Northwestern | McCORMICK SCHOOL OF ENGINEERING

Northwestern | BUFFETT INSTITUTE FOR GLOBAL AFFAIRS

Drone Urban Cyber-Defense Testbed

Drones are being increasingly used to carry out terrorist attacks. Groups such as ISIS, the PKK, Boko Haram, the Houthis, and Marawi militants, amongst many others, have deployed multiple drone-based attacks around the world. Independent research says that a total of 76 drone based terror attacks took place before 2019 of which 70% were successful. It is likely that autonomous vehicles such as drones, underwater autonomous vehicles, and autonomous land vehicles will be used in attacks in coming years. Of course, such drone based attacks can also be launched by one nation state on another.

NSAIL researchers have been developing a simulation platform called DUCK (Drone Urban Cyber-defense Testbed) to study the problem of drone attacks on urban areas and understand how best to combat them. The DUCK platform was developed jointly by researchers at NSAIL and Dartmouth College with several international partners.

DUCK was developd by an NSAIL-led multinational consortium that includes researchers from the University of Calibria (Italy), University of Durham (UK), Dartmouth College, and the Rochester Institute of Technology.

What DUCK Does

DUCK considers the case where a "Red" team (terrorists) attack a city such as New York City with a set of *m* drones. The DUCK platform allows *m* to consist of many different types of drones, each with their respective strengths and weaknesses. The attack is coordinated and managed by a Red Headquarters (HQ) which is assumed to lie outside the theater of battle.

The city attacked in this way is modeled via a graph whose nodes are neighborhoods and whose edges represent adjacency relationships in the city. The city itself is defended by a set of *n* drones. These drones are autonomous, but we require that:

i) The blue drones must exhibit *assured autonomy,* i.e. they must act in accordance with legal, ethical and other clearly articulated constraints. Moreover, they must do so in a provable manner.

ii) The drones must exhibit *autonomy*, i.e. they must be able to take actions that allow them to autonomous make their own judicious choice, while guaranteeing satisfaction of legal and ethic constraints.

iii) The drones must autonomously coordinate their behaviors both with each other and their headquarters, and

iv) The drones must protect the city as effectively as possible.

In addition, DUCK allows the HQs to hack drones belonging to the other team,

The DUCK simulation platform allows policy makers to answer questions such as: what percentage of city C is expected to be destroyed by an enemy with a certain drone configuration, given a drone configuration of the defender? Given a budget, what is the best allocation of resources for the blue team in order to minimize the expected damage caused by the red team? What budget is needed to ensure that no more than X% of the city is destroyed? How much of the city will be destroyed if the budget to defend the city is Y? These are just a few of the types of questions that DUCK will eventually help answer.

What DUCK Does Not Do

DUCK does not tell policy makers what operating constraints (do's and don't's) drones must follow. Nor does it tell policy makers about the advisability or morality of hacking drones belonging to terrorist groups. However, DUCK does provide policy makers with an experimental environment within which such questions can be answered.

How Does DUCK Work?

Each DUCK agent can perform some actions according to constraints in a deontic logic "agent program", e.g., a blue drone *bd* may be permitted (but not required) to fire at red drones within their firing range as long as they have payload. *bd m*ight be forbidden from firing at city nodes. Such constraints are encoded in a drone's agent program which imposes constraints on what the agent is permitted, forbidden, or obliged to do in different situations. Each agent autonomously chooses what actions to perform in a given state based on a set of objective functions to capture local objectives (e.g. a blue drone might minimize expected damage to only the

Figure 1: DUCK 3-Screenshot. The center screen visualizes the ground truth. Left and right screens show other technical details.

nodes it is protecting) and global objectives (e.g. minimize expected damage to the whole city). The key computation performed by an agent is to compute a Pareto-optimal set of actions at each time step. DUCK enables us to vary all of these parameters, e.g. number and types of drones, capabilities, positioning, agent actions, agent programs, objective functions and more. At any time *t*, a DUCK agent communicates with other agents during a communication phase, and then computes a Pareto-optimal set of actions to perform (e.g. hacking actions, firing, moving to another location, etc.) leading to a new state at time *t + 1*. DUCK includes new algorithms to compute such Pareto-optimal sets of actions.

DUCK System

DUCK allows a simulation to set the number and capabilities (e.g. payload, firing range, battery) of the blue and red drones, and the number of CCTVs. Figure 1 shows the DUCK system in action via 3 screens.

Left Screen. The top left window shows Red HQ's view of a red drone. The bottom left window shows how objective function values change as the simulation proceeds (e.g. if an agent's actions don't work out as expected because of another agent's actions). The bottom right view shows agents' optimal actions at time *t*. The top right window shows a real-world map with top-view of the drones.

Middle Screen This screen shows the ground truth at one intersection in NYC. Three blue drones (in the yellow circle) are shown firing at a target red drone (red circle) which is trying to destroy the region.

Right Screen. The top and bottom windows of the right part show Blue HQ's view (of one blue drone camera) and video from one CCTV. The center-left window shows the camera view controller which allows users to swap HQ camera feeds. The top-left window visualizes the state of the drones, e.g., GPS coordinates, battery, payload, hack status, alive/destroyed, etc. The launch button in the bottom center handles event-driven simulation. On each click, agents execute actions at that timestep.

Additional Information

T. Deb, J. Dix. M. Jeong, C. Molinaro, A. Pugliese, A. Quattrini Li, E. Santos, V.S. Subrahmanian, S. Yang, Y. Zhang. DUCK: A Drone-Urban Cyber-Defense Framework based on Pareto-Optimal Deontic Logic Agents, *Proc. AAAI 2023, Demo paper, to appear.*

https://sites.northwestern.edu/nsail/projects/duck/

Video

https://sites.northwestern.edu/nsail/videos/duck/

PARTICIPANTS

Lead: V.S. Subrahmanian

Current: Tonmoay Deb

Other Universities: Carly Beckerman, Mingi Jeong, Alberto Quattrini Li, Cristian Molinaro, Andrea Pugliese, Eugene Santos, Jay Yang, Youzhi ZHang

Northwestern | Security & AI Lab