Connoisseur of chaos

August 29 2012

Russ Tedrake*Photo: Dominick Reuter*

As a high school student in a Detroit suburb in the 1990s, Russ Tedrake did not fit the standard profile of a future computer science professor. Although he had a talent for math—"I won some of the little math competitions," he says—he spent his spare time playing football or soccer with friends rather than hacking code or even playing video games; in fact, he didn't get his first computer until he was a senior.

He got good grades, but he didn't find the work very demanding. The only calculus class offered at his school was geared to the easier of the two Advanced Placement tests offered by the College Board—although, against the advice of his teachers, Tedrake took the harder test anyway and did well on it. "I just sort of coasted through," he says.

Tedrake's father, who is from England, had moved to Michigan after

General Motors bought his employer, a British car company. But Tedrake's mother is a Michigan native, and, says Tedrake, "I was raised with maize-and-blue blood." So after graduation, he naturally headed to the University of Michigan.

At GM, Tedrake's father had developed antilock brakes for light trucks—"The joke was that my mousetrap car in high school didn't go very far, but it could stop on a dime," Tedrake says—and Tedrake had always assumed that he, too, would be an engineer. He enrolled at Michigan expecting to major in mechanical engineering, but when he took his first programming class—a prerequisite for most engineering majors—something changed. "Ultimately, I just like to build things," Tedrake says. "And the fact that I could build basically anything I wanted quickly on a computer was very appealing to me."

Tedrake had always had a passion for robotics, but at the time, Michigan didn't offer undergraduates much opportunity to work directly with robotic hardware. So he concentrated on the next-best thing, working on the algorithms that controlled the virtual robots—and humanoids—in video games and graphics programs.

Freshman at Ford

Tedrake's biggest chance to work with robots as an undergrad, in fact, came the summer after his freshman year, when he got an internship at a small company in his hometown that helped car manufacturers improve the efficiency of their paint shops. Tedrake spent the first part of the internship "climbing through paint sludge," but he somehow convinced his managers that he was just the person to develop a new control system for the paint shop air vents at Ford's Fort Wayne assembly plant—even though only a year earlier he hadn't known anything about computer programming.

"You've got a big robot shooting paint out of a gun, and there's some flow going through the chamber, and if it's not right, a lot of the paint is going to go down into the sewers," Tedrake explains. "You could actually save a lot of money if you just optimized the atmospheric conditions."

The system worked "fabulously," Tedrake says—until the day a paint shop worker tripped on an Ethernet cable and pulled it out of the computer running Tedrake's program. Sensibly, Tedrake had designed the system to shut down if anything went wrong, so the fans shut off, the temperature in the paint shop rose to a level that, under union rules, warranted evacuation, and the production line shut down for 10 minutes. "I got reamed by the line manager for about two hours about how much money I'd cost Ford," Tedrake says.

The internships Tedrake took during his remaining summers as an undergrad afforded him no less responsibility, but they didn't involve robots. For two summers, he worked at Microsoft, first on an image-viewing system for mobile phones and then on artificial-intelligence agents for video games; in both cases, some of his code found its way into commercial products.

He spent the summer after his senior year of college at the Santa Fe Institute in New Mexico, developing computer simulations of networks of neurons, and continued the work for an extra semester at Michigan, earning a master's in computer engineering. The following fall, he entered the PhD program in MIT's Department of Electrical Engineering and Computer Science, but did his research in the lab of professor Sebastian Seung in the Department of Brain and Cognitive Sciences.

Tedrake had chosen to investigate how humans learn to walk, and he and Seung spent some time discussing the empirical data on the topic, which was difficult to acquire and even more difficult to interpret. Ultimately, they concluded that the best approach was to build humanoid robots and try to train them to walk like humans. Through as roundabout a route as possible, Tedrake had become a roboticist.

Learning to walk

Tedrake began his new experiments with a variation on a children's toy, a wooden figure with moving legs that, placed at the top of a ramp, will walk down it, powered only by gravity. "Everybody's intuition is that if you have something that can walk down a ramp with no control, it should be easy to make it walk on the flat," Tedrake says. "But that was not true for any conventional control approach."

Although the construction of the ramp walker is simple, its limbs' response to external forces is very hard to describe mathematically. At the time, there were a few commercially available bipedal robots, but they used a large number of actuators not just for propulsion, but also to squelch the natural dynamics of their mechanical legs so that they would move in predictable ways. As a consequence, they used about 20 times as much energy as humans do when walking. Tedrake didn't want to suppress his walker's dynamics; he wanted to exploit them, to make its gait more efficient.

Tedrake reasoned that since the walker's design was clearly practical—under the right circumstances, it did walk—but also very simple, the control algorithm that would make it walk on a flat must be simple, too. The difficulty was in calculating the control signals, since that required modeling the complex dynamics of the walker's joints. So Tedrake didn't calculate them. Instead, he used machine-learning techniques to train the algorithm, and he got the walker to work on the flat.

As it turns out, at about the same time, three researchers at two other institutions had done complementary work on the problem of adding actuators to ramp walkers. All four researchers decided to write a paper together, and it was accepted to Science. At 26, Tedrake had solved an outstanding problem in robotics and published the results in a major journal. After finishing his PhD in 2004, he joined the faculty of MIT's Department of Electrical Engineering and Computer Science.

Tedrake's Robot Locomotion Group at the Computer Science and Artificial Intelligence Laboratory continues to explore how to harness the complex dynamics of mechanical systems to make robotic control more versatile and efficient. The group continues to work on bipedal walkers, but has also branched out to study the dynamics of flight. These days, Tedrake says, he's spending most of his time gearing up for the Defense Advanced Research Project Agency's new robotics challenge: to develop control algorithms that enable a humanoid robot to perform rescue operations during a disaster.

"I have been working tirelessly to make our approaches scale to the complexity of a full humanoid. I have humanoids on my screen any time you look," Tedrake says. "We're going to try to smash the control problem for that."

This story is republished courtesy of MIT News (web.mit.edu/newsoffice/), a popular site that covers news about MIT research, innovation and teaching.

Provided by Massachusetts Institute of Technology

Citation: Connoisseur of chaos (2012, August 29) retrieved 7 May 2024 from https://phys.org/news/2012-08-connoisseur-chaos.html

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.