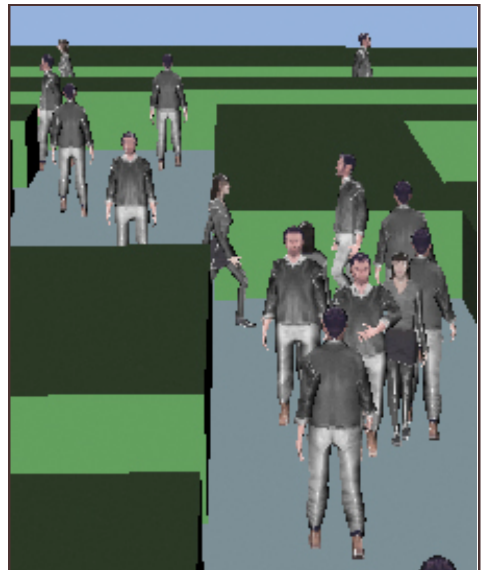
Endeavors magazine: Virtual Danger, Safer Places

Virtual Danger, Safer Places

Computer simulations can help people survive war and disaster.

by Margarite Nathe



Knowing how groups of people move with and around each other can help computer scientists develop realistic simulations for evacuation procedures, pedestrian walkways, and even sports arenas. Image courtesy of GAMMA Research Group.

Say you're a U.S. soldier on patrol in Baghdad. Your job: walk the perimeter line, maintain the peace, keep your eyes open. You're strapped into heavy-duty body armor and carrying an M16 in the crook of your arm. The streets are busy, the weather is warm, and the monotony of it all is making you a little sleepy.

Then the shooting starts. You have no idea who's doing it or where it's coming from. Before the smoke clears, the shooter vanishes, blending so seamlessly into the hysteria on the streets that you'll never know where he came from or where he went.

What were you doing in those few seconds? And what will you do in the moments to come? Really think about it. You could say that when the firing started you would have ducked and covered, or that you would've immediately returned fire, or that you would have done any number of things. But you never know for sure until you've been through it.

Training that uses virtual reality can help soldiers imagine themselves in situations like this before they ever happen. And simulation technology — the nuts and bolts software behind virtual reality — is at the core of everyday civilian lives, too: the patterns of our highways, the engines in our cars, the layouts of our doctors' offices. Being able to model how we do things as a society and as individuals can keep us safe, whether it's at home, at work, or in enemy territory.

In terms of sheer numbers of military personnel, North Carolina ranks fourth in the country. And for the past ten years, the U.S. Army has been investing billions of dollars into developing more advanced training for the thousands of troops they send to the Middle East. The Army put those two facts together and in 2003 approached Carolina computer scientists Ming Lin and Dinesh Manocha. By then, Lin and Manocha had already been working on algorithms for new kinds of simulation technology for at least a decade.

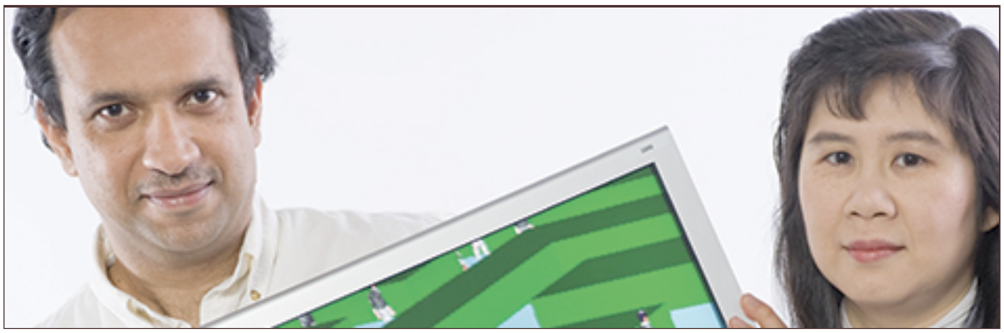
Now, through a modeling and simulation program called OneSAF (which stands for One Semi-Automated Forces), officials at over six hundred sites across the country have used Lin and Manocha's defense technology. This kind of simulation training can teach soldiers how to stay alive in the combat zone.

"When we say defense, we're not talking about hardware, like missiles or nuclear weapons," Lin says. "Rather, it's about simulation and training technologies that can actually train our military personnel so they can get away from their attackers or defend themselves."

"The nature of war has changed," Manocha says. "The kind of war they have in the Middle East and other parts of the world today is what we call urban warfare." The violence of today's wars takes place in the cities — in buildings and on streets — more often than in open battlefields where the armies can see each other coming. And while some war-making takes place from far afield — explosions caused by the push of a button thousands of feet above or hundreds of miles away from the resulting boom — most U.S. soldiers are at eye level with the action. Knowing how to respond quickly and calmly in a tight situation can save their lives.

"Before the Army sends someone from Fort Bragg to Iraq, they need to know how to train them for a very difficult and hostile environment," Manocha says. "So the Army has been using what we call training environments to give soldiers a feel for what it will be like there."

Most of these training exercises involve powerful computers (or even supercomputers) that are paired with some sort of display, whether it's a computer monitor or a huge projector. Some even use head-mounted displays. (One of Lin and Manocha's goals is to develop powerful software that could be used with hardware that's easier to get hold of — everyday desktop or laptop computers that you can buy in any electronics department.)



Dinesh Manocha (left) and Ming Lin. Photo by Steve Exum; ©2008 Endeavors.

But how can a simulation be anything like the real thing? Well, the more realistic it is, Lin and Manocha say, the better.

Take shadows. You can create a digital image of a potted plant at the end of a hallway, and most of us would be able to look at the monitor and tell what it's supposed to be. But if you manipulate the image to add in detailed, realistic shadows, we'll be able to tell what the lighting is like in the hallway, how close the plant is to the wall, and how close it is to where we're standing.

The simulation of sound in these virtual or training environments is especially important to get right, Lin says. "Soldiers use sound cues all the time. A lot of the sounds you hear in the movies are actually prerecorded, and they can be very expensive and dangerous to record — for example, explosions. And so we're using computers to simulate those sounds." By designing new algorithms, they try to get realistic sound propagation: Is that voice coming from a tiny room or a deep chamber? How big was that piece of glass that just shattered, and from how high did the pieces fall?

To make all this happen in a virtual environment — and happen *fast* — the software designer has to develop very efficient algorithms. An algorithm is a set of rules to make the software do certain things: first the foot moves forward, then it touches the ball, then there is a sound and the ball rolls away. Those instructions come in the form of a lengthy mathematical equation.

"That's what computer science is all about, you know," Manocha says. "Designing efficient algorithms."

One of the hardest things about creating a realistic virtual training environment, Manocha says, is that it has to be very fast, just like a video game. "If you have a joystick in your hand, or a mouse, and you take an action, the game responds to you right away."

Here's an example: You're a soldier inside a bombed-out building. It's dark, shadowy, and there's smoke in the air. There are several pounding footsteps on the floor above you, gunshots in the street outside, and the whispery sound of cloth rubbing against cloth — a stealthy movement — coming from just around the corner.

If this were a real-life scenario and the lurker just ahead of you jumped out from the shadows, your mind would give your body the command to defend itself — jump to the side, roll forward, take aim. In order to survive, your body would have to respond to that command instantaneously, in a fraction of a second at most.

That's what makes a realistic virtual training environment; if all of your senses are engaged and the environment is truly interactive — that is, things happen in real time — you can have a completely immersive training session. Things have to be detailed, have to look and feel real.

Some of the most engrossing simulations out there are a product of the video-gaming industry. Today's video and computer games can be so real, so intense, that more than one of them has given players nightmares. And gamers love that.

Software developers designing algorithms for simulation love it, too, and that's one of the reasons they sometimes borrow concepts from the video game industry. "The whole concept of a video game is that it runs at an interactive rate, and users are immersed in it because the display looks real and the behaviors feel real," Lin says. "And so we're thinking that if we develop equally efficient engines, like those that are behind the design of all these games — the graphics engine, the display engine, the physics engine, the behavior engine — we could start modeling all types of behavior." And Lin and Manocha *do* model different types of behavior; their work involves what they call simulation technology for dual use, which means that it can be used for dozens of things beyond defense.

Knowing how people and things move, interact, and function means that you can use different sets of algorithms to design cities, power plants, and commercial airplanes — Lin and Manocha's team worked closely with Boeing to make algorithms to model the 777 and 787 airplanes. And medical students are beginning to use virtual environments to learn how to perform surgeries; they can rehearse everything from incisions to suturing this way.

"Surgeons need to have their hands practiced," Lin says. "And you don't want to be the one they practice on."

Aside from stitches and open heart surgeries, there are plenty of things you don't want to have to learn on the fly. One of them is how to evacuate your collapsing office building.

Manocha points to a computer screen where dozens of colored cylinders are bumping and scooting around inside what looks like a maze. It's a floor plan, Lin says. The breaks in the lines are doorways, and the cylinders are people. ("In modeling these," Manocha says sheepishly, "we don't worry about looking realistic like in the Pixar movies. We want to perform these simulations in real-time, unlike Hollywood studios that take minutes to generate a single frame.")

Then he says, "Say you're designing a new building, and there are four floors and a hundred people in each floor. If there were two emergency exits, how long would it take to evacuate this building? How long if there are four exits? What is the hallway like leading to the exit? How large is the average person? These are the kinds of questions you want to ask *before* you design the building."

He pushes a button and the cylinders mob the doorway at the bottom of the screen. "You know how people go through exits," Manocha says. "The whole crowd gathers. There's a little bit of pushing going on. So the question is, if I made this doorway twice as wide, would my evacuation time go from four minutes to two minutes?" And with just a few tweaks to the algorithm, he and Lin can play with the layout of the floor plan, change the number of exits, make the cylinder-shaped people slimmer or fatter.

"Similar kinds of training technology can be used for training first responders, emergency response, police — things that are really important to what people call homeland security," Lin says. "It's not necessarily about going out there and fighting with another country, but even defending our own country, being able to respond to natural disasters. Some of the work we do with simulating crowds is targeted toward that area."

A good understanding of mob psychology, crowd flow, and transportation issues can help computer scientists develop simulations to predict how huge numbers of people — we're talking hundreds of thousands, entire cities' worth — will react in emergency situations.

Where do people go during a crisis? How do they get there? How well do they drive when they're panicking and in heavy traffic? How do crowds in a subway bottleneck to squeeze onto a subway car? How do people weave through crowded sports arenas without bumping into each other? Whether it's during a terrorist attack or a natural disaster, knowing the answers can help emergency responders do their jobs more quickly and effectively.

For example, officials in cities that are at high risk for flooding or storm damage could use simulated models of pedestrian flow — another project Lin and Manocha are working on — to design better roads and freeways. For that, the programmers need to know such things as how quickly most people cross the street, and how much wider the crosswalk would have to be to shave fifteen seconds off the crossing time.

As the start of the hurricane season approaches, Lin and Manocha are getting ready to work with Carolina's Renaissance Computing Institute to tackle a big question: Exactly how long would it take to evacuate the coastal city of Wilmington? No one really knows yet, Manocha says, and although a lot of their research is still in its youth, crowd simulation technology may provide the answer.

"People think games are fun, that games are for our kids," Manocha says. But simulation technology — whether it's designed to score points or to keep you safe — is a serious industry. The field is growing, too, especially in North Carolina. In October of 2007 in the Triangle area alone, Lin says, gaming companies reported roughly two hundred open positions — these are high-paying software development gigs. It's the same with defense contractors — companies and even the government can't fill the jobs fast enough.

Dinesh Manocha is the Phi Delta Theta/Matthew Mason Distinguished Professor and Ming Lin is the Beverly W. Long Distinguished Professor, both in the Department of Computer Science in the College of Arts and Sciences. Their work is funded by the Defense Advanced Research Projects Agency, the U.S. Army Research Office, the U.S. Army Research, Development, and Engineering Command, the Intelligence Advanced Research Projects Activity, the Office of Naval Research, the National Science Foundation, the Intel Corporation, the Walt Disney Company, the Microsoft Corporation, and the U.S. Department of Defense.

Other Carolina researchers working on realistic virtual training environments include Fred Brooks, Henry Fuchs, Greg Welch, Mary Whitton, and Herman Towles. Their teams include more than fifty undergraduate and graduate students.

Learn more:

- **dinesh manocha.**
- **ming lin.**
- **computer science at unc.**
- **UNC's gamma research group.**
- browse our archive for more stories in **computer science**