Nanomanipulator display come from a tiny probe poised over a surface's bumpy atomic landscape. In its original form, conceived 5 years ago by Williams and Warren Robinett, a UNC computer scientist, the system was hooked to a scanning tunneling microscope (STM)—a device that maps surface contours from variations in an electric current running from surface to tip. Besides seeing the surface in a head-mounted display, the user could "feel" it through force feedback on a mechanical arm electronically linked to the STM tip and even sculpt it by pulling a trigger on the handgrip. A pulse of voltage fired from the STM tip, says Williams, would "dig out chunks of the surface or deposit pieces of the STM tip."

To sculpt with more finesse, the Nanomanipulator collaboration substituted an atomic force microscope (AFM) for the STM. Unlike the STM, the AFM physically traces the surface, like a phonograph needle. And although its resolution is coarser than an STM's, the AFM tip can actually push objects around on the surface. Explains Williams, "If you bump the AFM tip into something, you feel it on the mechanical tracking arm, and so you can start pushing things around as you might push a soccer ball around with your hand."

In the last week of June, the Nanomanipulator team exploited that ability to build what they call a nano-Stonehenge from 100-atom colloidal gold particles deposited onto an otherwise smooth gold surface. According to Russell Taylor, a UNC graduate student who helped build the original system, it took just 40 minutes "to assemble an aggregate of [these] balls on the surface and sweep the others away." Other researchers building atomic-scale structures without the benefit of VR, in contrast, often spend days on such feats.

Williams thinks the group's achievement may be a first step toward creating nanometer-scaled electronic circuitry. He's also dreaming of manipulating and modifying biological molecules such as DNA and proteins. "We could use an STM or AFM tip as a very fine scalpel to dissect a protein molecule or DNA molecule," he says. "Take it apart, put it back together, see if we can do it all using physical manipulation rather than chemical." The project could start right away if his group had "a good partner," says Williams: "someone who would have the knowledge of the biological molecules and the patience to actually play around with this thing."

And play, Williams says unapologetically, is still a pretty good word for the use of VR by the Nanomanipulator group—and that of many others. Interesting as it is, he says, "it's still at the stage of a pretty far-out research toy."

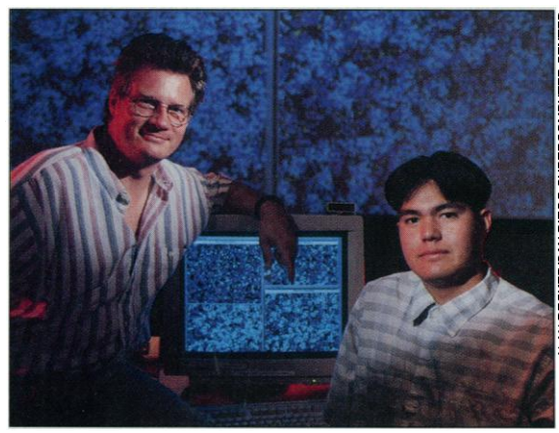
–Gary Taubes

BIOLOGICAL MODELING

Do Immunologists Dream of Electric Mice?

Call up Mosaic, a popular tool for exploring the Internet, and type http://bitmed.ucsd.edu. You'll see a logo that looks like a mouse and then a succession of menus. In a few moments, without leaving your desk, you'll be exploring the immune system of a mouse. With no more than a click of the (desktop kind of) mouse, you can manipulate a tiny syringe to inject cells or antigen, assay lymphocytes, and study how an immune response plays out in different tissues. Meet Cybermouse, an on-line virtual laboratory animal for exploring the immune system.

Cybermouse is the product of a collaboration between immunologist Don Mosier of the Scripps Research Institute in La Jolla, California, and mathematician Hans Sieburg of the University of California, San Diego and is just one example—albeit one of the most ambitious—of an expanding breed. In increasing numbers, computer scientists, physicists, and mathematicians are teaming up with what IBM physicist Phil Seiden calls "honest-to-God mouse-sticking immunologists" to study the immune system "in silico," as Sieburg puts it. The simulations range from sets of equations tracing population swings in white blood cells and antibodies to massive arrays of software agents known as "cellular automata" that mimic the interactions of different immune-system cells and tissues.



ALAN DECKER/SAN DIEGO SUPERCOMPUTER CENTER

Mouse and men. Flanking a display of a bone-marrow simulation from the Cybermouse model are Hans Sieburg, one of its developers *(left)*, and his student Cris Baray.

Regardless of which kind of model you're talking about, says Mosier, these computerized simulations can run experiments and test hypotheses that would be too time-consuming or difficult to do with live animals. "If we did them all in a mouse," he says, "it would take 10 years"—if they could be done at all. Adds Mel Cohen, a Salk Institute immunologist who learned his trade in the 1950s with Jacques Monod and François Jacob at the Pasteur Institute, "These computer models are the only way to take a large number of variables, take a system that is very complex, and study the effect of one variable on all the others."

Cohen and his colleagues aren't arguing that silicon-based immune systems can ever substitute fully for laboratory animals. But the simple, rigid logic of a computer is a good match to that of the immune system, argues Cohen's partner Rod Langman, a Salk Institute biologist. "[If I can] tell an incredibly stupid computer what to do... so that it can simulate immune response, then I must really have understood something about the immune system." Already, researchers have used computer simulations to make testable predictions about how the AIDS virus spreads through the immune system and how the system "remembers" a pathogen years after the original infection—and responds with a quick and vigorous counterattack.

One approach to putting the immune system on a computer was pioneered as early as the 1960s, when Los Alamos physicist George Bell developed a set of differential equations describing how antibody-producing white cells, called B cells, proliferate, differentiate, and secrete antibody in response to an antigen, in particular a foreign virus or bacteria. In 1986, theoretical immunologist Alan Perelson joined physicists Doyne Farmer and Norman Packard at Los Alamos to build on that strategy. As Perelson describes it, Bell's original equations followed the fate of some 40 B cell types, each one defined by a receptor able to recognize a different antigen. The actual immune system, however, has some 10 million different types of B cells, which is why it can recognize literally any antigen.

Rather than trying to write an equation for every possible B cell population and the antigen it responds to, Perelson, Packard, and Farmer developed a technique called metadynamics, in which the computer itself constructs the differential equations of the model. It does so in response to the interactions of bit strings representing the receptors of the various cell types. (Each cell and its progeny carry the same bit string—for in-

Devising Software Immune Systems for Computers

Most researchers who are building bridges between computer science and immunology are trying to understand the immune system (see main text). A handful, however, hope to reverse that logic and use what they know about the immune system to treat the diseases that plague computers. These researchers believe that a promising defense strategy against computer viruses is to endow computers with the digital equivalent of an immune system.

The immune system, after all, can recognize and defend the body against a vast range of threats without any prior instruction in how to recognize them. As Los Alamos theoretical immunologist Alan Perelson puts it, "It would be a disaster if we could only fight viruses we already knew about." But most existing computer-virus detectors do just that, says Perelson: "They have signature patterns of all the known viruses, and they look for those." The result is that a virus with an unexpected appearance can slip by.

One approach to broadening computers' defenses, taken by Perelson and computer scientist Stephanie Forrest of the University of New Mexico, mimics the body's own strategy of constant vigilance for anything that differs from the body's own tissues. In the immune system, explains Perelson, lymphocytes are born with a vast, random variety of receptors, able to bind a huge variety of molecules without regard for whether they belong to the body or to some foreign invader. They then undergo a process of "negative selection."

Cells destined to become T cells go to the thymus, where they interact with the body's own molecular signatures. Cells that have a high affinity for these "self" antigens—and would therefore tend to attack the body's own tissues—are triggered to die. "What you're left with," says Forrest, "are T cells more or less guaranteed not to match anything in self. Now if they ever bind to anything, the presumption is it's foreign, and so the eradication mechanism gets activated."

Similarly, Forrest and Perelson's anti-virus scheme generates thousands or millions of random bit strings—the equivalent of T cell receptors—and compares them to characteristic strings in existing programs. If an anti-virus string matches a self string, it is thrown away. If not, it's kept. Educated by this negative-selection procedure, the anti-virus system then goes on the lookout for viruses by continually comparing the remaining "receptor" strings to strings in the program code. Forrest and Perelson are talking to computer security experts who could turn this concept into marketable software.

Meanwhile, at IBM, computer scientist Jeff Kephart is working on an anti-virus program that recognizes unknown threats by a strategy based on both negative selection and an outdated theory of the immune system known as the instruction hypothesis. The instruction hypothesis held that the immune system doesn't have a repertoire of pre-existing receptors; instead, it creates them as needed to fit invading pathogens. The antigen serves as a template, molding the antibodies that gather around it into a shape that the immune system then mass-produces; the immune system is, in effect, "instructed" by the pathogen.

The instruction theory didn't hold up as understanding of the immune system deepened, but Kephart thinks it can improve on negative selection alone for computer virus detection. The negative selection scheme, he argues, requires vast numbers of recognizers—consuming "lots of memory," among other things—and defines everything nonself as an intruder, which makes it difficult to add legitimate new programs to the system. The IBM system, in contrast, only adds proven viruses to its archive of threats.

The system continually scans a computer's software for typical signs of viral infection, such as a sudden change in the length of a program. These signs trigger the release of decoy programs whose sole purpose is to become infected by the virus. After a decoy is infected, the system captures the virus, analyzes it, generates all possible recognizers for it, and eliminates those that also recognize self and other legitimate programs. It then adds the remaining, nonself recognizers to its anti-virus repertoire.

Kephart agrees that Perelson and Forrest's scheme is more faithful to biology. "But we're happy to take the discarded ideas of theoretical immunologists," he says, if they might work better.

–G.T.

stance, 11000110.) Antigens, too, carry these numerical fingerprints. Matches between bit strings determine the terms generated by the computer in the various differential equations. When a B cell encounters an antigen, for example, the simulation compares their bit strings; if they match, an immune response ensues, and the equation describing the proliferation of that B cell and the elimination of the antigen come into play. Using these methods, says Perelson, he and his colleagues can create models that follow the fate of tens of thousands of B cell clones.

Model of maturation. Lately, Perelson and biomathematician Thomas Kepler of North Carolina State University have been applying this kind of model to study affinity maturation. Affinity is the immunological term for the ability of antibodies to bind an antigen. During the course of an immune response, the B cell genes that determine how antibodies bind their targets undergo small "hypermutations"—random changes that ultimately yield new antibodies with increased affinity. By simulating the process, Perelson and Kepler hoped to learn the best mutation strategy—the one yielding the largest number of high-affinity antibodies.

According to their model, the best strategy turns out to be mutation in discrete bursts. Perelson explains that the body starts with only a few cells that can bind the antigen at all, "and mutation," he says, "generally is a deleterious process." Most hypermutation leads to antibodies less adept at binding the antigen than the original was; only a small fraction do it better. The model showed that if hypermutation set in as soon as a B cell recognized the antigen, it might quickly destroy the system's ability to respond at all. "So the optimal strategy," says Perelson, is to take "those few cells [that recognize antigen], expand them to a larger population, and only then turn on mutation. Then after you have created higher affinity antibodies, you turn off mutation, so you don't kill off the solutions, and you let them grow to a larger population" before mutating them again.

Researchers trying to trace this fitful mutation process in the body, say Perelson and Kepler, would find evidence of it in regions of the lymph nodes and spleen called germinal centers. The germinal centers are divided into light and dark zones; Perelson and Kepler speculate that B cells shuttle between these two zones, undergoing bursts of hypermutation in the dark zone and encountering antigen trapped in the light zone. Immunologist Garnett Kelsoe is testing that picture in his lab at the University of Maryland by radioactively labeling the B cells in the dark zone. If he and his colleagues later find doubly labeled B cells, says Kelsoe, they'll have "physical proof" that the cells went through the dark zone at least twice.

Local actions, global effects. That would represent a success for models based on differential equations, but some researchers have recently embraced a different strategy, based on cellular automata. Instead of coming up with a set of equations specifying

the system's overall behavior, these modelers allow global behavior to emerge from local interactions of many elements. Each element represents a cell or set of cells; what happens at each site depends only on the state of adjoining sites or the interactions between cells in the site itself. As the simulation unfolds, the computer repeatedly updates all the sites, assigning a new value to each one according to rules that take into account the adjoining sites.

According to New York University immunologist Franco Celada, this neighborly proceeding is "similar to what happens in an organ like a lymph node, where what happens in one spot depends on local interactions of nearby cells and molecules." It may also be a more effective strategy than sets of differential equations for parlaying simple immunological principles into a complex simulation, says Perelson. "With differential equations, you have to come up with terms for the rates of interactions [and for] the concentration dependencies of these interactions. In cellular automata you only have to come up with these rules, and you avoid having to have as much detailed knowledge of the actual system."

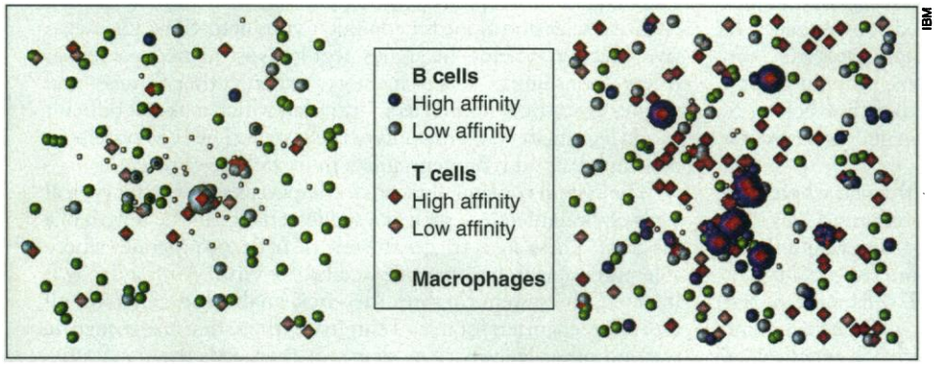
Lately Seiden and Celada have been using a cellular automaton model to understand one of immunology's central puzzles: the phenomenon of "immunologic memory." This memory appears as the immune system's ability to respond strongly to an antigen the second time it encounters it, even if decades have passed and the cells that first "saw" it have long since died. "There are a number of theories," says Seiden, "and we want to test them and see if we can come up with one self-consistent theory of memory."

One notion Seiden and Celada are testing is that antigen is kept in the system indefinitely, trapped in the germinal centers. Every so often, says Seiden, a little bit is released to restimulate antibody response. Another possibility is that when the immune system is flooded with antibodies during the initial response, it generates new antibodies that bind to the first set; these "anti-idiotypes," as they're known, resemble the original antigen. Long after the antigen is gone, new generations of anti-idiotypes keep its memory alive. If Seiden and Celada can come up with a variation of this mechanism that works in the model, Seiden says, immunologists could then try to confirm it in animals.

Immunologists using live animals don't limit themselves to studying generic immune responses; they're also concerned with how the system reacts to specific pathogens such as the AIDS virus, HIV. The same is true for the immune-system modelers. Modeling how the immune system reacts to a specific infection, however, requires a model elaborate enough to simulate different tissue types. That's the goal of Sieburg's Cybermouse.

One inspiration for Cybermouse dates back to 1986, when Sieburg was working in Mel Cohen's laboratory and realized that immunologists could benefit from computer simulations. With physicist Oliver Clay, now at the University of Zurich, Sieburg began developing a programming language that would allow them to "mass-produce" cellular automata modeled on different tissue types. The other inspiration came in 1988, when Mosier created the SCID mouse—severe combined immunodeficiency disease mouse—a flesh-and-blood mouse with a genetic defect that prevents it from making any lymphocytes whatsoever, leaving it with virtually no immune system. When human peripheral blood cells are transplanted into it, the SCID mouse will, for 3 months, sport a human immune system. Inject the SCID mouse with HIV, and the result is a relatively inexpensive animal model of AIDS.

But in conversations with Mosier, Sieburg says, they quickly realized that "you cannot do all experiments in the mouse. You cannot be complete and clean and do every detail on the wet bench. So that's where the idea of Cybermouse came in, to create various kinds of tissues such as lymph nodes, thymus, bone marrow, pieces of brain, gut, liver, and so forth, as artificial tissues and then infect them with HIV." For each artificial tissue, Sieburg, Mosier, and hematologist Christa Müller-Sieburg of the Medical Biology Institute in La Jolla developed a separate set of cellular automata representing a distinct collection of immune-system cells and molecules. Along with a variety of tissue types, Cybermouse includes what Sieburg calls "fluid microenvironments," such as blood.



IBM

Memory in a silicon immune system. The immune system gets a slow start *(left)* when it first encounters an antigen (*gray dots*) in this IBM simulation. Few cells are dividing *(mass at center)*, and high-affinity cells are scarce. The second exposure *(right)* brings a stronger response.

Mosier became sold on Cybermouse after running a series of experiments in the SCID mouse to identify the most efficient route of infection for the CD4+ T cells that are decimated in AIDS. At the time, most researchers believed that infected macrophages—an early target of the virus—are less efficient at spreading the disease to the T cells than is the "free" virus that floats in the blood. Mosier's SCID mouse experiments unexpectedly showed that the virus-infected macrophages were more deadly. And when Sieburg replicated the experiment in Cybermouse, he got the same result—along with insight into why macrophages might be at fault.

It turns out, says Mosier, "that the depletion of CD4+ T cells takes place most efficiently in local foci of infection. Macrophages are highly efficient at establishing these foci of infection. Free virus is less efficient. The virus goes in and hits single cells and may kill them very rapidly without allowing infection to spread to adjacent cells," whereas macrophages stay alive to provide an ongoing source of infection. This experiment, he adds, "really convinced me the simulator was going to be useful."

Lately, Sieburg and Mosier have been using Cybermouse and the SCID mouse in tandem to study how the AIDS virus spreads through the lymph nodes. Cybermouse, says Sieburg, reveals that the virus spreads like wildfire—jumping from one focus of infection to another. This finding implies that strategies for blocking these infectious hot spots might be extremely effective, says Mosier. "It also suggests," he says, "that studying human lymph nodes will have to be done very, very carefully." Researchers could easily focus on an uninfected segment of lymph node and overlook the CD4+ T cell depletion raging nearby.

Immunologists might once have resisted taking a cue from a computer. But Chris Phelps, a neuroendocrinologist at the University of South Florida who is collaborating with Sieburg on neuroimmunological applications of Cybermouse, argues that the complexity of the immune system leaves little choice. After years of looking at small pieces of the system—cells and signaling molecules—"immunologists are now faced with the challenge of putting it all back into a systems context."

–Gary Taubes