# Effects of Safety Protocols on Unmanned Vehicle Ground Operations

Jason C. Ryan[1]
*Massachusetts Institute of Technology, Cambridge, MA 02139*

M. L Cummings[2]
*Duke University, Durham, NC 27708*

**Recent advances in unmanned and autonomous vehicle technology are accelerating the push to integrate these vehicles into human-centered environments such as commercial aviation and public roads. Much of the current research into autonomous systems examines improving the performance of individual unmanned vehicles or improving the safety of their interactions with individual humans; very little examines the behavior of the broader system. For large-scale transportation systems, real-world field trials involving unmanned vehicles are difficult to execute due to concerns of cost, feasibility of construction, and the maturity of the technologies. This paper describes the use of an agent-based model of unmanned vehicle behavior in human-centered environments to explore the effects of their implementation in these domains. In particular, this work explores how safety protocols governing the integration of manned and unmanned vehicles affect performance in an aircraft carrier ground control environment. Three different types of futuristic unmanned vehicle control architectures are considered in conjunction with four different types of safety protocols: dynamic, area, temporal, and combined area+temporal separation. Results demonstrate that measures of safety vary widely across these systems, demonstrating distinct tradeoffs of safety and mission performance as well as across different safety measures.**

---

[1] PhD Candidate, Engineering Systems Division and member of the Humans and Automation Laboratory, 77 Massachusetts Ave Rm 35-220, Cambridge, MA 02139. AIAA Student Member.
[2] Associate Professor, Mechanical Engineering and Materials Science and Director of the Humans and Autonomy Laboratory, Durham, NC 27708. AIAA Associate Fellow.

## I. Introduction

Unmanned vehicles — in which there is no human operator seated within the vehicle and vehicle behavior is controlled remotely — have been used for years in a variety of domains, but it is only relatively recently that they have improved to the point that integrating them into human-centered civilian environments has become realistic. Of particular interest to the civilian population are the introduction of unmanned aircraft into the national airspace system and the introduction of increased autonomy into automobiles. While the FAA is currently focused on the integration of Unmanned Aerial Vehicles (UAVs) into the national airspace, unmanned aircraft must also collaborate and interact with flight crew and other aircraft in ground operations at airports. An additional similar domain is that of the aircraft carrier flight deck, for which the Navy is currently developing unmanned combat aircraft [1, 2].

The domains of automobile driving, airport operations, and aircraft carrier flight deck operations can all be characterized as heterogeneous manned environments — places in which humans work in close proximity to and in close collaboration with manually-controlled vehicles in a shared physical space. This work concerns the future versions of these domains, characterized as Heterogeneous Manned-Unmanned Environments (HMUEs), which will require that unmanned vehicles interact with personnel on foot and other manually-controlled vehicles. This transition may require the development of new paradigms of interaction for these personnel to work effectively with these unmanned vehicles. Changes to operating procedures may also be required to accommodate differences in capabilities between unmanned and manned vehicles. To this date, no research addressing what these changes might be and how these changes, and the introduction of unmanned vehicles that they enable, might affect the performance of the broader system has been observed.

Insight into the effects of unmanned vehicle integration in these larger-scale systems could be gained by field-testing prototype systems, but these environments are of such large scale (both in size and number of active elements) that testing them in any meaningful manner would require significant investments of time, money, and physical resources. Even if these resources could be provided, if the unmanned vehicle systems being tested are not yet technologically mature, the results may not be useful. Such tests might also place the autonomous systems, the researchers involved in the study, and any bystanders in unforeseen danger, as well as be severely limited in the conditions that could be explored.

### A. Agent-Based Modeling

Agent-based modeling [3] is one potential approach for addressing such limitations; these models attempt to replicate the individual decision-making, motion, and interactions of actors within a given environment. Constructing validated models of the real world system provides an avenue for testing unmanned vehicle performance under a variety of test conditions that may not be easily achievable. This research examines the use of an agent-based simulation of a candidate HMUE, the aircraft carrier flight deck, in order to examine how different types of **Safety Protocols (SPs)** could be employed to limit interactions between manned and unmanned aircraft and what the resulting effects of operational safety might be. Three different types of UAV **Control Architectures (CAs)** are explored, all modeling advanced autonomous systems that could be introduced in the future.

As described by Eric Bonabeau [3], agent-based modeling is as much a perspective as it is a methodology. Whereas the use of discrete-event or systems dynamics models elicits specific ideas of structure and constituent components, agent-based models may vary widely in content and in methods of application. Examples range from human decision-making (ex. [4]) to epidemiology and disease transmission (ex. [5]) to transportation systems involving both ground and aerial vehicles (ex. [6]), among many others. What matters is that, in each case, the agent-based modeling paradigm views the world "from the perspective of its constituent units" [3] — that the agents retain the same independence and employ similar decision-making strategies as the real-world entities they are intended to replicate. The agents are then placed into the simulated environment, given a set of goals (which may be self-generated during the simulation), and allowed to make decisions independently from that point forward.

While agent-based modeling has been used to validate control and sensing algorithms for a variety of UAV systems (e.g., [7]), these studies have primarily addressed systems involving the collaboration of multiple autonomous vehicles in performing tasks, such as searching for and tracking targets, and only include limited interactions with other entities. Another common, and related, area of interest is in the modeling and control of UAV "swarms" (large, coordinated groups of robots coordinating on a specific task) [8]. Agent-based models have also been built to test unmanned vehicle survivability [9] and single UAV search [10]. However, none of these examples has considered the role of the human in interacting with the vehicle system. While other non-agent-based models of human-UAV interaction have been constructed [11, 12], these models have focused on the human as a processing server working within a queuing system, considering the behavior of the vehicle only in the sense that it requires the operator to provide a task for it. In order to better understand the practical requirements for UAVs in these HMUEs, models of UAV motion in the world, UAV task execution, and UAV interactions with their operators, other human personnel, and other vehicles in the world must be generated.

### B. Agent-based Modeling in Vehicular Domains

Agent-based modeling for safety research has also occurred, addressing both the automobile [13–17] and airspace safety [18–21]. Traffic safety models have focused both on driver interactions at intersections [13] and highway traffic [14, 15]. Airspace safety analyses have dealt both the national airspace system in general [18] as well as more localized models of runway interactions at airports [19–21].

In these prior studies, the simulated drivers and pilots act independently in the environment based on a set of defined goals and behavioral rules, with metrics of performance focusing on both true accidents (where possible) and near-accidents — near collisions that suggest a high probability of an accident in reality. Airspace safety simulations have also investigated how changes in behavior affect the system, addressing both pilot [20, 21] and air traffic controller behaviors [19] affect the system.

However, at this time, few studies have focused on the interaction of manned and unmanned vehicles within the same shared physical environment, or on how the methods of controlling those unmanned vehicles and the rules of behavior within the environment affect the safety of operations.

The goal of the Multi-Agent Safety and Control Simulation (MASCS) model described within this paper was to generate an agent-based model of aircraft carrier flight deck operations that provided an ability to model not only various types on unmanned vehicle control architectures, but that also provided an ability to modify the rules of behavior in the environment: when and where manually-piloted and unmanned aircraft were allowed to interact with one another.

This paper begins with a description of the MASCS model and its modeling of aircraft carrier flight deck operations. The three different unmanned vehicle control architectures of interest are then described, followed by a discussion of current aircraft carrier flight deck operations and how they are modeled within the Multi-Agent Safety and Control Simulation (MASCS). Afterwards, the modeling of the unmanned vehicle control architectures within MASCS is described, followed by a description of the testing of these vehicles under a variety of mission conditions and process-level task definitions.

## II. The MASCS Agent-Based Model of Aircraft Carrier Flight Deck Operations

The MASCS model is an agent-based model of aircraft carrier flight deck operations that includes models of human flight deck crew, manually controlled aircraft operating on deck, unmanned aircraft, and supervisory control staff that oversee operations. Before defining the models of unmanned vehicle control architectures used in this work, a simulation of current aircraft carrier flight deck operations using manually-piloted aircraft was constructed and validated against empirical data, the process of which is described in [22]. Constructing the model required defining agents that replicate the behavior of human crew, pilots, aircraft, planning supervisors, flight deck equipment, as well as models of the tasks that they perform. Agent and task models consist of decision-making rules that govern agent behavior, parameters that describe how agents view and execute tasks in the world, and states that are viewable by other agents and are used in the decision-making routines.

The MASCS model was validated as part of prior work [22] that compared the performance of individual aircraft and missions of multiple aircraft against empirical data on operations. Validating the model in this fashion provided support for the validity of the modeling in replicating flight deck operations both in terms of system performance and in terms of the individual agent models.

The models of Manual Control (MC) vehicles were then used as templates for the creation of the unmanned vehicle control architectures of interest. The next section describes the basics of flight deck operations relevant to this work and how these operations were modeled within MASCS before discussing the modeling of the unmanned vehicle control architectures in Section II B.

### A. Aircraft Carrier Flight Deck Operations

A digitized map of the flight deck appears in Figure 1. Labels in the figure highlight the important features of the flight deck environment relevant to launch operations. At this time, the MASCS simulation focuses solely on launch operations, where future work may extend the modeling to other phases of flight. In launch operations, aircraft begin parked at the edges of the flight deck in the aft and starboard areas of deck; aircraft not able to fit in those areas are parked in the area known as "The Street." Aircraft will be assigned to launch from one of the four launch catapults (orange lines in the figure, numbered one to four). Each catapult is capable of launching a single aircraft at a time, but adjacent pairs of aircraft (1,2 or 3,4) are not able to operate in parallel, for two reasons. The first is that a single set of crew operates each pair of catapults and can only manage one catapult at a time. The second, and more important, is that the catapults are angled towards one another — simultaneous launches would result in a collision and loss of both aircraft. Catapults alternate launches within each pair, but the two pairs can process in parallel (e.g., 1 and 3 or 2 and 4). Operations typically allow one aircraft on the catapult and another waiting behind a protective barrier that rises from the deck during a launch. As such, each pair of catapults (if both are active) can queue up to four aircraft in the area at any time, but only one aircraft will be in the process of launching.

Getting aircraft from parking places to catapults requires interactions with human crew on the flight deck. A senior enlisted person, termed the Deck Handler, and his staff are responsible for creating the schedule of operations and allocating aircraft to launch catapults. The instantiation of the plan does not occur all at once: the decisions of where to send aircraft are made dynamically as new slots open. At the start of operations, assignments will be made such that queues at all catapults are filled. As launches occur and slots open at catapult queues, the next highest priority

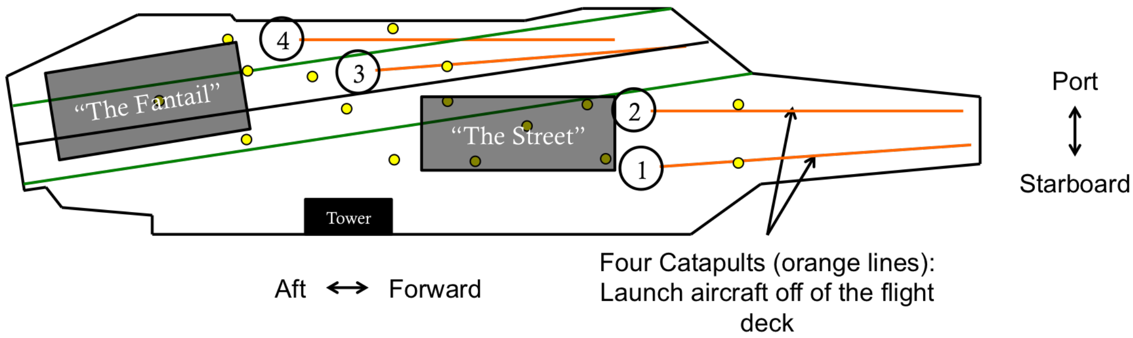


**Fig. 1 Digital map of the aircraft carrier flight deck. Labels indicate important features for launch operations. Four catapults (orange) accelerate aircraft to launch speed. The tower (black box) houses the Deck Handler, a senior supervisor, and a variety of other personnel. Aircraft Director crew (yellow dots) navigate aircraft around the flight deck. Two high-traffic areas known as "The Street" and "The Fantail" require that Directors carefully manage traffic flow.**

aircraft that has an open taxi route to this catapult is assigned. This preserves some flexibility in the face of the stochasticity of the launch preparation process.

Aircraft assignments are then passed to a set of crew on the flight deck, termed Aircraft Directors (yellow circles in Fig. 1), who provide taxi instructions to aircraft. These crew work in a form of "zone coverage," each controlling a specific area of the deck (which may overlap with others). A Director will taxi the aircraft through their area, then hand the aircraft over to an adjacent Director before accepting another aircraft into their area. Aircraft are passed between Directors until they reach their assigned catapult. Directors communicate instructions to pilots using a set of hand gestures, informing the pilot when to drive forward, stop, turn, and to whom they are being passed off. Directors also must maintain an understanding of current traffic conditions on the flight deck, delaying some aircraft to allow other aircraft to taxi by. This may be because the other aircraft has a higher priority, or that failing to do so would "lock" the deck, with aircraft unable to taxi to their require destinations.

Modeling these operations requires several agents within the MASCS simulation, each with their own sets of decision rules described each of the features above. The Deck Handler agent requires a model of their assignment heuristics, also developed as part of prior work [23]. Aircraft Director agents require a model of the other Directors they can pass aircraft to, whether or not they can currently hand the aircraft to the desired next Director or accept new aircraft themselves, as well as how to align to the aircraft they are currently instructing so that they are visible to the pilot. Directors also require models of their speed of movement on the flight deck, as do aircraft. Within MASCS, aircraft and pilots are modeled as a single unit at this time, requiring the definition of their speed and motion profiles, their ability to detect a Director, and their ability to execute tasks. Additional agents characterize the flight deck and the catapults used by aircraft during operations.

Task models provide the rules of behavior that dictate whether or not an agent can conduct a task and how it should be executed once execution can occur. For instance, for an aircraft motion task, rules determine (1) whether the aircraft can see its assigned Director, (2) whether the Director is allowing the aircraft to taxi because of traffic conditions, and (3) if another aircraft is physically in the way of the current vehicle. In extending MASCS to include unmanned vehicles into operations, different UAV types require changes to the models described above. The general planning methods do not differ substantially, nor do the models of physical motion. However, rules governing task execution for manual control operations or taxi routing on the flight deck may not apply to unmanned vehicle systems due to differences in the architecture of the systems. Other parameters describing the vehicles' physical characteristics might also vary from the baseline model of human Manual Control (MC). The differences that define the four unmanned vehicle control architectures used in this work are discussed in the next section.

### B. Models of Unmanned Vehicle Behavior

The Manual Control (MC) operations described in the previous section serve as the baseline comparison point for the three unmanned vehicle control architectures modeled in MASCS. The first of these unmanned architectures, Gesture Control (GC), replaces the human operator with a set of stereo vision cameras teamed with a sophisticated computer algorithm. This system replicates the same visual observation and interpretation tasks conducted by the pilot: the system observes deck crew, translating the hand and arm gestures provided by those crew into actions such as "taxi forward," and "turn left." As such, GC vehicles are a direct replacement for MC vehicles and no other changes to operations are required.

Advanced supervisory control systems require more substantial modifications to operations. In these systems, a single human operator issues commands to vehicles and receives feedback on their activity through a Graphical User Interface (GUI). Typically, commands are issued in a high-level, abstract form such as "taxi to this waypoint" or "monitor this area;" these commands are broken down into more specific inputs by the autonomy onboard the vehicle or contained in the operator's workstation.

These systems can be classified into two general forms. The first form is referred to here as Vehicle-based Human Supervisory Control (VHSC), in which supervisors supply commands to a single vehicle at a time, working iteratively through the queue of vehicles awaiting tasks (e.g. [24]). Here, the operator serves as a processing server in the system, managing a queue of vehicles awaiting new instructions that the operator processes at some distribution of service times under a certain queuing policy (e.g. first-in, first out). The larger the number of vehicles in the system, the longer the time between successive interactions with any individual vehicle.

For systems with greater than 8-12 vehicles, additional autonomy is often introduced in the form of planning and scheduling algorithms that simultaneously replan all tasks for all vehicles (ex. [25]). These are referred to as System-based Human Supervisory Control (SHSC) systems. The operator typically provides goals and constraints for the scheduling algorithm, which will calculate a new schedule and present it to the operator for review. The operator then has the option of accepting, rejecting, or modifying the schedule before it is transmitted to the vehicles for execution. Once accepted, new task schedules are uploaded for all vehicles, which begin executing their individual task lists.

In the context of flight deck operations and models of agent behavior, the important activities of a pilot that must be replicated by these three unmanned vehicle control architectures (GC, VHSC, and SHSC) can be summarized in terms of three key aspects: visual detection of an Aircraft Director that is attempting to provide instructions, correctly interpreting task instructions, and correctly executing the commanded tasks. Each of these corresponds to a related variable: the probability of observing the Director if within the vehicle's field of view, the probability of correctly interpreting the command, and the probability of correctly executing a task. For manual control, the likelihoods of these failures are so low that subject matter experts consider them as essentially nonexistent. While a fourth difference in behavior comes from the fact that the Director is providing instructions in this "zone coverage" routing; this does not occur for the advanced supervisory control systems, thus how vehicles are routed forms a fourth behavioral variable of "routing method."

Two other variables also affect vehicle behavior, relating to the physical characteristics of the vehicle: the field of view of the pilot/aircraft, which describes where the Director should be in order to provide commands, and the latency in processing commands. The field of view for manual operations is related to human sight range and the constraints of being in the cockpit; for unmanned vehicles, field of view is related to the cameras used on the vehicle. The latency in processing commands could come from a number of sources: delays in the communication of signals between operator and vehicle, the time required for the vehicle's computer system to process, or time required for the human operator to interact with the system. Each of these can also be modeled as variables within the MASCS simulation environment. The following paragraphs describe how these variables are defined for the control architectures used in this work, beginning with the set of variables that relates to visual processes.

#### *1. Visual Acquisition Parameters*

The first primary task of a pilot, visual acquisition, describes the process of visually identifying the current Director that is providing instructions to the pilot. GC systems must also perform this task, enabled by sensors and processing algorithms included on the aircraft. Associated with this task is a chance of failing to recognize the Director. Because these systems are still largely in the research and development phase, there is no true failure rate known for gesture control activity in the flight deck environment. A review of several papers, most of which include data for multiple tests and trials and reference multiple prior studies, provides a picture of the current general state of gesture recognition research [26–33]. Of the 146 different experimental results reported, 11 report results of 100% accuracy (often using data sets with a small number of very different gestures), 37 reported results better than 95%, and 91 reported better than 90% accuracy. Overall, the studies report an average failure rate of 11.18%.

It is not clear how this failure rate translates into realistic future operations; while the technology will certainly improve, a variety of environmental conditions (sun glare, obstructions, large numbers of moving vehicles) will limit performance in the real world. While future real-world failure rates are still not known, the median failure rate of 11.18% observed in the research is used as the failure rate of the baseline GC system. The GC system is modeled with the capability of failing repeatedly, modeled as a series of Bernoulli trials. In doing so, it is assumed that the system does not encounter cases in which it cannot determine a solution (or that it can at least be overridden by an Aircraft Director). For each failure, a time penalty of 5 seconds is applied for each failed acquisition of a Director and 5-10 seconds for failing to recognize a task.

#### *2. Latencies*

GC and VHSC systems are also expected to have latencies in task execution, albeit for two different reasons. The source of latency in GC systems comes from software processing as the system attempts to translate the hand gestures of the Director into understandable action commands. Prior research shows this requires anywhere <<1 to 3 seconds [26–33]. This latency is then modeled as a uniform distribution over the range $[0, 3]$, randomly sampled and applied to the start of every action commanded by a Director. VHSC systems incur latency related to the Handler's ability to process tasks through the supplied GUI. This interaction is described in the next section.

#### *3. UAV Routing Methods*

The final major difference between UAV Control Architectures involves how they are issued taxi commands and routed through the flight deck. As described earlier, GC systems still operate under "zone coverage," with crew communicating instructions through hand gestures. This is largely the same as the MC vehicles modeled for current operations, although the GC vehicles are affected by the failure rates and latencies described above.

VHSC systems do not rely on the crew for taxi commands, instead relying on commands issued from the Handler through a Graphical User Interface, which requires modeling the behavior of the Handler in operating such a system. As described previously, the operator functions as a queuing server, which requires the definition of both a service time and a queuing policy for managing tasks. For flight deck operations, experienced Handlers would likely operate under a "highest priority first" policy, prioritizing aircraft nearest to their assigned catapults in order to keep traffic flowing on the deck. If vehicles are of equal priority, the vehicle that has waited longest in the queue would be serviced.

The time required to process tasks is taken from prior research by Nehme [11], whose research involved operators controlling a team of unmanned vehicles through a VHSC-style interface. Nehme characterized the "idle" task, in which operators assigned a new transit task to a waiting vehicle, as normally distributed with a mean of 3.19 seconds and standard deviation of 7.19 seconds (bounded $[0, \infty]$). The idle task described by Nehme is analogous to the action of assigning taxi tasks to idling aircraft on the flight deck, so this same distribution will be used within MASCS for the baseline model of VHSC systems. This interaction time occurs for each individual vehicle in the queue; thus, there is a significant penalty for including additional vehicles in the VHSC queuing system that has been characterized in many previous studies (see [24, 34] for examples).

Additionally, since this style of routing does not rely on the network of crew on the flight deck, the Handler can also assign longer taxi tasks on the flight deck — taxiing from aft to forward, barring other constraints, could be done in a single motion rather than requiring multiple handoffs between Directors. The taxi paths, however, would largely stay the same. These same routing changes also would occur for the SHSC architecture, which enable a single operator, with the help of a scheduling algorithm, to simultaneously define task lists for all vehicles. As prior work has demonstrated that even complex planning algorithms have trouble decomposing the geometric constraints on the flight deck and provide little benefit over human planning [23], the same Handler planning heuristics applied to the other CAs are applied here. The SHSC architecture would also change some roles on the flight deck; Aircraft Directors would no longer be utilized in the same fashion as in current Manual Control (MC) operations, and some method of controlling the MC aircraft (either remotely in the same manner as the UAVs, or providing the human pilot with guidance indicators) must also be developed.

### C. Safety Protocols

Safety Protocols (SPs) address the ways in which the planning and administration of flight deck operations might be altered in order to limit the interactions of manned and unmanned aircraft and increase the safety of operations on the flight deck. Four different types of safety protocols are considered. The first, *Dynamic Separation only*, is currently used in today's manual control operations and is the least constrained form of operations: it relies only on rules that prevent vehicles from colliding with one another and makes no other changes to operations. If a vehicle comes too close to another vehicle in front of it, it stops until that vehicle moves sufficiently far away. The same rules used in manual control vehicles are applied to the unmanned vehicle control architectures.

The second safety protocol, *Area Separation*, segregates unmanned vehicles and manned vehicles into two different operating areas on the flight deck. Manned aircraft are moved forward (to the right side of the figure) and UAVs aft; aircraft are only assigned to catapults within their given area (Fig. 2). As long as at least one member of each group is on the flight deck, this segregation remains in place. The third option, *Temporal Separation*, separates vehicles based on schedule. Manned aircraft are again parked forward and unmanned parked aft, but all manned aircraft are launched first in the mission. Once the last manned aircraft has been assigned and cleared the interior of the deck, unmanned aircraft are allowed to begin operations. The fourth option combines the Temporal and Area protocols into one, launching all manned aircraft first from only the forward catapults, after which unmanned aircraft begin operations. The increase in constraints on operations from the Dynamic Separation only (DS) only to Temporal+Area (T+A) case should increase the safety of flight deck operations: interactions between aircraft should be minimized at this level, although it should come at a cost of productivity.

## III. Monte Carlo Simulation

Experimental tests simulated each of the three unmanned vehicle control architectures — Gesture Control (GC), Vehicle-based Human Supervisory Control (VHSC), and System-based Human Supervisory Control (SHSC) — in missions using 22 aircraft and each of the four safety protocols.

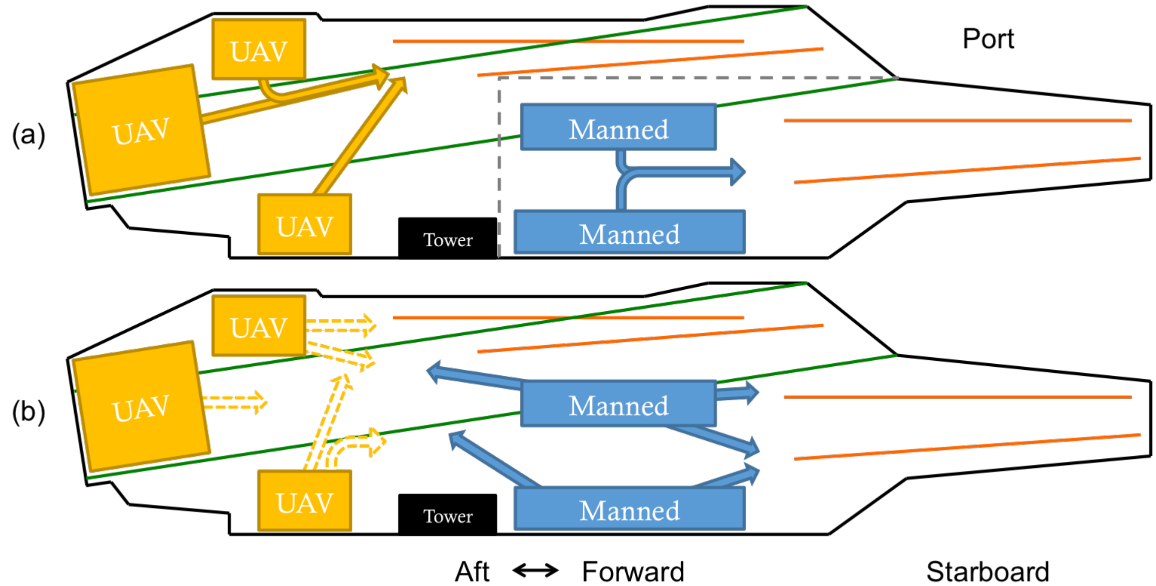


**Fig. 2 (a) Depiction of the Area Separation protocol assuming identical numbers of UAVs and manned aircraft. In this configuration, all manned aircraft parked forward are only assigned to the forward aircraft pair, while all unmanned aircraft parked aft are only sent to the aft catapults. When one group finishes all its launches, these restrictions are removed. (b) Depiction of Temporal Separation using identical numbers of UAVs and manned aircraft. All manned aircraft would be assigned and launched first with access to the entire flight deck. Once manned taxi operations are complete, unmanned aircraft would begin operations.**

Missions also used three different mission Compositions: 100% homogeneous (all 22 aircraft are the same), 50% unmanned (11 are unmanned), or 25% (5 unmanned). Differences in mission Composition lead to changes in how the safety protocols are applied to operations. Under Area Separation, a Composition of 50% unmanned vehicles means that equal numbers of manned and unmanned aircraft have access to equal numbers of catapults (the unmanned aircraft to the two aft catapults, manned aircraft to the two forward). These groups should then launch at the same rate, and the Area Separation constraint is applicable for generally the entire mission. For Area Separation using only a 25% unmanned Composition, the small number of unmanned aircraft will complete their launches relatively early on in the mission; after they have launched, the Area Separation constraint can be relaxed and manned aircraft can be allowed access to the entire flight deck. Similar differences occur for the Temporal+Area cases. In these, operations begin with only manned aircraft being sent to only forward catapults. This neglects half of the available catapults on the flight and should decrease the efficiency of operations; this means that the 25% Composition case (which uses more manned aircraft) should take much longer to complete that the 50% Composition case (and both should take longer than other Safety Protocol options).

An additional test of the original MC architecture was performed at 22 aircraft under the DS protocol as a baseline comparison point. This results in a total of 28 cases to be run: three unmanned vehicle Control Architectures times nine SP*Composition options, plus the one instance of the baseline Manual Control architecture. Each mission setting (CA+SP+Composition) was replicated thirty times within the simulation environment. For typical human subjects testing in real world systems (such as a real carrier and crew), performing 30 replications is a standard heuristic for ensuring that sufficient data is acquired for detecting normality and for making inferences on population means (since the population variance is not known exactly) [35]). Thirty replications are used for this simulation study as well. Output measures in the simulation addressed both safety and productivity. The primary measure of productivity is the total Launch event Duration (LD): the elapsed time from the start of the simulation to the launch of the last aircraft. A first set of safety measures addresses the relative distances between all aircraft in the system, tracking the number of times any vehicle or aircraft comes within one of three distance thresholds ("halos") around each aircraft (see Figure 3). For a given aircraft, a counter increases any time a crewmember or another vehicle comes within a given "halo" (area) around the vehicle. This concept also encompasses the tracking of other hazard/no-transgression zones around aircraft that are associated with engine intake and jet blast hazards, as well as and rotor/ propeller arcs. A timer also tracks the total duration during which the "halo" is violated. These measures of Primary (PHV), Secondary (SHV), and Tertiary Halo Violations (THV) can also be broken down into measures for Crew and Aircraft individually. Of particular interest in this work are the Tertiary Halo Violations due to Aircraft (THVA) measure, which was most effective in describing aircraft interactions, and the Primary Halo Violations (PHV) and Duration of Primary Halo Violations (DPHV) measures, which were most effective in understanding crew interactions with aircraft.

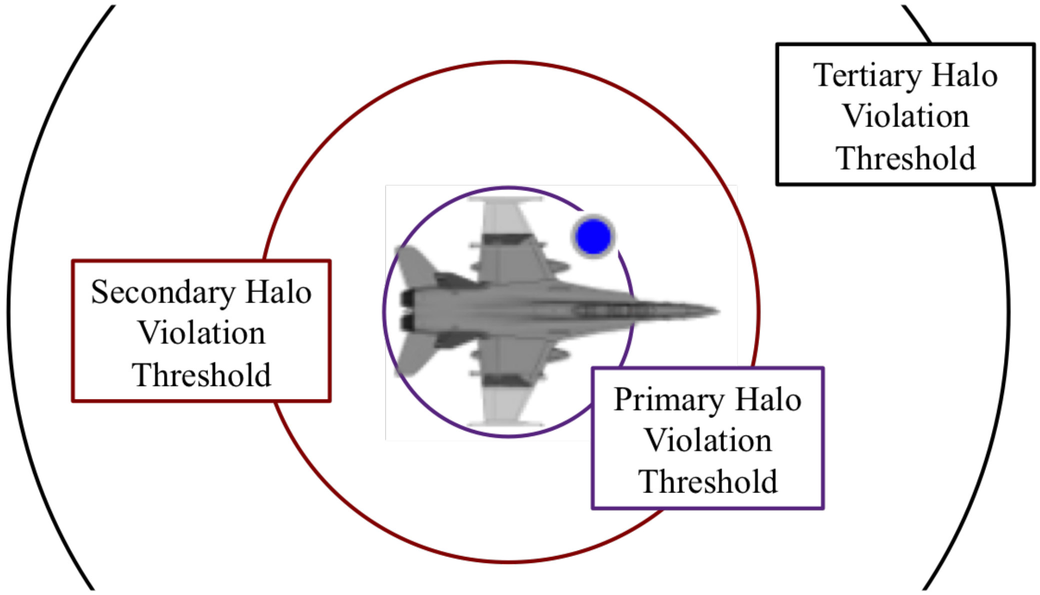


**Fig. 3 Diagram of "halo" radii surrounding an aircraft. In the image, the aircraft has its wings folded for taxi operations. A transparent image of the extended wings appears in the background.**

## IV. Results

### A. Effects on Aircraft Safety

Figure 4 shows the results of the Tertiary Halo Violations due to Aircraft (THVA) metric for all missions at the 22 aircraft Density level under all possible combinations of Safety Protocols, Control Architectures, and mission Composition. Within the figure, data point shapes are consistent across Control Architecture (e.g., all Gesture Control (GC) data are plotted as crosses). Data point colors are consistent across Safety Protocol-Composition combinations (e.g., 50% Temporal+Area is orange), given the interactions of these variables and their effects on mission execution. This coding of data points is used for all remaining Pareto frontier charts presented in this paper. Statistical tests used to examine the results were performed in SAS JMP PRO 10 unless otherwise noted.

The results in Figure 4 show an interesting range of values for the THVA measure. First, the results in the lower right corner of the figure suggest that the Temporal+Area (T+A) protocol provides the best performance for the THVA metric at a significant cost of Launch event Duration (LD). A Kruskal-Wallis test across safety protocols returns significant results ($\chi^2(3) = 333.43, p < 0.0001$); a Steel-Dwass non-parametric simultaneous comparisons test shows that five of the six possible pairwise comparisons are significant at $p < 0.0001$; only Temporal compared to Area Separation returns non-significance ($p = 0.2172$). These results indicate that Temporal+Area (mean rank = 194.35) is better than Temporal (368.136) and Area Separation (397.50), which are both better than the Dynamic Separation only protocol (601.41).

This trend in results suggests that, in and of itself, increasing the constraints on operations through the application of safety protocols reduces the number of interactions between aircraft on the flight deck (see Table 1). The unconstrained DS case provides the worst performance while the highly constrained T+A provides the best. However, the benefits of the increased constraints at the T+A level come at a cost: the average LD across all cases using the T+A protocol is 36.67 minute, significantly larger than the averages for Temporal (25.84), Area (24.71), and Dynamic Separation (24.60). Additionally, the use of 25% unmanned Composition under T+A is more detrimental than

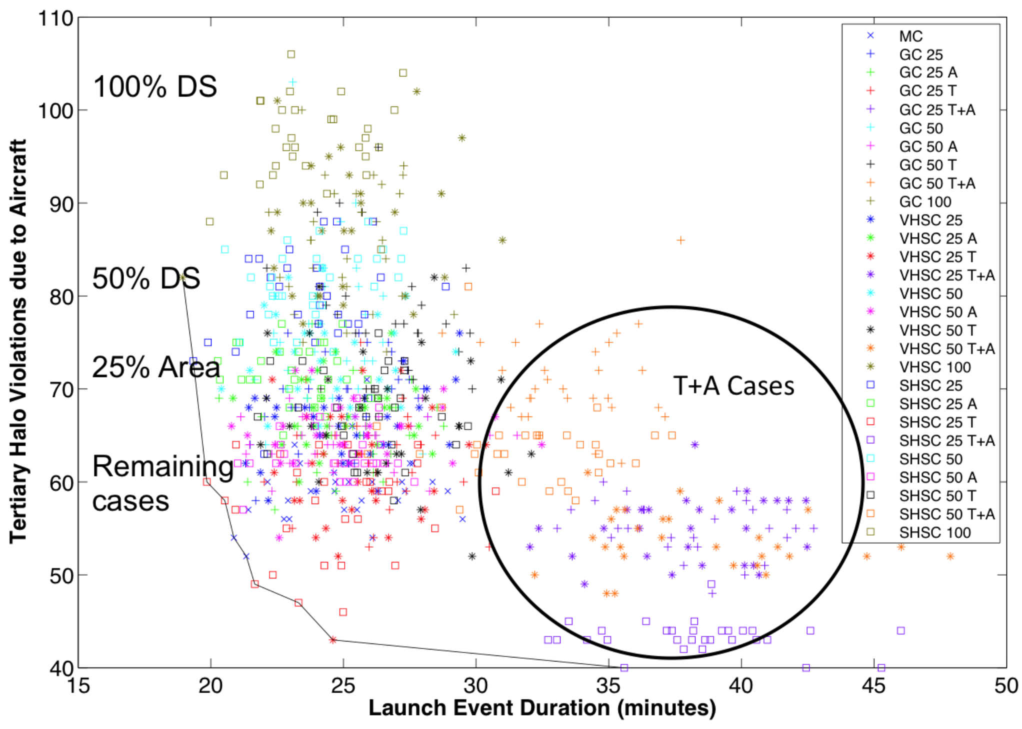


**Fig. 4 Pareto frontier plot of Tertiary Halo Violations due to Aircraft (THVA) versus LD for 22 aircraft missions. Results demonstrate that the choice of Safety Protocol, and to a lesser degree, of Composition are main drivers of performance. The highly-constrained Temporal+Area protocol provides the best results at the cost of mission completion times, while the unconstrained Dynamic Separation only protocol provides the worst overall performance.**

the 50% Composition setting. Because the former uses more manned aircraft in operations, it takes longer to complete the manned phase than under the 50% case. However, because the 25% case constrains operations for a longer period of time, it also produces smaller (better) THVA values for the SHSC and GC architectures.

**Table 1 Descriptive statistics for Tertiary Halo Violations due to Aircraft (THVA) values grouped by Safety Protocol.**

| Safety Protocol | Mean | Std Dev | Std Err Mean | Lower 95% | Upper 95% |
|---|---|---|---|---|---|
| Area Separation | 63.26 | 5.48 | 0.32 | 62.63 | 63.88 |
| Dynamic Separation | 71.33 | 12.52 | 0.57 | 70.21 | 72.46 |
| Temporal Separation | 62.54 | 8.42 | 0.49 | 61.58 | 63.50 |
| Temporal+Area | 54.54 | 8.01 | 0.46 | 53.63 | 55.45 |

A second interesting phenomenon is the performance of the SHSC control architecture, which appears as squares in Figure 4: SHSC provides both the best performance (under the T+A protocol) as well and the worst performance (under the DS protocol). These variations can also be described in terms of constraints. The SHSC control architecture is the least constrained of the control architectures, being limited neither by the "zone coverage" routing structure nor by interactions with a central operator. Aircraft are free to execute their own tasks and able to move through longer individual taxi segments on the flight deck. Not surprisingly, given this lack of constraints, SHSC performs the worst under the also unconstrained Dynamic Separation safety protocol. Interestingly, however, under the significant constraints of the T+A protocol it generates the lowest THVA values of any single test case.

This range of performance contributes to SHSC having some of the worst overall performance of any control architectures for the THVA metric (Table 2). A Kruskal-Wallis test demonstrates significant differences across architectures ($\chi^2(3) = 39.3551, p < 0.0001$), with MC demonstrating the best results overall (mean rank = 205.30), followed by VHSC (382.80), then by SHSC (443.915) and GC (458.69). A Steel-Dwass simultaneous comparisons tests shows that all pairwise comparisons are significant ($p \leq 0.0411$) except for the GC-SHSC pair ($p = 0.987$). In this case, the human operator providing individual instructions to aircraft in the VHSC architecture seems to regulate the motion of aircraft on the deck, providing additional spacing in operations and limiting vehicle interactions.

**Table 2 Descriptive statistics for Tertiary Halo Violations due to Aircraft (THVA) grouped by Control Architecture.**

| Level | Mean | Std Dev | Std Err Mean | Lower 95% | Upper 95% |
|---|---|---|---|---|---|
| Gesture Control (GC) | 69.56 | 10.12 | 0.62 | 68.35 | 70.76 |
| Local Teleoperation (LT) | 56.91 | 5.60 | 0.34 | 56.24 | 57.59 |
| Manual Control (MC) | 59.53 | 4.10 | 0.75 | 58.00 | 61.06 |
| Remote Teleoperation (RT) | 58.48 | 5.82 | 0.35 | 57.78 | 59.18 |
| SHSC | 69.28 | 15.13 | 0.92 | 67.47 | 71.09 |
| VHSC | 66.33 | 10.76 | 0.65 | 65.04 | 67.62 |

This is likely an artifact of the queuing policy or prioritizing aircraft nearest to catapults; because operations generally flow in the same direction, these vehicles are "in front" of most other aircraft. Moving these aircraft first provides space for the other aircraft to move, limiting the number of chances for vehicles to come too close to one another. While the GC system appears to encounter similar latencies, the random failures encountered by vehicles do not increase the spacing between vehicles; in fact, it appears to reduce the spacing. Because failures are random, there is a greater chance for vehicles "in front" of others to fail. When that vehicle fails, there is nothing preventing aircraft behind it from taxiing forward and coming close to the stopped vehicle, increasing the number of THVA violations.

### B. Effects on Crew Safety

Two other measures looked at the interactions of crew with aircraft during operations. While the safety protocols do not explicitly change how crew interact with aircraft, the changes in routing that result will affect the patterns of traffic and which crew are utilized. Figure 5 shows the results for Primary Halo Violations (PHV) values again plotted against LD values. A trend is immediately noticeable in Fig. 5: there is a clear separation between the VHSC and SHSC cases and the other control architectures (see Table 3 for descriptive statistics). What differentiates VHSC and SHSC cases from the rest are the fact that neither utilizes the "zone coverage" routing scheme on the flight deck; instead, they are given greater freedom to move around the deck while being assigned escorts to follow them through the deck. Under SHSC, manned aircraft also do not use the zone coverage mechanic; manned aircraft in VHSC do, however. Statistical tests verify the significance of the differences (Kruskal-Wallis test, $\chi^2(3) = 579.06, p < 0.0001$), and a Steel-Dwass simultaneous comparisons tests shows that VHSC (mean rank = 517.78) and SHSC (mean rank = 569.22, $p = 0.9976$) are not significantly different from one another, but they are significantly different from GC (mean rank = 155.28) and MC (107.48), all at $p < 0.0001$ level. MC and GC are also significantly different from one another ($p = 0.0217$).

**Table 3 Descriptive statistics for Primary Halo Violations (PHV) grouped by Control Architecture.**

| Level | Mean | Std Dev | Std Err Mean | Lower 95% | Upper 95% |
|---|---|---|---|---|---|
| Gesture Control (GC) | 67.82 | 6.75 | 0.41 | 67.01 | 68.62 |
| Local Teleoperation (LT) | 65.79 | 7.29 | 0.44 | 64.91 | 66.66 |
| Manual Control (MC) | 64.83 | 4.15 | 0.76 | 63.28 | 66.38 |
| Remote Teleoperation (RT) | 66.56 | 7.71 | 0.47 | 65.64 | 67.48 |
| SHSC | 131.23 | 16.29 | 0.99 | 129.28 | 133.18 |
| VHSC | 134.15 | 22.85 | 1.39 | 131.41 | 136.89 |

Viewing the effects of the safety protocols (Table 4), significant differences also appear (Kruskal-Wallis test, $\chi^2(3) = 82.56, p < 0.0001$). A Steel-Dwass simultaneous comparisons test shows that Temporal (mean rank = 503.32) and Temporal+Area (mean rank = 497.70) are not significantly different from one another ($p = 0.6113$), but are significantly different from Area (mean rank = 307.0) and DS only (mean rank = 392.58), all at $p < 0.0001$. DS only and Area Separation are also significantly different from one another ($p = 0.0011$). The increased values for Temporal and T+A can be attributed to how operations are conducted: each emphasizes routing aircraft through the "Street" area of the flight deck (Fig. 1). This area includes one more crew member than the Fantail area but in a smaller total area; emphasizing operations in this area requires more interaction with the crew, increasing the number of times that crew are too close to aircraft.

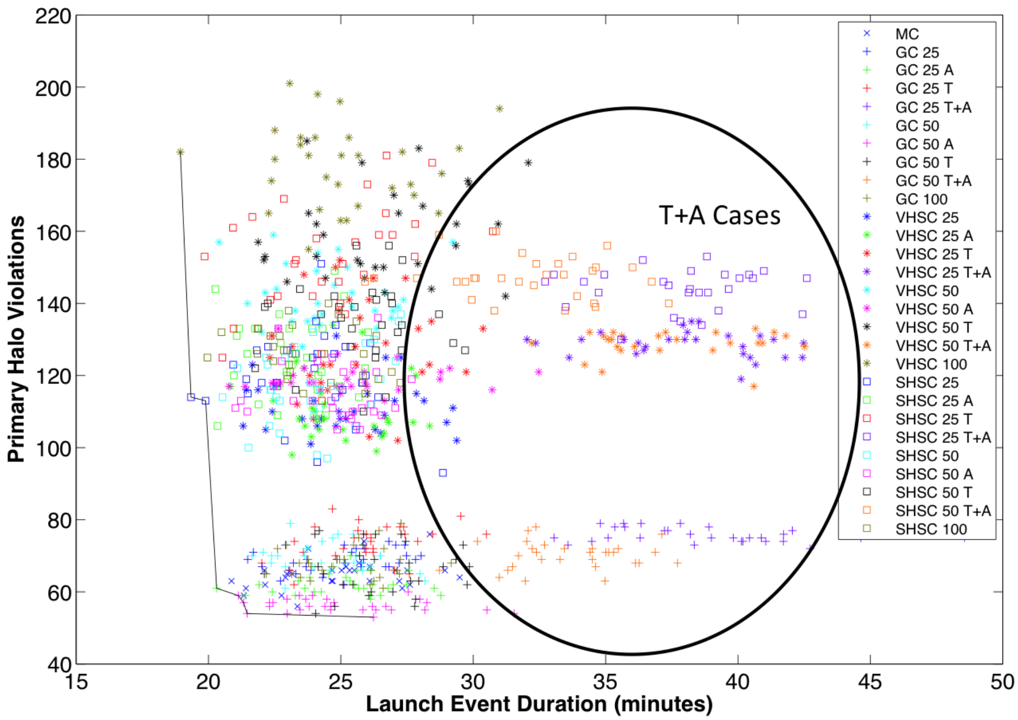


**Fig. 5 Pareto frontier plot of Primary Halo Violations (PHV) versus LD for 22 aircraft missions. Results indicate that the choice of Control Architecture is a main driver of results, with the two Supervisory Control cases (VBSC, SBSC) providing the worst performance due to the modeling of their "Escort" crew. For the remaining cases, the choice of Control Architecture provides further differences in performance, with Area Separation providing the best performance and Temporal the worst.**

**Table 4 Descriptive statistics for Primary Halo Violations (PHV) grouped by Safety Protocol.**

| Level | Mean | Std Dev | Std Err Mean | Lower 95% | Upper 95% |
|---|---|---|---|---|---|
| Area Separation | 81.58 | 30.18 | 1.74 | 78.15 | 85.01 |
| Dynamic Separation | 90.88 | 35.83 | 1.64 | 87.67 | 94.1 |
| Temporal Separation | 99.67 | 37.91 | 2.19 | 95.36 | 103.97 |
| Temporal+Area | 98.81 | 32.15 | 1.86 | 95.15 | 102.46 |

Interestingly, however, when tracking the total duration in which these violations occur, the results are quite different (Figure 6). From the figure, VHSC demonstrates larger DPHV values but no strong variations otherwise appear. Testing DPHV across control architectures (Tables 5), a Kruskal-Wallis test returns significance ($\chi^2(3) = 396.012, p < 0.0001$), with a Steel-Dwass test indicating four significant pairwise differences. The VHSC architecture (mean rank = 658.56) is significantly different from MC (302.15), GC (270.46), and SHSC (345.63) at $p < 0.0001$. GC is also different from SHSC ($p < 0.0001$), but MC is not different from GC ($p = 0.6154$) and SHSC ($p = 0.7974$). That MC and GC provide the best performance may be the result of the zone coverage routing framework and the way that Aircraft Directors align to and keep their distance from aircraft in motion.

**Table 5 Descriptive statistics for Duration of Primary Halo Violations (DPHV) grouped by Control Architecture.**

| Level | Mean | Std Dev | Std Err Mean | Lower 95% | Upper 95% |
|---|---|---|---|---|---|
| Gesture Control (GC) | 2495.25 | 243.96 | 14.85 | 2466.00 | 2524.50 |
| Local Teleoperation (LT) | 2431.19 | 283.12 | 17.23 | 2397.30 | 2465.10 |
| Manual Control (MC) | 2527.47 | 272.77 | 49.80 | 2425.60 | 2629.30 |
| Remote Teleoperation (RT) | 2420.98 | 268.63 | 16.35 | 2388.80 | 2453.20 |
| SHSC | 2598.54 | 270.47 | 16.46 | 2566.10 | 2630.90 |
| VHSC | 3598.66 | 898.27 | 54.67 | 3491.00 | 3706.30 |

The performance of the SHSC system as compared to VHSC is also interesting. It suggests that, although the escorts active for SHSC operations are continually moving in and out of the primary halo area, they are only in the area for short periods of time. The poor performance of VHSC can likely be attributed to the fact that it uses *both* escorts for unmanned aircraft and zone coverage for the remaining manned aircraft. It should also be noted that, for these two architectures, the escorts assigned to vehicles are optional: unmanned vehicles in these systems can taxi without human aid. This would reduce the PHV and DPHV values for both architectures, but for SHSC these values would be zero.

Comparing across safety protocols, a similar trend in results is observed (Tables 6). A Kruskal-Wallis test returns significance ($\chi^2(3) = 27.90, p < 0.0001$), with a Steel-Dwass simultaneous comparisons test showing that the Temporal Separation protocol (mean rank = 339.48) is significantly different from DS only (mean rank = 439.83, $p < 0.0001$), Area (mean rank = 425.60, $p = 0.0047$) and T+A (mean rank = 464.21, $p < 0.0001$). However, DS only, Area, and T+A are not signifi-

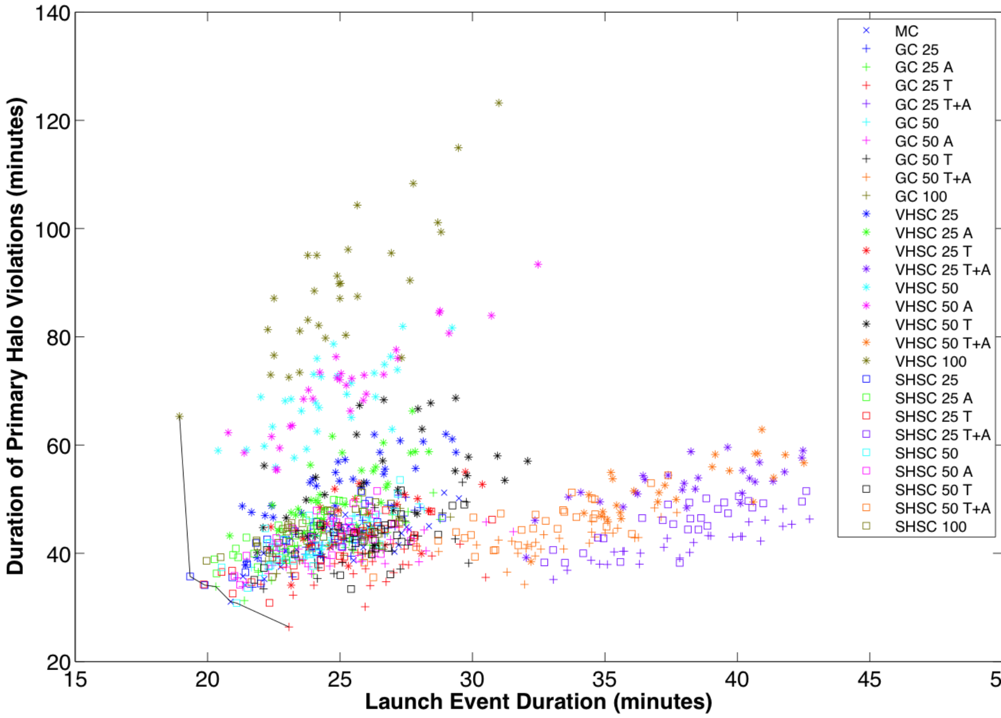


**Fig. 6 Pareto frontier plot of Duration of Primary Halo Violations (DPHV) versus LD for 22 aircraft missions.**

cantly different from one another. These results indicate a similar phenomenon as the SHSC system: that both Temporal and T+A cases results in a high number of violations, but the duration of each violation is comparatively smaller than under the Area and DS only protocols. Although the total number of violations is higher, the duration of each is far smaller under these two protocols.

**Table 6 Descriptive statistics for Duration of Primary Halo Violations (DPHV) grouped by Safety Protocol.**

| Level | Mean | Std Dev | Std Err Mean | Lower 95% | Upper 95% |
|---|---|---|---|---|---|
| Area Separation | 2688.27 | 646.19 | 37.31 | 2614.90 | 2761.70 |
| Dynamic Separation | 2848.83 | 824.40 | 37.63 | 2774.90 | 2922.80 |
| Temporal Separation | 2509.69 | 410.06 | 23.68 | 2463.10 | 2556.30 |
| Temporal+Area | 2686.81 | 397.26 | 22.94 | 2641.70 | 2731.90 |

### C. Discussion

For the safety of vehicle interactions, the Dynamic Separation only case generated far fewer violations when vehicles relied on the human crew for guidance in taxi operations (MC and GC), as opposed to the Supervisory Control architectures (VHSC and SHSC) that utilized guidance from a central supervisor. These results suggest that the use of the crew in taxi operations is an implicit form of safety constraint on the flight deck, providing some additional separation between vehicles during operations. For the supervisory control architectures operating in unconstrained operations, the Tertiary Halo Violations due to Aircraft (THVA) metric defined here suggests that these systems may be unsafe. However, given the limitations in available data on flight deck performance, it is not clear that the definition of the THVA metric is the most accurate in terms of true flight deck safety. Even so, it is interesting that the beneficial effects of using crew in taxi routing disappear as additional constraints are added in the Area and Temporal protocols.

For metrics involving crew safety, the choice of control architecture only had effects in terms of the two supervisory control architectures, and there only because of the definition of the crew "Escort" model. This model was developed solely for this work and is based on no prior experiments in Naval operations. It does suggest that, if Escorts are to be used in future operations, careful thought should be given to the specific definition of their roles in order to ensure that they are moving appropriately in and around aircraft. Alternatively, a complete removal of these Escort crew would drop SHSC PHV values to zero. For both vehicle and crew safety, simulations like MASCS provide significant utility in being able to explore close interactions between humans and vehicle systems without placing either in danger and can provide insight in where operations, and individual behaviors, might need to be adjusted in order to improve the safety of operations.

## V. Conclusions and Future Work

This paper has described the development of an agent-based model of aircraft carrier flight deck operations that includes models of futuristic unmanned vehicle systems and operational safety protocols that govern their interaction with other vehicle systems. Three different types of unmanned vehicle control architectures were tested, along with a baseline manual control model. Four types of safety protocols were modeled, ranging from a completely unconstrained Dynamic Separation only case to Area, Temporal, and combined Area+Temporal separation. Multiple tests using missions of 22 aircraft and different ratios of manned and unmanned vehicles showed significant interactions between the control architectures and safety protocols and their effects on both productivity and safety measures.

For aircraft carrier flight deck operations, these results suggest that safety is primarily influenced by the structure of operations and the constraints imposed therein. The structure of operations refers to the number of aircraft used in mission operations and the mix of unmanned and manned aircraft within that total. Constraints, in this work, relate to the safety protocols and the limitations they impose on when and where vehicles can be active on the flight deck. Interactions between these two factors require changes in the assignment of aircraft to resources on the flight deck, leading to characteristically different evolutions of missions. It is these differences in mission evolution that drive differences in safety measures.

Additionally, the types of delays and latencies that affect operations were also shown to be significant. The delays in VHSC systems pause aircraft in place while those closest to catapults are moved forward, freeing more space for movement. In GC operations, all vehicles have equal chances to become delayed due to system failures, allowing aircraft to taxi closer to those in front than would be desired. However, the undelayed and minimally constrained SHSC architecture demonstrated a wide range of performance: under no constraints, it provided some of the worst vehicle safety metrics while under high constraints provided the best.

Most importantly, these results demonstrate that a tradeoff exists in terms of the safety of flight deck operations in the current operating paradigm. Not only does this tradeoff occur between measures of safety and productivity, it occurs across different measures of safety. There is no single setting of control architecture, safety protocol, and mission Composition that provides clearly superior performance across all metrics. However, the factors that are shown to be a driver of safety performance at the mission level are characteristic of the flight deck system as a whole — the number and types of aircraft, the number of crew, the role of the crew, and the selection of safety protocols. The interactive effects of these settings that drive variations in performance in missions utilizing 22 aircraft operating simultaneously would not be observable when testing individual aircraft. In considering the introduction of unmanned vehicles in this and other similar environments, these factors should be taken into account. It is likely that completely reconfiguring flight deck operations — moving away from the current standard patterns of traffic, planning, and crew interactions — would be beneficial to the safety measures presented here.

Additionally, further exploration of the effects of vehicle performance parameters on operations is also warranted; the models used within MASCS testing assume mature vehicles and experienced vehicle operators, and the results of this work may not hold true in early phases of deployment. Similarly, these results are fully contingent on the definition of behavioral rules that govern the control architectures and safety protocols within the modeled flight deck environment. Changes to any of these rules might result in very different results than what has been observed here. The generalizability of these results is highly dependent on the similarity of the physical layout of the environment and the behaviors of human and vehicle agents in the environment. While it is difficult to know to what extent these results directly transfer to other related environments, given the unique characteristics and choreography on the aircraft carrier flight deck, the agent-based nature of MASCS enables it to be quickly extended to other domains through a simple redefinition of agents, behavioral rules, and task definitions. Extensions of MASCS to these domains are significant areas of future work.

## Acknowledgments

This research was supported by the Office of Naval Research Science of Autonomy program, Contract #N000140910625. The opinions expressed in this paper are those of the authors and do not necessarily reflect the opinions of the sponsoring agency.

## References

[1] McKinney, B., "Unmanned Combat Air System Carrier Demonstration (UCAS-D)," http://www.northropgrumman.com/Capabilities/X47BUCAS/Documents/X-47B_Navy_UCAS_FactSheet.pdf, 2011.

[2] Hennigan, W. J., "Navy drone X-47B lands on carrier deck in historic first," http://articles.latimes.com/2013/jul/10/business/la-fi-mo-navy-drone-x47b-20130709, July 2013.

[3] Bonabeau, E., "Agent-based Modeling: Methods and techniques for simulating human systems," *Proceedings of the National Academy of Sciences of the United States of America*, Vol. 99, No. Suppl. 3, 2002, pp. 7280–7287.

[4] Sycara, K. and Lewis, M., "Agent-Based Approaches to Dynamic Team Simulation," Tech. Rep. NPRST-TN-08-9, Navy Personnel Research, Studies, and Technology Division, Bureau of Naval Personnel, Millington, TN, 2008.

[5] Carley, K., Fridsma, D., Casman, E., Neal Altman, Chang, J., Kaminsky, B., Nave, D., and Yahja, A., "BioWar: Scalable Multi-Agent Social and Epidemiological Simulation of Bioterrorism Events," *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans*, Vol. 36, No. 2, 2006, pp. 252–265.

[6] Lee, S. M., Pritchett, A. R., and Corker, K. M., "Evaluating Transformations of the Air Transportation System through Agent-Based Modeling and Simulation," *Proceedings of the 7th FAA/Eurcontrol Seminar on Air Traffic Management Research and Development*, Barcelona, Spain, 2007.

[7] Rasmussen, S. J. and Chandler, P. R., "MultiUAV: A Multiple UAV Simulation for Investigation of Cooperative Control," *Proceedings of the 2002 Winter Simulation Conference*, San Diego, CA, 2002.

[8] Corner, J. J. and Lamont, G. B., "Parallel Simulation of UAV Swarm Scenarios," *Proceedings of the 2004 Winter Simulation Conference*, Washington, D. C., 2004.

[9] Weibel, R. E. and Hansman Jr., R. J., "Safety Considerations for Operation of Different Classes of UAVs in the NAS," *AIAA's 4th Aviation Technology, Integration, and Operations Forum*, Chicago, IL, 2004.

[10] Schumann, B., Scanlan, J., and Takeda, K., "Evaluating Design Decisions in Real-Time Using Operations Modelling," *Air Transport and Operations Symposium*, Delft, The Netherlands, 2011.

[11] Nehme, C., *Modeling Human Supervisory Control in Heterogeneous Unmanned Vehicle Systems*, Ph. D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, 2009.

[12] Mkrtchyan, A. A., *Modeling Operator Performance in Low Task Load Supervisory Domains*, S. M. Thesis, Massachusetts Institute of Technology, Cambridge, MA, 2011.

[13] Archer, J., *Methods for Assessment and Prediction of Traffic Safety at Urban Interactions and their Application in Micro-simulation Modelling*, Ph.D. thesis, Royal Institute of Technology, Stockholm, Sweden, 2004.

[14] Wahle, J. and Schreckenberg, M., "A Multi-Agent System for On-Line Simulations Based on Real-World Traffic Data," *Proceedings of the 34th Annual Hawaii International Conference on System Sciences*, Grand Wailea, Maui, HI, 2001.

[15] Paruchuri, P., Pullalarevu, A. R., and Karlapalem, K., "Multi Agent Simulation of Unorganized Traffic," *Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS'02)*, Bologna, Italy, 2002, pp. 176–183.

[16] Yuhara, N. and Tajima, J., "Multi-Driver Agent-Based Traffic Simulation Systems for Evaluating the Effects of Advanced Driver Assistance Systems on Road Traffic Accidents," *Cognition, Technology, and Work*, Vol. 8, 2006, pp. 283–300.

[17] Gettman, D. and Head, L., "Surrogate Safety Measures From Traffic Simulation Models Final Report," Tech. rep., U. S. Department of Transportation, McLean, VA, 2003.

[18] Pritchett, A. R., Lee, S., Huang, D., Goldsman, D., Joines, J. A., Barton, R. R., Kang, K., and Fishwick, P. A., "Hybrid-System Simulation for National Airspace System Safety Analysis," *Proceedings of the 2000 Winter Simulation Conference*, Orlando, FL, 2000, pp. 1132–1142.

[19] Xie, Y., Shortle, J., and Donahue, G., "Airport Terminal-Approach Safety and Capacity Analysis Using an Agent-Based Model," *Proceedings of the 2004 Winter Simulation Conference*, Washington, D.C., 2004, pp. 1349–1357.

[20] Blom, H. A. P., Stroeve, S. H., and de Jong, H. H., "Safety Risk Assessment by Monte Carlo Simulation of Complex Safety Critical Operations," Tech. Rep. NLR-TP-2006-684, National Aerospace Laboratory NLR, 2006.

[21] Stroeve, S. H., Bakker, G. J., and Blom, H. A. P., "Safety Risk Analysis of Runway Incursion Alert Systems in the Tower and Cockpit by Multi-Agent Systemic Accident Modelling," *The 7th USA/Europe ATM R&D Seminar*, Barcelona, Spain, 2007.

[22] Ryan, J. C. and Cummings, M. L., "Development of an Agent-Based Model for Aircraft Carrier Flight Deck Operations," *M&S Journal*, Spring 2014.

[23] Ryan, J. C., Banerjee, A. G., Cummings, M. L., and Roy, N., "Comparing the Performance of Expert User Heuristics and an Integer Linear Program in Aircraft Carrier Deck Operations," *IEEE Transactions on Cybernetics*, Vol. 44, No. 6, 2014, pp. 761–773.

[24] Cummings, M. L. and Guerlain, S., "Developing Operator Capacity Estimates for Supervisory Control of Autonomous Vehicles," *Human Factors*, Vol. 49, No. 1, 2007, pp. 1–15.

[25] Cummings, M. L., How, J., Whitten, A., and Toupet, O., "The Impact of Human-Automation Collaboration in Decentralized Multiple Unmanned Vehicle Control," *Proceedings of the IEEE*, Vol. 100, No. 3, 2012, pp. 660–671.

[26] Blank, M., Gorelick, L., Shechtman, E., Irani, M., and Basri, R., "Action as Space-Time Shapes," *Tenth IEEE International Conference on Computer Vision*, Beijing, China, 2005.

[27] Lin, Z., Jiang, Z., and Davis, L. S., "Recognizing Actions by Shape-Motion Prototypes," *12th International Conference on Computer Vision*, Kyoto, Japan, 2009.

[28] Jhuang, H., Serre, T., Wolf, L., and Poggio, T., "A Biologically Inspired System for Action Recognition," *IEEE 11th International Conference on Computer Vision*, Rio de Janeiro, Brazil, 2007.

[29] Scovanner, P., Ali, S., and Shah, M., "A 3-Dimensional SIFT Descriptor and its Application to Action Recognition," *ACM Multimedia*, Augsburg, Bavaria, Germany, 2007.

[30] Laptev, I., Marszalek, M., Schimd, C., and Rozenfeld, B., "Learning Realistic Human Action from Movies," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage, AK, 2008.

[31] Schindler, K. and van Gool, L., "Action Snippets: How many frames does human action recognition require?" *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[32] Yuan, J., Liu, Z., and Wu, Y., "Discriminative Subvolume Search for Efficient Action Detection," *IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL, 2009.

[33] Song, Y., Demirdjian, D., and Davis, R., "Continuous Body and Hand Gesture Recognition for Natural Human-Computer Interaction," *ACM Transactions on Interactive Intelligent Systems*, Vol. 2, No. 1, 2012.

[34] Cummings, M. L. and Mitchell, P. J., "Predicting Controller Capacity in Remote Supervision of Multiple Unmanned Vehicles," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, Vol. 38, No. 2, 2008, pp. 451–460.

[35] Bluman, A. G., *Elementary Statistics: A Step by Step Approach*, New York, NY, 5th ed., 2004.