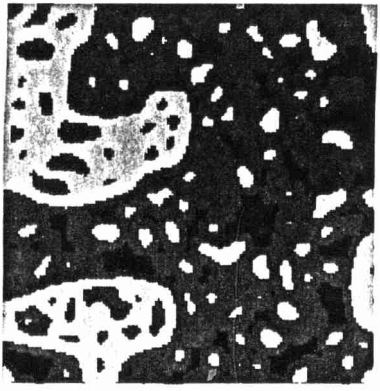
(b)



(d)

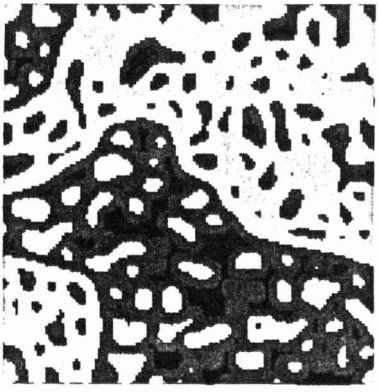


(e)

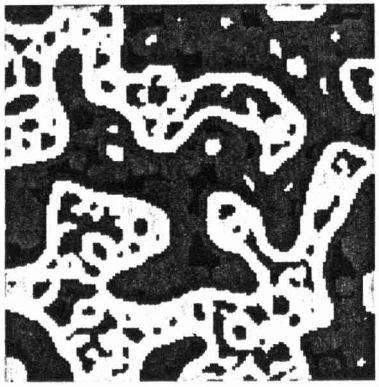


(g)

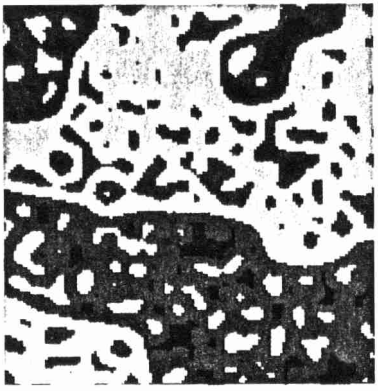
(h)



(c)



(f)

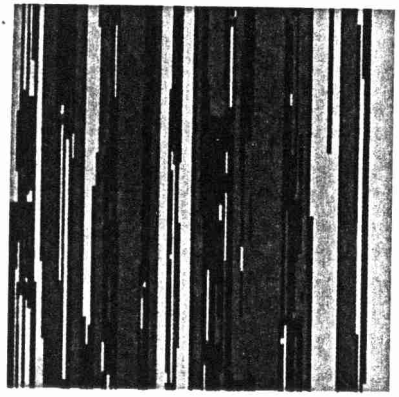


(i)

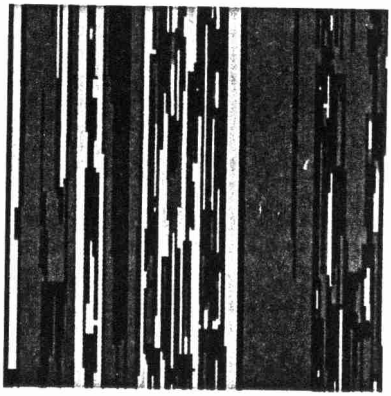
PLATE 2

Fields as a function of recruitment. The field, h_-, of the minus plane (i.e., of the green pattern in Plate 1) is color coded in three categories. Black means little or no stimulation, i.e., $h_- < \theta_1$. Green denotes a stimulatory field, i.e., $\theta_1 \leq h_- \leq \theta_2$. Red indicates a suppressive field, $h_- > \theta_2$. The nine panels show the fields for $\theta_1 = 1$, $\theta_2 = 80$, $P_R = 1, 2, 4, 8, 20, 40, 60, 80, 90$.

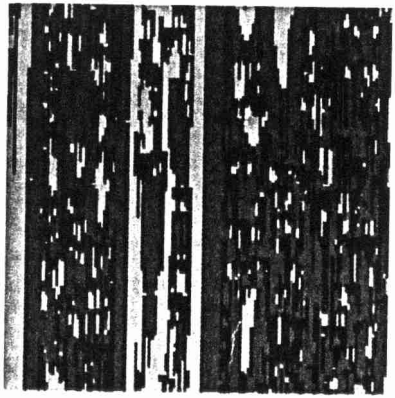
(a)



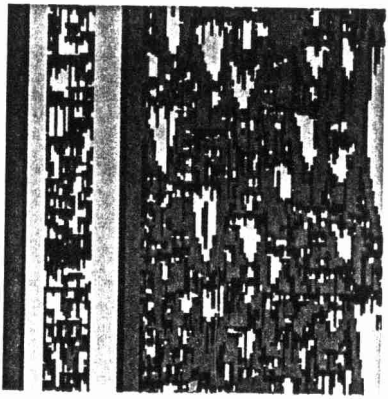
(b)



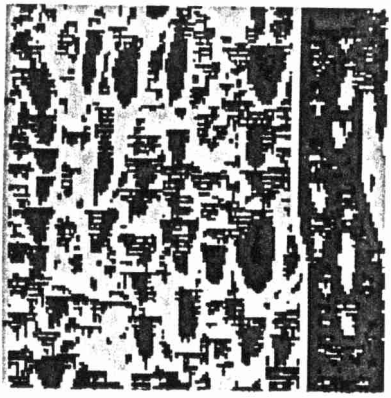
(d)



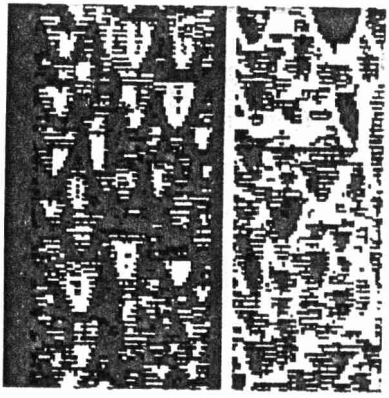
(e)



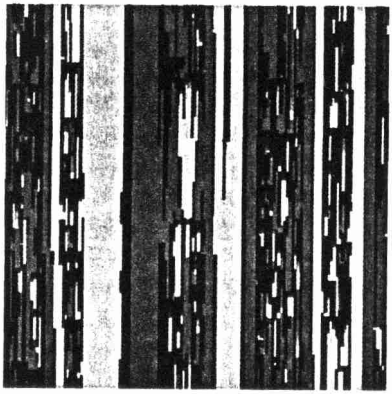
(g)



(h)



(c)



(f)

(i)

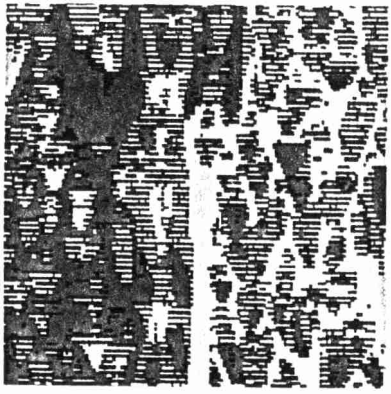


PLATE 3

Time series of a slice through the shape space. The panels are composed of 140 horizonal lines that correspond to horizontal slices through the center of the shape space at subseque ıt time steps. The slices correspond to the last 140 time steps of the simulation; time increases from top to bottom. The color coding is the same as in Plate 1. The nine panels show the time series for $\theta_1 = 1$, $\theta_2 = 80$, $P_R = 1, 2, 4, 8, 20, 40, 60, 80, 90$.

(a)

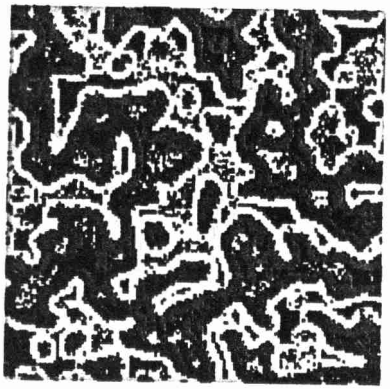
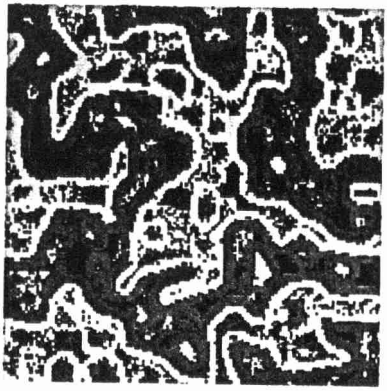


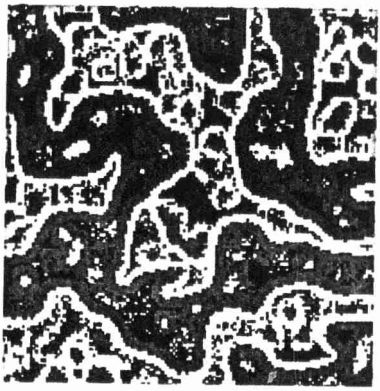
PLATE 4

The ontogeny of the pattern. For $\theta_1 = 1$, $\theta_2 = 80$, $P_R = 30$, Plate 4 shows five snapshots of the immune repertoire taken at day 15, 30,..., 75. Looking at the boundaries between the red and green areas, one can see that "capes" become eroded and that "bays" become filled. The color coding is the same as in Plate 1.

(b)



(c)



(d)

(e)

not contribute to the field, these novel clones do not interact and persist for one time step. This appears as a white line in the plot. Because the high recruitment has generated a high density, they suppress each other the next time step. As a consequence they all disappear. This appears as a black line in the plot This process also explains why in Plate 1 we observe a higher density of black and white regions: these are sustained by the oscillation.

The effect of the rate of recruitment on the fields is shown in Plate 2. Since the distribution of the fields follows that of the clones closely, the effect of P_R in Plate 2 is comparable to that in Plate 1. Additional effects are that increasing P_R increases the suppression, i.e., the part of the shape space covered in red. This reduces the responsivity of the system to external stimuli. A similar effect has been observed before in models based upon bitstrings.[7]

Summarizing, visual inspection of the results suggests that the system self-organizes on two spatial scales. Increasing the recruitment increases the scale of the large areas. Conversely, within the areas, a high rate of recruitment generates a smaller scale pattern of atolls. On the temporal scale the atolls become short lived when the rate of recruitment is high.

ONTOGENY OF THE PATTERN

A possible explanation for the increase in scale as a function of the rate of random recruitment is the similar effect that randomness has in voting rule systems (see Figure 1). In systems with simulated annealing and/or randomness, boundaries become fluid and small areas merge into larger ones. The temporal effects of annealing in voting rule systems are shown in Figure 2. This revealed that the patterns start at small scale and slowly increase in scale during subsequent generations.

We can test whether similar effects play a role in our immune network CA by studying the early evolution, i.e., the ontogeny, of the pattern. Thus, for $P_R = 30$, Plate 4 shows five snapshots of an immune repertoire taken after $15, 30, \ldots, 75$ time steps. This shows that in our immune network the scale of the patterns is increasing by a very similar process. Looking at the boundaries between the red and green areas, one can see that "capes" become eroded and that "bays" become filled. Thus, the effect of recruitment in the immune network is comparable to that of annealing or randomness in voting rule systems.

FILTERING BY MAJORITY VOTES

By defining a quantitative measure of scale, we here make our intuitive observations on the scale of the pattern a little more substantial. A simple scale variable is the ratio between border length and surface area (which is minimized by large discs). A problem with our patterns is that they are noisy. Red areas are not really red, and green areas are not really green. Thus, we resort to filtering techniques.

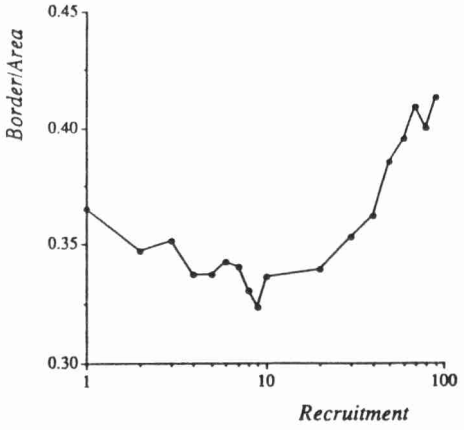


FIGURE 4 Scale, i.e., the border/area ratio, as a function of recruitment P_R. Each point corresponds to an average of the two fields of five simulations with $\theta_1 = 1, \theta_2 = 80$. For each simulation we take the minus plane after 400 time steps, normalize it, and perform majority voting until stabilization. For the filtered pattern the total number of bits in the borderlines and the total number of bits in areas are counted. The ratio of the two is plotted as a measure of scale.

An interesting technique for stressing the difference between regions is majority voting. Areas in which one of the colors dominates can be filled with that color by repeated majority votes. However, this procedure will only work if that dominant color occupies more than 50% of the sites. The average density in each of the shape planes is between 30% to 40% coverage. Further, the coverage increases somewhat with recruitment. Thus we first normalize the patterns to a coverage of 50%. This has two effects. First, areas in which one color dominates become covered with that color for more than 50%. Second, the coverage becomes independent of the rate of recruitment.

Thus, for each of the two shape planes, we count the coverage, and add random bits until the coverage is 50%. Actually, the normalization technique is (1) to calculate how many bits are required for attaining a coverage of 50%, (2) to make a random plane filled with that number of bits, and (3) to perform the logical OR operation on the original plane and the random plane. We then take this normalized version of the two planes of the immune CA as the initial configurations of a majority voting CA. This is expected to eliminate the noise and to connect some of the areas. We observed that this procedure preserves most of the characteristics of the original patterns (not shown). Subsequently, we calculate the border/area ratio of the pattern. A parallel technique for finding the borderlines is to filter out a plane of all bits that have less than eight neighbors. By counting the number of bits in the plane of borderlines and the number of bits in the plane subjected to normalization and voting, we know the total border length and the total surface area. We express scale as the ratio of the two, i.e., border/area.

We study the influence of the rate of recruitment on this measure of pattern scale. Thus, for $P_R = 1, 2, \ldots, 10, 20, \ldots, 90$, five simulations were carried out. Each dot in Figure 4 is the average border/area ratio of the two fields of the five simulations, i.e., each dot is based upon ten samples. The curve connecting these points

has a minimum around $P_R \approx 9$. Visually, the patterns indeed seem to be at the largest scale in this range of recruitment rates. Because of the emergence of many atolls, there is a steep increase of the border/area ratio at high rates of recruitment.

Biological data suggest that the rate of recruitment is very high. In the adult mouse, it is estimated that the production in the bone marrow amounts to 2 to 5×10^7 B cells per day.[11] This is sufficient for replacing the entire B-cell population in just a few days. Thus, if we assume that each day 20% of the repertoire can be replaced, an estimate for the probability of recruitment is $P_R = 20$. In this parameter range the patterns have a large scale (see Figure 4 and Plate 1(e)).

DISCUSSION

The model that we have developed is similar to a model developed by Stewart and Varela.[30] They also consider a two-dimensional shape space model in which clones are present or absent, and also implement complementarity by considering two planes of shapes. In their model, clones are randomly introduced in one of the planes and are maintained whenever they receive enough but not too much stimulation. The two-dimensional patterns that they report resemble those reported here. With our CA model we generalize the results of Stewart and Varela[30] and show that the scale of pattern depends on the rate of recruitment.

The network hypothesis that we have investigated here postulates that only part of the immune system is functionally connected to the immune network.[4,19,34,35] These ideas are (partly) based upon the percentage of activated lymphocytes in animals raised in "antigen-free" conditions.[20] Thus, it is estimated that 10–20% of the lymphocytes are functionally connected to the network, and that 80–90% remain disconnected.

In our model network we do find regions of the shape space that are functionally disconnected from the network. These regions correspond to the black areas in Plate 2 in which the field is low (i.e., $h_- < \theta_1$). The major problem with this result is, however, that the percentage of functionally disconnected clones is very small. Similar results have been obtained before with other network models.[7,8,30] Generically, the repertoires of the model networks are determined by a roughly complete coverage of shape space.[7] The networks are expected to encroach on areas that are poorly stimulated.

These results raise two questions. First, how can one explain the experimental observations that 80–90% of the clones remain functionally disconnected from the network? Second, what part of the immune system is responsible for the immune reactions, and for immunological memory?

With respect to the first question, we could use our results on the large parts of shape space that tend to be suppressed. Cells in these regions are not stimulated to grow and could be measured experimentally as nondividing lymphocytes.

Experimentally, such cells are interpreted as functionally disconnected cells. From our work, however, we know they are connected to the network by suppressive interactions, and cannot respond in an immune reaction. Another explanation that would account for a large fraction of apparently functionally disconnected clones is that the network might be composed of a special class of B cells, i.e., the Ly1 or CD5 B cells[16] which predominates during early life. The conventional B cells, which appear later, have several different properties. It is an open question whether or not the conventional B cells connect to the network. If they do not, they may account for the 80–90% nondividing cells.

With respect to the second question on the origin of immune reactions and memory, we interpret our results on the large stimulated areas that are only sparsely inhabited by clones. In these regions the presence of an antigen will allow more clones to be recruited and to respond in a primary immune reaction. This corresponds to a shift in the repertoire. If such a shift were to persist after the removal of the antigen, the network would account for memory. A secondary exposure with the same antigen would lead to a more vigorous immune reaction because that part of shape space would be densely populated. For our immune network CA, this speculation can be tested by extending our CA with an additional plane for the antigens. This distributed form of memory, i.e., one based upon repertoire shifts, has already been observed in previous models.[7] A more localized form of memory has been accounted for by network models with multiple stable points.[5,6,37]

The immune repertoire in this CA model, and that of previous network models,[7,8,9] is highly clustered. All clones in a cluster tend to have the same field and tend to be in the same state. This suggests that one could develop novel models in terms of "super clones" representing an entire cluster. In fact, such a super clone would correspond to the immunological concept of an "idiotype" which also comprises several clones. For such a super clone, the rate of recruitment corresponds to the average of many populations. Thus, it becomes realistic to model recruitment as a continuous, instead of a stochastic, process. This suggests that many earlier models that are based upon a continuous recruitment are realistic.

Our simple immune network CA has demonstrated that such clusters are likely to emerge because random recruitment increases the fluidity of the repertoire.

ACKNOWLEDGEMENTS

We thank Maarten C. Boerlijst and Alan S. Perelson for helpful discussions. This work is supported in part by grants to the Santa Fe Institute, including core funding from the John D. and Catherine T. MacArthur Foundation, the National Science Foundation (PHY-8714918), and the U.S. Department of Energy (ER-FG05-88ER25054).

APPENDIX I: CELLULAR AUTOMATA AS BITPLANES

One of the main properties of most CAs is their synchronicity or parallelism. This allows one to develop specialized hardware for fast CA simulations. Toffoli and Margolus[31] developed a Cellular Automata Machine (i.e., the CAM) that is based upon hardware bitplanes. In our bitplane approach we borrow a lot of their ideas. Thus, in our CAs, all operations are defined on the level of (software) bitplanes, and run for all individual bits simultaneously.

Because most computers use bitplanes for updating the screen, the bitplane approach has two advantages. First, the simulations are fast due to efficient algorithms in the system software. Second, it is fast and straightforward to visualize the state of the CA by copying a bitplane to the screen. Note that piles of bitplanes can be converted to color pixels. Visualization is a major advantage because one can actually watch the behavior of the CA and see its patterns emerge.

Operations on the level of bitplanes correspond to logical operations (e.g., OR, AND, and XOR). For example, one can perform the AND operation on two bitplanes to obtain a new bitplane that is black in those positions where both bitplanes are black. Note that whenever rules become complex, the translation into a sequence of logical operations may become quite complicated. Another operation is the SHIFT, which yields a copy of a bitplane with all positions shifted in some direction. Shift operations are helpful for getting neighborhood information. For instance, shifting a bitplane one bit to the right yields a novel bitplane with in each cell the states of its left neighbor. Shift operations require a specification of the boundaries of the CA. The conventional procedure is to use periodic boundaries that wrap the CA in the form of a torus. In our immune CA we assume fixed boundary conditions which means that all cells outside the boundaries are absent.

CAs with more than two states simply require more bitplanes. This is very common, e.g., the sum of neighbors in voting rule CAs ranges from zero to nine. We represent integer numbers in a pile of bitplanes by encoding it in the standard binary representation. Arithmetic operations on piles of bitplanes are also defined in terms of logical operations on the individual bitplanes (see Goldberg and Robson[15] for a more detailed explanation of counting by logical operations on bitplanes).

Random values are also obtained in parallel by running a random generator CA, i.e., the NOISE-BOX, described by Toffoli and Margolus.[31] At every generation the NOISE-BOX generates two planes with a random distribution of 50% ones and 50% zeros. Other percentages are obtained by running the NOISE-BOX for a few time steps and performing logical operations on several random planes.

By showing the code for the majority voting rule, we here illustrate the basic principles of bitplane CAs. To this end we define an *ad hoc* formal language. In this language, variables in normal typeface denote planes, variables in boldface denote piles of planes, and variables in italics denote integers. We have a library of several functions:

1. **Shift**(**Pile**,i,j) returns a copy of the **Pile** shifted i positions horizontally and j positions vertically.
2. **Sum**(**Pile**,**Pile**) returns a pile which is the arithmetic sum of its two arguments.
3. Random(i) returns a random Plane in which a percentage of i of the bits are black.
4. LargerEQ(**Pile**,i) returns a plane with ones for those positions where the integer encoded in the Pile is $\geq i$.

Having defined the library functions, the following function retrieves the sum of the 3×3 neighborhood of each cell in a plane in parallel:

```
GetNeighborhood(World) {
    N ← World                          initialize the neighborhood by the center cell
    N ← Sum(N, Shift(World, −1, −1))   add the NorthEast neighbor
    ⋮                                  shift eight times in total
    N ← Sum(N, Shift(World, 1, 1))     add the SouthWest neighbor
    GetNeighborhood ← N                return the sum
}
```

Having defined the neighborhood function, the program is simple. It consists of an iteration loop that calls the neighborhood function n times:

```
World ← Random(50)                     initialize with 50% random bits
Repeat n Times {
    World ← LargerEQ(GetNeighborhood(World), 5)
    Display(World)                     show the results on the screen
}
```

APPENDIX II: THE IMMUNE SHAPE SPACE CA

In the immune network CA, we require a large number of planes and piles for modeling the different parts of the system. The complementary shape space planes are represented in the two bitplanes Shape_+ and Shape_-. These planes are both initialized with a random distribution of 10% ones, i.e., 10% of the shapes is present. The field, i.e., the network stimulation, is retrieved by a GetWeightedNeighborhood function (see the code below). The result of this function (i.e., the weighted neighborhood) is stored in the piles $\textbf{Field}_+$ and $\textbf{Field}_-$. These piles are then filtered twice by the routine LargerEQ(). This returns a plane (i.e., Field_+ and Field_-, respectively) with ones at those positions where the weighted sum in the pile is between θ_1 and θ_2. Thus, Field_+ and Field_- correspond to the stimulatory fields.

The recruitment is realized by first creating two random planes (i.e., Marrow_+ and Marrow_-) with a certain percentage, P_R, of ones. The Marrow planes are added to the Shape planes by an OR operation on both planes. An AND operation on the new Shape plane and the corresponding Field plane determines which clones are to be maintained. Each time step the results are displayed on the computer screen.

For illustrative reasons we translate this formalism in our illustrative formal language. Below, the code for the Shape_+ plane is listed. The code for the Shape_- plane is very similar.

The program is based upon a GetWeightedNeighborhood function, which is in turn based upon the GetNeighborhood function defined in Appendix I. The main idea of the GetWeightedNeighborhood function is that shifting the standard neighborhood pile of a plane increases the size of the neighborhood and provides a weighting with respect to the distance. By repeating such a shift, one increases the neighborhood. Here we repeat the shift twice.

```
GetWeightedNeighborhood(Shape) {
      N ← GetNeighborhood(Shape)
      P ← N                                  initialize with the standard neighborhood
      P ← Sum(P, Shift(N, −1, −1))                          now shift four times
      P ← Sum(P, Shift(N, 1, −1))
      P ← Sum(P, Shift(N, −1, 1))
      P ← Sum(P, Shift(N, 1, 1))
      Q ← P                                  initialize with the shifted neighborhood
      Q ← Sum(Q, Shift(P, −1, 0))                           shift another four times
      Q ← Sum(Q, Shift(P, 0, −1))
      Q ← Sum(Q, Shift(P, 1, 0))
      Q ← Sum(Q, Shift(P, 0, 1))
      GetWeightedNeighborhood ← Q                          return the weighted sum
}
```

In terms of these functions, the program for the plus planes in the immune network CA becomes

```
Shape+ ← Random(10)
Repeat 400 Times {
      Field+ ← GetWeightedNeighborhood(Shape−)
      Field+ ← LargerEQ(Field+, θ1) AND NOT LargerEQ(Field+, θ2+1)
      Marrow+ ← Random(PR)
      Shape+ ← (Shape+ OR Marrow+) AND Field+          determine next state
      Display(Shape+)
}
```

REFERENCES

1. Avrameas, S. "Natural Antibodies: From 'Horror autotoxicus' to 'Gnothi seauton.'" *Immunol. Today* **12** (1991): 154–159.
2. Bak, P. "Self-Organized Criticality and Gaia." This volume.
3. Boerlijst, M. C., and P. Hogeweg. "Spiral Wave Structure in Prebiotic Evolution: Hypercycles Stable Against Parasites." *Physica D* **48** (1991): 17–28.
4. Coutinho, A. "Beyond Clonal Selection and Network." *Immunol. Rev.* **110** (1989): 63–87.
5. De Boer, R. J., and P. Hogeweg. "Memory but No Suppression in Low-Dimensional Symmetric Idiotypic Networks." *Bull. Math. Biol.* **51** (1989): 223–246.
6. De Boer, R. J. "Recent Developments in Idiotypic Network Theory." *Neth. J. Med.* **39** (1991): 254–262.
7. De Boer, R. J., and A. S. Perelson. "Size and Connectivity as Emergent Properties of a Developing Immune Network." *J. Theor. Biol.* **149** (1991): 381–424.
8. De Boer, R. J., P. Hogeweg, and A. S. Perelson. "Growth and Recruitment in the Immune Network." In *Theoretical and Experimental Insights into Immunology*, edited by A. S. Perelson, G. Weisbuch, and A. Coutinho. New York: Springer, 1992 (in press).
9. De Boer, R. J., L. A. Segel, and A. S. Perelson. "Pattern Formation in One- and Two-Dimensional Shape Space Models of the Immune System." *J. Theor. Biol.* **155** (1992): 295–333.
10. De Boer, R. J., A. U. Neumann, A. S. Perelson, L. A. Segel, and G. W. Weisbuch. "Recent Approaches to Immune Networks." In *Proceedings First European Biomathematics Conference*, edited by V. Capassso and P. Demongeot. Berlin: Springer, 1992 (in press).
11. Freitas, A. A., B. Rocha, and A. A. Coutinho. "Lymphocyte Population Kinetics in the Mouse." *J. Immunol.* **91** (1986): 5–37.
12. Gardner, M. "Mathematical Games: The Fantastic Combinations of John Conway's New Solitaire Game 'Life.'" *Sci. Am.* **223(4)** (1970): 120–123.
13. Gardner, M. "Mathematical Games: On Cellular Automata, Self-Reproduction, the Garden of Eden and the Game 'Life.'" *Sci. Am.* **224(2)** (1971): 112–117.
14. Gerhardt, M., H. Schuster, and J. J. Tyson. "A Cellular Automaton Model of Excitable Media Including Curvature and Dispersion." *Science* **247** (1990): 1563–1566.
15. Goldberg, A., and D. Robson. *Smalltalk-80, the Language and Its Implementation*, 412–413. Redwood City, CA: Addison-Wesley, 1983.
16. Herzenberg, L. A., A. M. Stall, P. A. Lalor, C. Sidman, W. A. Moore, D. R. Sparks, and L. A. Herzenberg. "The Ly-1 B Cell Lineage." *Immunol. Rev.* **93** (1986): 81–102.

17. Hogeweg, P. "Cellular Automata as a Paradigm for Ecological Modeling." *Appl. Math. Comp.* **27** (1988): 81–100.
18. Hogeweg, P. "Local T–T and T–B Interactions: A Cellular Automaton Approach." *Immunol. Lett.* **22** (1989): 113–122.
19. Holmberg, D., Å. Andersson, L. Carlson, and S. Forsgen. "Establishment and Functional Implications of B-Cell Connectivity." *Immunol. Rev.* **110** (1989): 89–103.
20. Hooykaas, H., R. Benner, J. R. Pleasants, and B. S. Wostmann. "Isotypes and Specificities of Immunoglobulins Produced by Germ-Free Mice Fed Chemically Defined 'Antigen-Free' Diet." *Eur. J. Immunol.* **14** (1984): 1127–1130.
21. Jerne, N. K. "Towards a Network Theory of the Immune System." *Ann. Immunol. (Inst. Pasteur)* **125C** (1974): 373–389.
22. Kearney, J. F., and M. Vakil. "Idiotype-Directed Interactions During Ontogeny Play a Major Role in the Establishment of the Adult B Cell Repertoire." *Immunol. Rev.* **94** (1986): 39–50.
23. Kirkpatrick, S., C. D. Gelatt, and M. P. Vecchi. "Optimization by Simulated Annealing." *Science* **220** (1983): 671–680.
24. Langton, C. G. "Life at the Edge of Chaos." In *Artificial Life II*, edited by C. G. Langton, C. Taylor, J. D. Farmer, and S. Rasmussen. SFI Studies in the Sciences of Complexity, Vol. X, Part Two, 41–91. Redwood City, CA: Addison-Wesley, 1991.
25. Neumann, A. U., and G. Weisbuch. "Window Automata Analysis of Population Dynamics in the Immune System." *Bull. Math. Biol.* **54** (1991): 21–44.
26. Perelson, A. S., and G. F. Oster. "Theoretical Studies on Clonal Selection: Minimal Antibody Repertoire Size and Reliability of Self–Non-Self Discrimination." *J. Theor. Biol.* **81** (1979): 645–670.
27. Perelson, A. S. "Some Mathematical Models of Receptor Clustering by Multivalent Ligands." In *Cell Surface Dynamics: Concepts and Models*, edited by A. S. Perelson, C. DeLisi, and F. M. Wiegel, 223–276. New York: Marcel Dekker, 1984.
28. Segel, L. A., and A. S. Perelson. "Computations in Shape Space: A New Approach to Immune Network Theory." In *Theoretical Immunology*, Part Two, edited by A. S. Perelson. SFI Studies in the Science of Complexity, Vol. III, 321–343. Redwood City, CA: Addison-Wesley, 1988.
29. Segel, L. A., and A. S. Perelson. "Shape-Space Analysis of Immune Networks." In *Cell to Cell Signalling: From Experiments to Theoretical Models*, edited by A. Goldbeter, 273–283. New York: Academic Press, 1989.
30. Stewart, J., and F. J. Varela. "Morphogenesis in Shape-Space. Elementary Meta-Dynamics in a Model of the Immune Network." *J. Theor. Biol.* **153** (1991): 477–498.
31. Toffoli, T., and N. Margolus. *Cellular Automata Machines. A New Environment for Modeling.* Cambridge, MA: MIT Press, 1987.

32. Ulam, S. "Random Processes and Transformations." *Proc. Intl. Congr. Math.* **2** (1952): 264–275.
33. Varela, F. J., A. Coutinho, B. Dupire, and N. N. Vaz. "Cognitive Networks: Immune, Neural, and Otherwise." In *Theoretical Immunology*, Part Two, edited by A. S. Perelson. SFI Studies in the Science of Complexity, Vol. III, 359–375. Redwood City, CA: Addison-Wesley, 1988.
34. Varela, F. J., and A. Coutinho. "Second Generation Immune Networks." *Immunol. Today* **12** (1991): 159–166.
35. Varela F. J., A. Coutinho, and J. Stewart. "What is the Immune Network for?" This volume.
36. Vichniac, G. Y. "Cellular Automata Models of Disorder and Organization." In *Disordered Systems and Biological Organization*, edited by E. Bienenstock, F. Fogelman Soulié, and G. Weisbuch, 1–20. Berlin/Heidelberg: Springer-Verlag, 1986.
37. Weisbuch, G. "A Shape Space Approach to the Dynamics of the Immune System." *J. Theor. Biol.* **143** (1990): 507–522.
38. Wolfram, S. "Universality and Complexity in Cellular Automata." *Physica D* **10** (1984): 1–35.