Defining Collaborative Control Interactions Using Systems Theory

Andrew N. Kopeikin
Massachusetts Institute of Technology
77 Mass. Ave, Cambridge, MA 02139, USA
617-258-7537
kopeikin@mit.edu

Nancy G. Leveson
Massachusetts Institute of Technology
77 Mass. Ave, Cambridge, MA 02139, USA
617-258-7537
leveson@mit.edu

Natasha A. Neogi
National Aeronautics and Space Administration
1 NASA Dr., Hampton, VA 23666, USA
757-864-1000
natasha.a.neogi@nasa.gov

Abstract. Human teams collaborate by establishing roles, changing functional authorities, maintaining team cognition, coordinating, and mutually helping one another close control loops. These complex interactions are inspiring novel concepts to improve human-machine and multi-machine collaboration. However, these new systems face engineering gaps in modeling, analysis, design, and assurance. As such, few have been fielded in safety-critical domains like aerospace. To analyze safety, this paper introduces a system-theoretic framework to describe interactions that are—or are planned to be—used in multi-controller systems. It outlines a taxonomy of seven structural dimensions that influence controller interactions and nine dynamics observed in collaborative control that are defined using Systems Theory. An analyzed set of 101 controller interactions in aerospace systems demonstrates how to apply the framework and that designers are trying to create more sophisticated systems. This framework provides the foundation to extend system-theoretic hazard analysis techniques to systematically find collaborative-control causal factors.

Introduction

A *system* is "a set of components that act together as a whole to achieve some common goal, objective, or end". Aerospace systems components consist of hardware, software, and humans. Human to human interactions in systems can be complex. The social process of *teaming* involves dynamics in establishing roles, changing functional authorities, team cognition, coordination, and mutually helping one another close feedback control loops (Johnson et al., 2014). These mutually influential interactions allow teammates to leverage each other's contributions to joint activity to improve performance in a task or to achieve something they cannot do alone.

Despite amazing technological progress and expanding applications, the basic interactions between humans and software controllers have remained relatively limited in fielded aerospace systems. Most follow the same basic concept. Automated machines control various processes using feedback loops. Humans generally set goal conditions for the control effort, supervise the machines, and in some cases intervene.

Technological advances, limitations of current systems, and market pressures are all energizing interest in developing systems with human-machine or multi-machine interactions inspired by human teams. For example, in Urban Air Mobility (UAM), the industry envisions partnering humans with future automated controllers that would be certified to safely take on some of the operating responsibilities (GAMA, 2019). It is also exploring how teams of ground-based human operators can

share remote control of multiple UAM aircraft, possibly without any vehicle operator onboard (Holbrook et al., 2020). The airline industry is also weighing similar control concepts (Battiste et al., 2018; Shively, 2017). Similarly, civil aviation authorities are exploring how distributed decision-making and dynamic switching between human and automated control can help integrate new classes of aircraft into the National Airspace System (Holbrook et al., 2020; NASA, 2017).

The defense sector has similar ambitions. New concepts include collaboratively pairing pilots with automation that acts as an additional crewmember and dynamically offloads some operating tasks (Mosier et al., 2017). Numerous early prototypes of distributed control multi-Unmanned Aircraft System have demonstrated they can achieve group effects (Hudson, 2019). Several designs team human-piloted aircraft with unmanned aircraft to jointly execute complex missions (Fabijanowicz, 2020). Finally, some argue the military should increase its use of smaller collaborative heterogeneous unmanned systems to deliver more overwhelming effects (Clark et al., 2020).

Despite high interest in engineering aerospace systems with these more complex component interactions, none (except for the simplest designs) have been fielded. Options are constrained by current systems engineering methods, which cannot effectively model, analyze, design, and assure such systems. To do so, requires both a theoretical framework to categorize collaborative designs and a process built on that framework to do analysis. This paper suggests a theoretical framework to support the safety analysis and design of aerospace systems with these complex interactions. The framework is grounded in Systems Theory and consists of (1) a taxonomy of the structure of controller interactions and (2) a set of dynamics observed in collaborative control. Its goal is to create a foundation to develop the extended hazard analysis methods needed to systematically identify causal factors associated with these interactions. Later papers will present the extended hazard analysis methods.

The remainder of the paper is organized as follows. The next section describes a selection of the literature relevant to this work. Subsequently the proposed framework to describe collaborative control interactions using Systems Theory is outlined. Then, the framework is applied to a sample of aerospace systems to demonstrate its applicability. Finally, the last section describes future work that extends hazard analysis techniques using this collaborative control framework.

Relevant Literature Review

Theoretical Foundations of Teaming. The literature can be categorized with respect to human teams, human-machine teams, or teams made up only of machines. The following is a brief overview of previous research followed by some observed gaps of existing models.

The foundational human teaming literature is mostly grounded in human factors and organizational psychology, and provides a wide variety of models (Paris et al., 2000). One widely cited model, which describes “five core components” and “three coordinating mechanisms” for effective teamwork (Salas et al., 2005), helped identify some key considerations for the framework proposed in this paper. In addition, two complementary theories of team cognition were also considered in the framework. *Shared* situational awareness (SA) focuses on information held in common between collaborators (Endsley, 1995). *Distributed* SA is centered on SA of who knows what and when on the team (Stanton et al., 2006).

Human-Machine Team (HMT) concepts generally seek to extend Human Automation Interaction (HAI) from traditional human supervisory control over to more collaborative partnerships. One study groups different aviation HAIs by increasing interdependence with humans (Mosier et al., 2017). The groups range from basic information automation, to more elaborate supervisory control systems, and finally to interdependent HMTs that are not yet fielded. While this categorization is helpful to differentiate some HMTs from more basic HAIs, they are insufficient to identify fundamental architectural differences.

HMT research is often derived from human teaming theory even though automated software-based entities are not sentient beings. Some have suggested they should never be labeled as a teammate because they lack affective and cognitive processes comparable to those of humans (Pritchett, 2009). For example, machines do not have a sense of responsibility, motivation, loyalty, or values to guide critical problem-solving (Pritchett et al., 2018). When they face the unknown, they are unable to perform effectively beyond their programmed bounds (Leveson, 2023). They are bad at anticipating human needs (Klein et al., 2004). They handle information processing and timing differently than humans (Connors, 2017). Machines are challenged by both syntactic and semantic nuances that humans can handle in natural languages (Laird et al., 2019). Finally, machines lack expected team etiquette and often unduly interrupt humans (Battiste et al., 2018).

In machine-machine teams, many key attributes that enable distributed control of unmanned systems are similar to those previously described in human teams and HMTs. Like other teams, distributed systems must be directable and collaborate with one another to achieve certain behaviors (Ferber and Weiss, 1999). The concept of distributed control relates to that of shared authority in teaming because multiple controllers can influence a common process. Distributed systems also rely on coordination and collaboration to achieve synergistic effects (Ferber and Weiss, 1999). These interactions require bi-directional information exchange between systems (Murphey and Pardalos, 2002; Ren and Beard, 2008). They employ algorithmic processes to reach information consistency, or consensus in team cognition (Olfati-Saber and Murray, 2004).

The above body of research helps explain how teams function, but it does have some gaps. First, most models aim to improve team performance and do not focus on safety. Second, studies rarely provide tangible guidance to systematically analyze or design teaming systems. Many provide a model, but no instructions on how to use it. Third, the available direction is even sparser on how to analyze or design a team that interacts within a larger socio-technical system. Few models explore human teaming through a system-theoretic lens (Paris et al., 2000), and those that do only do so superficially.

Prior Taxonomies of System Interactions. A few taxonomies have been proposed to describe system and component interactions to support system safety analysis. One classifies system interactions using two dimensions of coupling and complexity (Perrow, 1984). However, the framework is highly subjective and emphasizes that accidents originate from component failures, which does not align with more modern views system safety (see next subsection). Another more recent study developed a 9-dimensional taxonomy for socio-technical system interactions designed to feed Functional Resonance Analysis Method (FRAM) (Saurin and Patriarca, 2020). Unfortunately FRAM assumes sequential, or acyclic, causality, which is an oversimplification of non-linear interactions that contribute to accidents (Leveson, 2023).

Other research domains have categorized system interactions. HAIs are often described along the popular Levels of Automation (LOA) axis (Sheridan and Verplank, 1978). This construct is helpful to conceptualize how automation is intended to support humans in a design, but it is also too simplistic and does not characterize collaborative behavior (Dekker and Woods, 2002). Distributed control theory has been used to define increasingly coupled control interactions, ranging from collective to coordinated (Murphey and Pardalos, 2002). Finally, multi-agent control theory has been employed to classify interactions according to the different types of communication, intent, and congruence of system-level goals (Parunak et al., 2004).

While various elements of these frameworks helped inspire the formulation proposed in this paper, these sources are still insufficient to describe the collaborative control interactions found in designed systems concepts. This requires a new taxonomy to be developed, such as the one introduced in this paper, which is based on Systems Theory.

Systems Theory. *Systems Theory* augments the scientific method in the study of increasingly complex systems developed by humans. Modern systems can no longer be analyzed using only reductionist approaches that systematically break them down into smaller components, analyze their individual behaviors, and synthesize the results to predict the behavior of the system as a whole. Such analytical decomposition limits the analysis of system behavior (Leveson, 2020). In addition, modern systems also have too much structure to assume random behavior. As a result, analysis performed using the statistical methods appropriate for studying much larger natural systems, like populations, can be misleading or incorrect (Weinberg, 2001).

Systems Theory focuses on the system as a whole rather than a conglomeration of parts. It has two guiding principles: *emergence and hierarchy*, and *communication and control* (Checkland, 1999). The first principle means that systems are modeled using hierarchal levels, where each level is more complex than the level below. The second principle implies that controls can be applied from any level in the hierarchy onto the level below to constrain degrees of freedom on its emergent behavior. These two concepts are at the core of System-Theoretic Accident Model and Processes (STAMP).

STAMP is a relatively new causality model grounded in Systems Theory (Leveson, 2011), which treats safety as a control problem rather than a reliability problem. STAMP assumes that accidents are caused by unsafe component behaviors as well as unsafe interactions among system components. These components can be hardware, software, and human in nature. Systems are modeled using control structures with feedback control loops between different hierarchal levels (Figure 1). STAMP, and its analysis tools, such as System-Theoretic Process Analysis (STPA) (Leveson and Thomas, 2018), have recently gained increasing popularity in a large number of industries.

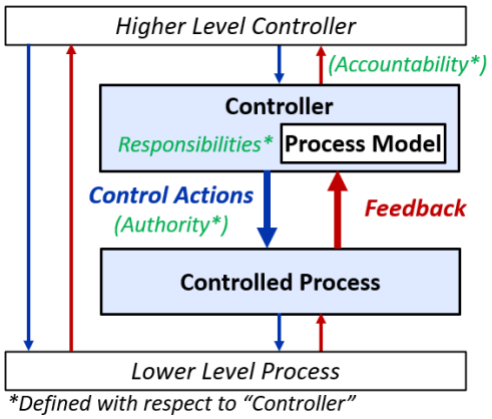

Figure 1: Example STAMP Hierarchical Control Structure

There are several reasons why Systems Theory and STAMP are well-suited to address some of the challenges in fielding novel aerospace systems with more complex component interactions. First, the STAMP causality model can handle the more complex (including non-linear, or circular) relationships between system components inherent in collaborative control. Second, it can explore interactions between different types of system components and controllers, including hardware, software, and humans. Third, it does not assume that accidents are only the result of component failures like most hazard analysis techniques. Fourth, STAMP emphasizes how a system is integrated within its larger socio-technical context. Finally, it uses abstraction to manage complexity.

Despite these strengths, many types of complex interactions that may be designed into systems have not been defined using Systems Theory. This paper aims to reason about some of the interactions observed in collaborative control so that they can be more completely analyzed using STAMP and the modeling and analysis tools built on it.

System-Theoretic Definition of Collaborative Control

A widely cited *team* definition is: "A team consists of two or more entities who interact dynamically, interdependently, and adaptively toward a common and valued goal, with unique roles and functions to perform" (Salas et al., 1992). It is nearly identical to that of a *system* (see Introduction) and therefore, is not very useful by itself in distinguishing teams from other systems.

What makes teams different and so challenging to put together is that their component interactions are more complex than those in previously fielded designed systems in safety-critical applications. These interactions can be described using Systems Theory so they can be modeled and analyzed using state-of-the-art methods in system safety. This section proposes a way do so.

Taxonomy of Controller Interaction Structure. The dynamics between multiple interacting system entities, including those observed in collaborative control, are influenced by the structure of the interaction between the controllers. The following is a proposed taxonomy of seven structural dimensions to consider in system analysis (Figure 2).

Categorizing the interactions according to these structural dimensions does not mean assigning a value on a numerical scale. The axes are ordinal. Furthermore, there is no claim that these axes are independent. Some interactions may exhibit multiple labels on each axis. The purpose of this taxonomy is to qualitatively consider structural factors that influence causality between multiple interacting controllers.

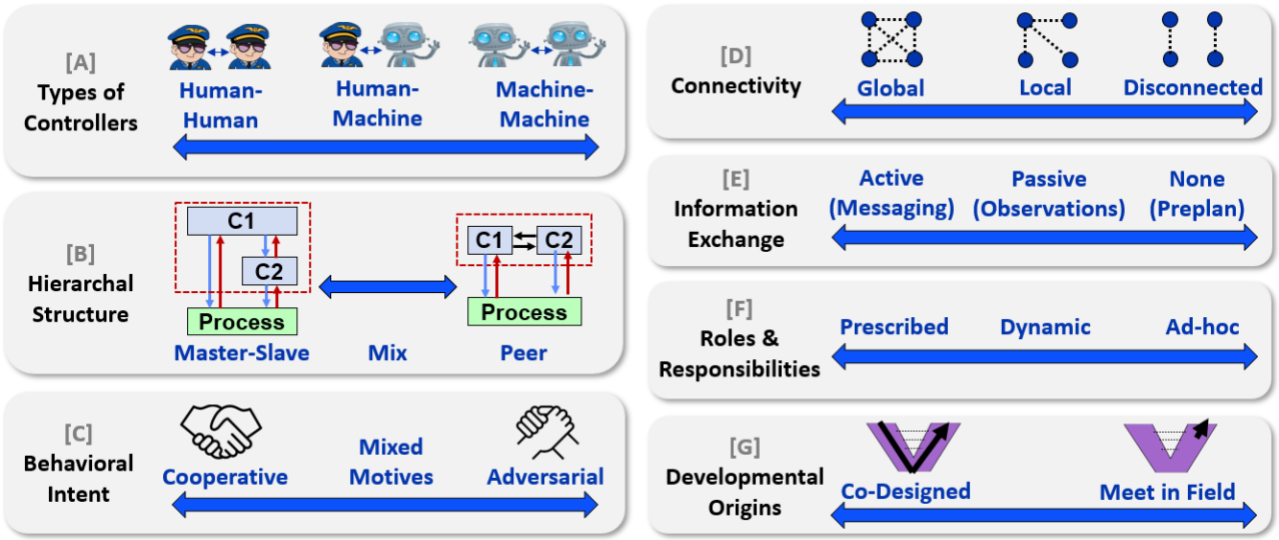

Figure 2. Taxonomy of Structure of Interactions between Multiple Controllers

First, the dynamics are shaped by the *types of controllers* [A] interacting, which may be humans only, human(s) and machine(s), or machines only. These interactions can also form the basis of higher-level systems, such as human organizations, teams of teams, or multiple human-machine systems interacting as a whole. Finally, the types of controllers may not be determined yet if the system is in the design stage. The nature and differences between interacting controllers have significant implications for how they coordinate, share information, and make control decisions.

Next, the *hierarchal structure* [B] between controllers can vary from a *master-slave* control relationship to a partnership between *peers*. Some interactions can exhibit a mix of both traits. For example, interactions among human pilots involve both supervisory control and peer partnership at different levels of work. The captain directs the roles and responsibilities of the first officer. However, both pilots also work together at a lower level to aviate, navigate, and communicate.

Entities can exhibit a range of *behavioral intent* [C] toward one another. Designed systems often consist of components that are mutually cooperative to achieve a common objective. On the opposite end, some systems contend with adversarial interactions, such as those found in competition and conflict (Parunak et al., 2004). Between these two ends, there are also interactions without reciprocal intent. Two entities may consist of one behaving cooperatively whereas the other is not. Adversarial intent can also be unilateral. Furthermore, interactions may also be influenced by mixed motives.

Multi-controller interactions are also influenced by their types of *connectivity* [D]. Connectivity may be *global,* in which all controllers involved in joint activity can directly exchange information with one another. It can also be *local*, where two controllers that are not directly connected may exchange information indirectly through other entities. Finally, some controllers or groups of controllers may be disconnected and not be able to exchange information.

Next, there are different types of *information exchange* [E] that can occur along these connections. Communications can involve active messaging where content, such as controller states and intentions, is encoded and shared. Communication can also be passive, in which a controller observes the behavior of another to estimate its state (Murphey and Pardalos, 2002). Finally, the existence of lines of communication does not mean information exchange actually takes place. The activity of multiple controllers can be coordinated a priori in an open-loop via preplanned policies only.

The roles and responsibilities [F] of interacting entities can range from prescribed to ad-hoc. In simpler systems, the responsibility, authority, and accountability for controllers are prescribed using modes so that control boundaries are defined to avoid overlaps. For example, a pilot can turn on "heading hold" mode on an autopilot to entirely delegate that task to the machine.

Other systems may intentionally allow more dynamic control boundaries with overlap between multiple entities to improve performance. For example, there are many concepts of teams of multiple robots that share a common set of tasks and must coordinate and deconflict task allocation during operation. Furthermore, roles and responsibilities may even be ad-hoc and determined only during execution. This ad hoc teaming occurs in human interactions, for example, when a team forms to handle a roadside accident.

Finally, interactions are also influenced by their *developmental origins* [G], i.e., the time when they are considered in the engineering life-cycle. Some controllers are co-designed to interact from the beginning, while for others, new functional interactions are only considered post-fielding. In between, this axis also includes separate designs of interacting controllers according to a common specification. It also contains unilateral relationships, where one controller is developed to interact with another that is already fielded.

Collaborative Control Dynamics. Nine collaborative control dynamics have been derived from teaming fundamentals (previous section) and the study of certain novel aerospace concepts (see Introduction and next section). When present, their effects must be systematically considered to identify potential causes of hazardous control. The dynamics are illustrated in Figure 3 and then described using the key principles of Systems Theory and STAMP.

The concept of emergence implies that collaborative control dynamics can only be described in terms of the interactions among multiple entities. In this case, the interacting entities are two or more controllers engaged in joint activity. The collaborative control dynamics are irreducible in that they have no meaning for any individual controller. The concepts of hierarchy, communication, and control are all explicitly represented in the models shown in Figure 3.

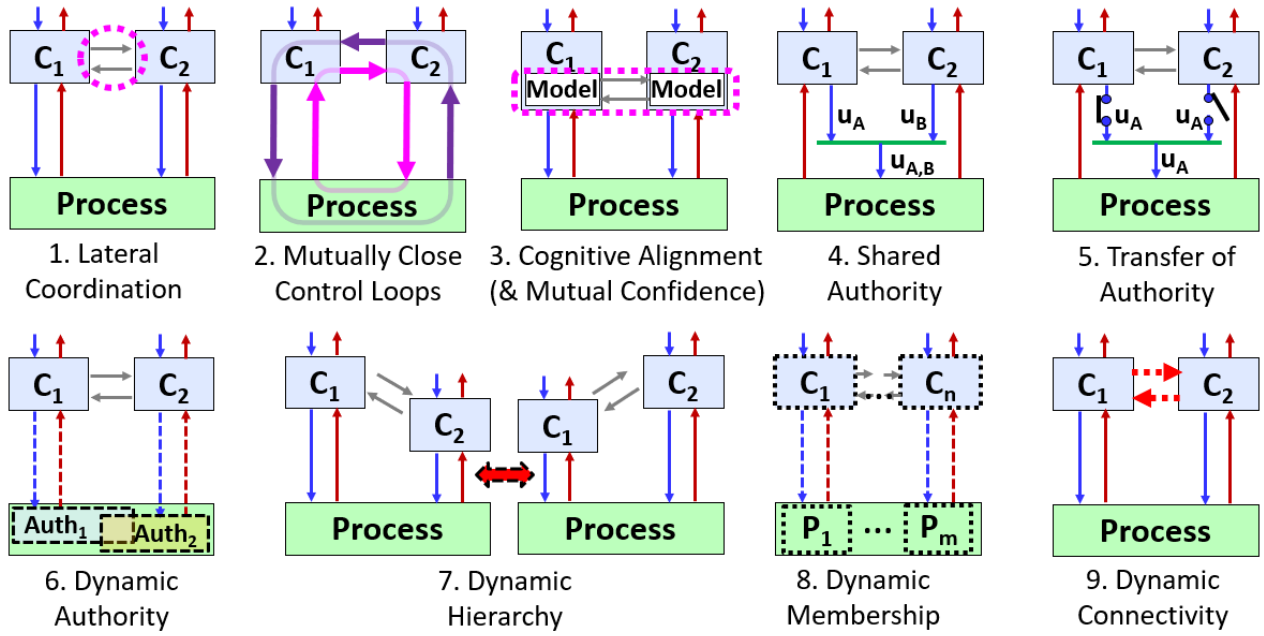

Figure 3. System-Theoretic Collaborative Control Dynamics

1. Lateral coordination is a causal process between peers that does not imply control (Johnson, 2017; Parunak et al., 2004). This dynamic is enabled by the hierarchal structure of the interacting controllers, as described in the taxonomy. Lateral coordination involves multiple controllers that have some independence in decision-making to achieve their goals.

2. Mutually closing control loops is central to interdependence analysis in coactive design (Johnson et al., 2014). Controllers observe, predict, and direct each other bi-directionally to execute joint activity. This dynamic implies that a controller may issue a command to a process and depend on feedback from another controller to determine its impact. It also means a controller may influence another controller to control a process and depend on feedback from the process to assess the influence. This dynamic implies that control loops between any one controller and the controlled process cannot be analyzed individually.

3. Cognitive alignment is the process of establishing and maintaining consistent shared (Endsley, 1995) and distributed (Stanton et al., 2006) process models between collaborating controllers. Accidents often occur because controllers have flawed models of the process they are controlling (Leveson, 2011). In collaborative control, these models include estimates of the states and behaviors of the controlled process and collaborating controllers involved. Any one controller may not have a full model of the controlled process and may depend on model variables held by those other controllers.

Mutual confidence is also part of cognitive alignment. It relates to mutual trust, a key mechanism for effective teamwork (Salas et al., 2005). Confidence is relevant in this research because it is encodable in machines, unlike trust, which is inherently a human property. As proposed for trust by Chancey et al. (Chancey et al., 2021), confidence is an element of the process model for each controller regarding the expected behavior of other interacting controller(s). A controller's confidence in other controllers may influence its control decisions.

4. Shared authority occurs when multiple controllers have authority over one process or multiple interdependent subprocesses. This dynamic is common in hierarchal systems, in which a supervisor (typically human) can delegate certain functions to another controller under its supervision (often automated systems) but can also intervene and reclaim control. However, shared authority can also occur when controllers interact as peers. In some cases, a system may require simultaneous inputs

from multiple controllers to control the same process. A key implication of shared authority is that the control actions from multiple controllers cannot be analyzed individually.

5. Transfer of authority is a dynamic that can only occur in the presence of *shared authority*. It specifically describes how some systems perform handoffs of a common control action between multiple controllers over time. Handoffs may be done to dynamically reallocate control authorities to address shortfalls identified during operation. Not all systems with *shared authority* exhibit *transfer of authority.*

6. Dynamic authority also requires the presence of, but is not implied by, *shared authority*. Dynamic Authority enables multiple controllers with overlapping control boundaries to shift the division of labor during execution of joint activity. As previously described, some systems are intentionally designed with this overlap to coordinate adjustments to task allocation to improve performance. The ability to adapt, redirect resources, and change collective behavior is important in effective teamwork (Salas et al., 2005). However, this dynamic can lead to conflicts or gaps in authority, which are well-known causal factors for hazardous behavior in coordinated control (Leveson, 2011).

7. Dynamic hierarchy occurs in Mixed-Initiative Interactions (Allen et al., 1999). Controllers alternate in leading the team in the execution of various parts of the joint activity. This dynamic can be beneficial in improving performance when certain controllers are contextually better suited to control others. Such interactions are commonly observed in human dialog and are of interest for designing collaborative systems (Allen et al., 1999).

8. Dynamic membership means the set of controllers engaged in joint activity can change over time. Collaborative teams may lose, swap out, or gain members during activity. This dynamic may require controllers to track the set of active collaborators as a process model variable. Similarly, the controlled process may be composed of dynamic subprocesses that come and go over time. For this reason, dynamic membership may require the system to exhibit dynamic authority to operate safely.

9. Dynamic connectivity indicates the network topology between controllers changes over time while the controllers are interacting. Most systems may be subject to failed channels of control, feedback, and information flow. However, this dynamic specifically focuses on systems for which these channels are expected to vary as part the operation.

Analyzing Systems for Collaborative Control Dynamics

A sample of 101 interactions found in aerospace systems[1] was used to evaluate the ability to use the categorization (framework) introduced in the previous section. The sample contains both fielded systems and unfielded systems, e.g., systems in concept development, systems that have been prototyped but not yet fielded, etc. While this set consists of systems encountered in the teaming literature, including all those introduced in the introduction, it is not necessarily representative of all possible and actual aerospace systems. This section provides (1) a demonstration of how to categorize controller interactions according to the framework and (2) a comparison of the presence of collaborative control dynamics in systems that are fielded or under development today.

Demonstration of the Framework. The framework was applied to the main component interactions of analyzed systems. An example of how this was done is provided for the Airborne Collision Avoidance System (ACAS-X) and specifically for interactions between ACAS-X units on different airplanes. ACAS-X also interacts with flight crews and with Air Traffic Controllers (ATC), but those interactions are categorized differently according to the framework.

[1] The full dataset is available at: http://sunnyday.mit.edu/

ACAS-X is a series of developmental upgrades to the fielded Traffic Collision Avoidance System (TCAS) to handle a wider range of aircraft operations (SC-147, 2020). ACAS-X units on different aircraft interact as a collaborative machine team to jointly ensure collision avoidance. The categorization of these interactions is shown in Figure 4 and described below.

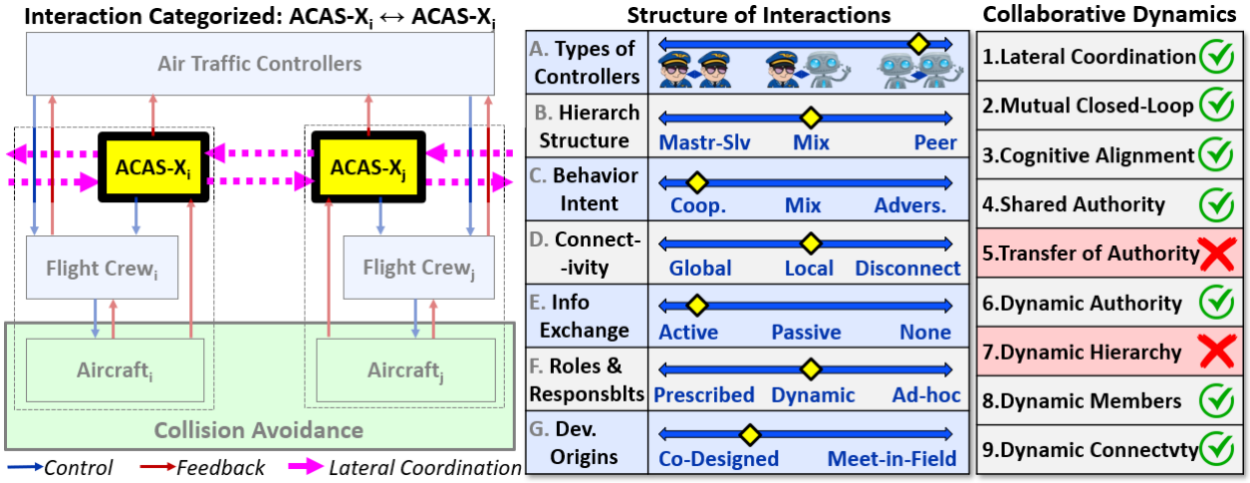

Figure 4. Categorization of Interactions between ACAS-X Aircraft

The structure of interactions is categorized according to dimensions [A-G] as follows. First, the types of controllers [A] are machines, which consist of transponder-computer units that autonomously exchange information with one another. These machines communicate their decisions to human flight crews and ATC using other interactions.

The hierarchal structure [B] between these controllers exhibits both hierarchal control and peer-level interactions. A hierarchy is established between aircraft based on their transponder identifiers to allow the higher-ranked aircraft to overrule the lower-ranked aircraft in certain situations. However, the lower-ranked aircraft may also share its resolution intent first, in such a way that influences the higher-ranked aircraft's decision.

The behavioral intent [C] between ACAS-X controllers is cooperative. The connectivity [D] in networks of ACAS-X aircraft is local as resolution advisories are coordinated between aircraft pairs only. The information exchanged [E] consists of active coordination with content-based messages.

The roles and responsibilities [F] of the controllers are dynamic. In every encounter, the entities must determine their relative hierarchy, which specifies their respective responsibilities and authorities. Finally, the developmental origins [G] of individual ACAS-X systems are not all co-developed, but they are all designed according to a common standard intended to work collaboratively.

Aircraft interactions using ACAS-X exhibit seven of the nine collaborative control dynamics. They coordinate laterally as peers by exchanging resolution advisories and sharing their positions. They mutually close each other's control loop of collision avoidance by selecting both aircraft control inputs and messages to one another based both on their relative position and their exchanged intent. They facilitate cognitive alignment through shared information about the state and intended behavior of each entity. This alignment also includes mutual confidence because logic is in place to assess whether a peer is executing the expected deconfliction maneuver.

This multi-controller system exhibits shared authority over the controlled process of collision avoidance. Dynamic authority is also observed as each ACAS-X controller pair must determine its hierarchal structure, and as a result, define during operation the control boundaries of each participant. The team also has dynamic membership as the set of aircraft that needs to mutually

deconflict changes throughout execution. Finally, there is dynamic connectivity given that ACAS-X aircraft cannot assume fully reliable communications.

Some collaborative control dynamics are not observed between interacting ACAS-X aircraft. First, ACAS-X controllers do not transfer authority of their controlled process over to other ACAS-X controllers. When interacting, each ACAS-X controller maintains its control authority over collision avoidance, which is actuated by its flight crew controlling the aircraft according to its resolution advisories. Similarly, the system does not exhibit dynamic hierarchy. While two ACAS-X units interact, there is no provision for one to be the ranking unit for part of the time, and then dynamically switch this role with the other.

For some cases in the analyzed set of 101 interactions, insufficient information was available to label one or more of the dimensions, and those items were not evaluated for that interaction. In other cases, the description made it possible for different versions of the system to be categorized differently. In such instances, dynamics were rated as 0.5 on the 0-1 scale of whether or not the dynamic is present. Similarly, multiple labels were assigned for the structure of interaction in such cases.

Results of the Categorization. Observations noted in categorizing sampled interactions are grouped into four categories: (1) human-machine (HM) or machine-machine (MM) interactions in fielded systems, (2) HM or MM interactions in conceptual unfielded systems, (3) human to human (HH) interactions in fielded systems, and (4) HH interactions in unfielded systems.

These four groups imply a categorization in the *types of controllers* dimension of the interaction structure taxonomy. Figure 5 shows the relative distribution of how the other six dimensions were labeled for each of the four groups. In Figure 5 and Figure 6, ⌐*fielded* means *unfielded*.

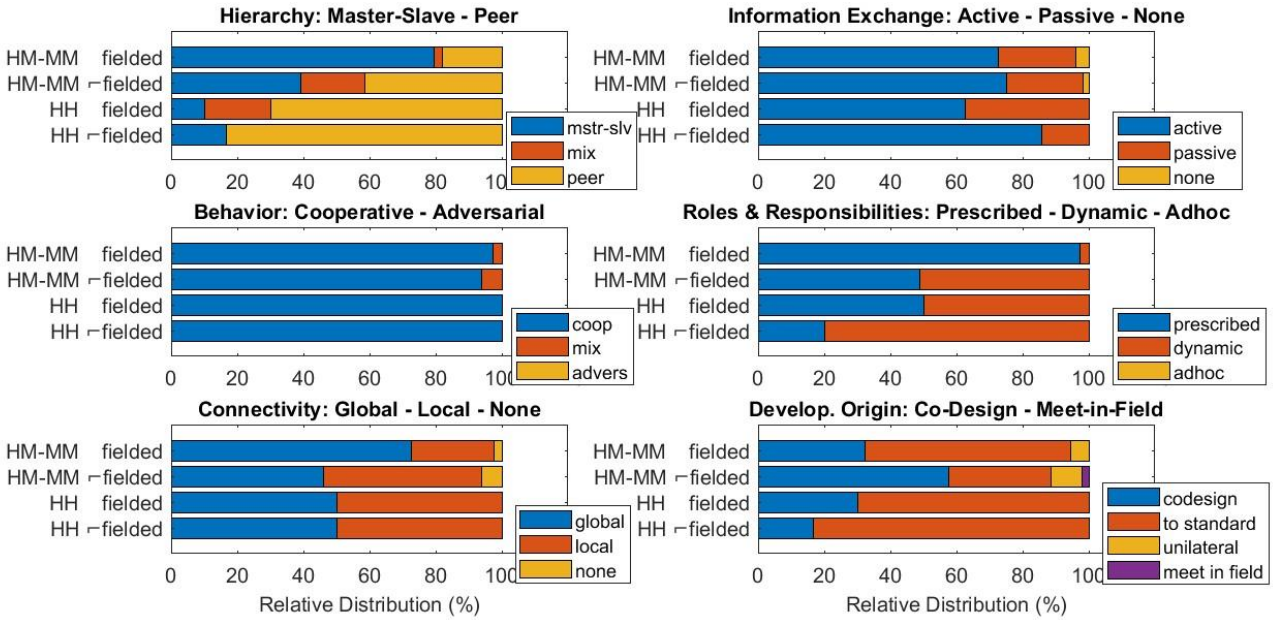

Figure 5. Comparison of the Structure of Controller Interactions

Three dimensions have distributions for which HM and MM interactions in unfielded systems are closer to those found in HH interactions, fielded and not, than those seen in HM and MM interactions of fielded systems. The sampled conceptual systems have more *peer-level structures*, *local connectivity*, and *dynamic roles and responsibilities* in their interactions than the currently fielded systems in the sample.

Two dimensions have similar distributions across the all groups. Nearly all of the systems studied had *cooperative intent*, and most of the systems used *active communications* to coordinate. This

observation may indicate that differences in these structural elements of multi-controller interaction are less pronounced between systems with collaborative control and those with simpler relationships.

Finally, the developmental origin axis actually aligned fielded HM and MM interactions more closely with HH interactions. In these cases, most fielded HM and MM interactions systems were developed separately to a *common standard*. Conversely, the majority of unfielded HM and MM interactions occurred between co-designed controllers. This outcome is more than likely influenced by the very nature of whether a system is fielded or still in design. Many novel systems do not yet have common standards.

Next, the prevalence of the nine collaborative control dynamics is compared between the four groups. Figure 6 captures the percentage of interactions sampled that exhibit each dynamic. For the most part, HH interactions in the sampled set, fielded and not, exhibit more of each of these dynamics than HM and MM interactions. However, the results also indicate that HM and MM interactions in unfielded systems exhibit more of every one of the dynamics than in fielded systems in the sample.

These results are not intended to quantitatively suggest that all aerospace systems follow these trends. The systems analyzed were chosen from the literature related to teaming, and they are therefore not representative of the overall population of systems.

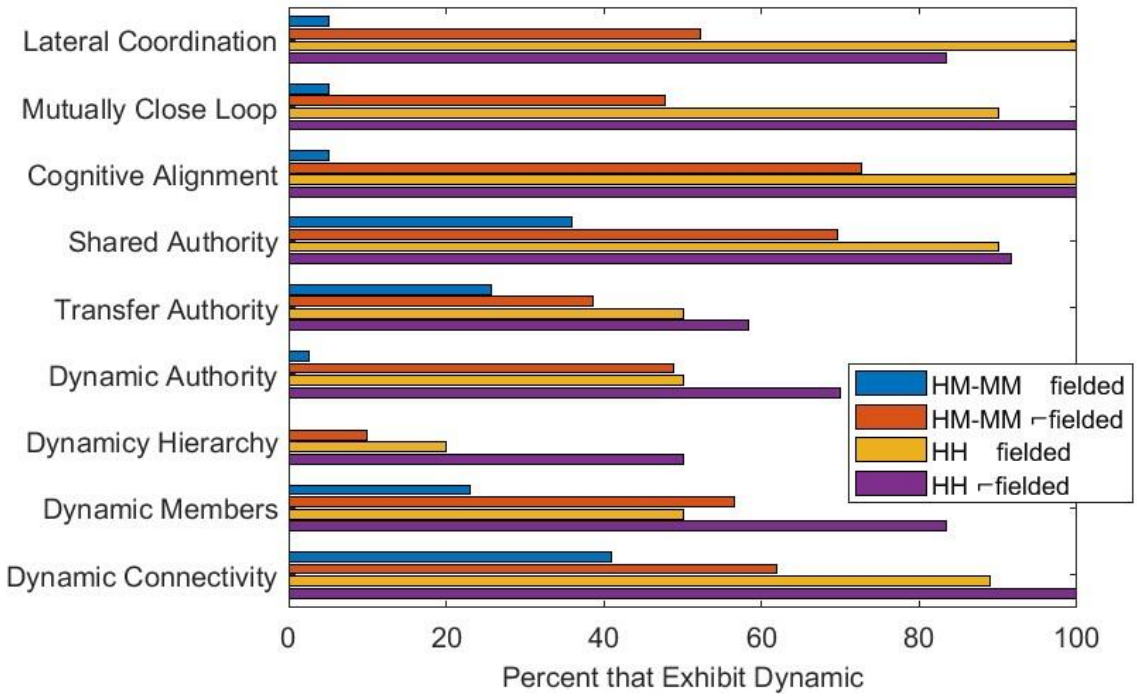

Figure 6. Percentage of Interactions that Exhibit Each Collaborative Control Dynamic

However, there are two important takeaways from this analysis. First, there is evidence in the literature that systems are being designed to exhibit each of these complex collaborative control dynamics. And second, of the systems sampled, those that have not yet been fielded tend to exhibit more of these complex interactions. These points support the argument that causal factors associated with these dynamics must be considered in safety analysis and design.

Conclusions and Future Work

The aerospace community is designing new systems that improve how human-machine and machine-machine teams work collaboratively. Unfortunately, our current Systems Engineering processes

cannot model, analyze, design, and assure systems with multiple, complex human-machine interactions to be safe. As a result, these systems have not yet been fielded.

This research aims to overcome some of these limitations by developing a rigorous and systematic technique to analyze system safety of collaborative multi-controller teams. To accomplish this goal, the diverse types of collaboration that multi-controller teams require for safety need to be defined. This paper introduced a system-theoretic framework to define the different types of interactions that are—or are planned to be—designed into aerospace systems. The taxonomy helps reason about seven different structural dimensions in controller interactions. In addition, a set of nine dynamics observed in collaborative control are defined using Systems Theory.

To determine whether and how this taxonomy relates to fielded and proposed systems, a set of 101 system interactions studied in this work was categorized using the proposed framework. The associated analysis supports the hypothesis underlying the taxonomy that each of the collaborative control dynamics is found in system concepts the aerospace community wants to field. It also shows that these dynamics are more prevalent in conceptual systems than in those already fielded.

Unfortunately, existing hazard analysis techniques are not well suited to systematically handle collaborative control interactions. Most methods assume acyclic causality, in which one element can influence another, but not vice versa. They often also focus on hardware failures rather than unsafe behaviors that can result from component interactions, which may include human and software elements (Leveson, 2011). These assumptions fall short for the analysis of collaborative control.

STAMP-based techniques, such as STPA, overcome these challenges by treating safety as a feedback control problem. However, even these techniques must be extended to systematically address all the dynamics present in collaborative control. The system-theoretic definition of collaborative control dynamics proposed in this paper provides the necessary foundation for such extensions. To that end, several efforts are in development to systematically identify causal factors associated with these interactions. These extensions will be presented in future work.

Finally, while the present research is scoped to enhance safety-guided design, the same system-theoretic foundation can also help analyze other emerging properties in collaborative systems. For example, STPA is also used to engineer security into systems (Young and Leveson, 2014). The same principle holds true in this work, and the developed framework will provide an avenue for future research to expand on its application.

Acknowledgments

The authors acknowledge the following researchers for their perspectives and examples on teaming systems. Discussions with them helped, in part, shape the taxonomy introduced in this paper: Emily Cowen, Andrew Heier, Kyle Ingols, Reed Jensen, Kevin Lahey, Caroline Lamb, Vincent Mancuso, Mike Pietrucha, and Rohan Paleja.

References

Allen, J.E., Guinn, C.I., Horvtz, E., 1999. Mixed-initiative interaction. IEEE Intell. Syst. Their Appl. 14, 14–23. https://doi.org/10.1109/5254.796083

Battiste, V., Lachter, J., Brandt, S., Alvarez, A., Strybel, T.Z., Vu, K.-P.L., 2018. Human-Automation Teaming: Lessons Learned and Future Directions, in: Yamamoto, S., Mori, H. (Eds.), Human Interface and the Management of Information. Information in Applications and Services. Springer International Publishing, pp. 479–493.

Chancey, E.T., Politowicz, M.S., Le Vie, L., 2021. Enabling Advanced Air Mobility Operations through Appropriate Trust in Human-Autonomy Teaming: Foundational Research Approaches and Applications. Presented at the AIAA Scitech 2021 Forum. https://doi.org/10.2514/6.2021-0880

Checkland, P., 1999. Systems Thinking, Systems Practice. Wiley.
Clark, B., Patt, D., Schramm, H., 2020. Mosaic Warfare: Exploiting Artificial Intelligence and Autonomous Systems to Implement Decision-Centric Operations.
Connors, M.M., 2017. Concepts for the Design of Human-Autonomy Systems. NASA Ames Res. Cent.
Dekker, S.W.A., Woods, D.D., 2002. MABA-MABA or Abracadabra? Progress on Human-Automation Co-ordination. Cogn. Technol. Work 4, 240–244. https://doi.org/10.1007/s101110200022
Endsley, M.R., 1995. Toward a Theory of Situation Awareness in Dynamic Systems. Hum. Factors 37, 32–64. https://doi.org/10.1518/001872095779049543
Fabijanowicz, J.R., 2020. Design of Experiments for Air Launched Effects Unmanned Aerial Vehicles (Masters Thesis). Naval Postgraduate School, Monterey, CA.
Ferber, J., Weiss, G., 1999. Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence. Reading: Addison-Wesley.
GAMA, 2019. A Rational Construct for Simplified Vehicle Operations (SVO).
Holbrook, J., Prinzel, L.J., Chancey, E.T., Shively, R.J., Feary, M., Dao, Q., Ballin, M.G., Teubert, C., 2020. Enabling Urban Air Mobility: Human-Autonomy Teaming Research Challenges and Recommendations. Presented at the AIAA AVIATION 2020 FORUM. https://doi.org/10.2514/6.2020-3250
Hudson, A., 2019. The Looming Swarm. Air Force Mag. URL www.airforcemag.com/article/the-looming-swarm/ (accessed 1.21.22).
Johnson, K.E., 2017. Systems-Theoretic Safety Analyses Extended for Coordination (PhD Dissertation). Massachusetts Institute of Technology, Cambridge, MA.
Johnson, M., Bradshaw, J.M., Feltovich, P.J., Jonker, C.M., Van Riemsdijk, M.B., Sierhuis, M., 2014. Coactive Design: Designing Support for Interdependence in Joint Activity. J. Hum.-Robot Interact. 3, 43. https://doi.org/10.5898/JHRI.3.1.Johnson
Klein, G., Woods, D.D., Bradshaw, J.M., Hoffman, R.R., Feltovich, P.J., 2004. Ten challenges for making automation a "team player" in joint human-agent activity. IEEE Intell. Syst. 19, 91–95. https://doi.org/10.1109/MIS.2004.74
Laird, J., Ranganath, C., Gershman, S., 2019. Future Directions in Human Machine Teaming Workshop. Presented at the Future Directions Workshop Series Sponsored by Office of Under Secretary of Defense for Research and Engineering, Arlington, VA.
Leveson, N.G., 2023. Introduction to System Safety Engineering, In Production. ed. MIT Press, Cambridge, MA.
Leveson, N.G., 2020. Safety III: A Systems Approach to Safety and Resilience.
Leveson, N.G., 2011. Engineering a safer world: systems thinking applied to safety, Engineering systems. MIT Press, Cambridge, MA.
Leveson, N.G., Thomas, J.P., 2018. STPA Handbook.
Mosier, K.L., Fischer, U., Burian, B.K., Kochan, J.A., 2017. Autonomous, Context-Sensitive, Task Management Systems and Decision Support Tools I: Human-Autonomy Teaming Fundamentals and State of the Art. NASA Rep. https://doi.org/10.13140/RG.2.2.25859.60966
Murphey, R., Pardalos, P.M., 2002. Cooperative Control and Optimization, Applied Optimization. Kluwer Academic Publishers.
NASA, 2017. NASA Aeronautics Strategic Implementation Plan.
Olfati-Saber, R., Murray, R.M., 2004. Consensus Problems in Networks of Agents With Switching Topology and Time-Delays. IEEE Trans. Autom. Control 49, 1520–1533. https://doi.org/10.1109/TAC.2004.834113
Paris, C.R., Salas, E., Cannon-Bowers, J.A., 2000. Teamwork in multi-person systems: a review and analysis. Ergonomics 43, 1052–1075. https://doi.org/10.1080/00140130050084879

Parunak, H.V.D., Brueckner, S., Fleischner, M., Odell, J.J., 2004. A Design Taxonomy of Multi-Agent Interactions, in: Giorgini, P., Müller, J.P., Odell, J.J. (Eds.), Agent-Oriented Software Engineering IV: 4th International Workshop, AOSE. Springer, New York, pp. 132–146.

Perrow, C., 1984. Normal Accidents: Living with High Risk Technologies. Princeton University Press.

Pritchett, A., Portman, M., Nolan, T., 2018. Research & Technology Development for Human-Autonomy Teaming Final Report: Literature Review and Findings from Stakeholder Interviews.

Pritchett, A.R., 2009. Aviation Automation: General Perspectives and Specific Guidance for the Design of Modes and Alerts. Rev. Hum. Factors Ergon. 5, 82–113. https://doi.org/10.1518/155723409X448026

Ren, W., Beard, R.W., 2008. Distributed Consensus in Multi-vehicle Cooperative Control, Communications and Control Engineering. Springer London, London. https://doi.org/10.1007/978-1-84800-015-5

Salas, E., Dickinson, T., Converse, S., Tannenbaum, S., 1992. Toward an understanding of team performance and training, in: Teams: Their Training and Performance, Teams: Their Training and Performance. Ablex Publishing, Westport, CT, US, pp. xvi, 415.

Salas, E., Sims, D.E., Burke, C.S., 2005. Is there a "Big Five" in Teamwork? Small Group Res. 36, 555–599. https://doi.org/10.1177/1046496405277134

Saurin, T.A., Patriarca, R., 2020. A taxonomy of interactions in socio-technical systems: A functional perspective. Appl. Ergon. 82, 102980. https://doi.org/10.1016/j.apergo.2019.102980

SC-147, 2020. DO-386: Minimum Operational Performance Standards for Airborne Collision Avoidance System Xu (ACAS Xu).

Sheridan, T.B., Verplank, W.L., 1978. Human and computer control of undersea teleoperators. Massachusetts Inst of Tech Cambridge Man-Machine Systems Lab.

Shively, J., 2017. Tech Activity Update US (NASA) - Human Autonomy Teaming - Model, Agent, Principles & Patterns. NASA Ames Res. Cent.

Stanton, N.A., Stewart, R., Harris, D., Houghton, R.J., Baber, C., McMaster, R., Salmon, P., Hoyle, G., Walker, G., Young, M.S., Linsell, M., Dymott, R., Green, D., 2006. Distributed situation awareness in dynamic systems: theoretical development and application of an ergonomics methodology. Ergonomics 49, 1288–1311. https://doi.org/10.1080/00140130600612762

Weinberg, G., 2001. An Introduction to General Systems Thinking. Dorcet House Publishing, New York, NY.

Young, W., Leveson, N.G., 2014. An integrated approach to safety and security based on systems theory. Commun. ACM 57, 31–35. https://doi.org/10.1145/2556938

Biography

Andrew N. Kopeikin is a Ph.D. Candidate at MIT in the Department of Aeronautics and Astronautics. His research interests lie at the intersection of systems engineering, system safety, and autonomous collaborative control systems. Andrew is also a Technical Staff at MIT Lincoln Lab, a USAF Reserve Officer, an active flight instructor, and prior to starting at MIT taught at the US Military Academy at West Point. He has a M.S. from MIT and a B.S. from University of Illinois Urbana Champaign, both in Aerospace Engineering.

Nancy G. Leveson is a Professor of Aeronautics and Astronautics and also Professor of Engineering Systems at MIT. She is an elected member of the National Academy of Engineering (NAE). Prof. Leveson conducts research on the topics of system safety, software safety, software and system engineering, and human-computer interaction. She has received the ACM Allen Newell Award for outstanding computer science research, the AIAA Information Systems Award for "developing the field of software safety", and the ACM Sigsoft Outstanding Research Award. She has published more than 200 research papers and is the author of two books. She consults extensively in industries on ways to prevent accidents.

Natasha A. Neogi is currently NASA's Acting Senior Technologist for Intelligent Flight Systems and the Subproject Manager of NASA's System-Wide Safety Project Safety Demonstrator Series. Her primary research interests are in the verification and validation of software-intensive safety-critical infrastructure systems, as well as certification issues concerning airworthiness of non-conventionally piloted vehicles. She received her Ph.D in Aeronautical and Astronautical Engineering from the Massachusetts Institute of Technology. She was the recipient of NASA's 2021 Outstanding Leadership Medal and has numerous awards and publications in AIAA, IEEE and ACM conferences and journals.

Appendix. System Sample Categorization Data

Table A2 shows categorization data of major controller interactions for systems sampled from the literature reviewed for this work. Each interaction is categorized according to the taxonomy of structure of interactions between controllers, and for the presence of the collaborative control dynamics described in Section 3. The legend in Table A1 explains the meaning of symbols in the data.

Table A1. Legend for Categorization Data

Header	**Symbol and Meaning**
Fielded	0: system is not fielded (still in various development stages) 1: system is fielded
Type of Controllers	HH: Human – Human Interaction HM: Human – Machine Interaction MM: Machine – Machine Interaction SS: Higher-level abstractions with combination of above
Hierarchical Structure	H: Hierarchal or Master-Slave Interaction P: Peer Interaction HP: Mix of Hierarchal and Peer Interactions
Behavioral Intent	1: Primarily cooperative in intent 2: Primarily unknown and/or mixed-motive intent 3: Primarily adversarial in intent
Connectivity	G: Controllers globally connected L: Controllers locally connected N: Controllers not connected *Multiple: different modes of interaction can exist*
Information Exchange	A: Active (deliberate content-based messaging) P: Passive (observation only) N: No Information Exchange *Multiple: different modes or multiple simultaneous types of exchanges*
Roles & Responsibilities	P: Prescribed (control boundaries delineated to avoid overlap per mode) D: Dynamic (control boundaries overlap coordinated during execution) A: Ad-hoc (control boundaries not predefined, determined in execution)
Collaborative Control Dynamics	0: Dynamic not exhibited 1: Dynamic exhibited 0.5: Dynamic may or may not be exhibited in different system versions

Table A2. Categorization Data for Interactions in Systems Sampled

#	System (Interacting Entities)	Fielded	Structural Dimensions of Controller Interaction							Collaborative Control Dynamics								
			Type Controllers	Hierarchal Struct	Behavior Intent	Connectivity	Info Exchange	Roles & Resp	Dev Origins	Lateral Coord	Mutual Close Loop	Cognitive Align	Share Authority	Transfer Authority	Dyn Authority	Dyn Hierarchy	Dyn Membership	Dyn Connectivity
1	Cockpit Situation Display (Pilot – CSD) [21]	1	HM	H	1	G	A	P	2	0	0	0	0	0	0	0	0	0
2	Enhanced Vision Sys (Pilot - EVS) [21]	1	HM	H	1	G	A	P	2	0	0	0	0	0	0	0	0	0
3	Synthetic Vision Sys (Pilot - SVS) [21]	1	HM	H	1	G	A	P	2	0	0	0	0	0	0	0	0	0
4	Combine Vision Sys (Pilot - CVS) [21]	1	HM	H	1	G	A	P	2	0	0	0	0	0	0	0	0	0
5	External Vision Sys (Pilot - XVS) [21]	1	HM	H	1	G	A	P	2	0	0	0	0	0	0	0	0	0
6	Heads Up/Mounted Display (Pilot - HUD/HMD) [21]	1	HM	H	1	L	A	P	2	0	0	0	0	0	0	0	0	0
7	Ground Prox Warning Sys (Pilot - GPWS) [21]	1	HM	H	1	G	A	P	2	0	0	0	0	0	0	0	0	0
8	Terrain Awareness & Warning Sys (Pilot - TAWS) [21]	1	HM	H	1	G	A	P	2	0	0	0	0	0	0	0	0	0
9	Cockpit Display of Traffic Info (Pilot–Display) [21]	1	HM	H	1	G	A	P	2	0	0	0	0	0	0	0	0	0
10	Auto Dependent Surv Broadcast In (Pilot - ADS-B In) [21]	1	HM	H	1	G	A	P	2	0	0	0	0	0	0	0	0	1
11	NextGen SURF-IA (Pilot – Display) [21]	1	HM	H	1	G	A	P	2	0	0	0	0	0	0	0	0	0
12	RNAV & RNP Display Tools (Pilot – Tool) [21]	1	HM	H	1	G	A	P	2	0	0	0	0	0	0	0	0	0
13	NextGen On-Demand NAS Info (Human – Tool) [21]	1	HM	H	1	G	A	P	2	0	0	0	0	0	0	0	0	0
14	Overrun Protection Sys (ROPS) (Pilot – Tool) [21]	1	HM	H	1	G	A	P	2	0	0	0	0	0	0	0	0	0
15	Emergency Landing Planner (Pilot – ELP) [21]	0	HM	H	1	G	A	P	2	0	0	0	0	0	0	0	0	0
16	Electronic Centralized A/C Monitor (ECAM) [21]	1	HM	H	1	G	A	P	1	0	0	0	0	0	0	0	0	0
17	Auto Ground Collision Avoidance Sys (Pilot – AGCAS) [21]	1	HM	H	1	G	A	P	2	0	0	0	1	1	0	0	0	0
18	NextGen Collaborative ATM (ATC-Dispatchers) [79]	1	HH	P	1	G	A	P	2	1	1	1	1	0	0	0	1	1
19	NextGen Collab ATM (ATC or Dispatchers - Pilot) [79]	0	HH	H	1	L	A	P	2	0	1	1	1	0	0	0	1	1
20	NextGen Collab ATM (Humans – Decision Tool) [79]	0	HM	H	1	G	A	P	2	0	0	0	0	0	0	0	1	1
21	Time-Based Flow Management (ATC - ATC) [80]	1	HH	H	1	L	A	P	2	1	1	1	1	1	0	0	1	1
22	Time-Based Flow Management (ATC - Decision Tool) [80]	1	HM	H	1	G	A	P	2	0	0	0	0	0	0	0	1	1
23	Traffic Collision Avoid Sys (TCAS Aircraft – TCAS A/C) [74]	1	MM	HP	1	L	A	D	2	1	1	1	1	0	1	0	1	1
24	Traffic Collision Avoid Sys (TCAS – Flight Crew) [74]	1	HM	H	1	G	A	P	2	0	0	0	0	0	0	0	0	0
25	Traffic Collision Avoid Sys (TCAS – ATC) [74]	1	MH	P	1	G	A	P	2	0	0	0	1	1	0	0	1	1
26	ACAS-X Active & Passive Coord (Peer-Peer Aircraft) [75]	0	MM	HP	1	L	A	D	2	1	1	1	1	0	1	0	1	1
27	ACAS-X Responsive Coord (Senior - Junior Aircraft) [75]	0	MM	HP	1	L	A	D	3	1	1	1	1	0	1	0	1	1
28	ACAS-X (ACAS Aircraft – Non-ACAS Aircraft) [75]	0	MM	P	12	L	P	D	3	0	0	0	1	0	0	0	1	1
29	ACAS-X (ACAS – Flight Crew, ACAS - Autopilot) [75]	0	HM	H	1	G	A	P	2	0	0	0	0	0	0	0	0	0
30	ACAS-X (ACAS – ATC) [75]	0	HM	P	1	G	A	P	2	0	0	0	1	1	0	0	1	1
31	UAS Detect And Avoid (Operator - DAA) [72]	0	HM	H	1	G	A	P	12	0	0	0	0	0	0	0	0	0
32	UAS Detect And Avoid (UAS - other Aircraft) [72]	0	SS	HP	12	L	A	D	3	1	1	1	1	0	1	0	1	1
33	UAS Detect And Avoid (ATC - Traffic) [72]	0	SS	H	1	L	AP	P	2	0	0	1	0	0	0	0	1	1
34	Patriot Friendly Fire (Aircraft - Patriot Battery) [72]	1	SS	P	1	L	AP	P	3	1	0	1	0	0	0	0	1	1
35	Patriot Friendly Fire (JFAC Air Cmd – JFLC Land Cmd) [72]	1	HH	P	1	L	AP	D	2	1	1	1	1	0	1	0	0	1
36	“Playbook” (Human - Automation) [81]	0	HM	H	1	GL	A	D	2	0	0	1	1	1	1	0	0	0
37	Common Autopilot or Auto-throttle (Pilot – Auto)	1	HM	H	1	G	AP	P	1	0	0	0	1	1	0	0	0	0
38	Full Authority Digital Engine Control (Pilot - FADEC)	1	HM	H	1	G	A	P	1	0	0	0	0	0	0	0	0	0
39	Future Multi-Rotor Flight Control System (Pilot - FCS)	0	HM	H	1	G	AP	P	1	0	0	0	0	0	0	0	0	0
40	Terrain Following Radar (Pilot - TFR)	1	HM	H	1	G	AP	P	1	0	0	0	1	1	0	0	0	0
41	Airliner Autoland (Pilot - Autoland)	1	HM	H	1	G	AP	P	1	0	0	0	1	1	0	0	0	0
42	Triple Modular Avionics on Adv Aircraft (Modules)	1	MM	P	1	G	A	P	1	0	0	0	1	0	0	0	0	0
43	Emergency Descent Mode (Pilot - EDM) [Garmin Ltd.]	1	HM	H	1	G	AP	P	2	0	0	0	1	1	0	0	0	1
44	Electronic Stability and Protection (Pilot - ESP) [Garmin Ltd.]	1	HM	H	1	G	AP	P	2	0	0	0	1	1	0	0	0	1
45	Emergency Autoland (Human – Autoland) [Garmin Ltd.]	1	HM	H	1	G	AP	P	2	0	0	0	1	1	0	0	0	1
46	Brake Control Unit (Flight Crew - BSCU) [70]	1	HM	H	1	G	AP	P	2	0	0	0	1	1	0	0	0	0
47	Common Single UAS Op (Operator - Autopilot)	1	HM	H	1	GL	AP	P	1	0	0	0	1	1	0	0	0	1
48	Common Single UAS Op (Operator – Operator Team)	1	HH	P	1	G	AP	D	2	1	1	1	1	1	1	1	1	1
49	Communications Denied UAS (Operator - Autopilot)	0	HM	H	1	LN	AN	P	1	0	0	0	1	1	0	0	0	1
50	Auto Track Cinematography UAS (Operator - UAS) [Skydio]	1	HM	H	1	L	AP	P	134	0	1	0	1	0	0	0	0	1
51	Multi-Munition (Munitions) [23]	0	MM	P	1	L	A	D	1	1	1	1	1	0	1	0	1	1

#	System (Interacting Entities)	Fielded	Structural Dimensions of Controller Interaction							Collaborative Control Dynamics								
			Type Controllers	Hierarchal Struct	Behavior Intent	Connectivity	Info Exchange	Roles & Resp	Dev Origins	Lateral Coord	Mutual Close Loop	Cognitive Align	Share Authority	Transfer Authority	Dyn Authority	Dyn Hierarchy	Dyn Membership	Dyn Connectivity
52	Multi-Munition (Operator - Munitions) [23]	0	HM	H	1	L	A	D	1	0	0	-	1	1	1	0	1	1
53	Common Distributed Multi-UAS (Operator - UAS) [82], [83]	0	HM	H	1	L	A	D	1	0	0	1	1	1	1	0	1	1
54	Common Distributed Multi-UAS (UASs) [82], [83]	0	MM	P	1	G	A	D	1	1	1	1	1	0	1	0	1	1
55	HIPC Multi-UAS Distributed Control (UASs) [58]	0	MM	P	1	L	A	D	1	1	1	1	1	0	1	0	1	1
56	USMA Service Academy Swarm Challenge (UASs) [84]	0	MM	P	1	G	A	D	1	1	1	1	1	1	1	0	1	1
57	Multi-UAS Hazard Analysis (UASs) [76]	0	MM	P	1	L	A	D	1	0.5	0.5	0.5	1	0	1	0	1	1
58	Leader-Subordinate Multi-UAS Control (UASs) [85]	0	MM	HP	1	L	A	D	1	0.5	0.5	0.5	1	1	1	0.5	1	1
59	Multi-UAS Light Shows (UASs)	1	MM	P	1	L	N	P	1	0	0	0	0	0	0	0	1	1
60	Multi-UAS Light Shows (Operator - UAS)	1	HM	H	1	L	A	P	1	0	0	0	0	0	0	0	1	1
61	UAS-UGV Teaming (UAS – UGV)	0	MM	P	1	G	A	P	1	1	0.5	1	0	0	0	0	0	1
62	Human Piloted Flight Formation	1	HH	HP	1	G	AP	P	1	1	1	1	1	0	0	0	1	1
63	Manned Unmanned Teaming (MUM-T) (UASs) [86]	0	MM	P	1	L	A	D		1	1	1	1	-	1	-	1	1
64	MUM-T (Team Lead - GCS) [86]	0	HH	P	1	GL	A	D	2	1	1	1	1	1	1	0.5	1	1
65	MUM-T (TL or GCS - UAS) [86]	0	HM	HP	1	L	A	P	1	1	1	1	1	-	1	0	1	1
66	MUM-T (Neural Net Control Sys – Runtime Assurance) [87]	0	MM	P	1	G	AP	P	2	1	1	1	1	1	0	0	0	0
67	MUM-T (Safety Pilot – Automated Formation Control) [87]	0	HM	H	1	G	AP	P	1	0	-	-	1	1	0	0	0	0
68	Tethered UAS (PIC - UAS) [88]	0	HM	HP	1	L	A	P	1	1	1	1	1	0	0	0	1	1
69	Tethered UAS (UASs) [88]	0	MM	P	1	G	A	D	1	1	1	1	1	0	1	0.5	1	1
70	Satellite Constellation (Satellites) [89]	1	MM	P	1	N	N	P	1	0	0	0	0	0	0	0	1	1
71	Satellite Constellation (Orbital - Payload Teams) [89]	1	HH	P	1	G	A	P	1	1	1	1	1	0	0	0	0	-
72	Satellite Constellation (Operator - Satellite) [89]	1	HM	H	1	L	A	P	1	0	0	0	0	0	0	0	1	1
73	Distributed Satellite Constellation (Satellites) [89]	0	MM	P	1	L	A	P	1	1	1	1	1	1	1	0	1	1
74	Distributed Formation Control Astrobee [90]	0	MM	P	1	G	AP	P	1	1	1	1	1	0	1	0	0	0
75	Astrobee ConOps(Human-Auto) [91]	1	HM	H	1	L	AP	P	1	0	0	0	0	0	0	0	0	1
76	Combined Product Transportation Sys (Operators) [92]	0	HH	P	1	G	A		1	1	1	1	1	0	-	-	0	-
77	Combined Product Transportation Sys (Operator - AGV) [92]	0	HM	H	1	L	A	P	1	0	0	0	0	0	0	0	0	0
78	Combined Product Transportation Sys (AGV - AGV) [92]	0	MM	P	1	L	P	P	1	1	0.5	1	1	0	0	0	0	1
79	Supply Vessel Dynamic Positioning (Operators) [93]	0	HH	P	1	L	A	D	2	1	1	1	1	1	1	1	1	1
80	Supply Vessel Dynamic Positioning (AGV-AGV) [93]	0	MM	P	1	L	AP	D	1	1	1	1	1	0	1	0	1	1
81	Supply Vessel Dynamic Positioning (Operator - AGV) [93]	0	HM	H	1	G	AP	P	1	0	0	0	1	1	0	0	0	0
82	Airliner Flight Crew (Pilots) [94]	1	HH	HP	1	G	AP	D	2	1	1	1	1	1	1	1	1	0
83	Integrated Modular Avionics (Flap - Thrust) [95]	0	MM	P	1	G	A	P	2	1	0	1	0	0	0	0	0	0
84	In-Trail Procedure (ITP) (Aircraft-Aircraft) [96]	1	SS	P	2	L	A	P	3	0	0	0	0	0	0	0	1	1
85	In-Trail Procedure (ITP) (ATC Controllers) [96]	1	HH	P	1	L	AP	D	2	1	1	1	1	1	1	0	0	1
86	NASA & JAXA Coordination (2 Organizations) [70]	1	HH	P	1	L	A	P	2	1	0	1	0	0	0	0	0	1
87	1994 Friendly Fire (AWACS controllers) [35]	1	HH	P	1	L	AP	D	1	1	1	1	1	1	1	0	0	1
88	Embedded Vehicle Software Subsys (Subsystems) [97]	1	MM	P	1	G	AP	P	23	0	0	0	1	0	0	0	0	0
89	Digital Copilot [Collaborative Checklist] (Pilot – Auto) [20]	0	HM	HP	1	G	AP	D	1	1	1	1	1	1	1	1	0	0
90	Digital Copilot [Cognitive Assistance] (Pilot – Auto) [98]	0	HM	H	1	G	A	P	2	0	0	1	0	0	0	0	0	0
91	Pilot Assistant [NATO] [30]	0	HM	H	1	G	A	D	12	0	0	1	0	0	0	0.5	0	0
92	Reduced Crew Ops Test (Pilot - Remote Pilot) [11]	0	HH	P	1	G	AP	D	2	1	1	1	0.5	0.5	0.5	-	1	1
93	Reduced Crew Ops Test (Human - Autonomy) [11]	0	HM	H	1	G	A	P	2	0	0	1	1	1	0	0	0	0
94	Co-Active Design Model (TBD - TBD) [5]	0	any	P	1	GLN	AP	PD	1	1	1	1	1	1	0.5	0	0	0.5
95	UAM Simplified Vehicle Ops (Human - Autonomy) [6], [8]	0	HM	HP	1	G	AP	D	1	1	0.5	1	1	1	1	1	0	0
96	UAM Remote Supervisory Ops (Humans) [6]	0	HH	P	1		A	D	2	1	-	1	1	1	1	-	1	1
97	UAM Remote Supervisory Ops (Human - Auto) [6]	0	HM	H	1	G	A	D	1	0	0	1	0	0	0	0	1	1
98	UAM Remote Supervisory Ops (Machines) [6]	0	MM	P	2	N	N	P	2	0	0	0	0	0	0	0	1	0
99	UAM Distributed ATM Concept (ATM-ATM) [99]	0	MM	P	1	L	A	D	1	1	1	1	1	1	1	0	1	1
100	UAM Distributed ATM Concept (ATM-UAM) [99]	0	MM	H	1	L	AP	D	1	0	0	1	0	0	0	0	1	1
101	Mosaic of Warfare (Overall Concept) [27]	0	HM	HP	1	L	A	D	1-4	1	1	1	1	1	1	1	1	1