



MATHEMATICS



SWARMING DRONES, SELF-DRIVING VEHICLES, AND THE ALGORITHMS THAT LIGHT THE WAY

ANY DRIVER KNOWS THE BIGGEST PROBLEM ON ROADS: OTHER DRIVERS. Each independently—and with highly variable skill—pilots a vehicle weighing an average of two tons at high speed in a complex environment that includes road conditions, weather, obstacles and the constant jockeying of other cars and trucks. Drivers ideally keep plenty of space between them to allow reaction time in case of erratic or surprising developments.

The advent of self-driving cars may minimize some of the risks associated with driver inconsistency. Rather than battling a chaotic jumble of people ranging from sedate slowpokes to crazed speed demons, you may someday—perhaps sooner than you think—belong to what's called a platoon of cars that travel in a self-regulating group. Each car would communicate with cars nearby so they'd be attuned to what others are doing and be able to maneuver optimally to reach a given destination in the most efficient amount of time possible.

He and other Lehigh researchers are at the forefront of research into complex systems dealing with constantly changing variables—an area important to development of robotics, automation and machine learning. “There’s great interest in these areas at Lehigh,” says Subhrajit Bhattacharya, assistant professor of mechanical engineering and mechanics, who explores motion planning and control of autonomous, intelligent systems. “What’s happening here is very exciting.”

DISTRIBUTING CONTROL

Crucial to a system’s integrity, autonomy and efficiency is how it’s controlled. And distributing control across multiple agents can make a network more robust in responding to local failures. Take the example of drones, which can be flown in networked arrays—sometimes referred to as swarms—that have a variety of potential applications such as traffic reporting, surveillance and military missions.

IN MOTION

“We envision a team of high-speed, driverless cars, safely distanced from one another, traveling from city to city,” says Nader Motee, associate professor of mechanical engineering and mechanics. “The most desirable feature for these networks is long-term autonomy. Yet this independence introduces the concepts of risk and fragility.”

Risk and fragility, in fact, are key areas of Motee’s work in design, control and optimization of distributed control and dynamical systems. “When we’re talking about the kinds of systems where you have a group of agents—self-driving cars, drones, power plants—aiming to work as a team to accomplish a task, we’re talking about networks,” Motee says. “These are highly dynamic systems where a small time delay or an incorrectly computed response to an external disturbance can lead to catastrophic consequences.”

Relying on a central control system such as a ground-based command center introduces potential vulnerabilities. “Each agent needs a communication channel with the base, which may be far from where the drones do their mission,” Motee says. “If for some reason there’s a failure at the base—power outage or hackers get into the central server—then all the drones are in trouble.” Fragility is high.

Distributing control among the agents shores up some of these weaknesses. “If drones don’t communicate to a central base but are just in touch with their neighbors and make decisions using local information, you can endure loss of one drone and still go on with the mission because the remaining drones continue to exchange information among each other,” Motee says. “The network is less fragile and uncertain events become more tolerable.”





Nader Motee with Ph.D. students Yaser Ghaedsharaf and Shima Dezfulian, adjunct professor Babak ShirMohammadi, and Ph.D. student Arash Amini (l to r, above); Subhrajit Bhattacharya and students develop algorithms that guide the motion of robots and other devices and vehicles (opposite.)

FORESEEING UNCERTAINTY'S OUTCOMES

Yet vulnerabilities remain. Part of what can make systems fragile is what researchers call noise—variables from outside the system that can degrade performance or make agents fail, increasing the risk that remaining nodes will also fail in a domino-like cascade. A car platoon that hits a road obstacle might suffer a collision that becomes a pileup. A fleet of drones flying south to north that hits a strong east-west wind might get thrown off course.

Sensing and communication become important elements in overcoming noise—but too much communication between agents becomes its own vulnerability. “As agents talk more and establish more communication channels, noisy information circulates more and gets amplified across the network,” Motee says. So does the potential for intruders to hack the network and disrupt signals.

“Using our framework, relevant features can be abstracted away from these domain-specific applications into dynamic models suitable for risk and fragility analysis.” —Nader Motee

“You want to minimize the number of talking points across the network by dropping communication links but at the same time not lose performance,” Motee says. This process, known as sparsification, presents difficult mathematical problems related to identifying where a system is most likely to fail and which channels between which agents should be used. “We have come up with some tractable algorithms to solve this inherently difficult problem,” Motee says.

One approach is to use randomized algorithms. “We randomly pick a pair of agents and drop their communication channel, but in a way that allows performance of the network to remain the same,” Motee says. So while performance doesn’t drop, complexity does.

Motee and his students have also developed a computational framework based on systemic risk measures that can be applied to design a wide variety of networks. Systemic risk measures allow for constraints and uncertainties to be treated

in terms of safety margins instead of exact requirements.

“Our methodology allows for all these seemingly different problems to be viewed in a rather unified manner,” Motee says. “Autonomy and fragility issues in self-driving cars are different from drone problems, which are different from smart-grid problems, but the mathematical challenges are similar. Using our framework, relevant features can be abstracted away from these domain-specific applications into dynamic models suitable for risk and fragility analysis.”

IN THE COLISEUM

Motee’s lab is now researching advanced control capabilities in a 4,000-square-foot Lehigh space known as the ‘Flying Machines Coliseum.’ It’s equipped with a high-precision motion capture system, a central server, a wireless communication network, and custom software that can monitor drones and provide information about their location in space along with dynamical measures such as the roll and pitch of each quadcopter in a networked array.

One new initiative is to design drones that can fly and navigate on their own by learning from what they see—without the aid of GPS. “This is particularly useful in providing fast and reliable information for emergency or search and rescue operations, or even reconnaissance missions,” Motee says. “These drones can navigate through collapsed structures or even hazardous conditions with radiological, biological or chemical contamination.”



Motee’s team is developing a design methodology to optimize drone size, flight controller, and on-board sensors by integrating and co-designing control, navigation, and vision algorithms.

“We blend tools from control theory, probabilistic modeling, computer vision, and embedded systems,” he says.

“We aren’t as focused on the drones themselves or planning their motion as we are on understanding long-term autonomy for a wide range of dynamic network applications.”

Motee’s work is funded through grants associated with his 2016 Office of Naval Research Young Investigator Program award, as well as his 2015 National Science Foundation CAREER award.

MAPPING MOTION

A self-driving car needs to know where it’s going from the start of its journey to the finish and the exact route to travel at every moment along the way. If the road curves, it needs to steer through the turn; if people are in the middle of a street, it needs to avoid them. “This is not only an engineering problem, it’s a mathematical problem,” Bhattacharya says. “Motion planning finds a sequence of action in the system’s configuration space.”

A rudimentary way to compute the motion of a robot navigating on a plane is to construct a line between its start