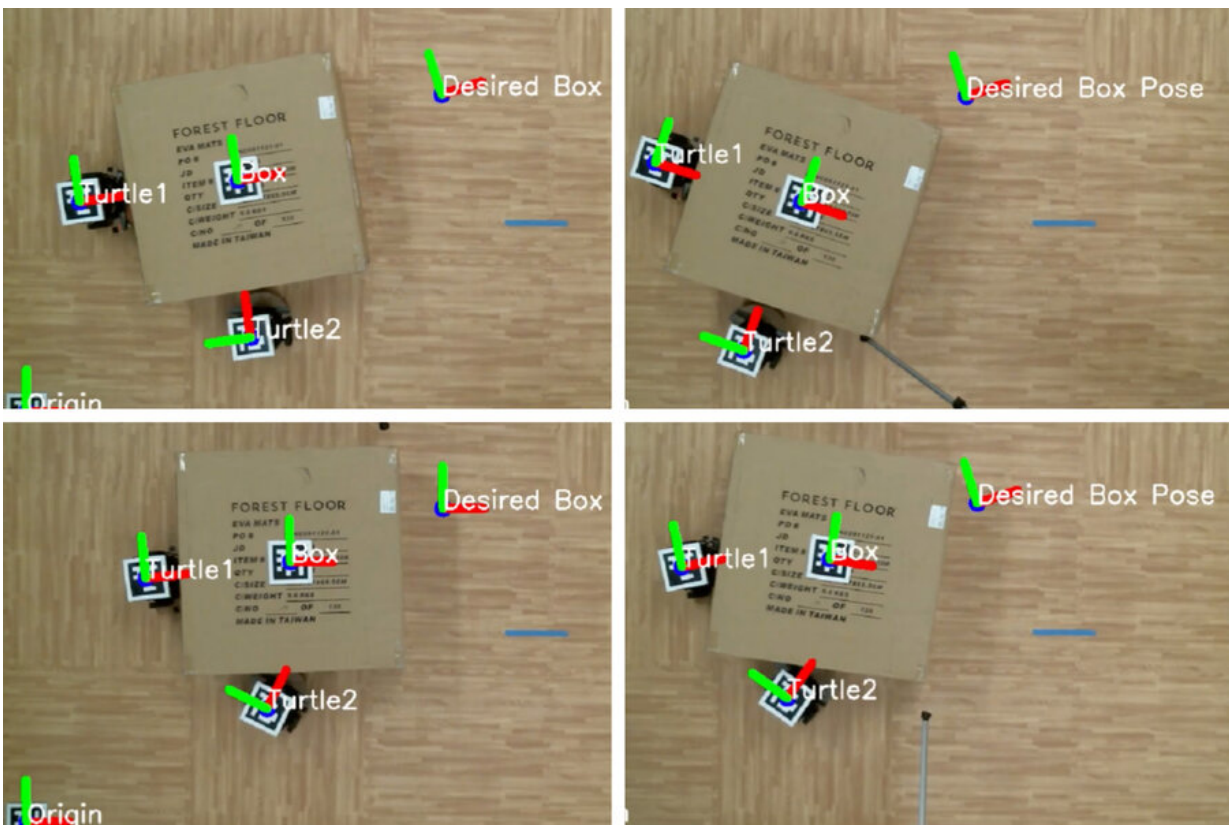# Engineers devise a recipe for improving any autonomous robotic system

June 21 2022, by Jennifer Chu

A new general-purpose optimization tool can improve the performance of many autonomous robotic systems. Shown here is a hardware demonstration in which the tool automatically optimizes the performance of two robots working together to move a heavy box. Credits: Courtesy of the researchers

Autonomous robots have come a long way since the fastidious Roomba.

In recent years, artificially intelligent systems have been deployed in self-driving cars, last-mile food delivery, restaurant service, patient screening, hospital cleaning, meal prep, building security, and warehouse packing.

Each of these robotic systems is a product of an ad hoc design process specific to that particular system. In designing an autonomous robot, engineers must run countless trial-and-error simulations, often informed by intuition. These simulations are tailored to a particular robot's components and tasks, in order to tune and optimize its performance. In some respects, designing an autonomous robot today is like baking a cake from scratch, with no recipe or prepared mix to ensure a successful outcome.

Now, MIT engineers have developed a general design tool for roboticists to use as a sort of automated recipe for success. The team has devised an optimization code that can be applied to simulations of virtually any autonomous robotic system and can be used to automatically identify how and where to tweak a system to improve a robot's performance.

The team showed that the tool was able to quickly improve the performance of two very different autonomous systems: one in which a robot navigated a path between two obstacles, and another in which a pair of robots worked together to move a heavy box.

The researchers hope the new general-purpose optimizer can help to speed up the development of a wide range of autonomous systems, from walking robots and self-driving vehicles, to soft and dexterous robots, and teams of collaborative robots.

The team, composed of Charles Dawson, an MIT graduate student, and ChuChu Fan, assistant professor in MIT's Department of Aeronautics and Astronautics, will present its findings later this month at the annual

Robotics: Science and Systems conference in New York.

Inverted design

Dawson and Fan realized the need for a general optimization tool after observing a wealth of automated design tools available for other engineering disciplines.

"If a mechanical engineer wanted to design a wind turbine, they could use a 3D CAD tool to design the structure, then use a finite-element analysis tool to check whether it will resist certain loads," Dawson says. "However, there is a lack of these computer-aided design tools for autonomous systems."

Normally, a roboticist optimizes an autonomous system by first developing a simulation of the system and its many interacting subsystems, such as its planning, control, perception, and hardware components. She then must tune certain parameters of each component and run the simulation forward to see how the system would perform in that scenario.

Only after running many scenarios through trial and error can a roboticist then identify the optimal combination of ingredients to yield the desired performance. It's a tedious, overly tailored, and time-consuming process that Dawson and Fan sought to turn on its head.

"Instead of saying, 'Given a design, what's the performance?' we wanted to invert this to say, 'Given the performance we want to see, what is the design that gets us there?'" Dawson explains.

The researchers developed an optimization framework, or a computer code, that can automatically find tweaks that can be made to an existing autonomous system to achieve a desired outcome.

The heart of the code is based on automatic differentiation, or "autodiff," a programming tool that was developed within the machine learning community and was used initially to train neural networks. Autodiff is a technique that can quickly and efficiently "evaluate the derivative," or the sensitivity to change of any parameter in a computer program. Dawson and Fan built on recent advances in autodiff programming to develop a general-purpose optimization tool for autonomous robotic systems.

"Our method automatically tells us how to take small steps from an initial design toward a design that achieves our goals," Dawson says. "We use autodiff to essentially dig into the code that defines a simulator, and figure out how to do this inversion automatically."

Building better robots

The team tested their new tool on two separate autonomous robotic systems, and showed that the tool quickly improved each system's performance in laboratory experiments, compared with conventional optimization methods.

The first system comprised a wheeled robot tasked with planning a path between two obstacles, based on signals that it received from two beacons placed at separate locations. The team sought to find the optimal placement of the beacons that would yield a clear path between the obstacles.

They found the new optimizer quickly worked back through the robot's simulation and identified the best placement of the beacons within five minutes, compared to 15 minutes for conventional methods.

The second system was more complex, comprising two wheeled robots working together to push a box toward a target position. A simulation of

this system included many more subsystems and parameters. Nevertheless, the team's tool efficiently identified the steps needed for the robots to accomplish their goal, in an optimization process that was 20 times faster than conventional approaches.

"If your system has more parameters to optimize, our tool can do even better and can save exponentially more time," Fan says. "It's basically a combinatorial choice: As the number of parameters increases, so do the choices, and our approach can reduce that in one shot."

The team has made the general optimizer available to download, and plans to further refine the code to apply to more complex systems, such as robots that are designed to interact with and work alongside humans.

"Our goal is to empower people to build better robots," Dawson says. "We are providing a new building block for optimizing their system, so they don't have to start from scratch."

More information: Paper: roboticsconference.org/program/papers/037/

Provided by Massachusetts Institute of Technology