

The war was going well. City after city had fallen before my advancing army. Although I had been forced to commit nearly all my military units, I was now within striking distance of the enemy capital. Suddenly, disaster! A transport ship had sneaked past my naval blockade and was unloading a division of tanks onto the shore beside my own capital. The city was far from the front, and I had thought it safe, but now I watched in horror as the enemy easily overpowered my troops and occupied the metropolis, plunging my nation into chaos.

I had just been outwitted by an artificial intelligence (AI).

The video game in which this AI was embedded was *Civilization III* [see screenshot, opposite page]. The creation of Firaxis Games (Hunt Valley, Md.), *Civilization III* sets the player and various computer opponents to fighting military, economic, and cultural battles across a randomly generated set

of continents and oceans. While game AIs are most visible when they take on the role of an opponent to a human player, they can also be used to work in the game's background, making sure the digital environment runs smoothly, and even on the player's behalf in roles like tutors or wingmen.

An increasing number of other games are taking advantage of changes in computer architecture and the growth in processing power to become smarter than ever before. Although most of these games use relatively unsubtle AI techniques, a few pioneers are even showing the academic AI community a trick or two.

While AI originated in the laboratory, it has now been co-opted by designers of video games, and work is under way on increasing the learning powers of a video game's cast of char-

BY STEPHEN CASS
Associate Editor

Mind Games

To beat the competition, video games are getting smarter



acters and refining their social interactions with one another and with human players, too. At this point, even cinematographers and the military are showing interest in possible applications.

Substance over style?

Video games are big business. In 2001, U.S. sales of 225 million units generated US \$6.35 billion, according to Interactive Digital Software Association (IDSA, Washington, D.C.), a trade organization for the industry.

Traditionally, the heaviest emphasis in creating a video game has been on strong visual images, not least to entice people looking at screenshots in a magazine review or on a product box. Ultimately, this emphasis on the visual advanced not just the software used by the games, but the hardware available as well [see “The Wizardry of Id,” *IEEE Spectrum*, August, pp. 42–47].

Now, though, as realistic and detailed three-dimensional graphics become commonplace, “you’ve got to find something else to make your game stand out,” explains Steven M. Woodcock. The editor of GameAI.com, a Web site devoted to the small but growing game AI community, Woodcock noticed a growing interest in building better game AI in the mid-1990s, when talks on the subject started drawing packed crowds at the industry’s premiere gathering, the annual Game Developers Conference (GDC). By 1998 developers “were trying just everything they could think of...there was a lot of touting that this AI is going to be the last AI you’ll ever play,” says Woodcock. “That didn’t work out.”

At the time PCs and video game consoles still lacked the computational horsepower to deliver on the promise of AI. In addition many techniques in vogue, such as neural networks, were ill chosen or misapplied by game programmers. “Most game programmers...don’t know how to take techniques from academic AI research and make them work in a practical way,” says Steve Rabin, editor of the book, *AI Game Programming Wisdom*. “Typically, academic work goes into games only when someone with a master’s or a Ph.D. moves to a game company and brings their experience with them.”

Fortunately, most graphics processing had by then moved onto dedicated graphics cards, and CPU resources and memory—already increasing dramatically, thanks to Moore’s law—were being freed up for computationally intensive and hitherto impractical tasks, such as better AI.

Quite a difference

Game and academic AI development are different, agrees Soren Johnson, who works at Firaxis and was responsible for the AI in *Civilization III*. In games “it’s like this big loop: build a little bit of AI, see how it does, make it better, see how it does, continually and continually testing,” he says, “whereas in the academic community, there is no real reason to do any sort of iterative process. The important thing there is that your AI is as close as possible to the model that you’re trying

to study.” Academic AIs have a related advantage over their game cousins—they do not have to be shipped on time as an integrated component of a polished product.

“The hardest thing in game AI is just making sure that the game never looks dumb. You’d be better off having an AI that was just above average all the time, rather than one that was brilliant 98 percent of the time and stupid 2 percent of the time,” points out editor Rabin. Neural networks and genetic algorithms might seem to be useful (the former attempts to model how neurons behave in the brain, and the latter uses digital evolution and selection to develop a solution to a problem by trial and error). But when “something misbehaves with one of these technologies, it’s not easy to fix. You can’t exclude the one thing that’s broken without destroying all of the other beautiful things in there. It’s all or nothing, which is a very difficult situation when deadlines approach,” he concludes.



MANAGING A CIVILIZATION. Screen of the strategy game *Civilization III* allows the player to manage a city’s food supply, industrial production, luxuries, and aspects of commerce. Or the player can hand off management chores to an AI-based “city governor.”

Currently the mood of the game industry is pragmatic. Rather than rush to new technologies at the 2002 Game Developers Conference, “developers were mostly focused on digesting what they had,” Woodcock told *Spectrum*. Two tried and true technologies dominate game AI today: finite-state machines and scripting.

Finite intelligence

Non-player characters (NPCs), or agents, such as monsters, enemy guards, or friendly wingmen, can be controlled by a finite-state machine. In this AI model, a limited number of behaviors, or states, are pre-defined by the game designer. The actual state that the NPC is in is made to switch with seeming intelligence, depending on criteria also pre-defined by the designer. These could include such things as whether the NPC can see the player or how much ammunition it has left. To add an element of unpredictability to NPCs, many designers incorporate fuzzy logic or random weighting into the decision-making process.

Consider a simple finite-state machine for a hostile, sword-carrying monster guard [see illustration, below]. The monster begins in the Wait state, in 100 percent health, and stays put until the player comes into view. It then enters a Charge state and charges toward the player, until it's within striking range, when Charge changes to Attack and the monster tries to kill the player with its sword. If it succeeds, it will perform a Victory Dance. Game over.

If at any time the player's counterattack reduces the monster's health to less than 20 percent, the monster will shift to a Flee state and run away from the player. Flee lasts until either the monster is dead or it can no longer see the player, at which point it shifts back into the Wait state.

Scripting is a technique in which the control of game events and NPCs is not hard coded into the game, but defined using a high-level language. Scripting allows game designers—who set the overall goals and playing methods of the game—to work semi-independently of the programmers, who work on the nuts and bolts implementation of the games, shortening the development process.

To allow scripting, the game engine, which maintains and updates the data structure that defines the game environment, usually has an interface that understands a collection of primitive commands and queries that change the position of an NPC, use one kind of animation or another, determine what key the player is pressing, and so on. Feeding the engine different scripts produces different behaviors. In this way, a game company can even use the same game engine to produce two completely different games by designing different scripts.

Scripts can vary widely. As the name implies, they can be a fixed, linear sequence of actions and are often used as such to direct so-called cut scenes, where some piece of exposition is presented to the player—for example, having a wizard NPC explain what a magic item does. Or they can be complex programs in themselves, even implementing finite-state machines or more exotic types of AI.

Scripts can be written in a special-purpose language created for a specific game, or in standard general-purpose languages such as Perl or Java. For the sake of efficiency, the high-level script is not interpreted while the game is running. Instead, the script is pre-compiled into a stream of low-level codes corresponding to the various primitive commands understood by the game engine.

If you can't beat 'em, cheat 'em

Game AIs also have two big advantages over their academic counterparts—they can cheat, and they can get a lot of help from the world they find themselves in.

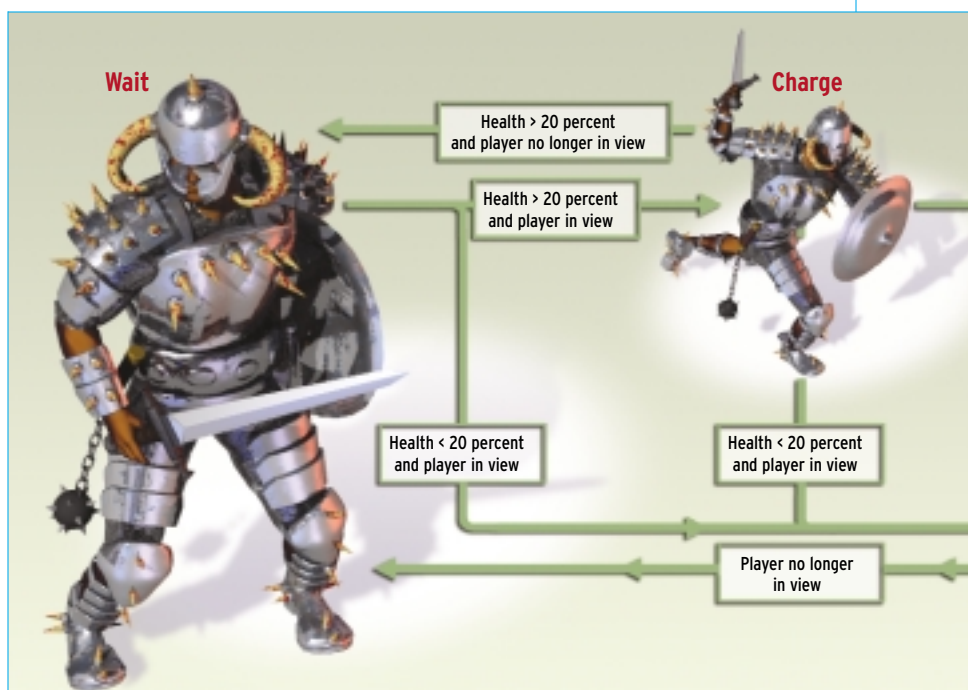
AIs cheat by having some of the constraints the player must face removed from them. To build a tank in a military strategy game, for instance, the computer may need to gather only half as many resources as the player.

"It's OK to cheat, so long as the player never knows you're cheating, or you tell them up front that the game cheats in order to be a worthwhile opponent," says Rabin. Woodcock agrees: if you don't tell them and then they "figure out that it cheats, it really frustrates the players and makes them mad. You see it in letters to the editor of the game magazines."

Also unlike academic AIs, which are built with a view to coping with the real world's messy lack of clues, game AIs operate in the much neater and simpler realm of the game designers' creation. Clues and commands can be built into these worlds. Imagine the problem of moving a patrolling guard through a number of rooms in a castle. The designer can sidestep the problem of teaching the guard how to work out the best path between rooms by simply drawing an invisible track for the guard to follow through the castle.

The ability to cross the traditional division between an AI and its environment allowed Will Wright at Maxis, a division of Electronic Arts Inc. (Redwood City, Calif.), to turn the concept of an AI inside out. In Wright's game, *The Sims* [see screenshot, next page], one of the most popular of all time, players run a modern version of an ant farm by overseeing the lives of a number of autonomous simulated people known as Sims. The players can build houses for their Sims and fill them with objects such as television sets and pets to keep them occupied. New objects can be created and shared over the Internet. "You download a beach ball and your Sims will start playing with it," says Woodcock.

But the intelligence to interact with any of these objects is not built into the Sims themselves. Rather, they are equipped with a few basic needs, for food, say, or entertainment. Objects within the game advertise their ability to satisfy some of these needs to any Sims who wander nearby. Wright calls it "smart terrain. A Sim is looking around for fun and spots the beach ball...and all the instructions about how to bounce the ball are there as part of the ball's code," he continues.



Well done is more fun

It is strategy and open-ended games like *The Sims*, where game play soon goes beyond any developer’s capacity to pre-program situations, that have the greatest need for advanced AI technologies. “Open-ended games are becoming fashionable because they’re just more fun,” says Richard Evans of Lionhead Studios (Guildford, UK). “Instead of restricting you to a linear path, they just simulate a world and enable you to emerge with whatever behavior you want.” Evans notes that “people [publish on the Internet] the little stories they’ve experienced in these games because they’re unique.” Lionhead Studios is best known for its AI-based game *Black and White*, in which the player takes on the role of either a vengeful or benevolent god watching over its followers and competing with rival deities for complete control [see screenshot, next page].

Two other genres stand to benefit from improved game AI: the increasingly popular massively multiplayer role-playing games (MMRPGs) and so-called first-person shooters. In the first, players take on the personas of adventurers in a persistent Internet world (meaning one that exists independently of the players’ presence or absence). The first-person shooters typically pit players against each other in kill-or-be-killed shootouts.

Woodcock explains why a world filled with human players would benefit from better AI. “Everyone wants to be the hero, the Captain Kirk character. No one wants to be the guy in the red shirt who’s killed right away....That means you need supporting characters.” A current MMRPG like *Ultima Online* does supply supporting characters such as shopkeepers for its medieval fantasy world, but the presence of human-controlled (albeit graphically identical) characters side by side with the AI-based supporting cast makes the limited AI of those agents conspicuous. Dealing with the agents is a “jarring experience...so the human players congregate together and ignore all the hard work the developers did to make a believable village,” says Woodcock.

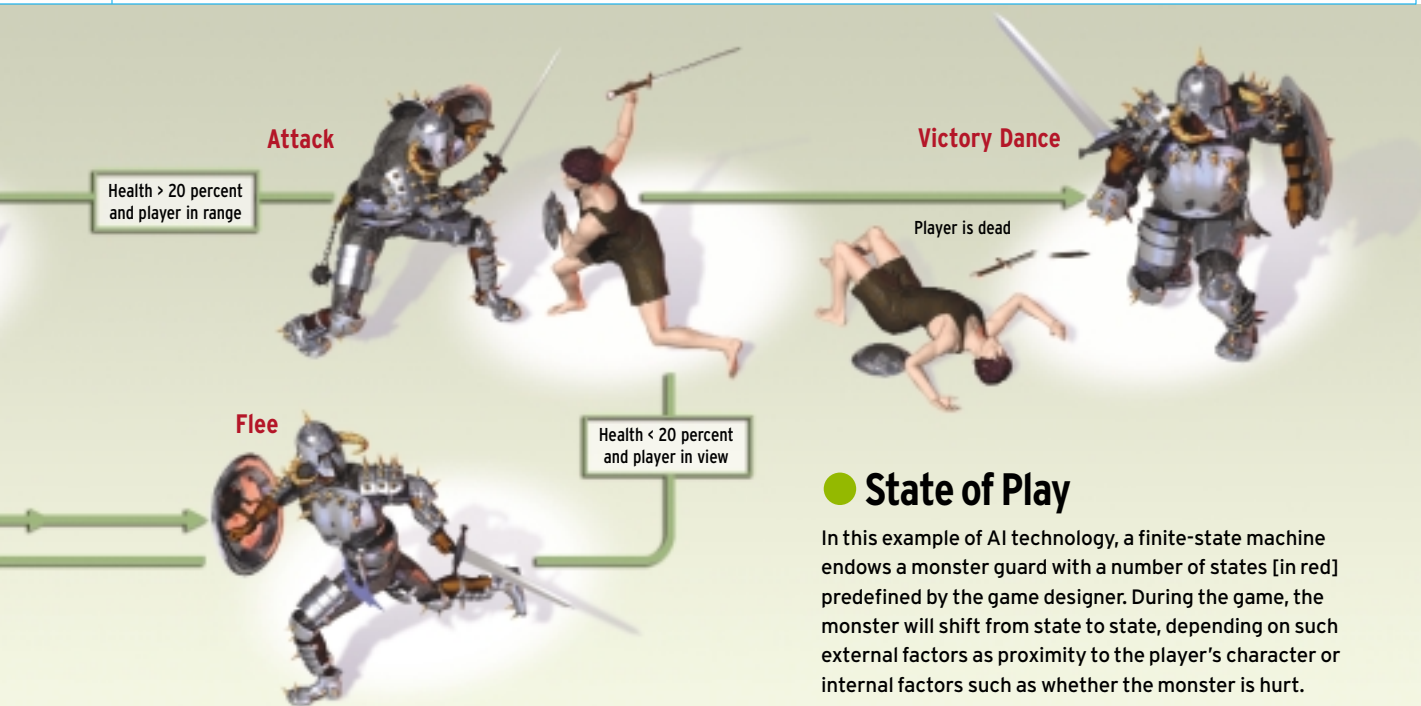


A 21ST CENTURY ANT FARM: In *The Sims*, players oversee the lives of simple artificial people who can interact with others and objects in their environment in unexpected ways.

Adaptive technologies, such as neural networks, which learn from experience, may after all play a role here. They will allow agents to respond to players and accommodate changes in their behavior without the need for the MMRPG’s designer to explicitly program every possibility into the agent.

First-person shooter games such as *Duke Nukem 3D* and *Quake* had their beginnings as single-player games, but soon latched onto the appeal of multiplayer gaming. “There was a phase where developers were going, ‘Hah, we don’t need a good AI because players are going to play with other people,’” says Woodcock.

They were wrong. Players often can’t get a fast enough Internet connection to compete, and even when they can, “people want a good single-player experience to practice before they go on-line because they feel stupid when they get their butt kicked by a 12-year-old in Ohio,” laughs Michael Zarozinski.



● State of Play

In this example of AI technology, a finite-state machine endows a monster guard with a number of states [in red] predefined by the game designer. During the game, the monster will shift from state to state, depending on such external factors as proximity to the player’s character or internal factors such as whether the monster is hurt.

Zarozinski is the founder of Louder Than A Bomb! Software (Gardner, Mass.), which sells a relatively new product—AI middleware. Ideally, AI middleware would provide game developers with a ready-made AI that they could easily customize and drop into their game to control agents. Middleware like Zarozinski's *Spark!* software may become a vital tool in managing the complexities of creating modern games.

Building the pipeline

Middleware “hides all the ugly nasty little details” of implementing an AI, explains Zarozinski. Developers “can have *Spark!* running alongside the game...and tweak the rules in real time and see the impact of the changes without having to stop and recompile,” as they would with an AI built directly into the game. For example, the precise noise threshold that activates a guard's attack pattern can be tuned so that a player can open a door and creep up on the guard without alerting it, whereas firing a weapon nearby will give the player away. Middleware allows the developer to do this tuning without having to reprogram the AI directly.

One of Zarozinski's biggest challenges in selling his middleware has not been technical, but psychological. “There is a huge not-invented-here syndrome in the game industry. People do not want to buy middleware, they want to write it themselves,” he told

Spectrum. In the last couple of years, though, the situation has improved, thanks to the example of commercially available third-party graphics middleware such as the *Quake* or *Unreal* engines.

Still, there are those who are skeptical that generic AI middleware will become ubiquitous. “It's a noble pursuit,” says Rabin, but “AI relies so heavily on a game's data structures, it's difficult to create efficient middleware in the absence of well-established standards for those structures in the industry.”

AI.implant is an AI middleware product sold by BioGraphic Technologies (Montreal). Its president, Paul Kruszewski, acknowledges that his software cannot be used in every type of game. “For example, sports games are extremely specific, there's so many hacks” needed to reproduce all the rules of the particular sport, he said, noting that because *AI.implant* is centered on controlling relatively sophisticated individual agents, it's also not suited for real-time strategy games in which many simple, identical units must be managed. “But we are excellent at things like racing games or action/adventure games,” says Kruszewski.

Although initially targeted at game developers, Kruszewski adds, *AI.implant* is finding applications outside games. Two large studios in California have recently signed on to use the software to generate special effects, such as large crowd scenes in movies. “The film people want intelligent extras,” he explained.

There has also been interest from the military in using it in training simulations. “We have serious enquiries from the U.S. Marine Corps,” says Kruszewski [see “Games Soldiers Play,” *Spectrum*, March 2002, pp. 32–37].

Better, smarter, faster

The next big challenge for game AI may be getting a game's cast of characters better at learning and social interaction. Here, Richard Evans at Lionhead Studios is leading the charge. Lionhead's *Black and White* god game broke new ground by giving the player an in-game representative, a Creature. The player's Creature acts autonomously, but it can be trained by a system of punishments and rewards. Most notably, it watches the player's actions and attempts to divine the intent behind them, a technique called empathic learning. Suppose you throw fireballs

against another tribe's village; then the Creature will guess that you don't like foreign tribes and spontaneously try to trample foreigners. Later it can be taught the difference between hostile and friendly foreigners and to trample only the former. Rabin notes that he's seen recent academic presentations on AI learning that have only “a fraction of what Richard's done with *Black and White*.”

For a project code-named Dmitri, Evans is now focused on improving the ability of AI to interact socially. Agents' behavior will be controlled by their membership in overlap-



HOUSETRAINING: In *Black and White*, the player plays a god. The game features an autonomous AI-based Creature that can be trained to act on the player's behalf.

ping social groups. Each group will have its own set of needs and attitudes toward other groups. Its members reduce demands on memory and processing power by sharing information among themselves, and because they belong to multiple groups to varying degrees, the agents are individualized. Evans expects to release a working game in two or three years.

Another attempt at good AI on a large scale is being pioneered by Majorem Ltd. (Midreshet Sde Boker, Israel). Majorem is developing an Internet-based massively multiplayer strategy game called *Ballerium*. Strategy games, which operate in real time (as opposed to a game in which the computer and player take turns, such as *Civilization III*), are the most computationally demanding of all AI-based games. While most multiplayer online games use the connected players' client software just to render graphics and transmit commands, *Ballerium*'s ambition is to harness the power of the computers of logged-in players. Difficult problems, such as periodically evaluating a tactical position, will be farmed out to client computers in a novel example of distributed computing.

But finally, the value of all game AI technologies is measured against one simple standard. “Players want to feel like they were almost beat and then triumphed,” says Rabin. “Players don't really care about AI itself, only about whether a game is any *fun*.” ●



PAUL E. GREEN JR. (LF) [p. 13] is a communications consultant based in Mount Kisco, N.Y., and former president of two IEEE Societies: Information Theory and Communications. While at the MIT Lincoln Laboratory in the 1950s and 1960s, he introduced the first operational spread-spectrum system and invented planetary radar range-doppler mapping.

ROGER PARLOFF [p. 16, 17] has written for *The New York Times Magazine*, *The Wall Street Journal*, and *Harper's*. Earlier, he was a staff writer for *The American Lawyer* and *Brill's Content*.

ALAN S. BROWN [p. 20] is a freelance writer who has written extensively about advanced materials, technology, and technology management (AlanSBrown@comcast.net).

JONATHAN BRODER [p. 21] is senior defense and foreign policy editor at *Congressional Quarterly*, an independent news organization in Washington, D.C.

JEN LIN-LIU [p. 22], a journalist in Shanghai, writes for *The Chronicle of Higher Education*.

WILLIAM LEVENTON [p. 28] is a technology writer based in Somers Point, N.J. Over the last 15 years, he has written hundreds of articles for a variety of magazines, newspapers, and Web sites.

ROBERT CARLSON [p. 46] works on biological technologies in Berkeley and Seattle. He is currently a Visiting Scholar in the Comparative History of Ideas Program at the University of Washington.

SCOTT KARIYA [p. 52] writes frequently about career strategy as a corporate recruiter in New York City (stkariya@yahoo.com).

TERRY COSTLOW [p. 56] writes about the electronics industry, covering technology and its societal impact.

PAUL MCFEDRIES [p. 68] is a technical and language writer with over 40 books to his credit. In addition, he runs Word Spy, a Web site and mailing list that tracks new words and phrases (<http://www.wordspy.com/>).

AT IEEE SPECTRUM ONLINE

Go to <http://www.spectrum.ieee.org> for:

Tissue Engineering's Troubles

Bankruptcies Rock Biotechs

Pioneering companies take a financial beating

PLUS:

California Warns of Powerline Health Risks

Yet final report's methodology may be faulty

Startup Company Has Holographic Screening System

Could supplement metal detectors, body searches

Pentagon Refines Thermobaric Bombs

Can they take out buried weapons of mass destruction, or are they too indiscriminate?

EDITOR Susan Hassler

EXECUTIVE EDITOR Glenn Zorpette MANAGING EDITOR Alfred Rosenblatt
SENIOR ENGINEERING EDITOR Michael J. Riezenman SENIOR NEWS EDITOR William Sweet

SENIOR TECHNICAL EDITOR Linda Geppert SENIOR EDITOR Tekla S. Perry
SENIOR ART DIRECTOR Mark Montgomery

ASSISTANT MANAGING EDITOR Elizabeth A. Bretz

SENIOR ASSOCIATE EDITORS Steven M. Cherry, Harry Goldstein, Jean Kumagai
ASSOCIATE EDITORS Stephen Cass, Samuel K. Moore

ASSISTANT EDITOR Willie D. Jones

CHIEF COPY EDITOR Margaret Eastman COPY EDITOR Sally Cahur

EDITORIAL RESEARCHER Alan Gardner

ADMINISTRATIVE ASSISTANT Nancy T. Hantman

IEEE FOUNDATION JOURNALISM INTERN Holli A. Riebeck

CONTRIBUTING EDITORS Jonathan Broder, Alan S. Brown, Robert Carlson,
Terry Costlow, Scott Kariya, William Leventon, Jen Lin-Liu,
Robert W. Lucky, Paul McFedries, Roger Parloff

ADMINISTRATOR, EDITORIAL SUPPORT SERVICES Rita Holland

SENIOR EDITORIAL ASSISTANTS Ramona Gordon, Desiree Noel

ASSISTANT ART DIRECTOR Michael R. Vella

DIRECTOR, PERIODICALS SERVICES Alan B. Schafer

DIIRECTOR, PERIODICALS PRODUCTION SERVICES Robert Smrek

EDITORIAL & WEB PRODUCTION MANAGER Roy Carubia

ELECTRONIC LAYOUT ARTIST Darlene Alberti

WEB PRODUCTION COORDINATOR Robin Edwards

WEB PRODUCTION ASSISTANT Jacqueline Parker

IEEE MEDIA

PUBLISHER James A. Vick

ASSOCIATE PUBLISHER, SALES & ADVERTISING DIRECTOR Marion Delaney

BUSINESS DEVELOPMENT MANAGER, SPECTRUM ONLINE Michael Buryk

BUSINESS MANAGER Robert T. Ross

MARKETING & PROMOTION MANAGER Blanche McGurr

MAIL LIST SALES MANAGER Iliia Rodriguez

LIST SALES ASSISTANT Simra Davis

REPRINT SALES +1 800 482 1450

ADMINISTRATIVE ASSISTANT Malinda Bell

ADVERTISING SALES +1 212 419 7760

ADVERTISING PRODUCTION SUPERVISOR Felicia Spagnoli

ADVERTISING PRODUCTION COORDINATOR Nicole Evans

ADVERTISING PRODUCTION ASSISTANT Deborah Grant

ADVERTISING PRODUCTION +1 732 562 6334

STAFF EXECUTIVE, PUBLICATIONS Anthony Durniak

EDITORIAL BOARD

Susan Hassler, *Chair*; Tariq S. Durrani, Kenneth R. Foster, Alex Hills,
W. Dexter Johnston Jr., Pradeep K. Khosla, Stuart A. Long,
Raymond C. Montgomery, M. Granger Morgan, Jon M. Peha,
S. Tom Picraux, Mark Saffian, Larry Stern, Lewis M. Terman, Jill Tietjen,
Harry L. (Nick) Tredennick, Michael W. Vannier, Yervant Zorian

IEEE SPECTRUM HEADQUARTERS

3 Park Ave., 17th floor, New York, NY 10016-5997

Tel: +1 212 419 7555 Fax: +1 212 419 7570

www.spectrum.ieee.org

BUREAU: Palo Alto, Calif. Tekla S. Perry, +1 650 328 7570