Urban Air Mobility Conflict Resolution: Centralized or Decentralized?

Min Xue*
NASA Ames Research Center, Moffett Field, CA 94035

This work begins to address one of the critical questions in the urban air mobility and small unmanned aircraft communities: Should the en-route conflict resolution function in an urban air mobility traffic system be centralized or decentralized? Three conflict resolution architectures are modeled and analyzed: centralized, decentralized with uniform rules, and decentralized with mixed rules. This study compares these architectures and investigates their robustness to communication and state information errors in terms of safety and efficiency metrics. Experiments are conducted using a high-fidelity Monte Carlo traffic simulator and a generic set of traffic scenarios with increasing traffic density. When no errors were modeled, the centralized architecture marginally outperformed the decentralized architecture. However, performance of the centralized architecture was found to be adversely affected by the modeled input errors to a greater degree than was the decentralized architecture. Performance of the centralized architecture also was degraded significantly by the modeled transmission errors of the centralized resolution maneuvers. In the decentralized architecture, uniform rules outperformed mixed rules because, in the mixed rules case, system safety performance was undermined and dominated by the poor performers.

I. Introduction

As the ground transportation becomes increasingly congested, the Urban Air Mobility (UAM) concept [1, 2] has attracted a great deal of attention for its potential to significantly improve the movement of people and goods. Besides the challenges of building and certifying safe and quiet electric powered vertical takeoff and landing aircraft (eVTOL), developing a safe, efficient, and scalable traffic management system faces great challenges as well. The envisioned high density of traffic operations of diverse aircraft types in complex urban environments mandates a paradigm shift from the conventional human-centric air traffic management. Similarities between UAM and small unmanned aerial vehicle (UAV) operations have led to efforts to adapt the federated autonomous Unmanned aircraft system Traffic Management (UTM) system [3–6] developed by NASA, FAA, and industry partners to become a foundation for the UAM traffic system. However, there are still many crucial questions needed to be addressed to enable UAM operations, such as how to interact with conventional manned aviation, if/when/where route structures should be placed, and should core functions like conflict resolution be centralized or decentralized?

As the core function in airspace management system, the architecture of conflict resolution is more critical for the early stage developement of other systems: for instance, if a decentralized architecture is favored, then the conflict resolution function needs to reside in the UAM vehicle system and a close-range vehicle-to-vehicle(V2V) communication capability becomes paramount. Whereas if a centralized architecture is decided, a reliable and high-bandwidth air-ground communcation becomes more important than the V2V communcation with easier requirements on the vehicle system.

To support the development of the UAM concept, this work analyzes and evaluates various architecture options for conflict resolution strategy and service, which is one of the core capabilities to enable UAM operations. To help understand the analysis in this study, the conflict resolution architectures are categorized into four types. The first one is the *Centralized* architecture, which refers to a conflict resolution architecture where a single centralized and ground-based conflict resolution service sends trajectory amendments to aerial vehicles to resolve projected conflicts with other aircraft. This is analogous to the role of human air traffic controllers in current air traffic management. The second is called *Uniform rule-based* architecture, which refers to a decentralized conflict resolution architecture where each individual aerial vehicle follows a uniform set of predefined rules or protocols when resolving conflicts. "Autonomous Flight Rules" [7], protocol-based conflict resolution [8], distributed conflict resolution with coordinated strategy [9], and agent-based cooperation [10] are all examples of this category. The third group is the *Coordination-based* architecture, which is similar to the *Uniform rule-based* strategy except that it allows aircraft to coordinate with each other via a predefined, real-time negotiation mechanism. The difference from the *Uniform rule based* system is that, for a given

*Aerospace Research Engineer, Aviation Systems Division. Mail Stop 210-15. AIAA senior member.

conflict situation, resolutions from the *Coordination-based* system may differ due to the dynamic nature of real-time coordination, whereas the resolutions from the *Uniform rule-based* system are always the same. The collaborative trajectory options program (CTOP) [11] may be the closest example in the current manned aviation system. The fourth group is called the *Mixed rule-based* architecture, representing a completely decentralized mechanism with separated objectives or rules for different sets of aircraft. Aircraft-based conflict resolutions in which each aircraft has its own resolution objective [12] is one extreme example. The last three categories, *Uniform rule-based*, *Coordination-based*, and *Mixed rule-based*, are all decentralized; they differ in their levels of decentralization.

There have been many studies in comparing centralized and decentralized systems, although most are from outside the aviation domain. For instance, based on mathematical models of incentives, bias, coordination, and communication, Alonso et. al [13] showed that in multidivisional organizations decentralization can outperform centralization even when coordination is extremely important. Using simulations based on a simple sorting problem, Veetil [14] showed that a decentralized system was more robust than a centralized system in the presence of communication errors, whereas the centralized system required less coordination. Although there are many studies proposing different architectures for conflict resolution [7–10, 12], only a few studies quantitatively compared centralized and decentralized systems in the air traffic management domain. Bicchi et. al. [15] showed that a decentralized conflict resolution is more robust to conflict resolution failures from the perspective of trajectory optimization. Conversely, Krozel et. al. [16] showed that centralized separation outperforms decentralized separation in terms of system stability and efficiency metrics proposed by the authors based on the assumption of perfect information exchange and the presence of no errors or delays.

This study makes no such assumption, and further extends the study of conflict resolution architectures to small UAM vehicle types. The contribution of this paper is to quantify safety and efficiency performance differences between centralized and decentralized architectures for conflict resolution, and to assess their robustness to input and output errors. To gain better understanding of the differences among these architectures through quantitative analysis, this work investigates three of the above four architectures: *Centralized*, *Uniform rule-based*, and *Mixed rule-based*. The high fidelity Fe^3 [17] simulator, which has the capability of modeling various sources of uncertainty is used to run the experiments.

Section II introduces the models and parameters for these three architectures. Section III presents the experiments setup. Section IV analyzes and compares three conflict resolution architectures in terms of system safety, efficiency, robustness, and scalability. Section V presents the findings of this study.

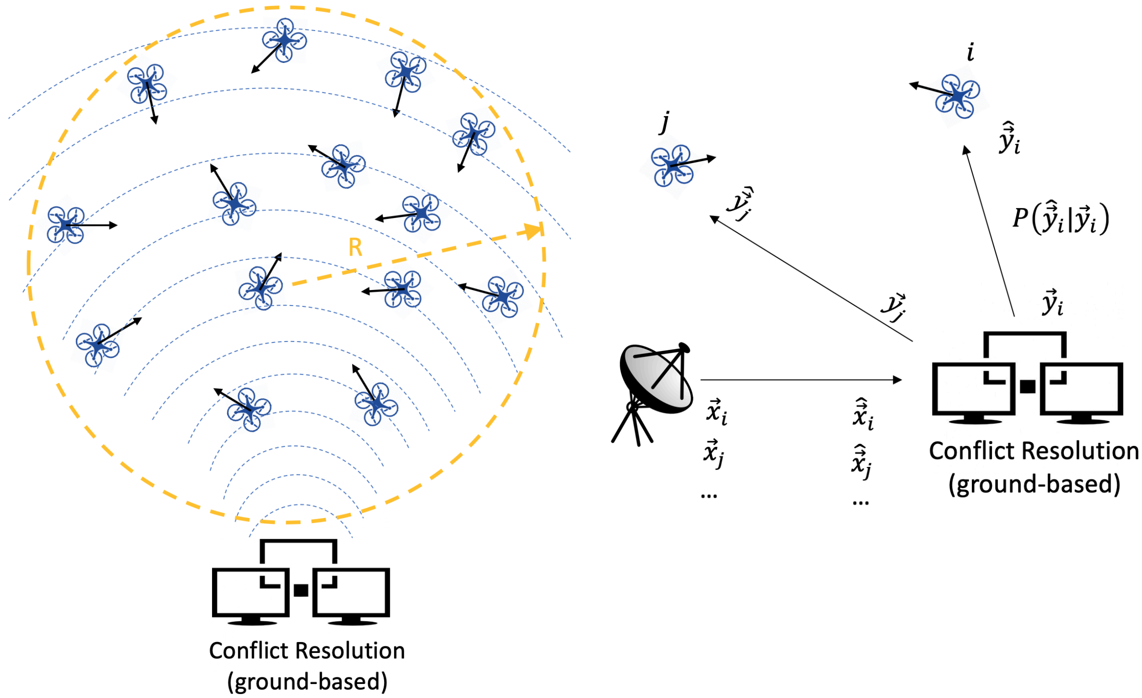
II. Conflict Resolution Architecture Modeling

In general, conflict resolution includes four phases: *Strategic deconfliction* assigns delays to resolve potential conflicts typically before departure; *Separation assurance* modifies trajectories after departure to several minutes before potential losses of separation could happen; *Detect and avoid* initiates tactical maneuvers several minutes before the loss of separation or loss of well-clear; and *Collision avoidance* activates final maneuvers to avoid near-mid-air-collision(NMAC) once the loss of well-clear/loss of separation happens. Conflict resolution in this study refers to all three airborne phases including *Separation assurance*, *Detect and avoid*, and *Collision avoidance*. This section will describe the models for the three architectures introduced previously: *Centralized*, *Uniform rule-based*, and *Mixed rule-based*.

A. Centralized Architecture

Centralized architecture is widely used in the national airspace system (NAS) operations, especially at the sector level, where typically one (a radar-side controller) or two (a radar-side controller and a data-side controllers) ground-based human controllers provide conflict resolution services [18]. Fig. 1(a) shows a notional picture of a centralized UAM conflict resolution architecture, where the ground-based conflict resolution service collects vehicle information, resolves potential conflicts, and sends trajectory amendments to each individual UAM vehicle. Due to the scalability limitations associated with human-centric conflict resolution, it is envisioned that human controllers may not be responsible for conflict resolution in UAM operations, as in the UTM concept [3, 4].

Fig. 1(b) shows the information flows in such a centralized architecture. In this figure, the aircraft state information $\vec{x}$ is provided by either surveillance systems or operator (through air-to-ground or ground-to-ground communications) and fed into the centralized conflict resolution service. There will be some errors for the information fed into the conflict resolution service because of imperfect detection, navigation, and communication. Although many tracking and filtering algorithms [19–21] have been developed to minimize these errors, they cannot eliminate errors. If the final aircraft state information (such as aircraft position, velocity, intent, etc.) received by the conflict resolution is referred to as $\hat{\vec{x}}$ representing the approximated aircraft states, then the difference between $\vec{x}$ and $\hat{\vec{x}}$ is represented as $\tilde{\vec{x}}$ (shown in Eqn. 1



(a) Centralized System

(b) Data Flow in the Centralized System

Fig. 1 Notional Graph and Data Flow for the Centralized System

for aircraft i at time t).

$$\hat{\vec{x}}_i(t) = \vec{x}_i(t) + \tilde{\vec{x}}_i(t) \tag{1}$$

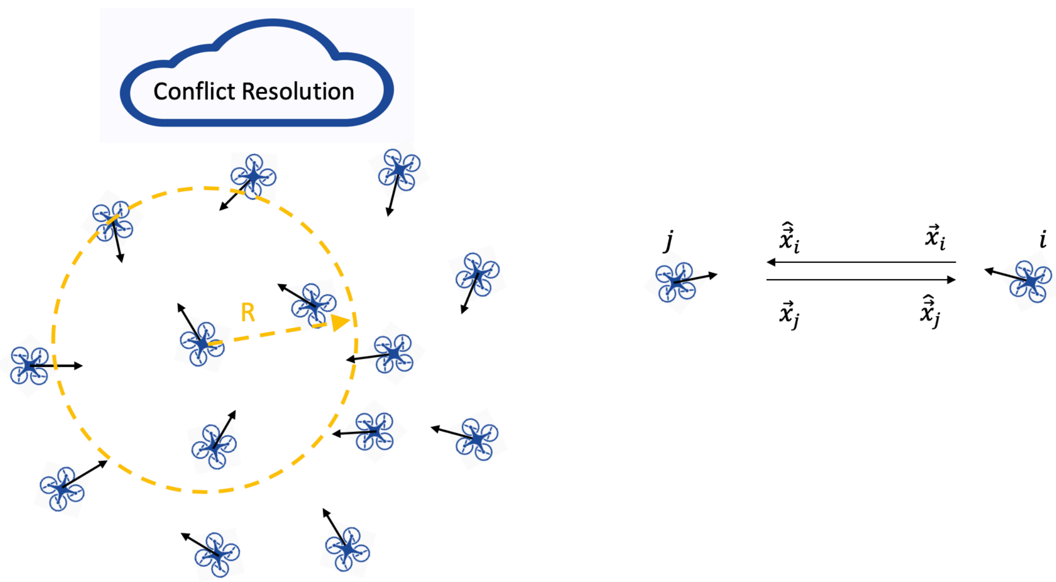
After computing, the centralized service needs to send resolutions, such as waypoints, updated trajectory, heading changes, and speed changes, to remote individual aircraft or operators, typically from a ground-based station. Again due to communication errors, there will be a difference between the true resolutions $\vec{y}$ transmitted by the centralized service and those received by aircraft or operators (represented by $\hat{\vec{y}}$). The communication may also be degraded by packet loss. Let the reception probability $P\{\hat{\vec{y}}(t)|\vec{y}(t)\}$ represents the probability of receiving $\hat{\vec{y}}$ when a resolution $\vec{y}$ is transmitted by the centralized service at time t. Eqn. 2 shows the final conflict resolution instruction $\hat{\vec{y}}_i$ at time t.

$$\hat{\vec{y}}_i(t) = [\vec{y}_i(t) + \tilde{\vec{y}}_i(t)] \cdot P\{\hat{\vec{y}}_i(t)|\vec{y}_i(t)\} \tag{2}$$

To capture the difference in the objectives between centralization and decentralization, it is assumed in this study that the objectives are the same for both centralized and decentralized systems, except that in the centralized service it first prioritizes conflicts based on their estimated time until conflicts and then resolves these conflicts in sequence according to their priorities. For instance, if there are two conflicts, the conflict is projected to occur first will be resolved first before the conflict that is projected to occur second. The coverage for both centralized and decentralized architectures is assumed to be the same long range, which is not the focus of this work.

B. Decentralized with uniform rules

A decentralized system is usually envisioned to provide flexibility, data privacy, and robustness, where each division has the flexibility to optimize its objective function based on its own needs while only sharing necessary data across different divisions. Fig. 2(a) presents a notional picture of a decentralized system with uniform rules and R is the detection/communication range of a given aircraft. Each individual aircraft is required to follow the uniform rules/protocols and determine conflict resolution maneuvers by itself based on predefined rules/protocols when a loss of separation is about to happen. With decentralization, the input flow is the only data flow, either coming from vehicle-to-vehicle communications or onboard sensors as shown in Fig. 2(b). Similar to the input flow in centralization, because of errors in communication, sensor, and navigation, there will exist the error $\tilde{\vec{x}}$ between the approximated states $\hat{\vec{x}}$ and the true states $\vec{x}$. As the aircraft resolve conflicts onboard and all the aircraft already have the pre-defined resolution strategies, there is no need to transmit resolutions when using decentralized strategies.



(a) Decentralized with Uniform Rules

(b) Data Flow in the Decentralized System

Fig. 2 Notional Graph and Data Flow for the Decentralized System with Uniform Rules

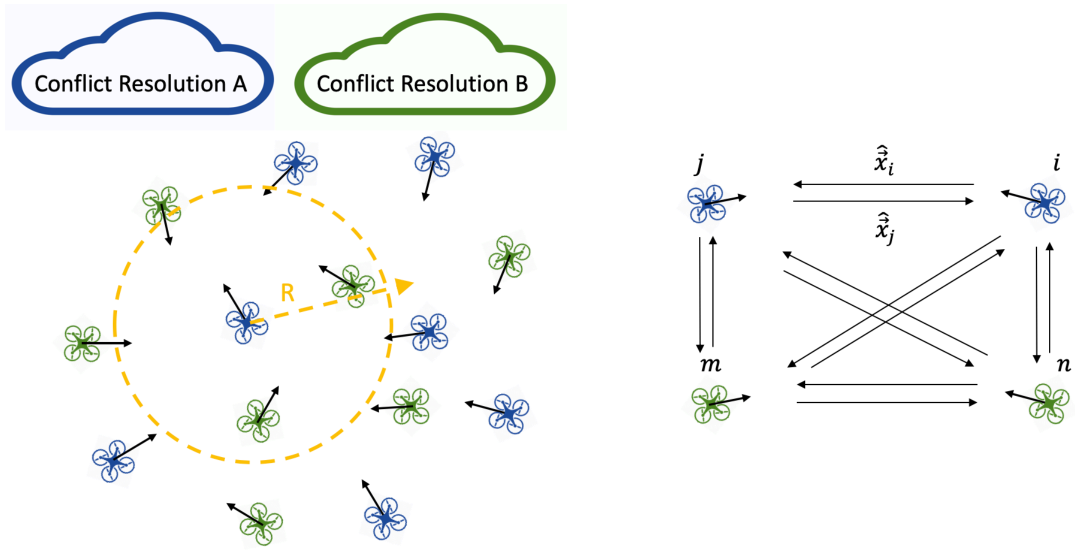
C. Decentralized with mixed rules

Another decentralized architecture allows multiple different sets of conflict resolution rules operate in one area. Fig. 3(a) shows a notional picture, where the blue aircraft follow conflict resolution rule set A and the green aircraft follow conflict resolution rule set B. An analogy in the UTM concept would be multiple service providers (A, B, ...) providing conflict resolution services to different groups of operators. Although the aircraft abide by one of the two different sets of rules, they are still assumed to share their information with all proximate aircraft regardless of the group to which they belong (as shown in Fig. 3(b)). The data flow and objectives of conflict resolutions are the same as in the decentralized system with uniform rules.

D. Conflict resolution algorithms

There is a large body of research on en-route conflict resolution algorithms [22]. Representative algorithms include the horizontal vector turn method [23, 24], the potential field method [25], the rule-based methods [26, 27], and the partially observable Markov decision process method [28], and these algorithms cover various time horizons. The scope of this work is to study conflict resolution architectures instead of designing or evaluating a specific conflict resolution algorithm. Therefore, without loss of generality, rule-based algorithms are applied here. In addition to basic conflict resolution parameters, like look-ahead time, conflict detection threshold, etc., a rule-based algorithm also provides a set of flight rules to define the maneuver options for the responsible party under given conditions. If there exist multiple options, a simple objective (e.g., minimum deviation) may be used to decide the final resolution maneuvers.

Table 1 shows an example of conflict resolution rules defined in a rule-based algorithm. These rules define the responsibility and specify a range of resolution maneuvers under various conditions. With these rules, a final resolution maneuver is then identified with minimum deviation. In this table, θ_{RPA} represents the relative position orientation



(a) Decentralized with Mixed Rules

(b) Data Flow in the Decentralized System

Fig. 3 Notional Graph and Data Flow for the Decentralized System with Mixed Rules

Table 1 Sample rules in a rule-based conflict resolution algorithm

Rule	Conditions	Action
1	$\theta_{RPA} \in [-10.0°, 10.0°] \wedge (\theta_{RHA} \in (160.0°, 180.0°] \vee \theta_{RHA}(-180.0°, -160.0°))$	$\phi \in [0°, 180.0°]$
2	$\theta_{RPA} \in (-90.0°, 0.0°) \wedge \theta_{RHA} \in (-180.0°, 0.0°)$	$\phi \in [0.0°, 90.0°]$
3	$\theta_{RPA} \in (-180.0°, -90.0°) \wedge \theta_{RHA} \in (-180.0°, 0.0°)$	$\phi \in [-90.0°, 0.0°]$
4	$\theta_{RPA} \in (90.0°, 180.0°] \wedge \theta_{RHA} \in [0.0°, 180.0°]$	$\phi \in [0°, 90.0°]$
5	$\theta_{RPA} \in (0.0°, 90.0°] \wedge \theta_{RHA}[0.0°, 180.0°]$	$\phi \in [-90°, 0.0°]$

of the intruder with respect to the ownship, θ_{RHA} refers to the relative heading angles between two aircraft with the ownship heading as the reference, and ϕ denotes the action options, which are the allowed heading angle changes of the aircraft. In this paper, two different conflict resolution algorithms - each with their own sets of rules - were implemented as decentralized conflict resolution algorithms. The centralized algorithm was similar to the decentralized algorithm, except that the centralized algorithm prioritized the conflicts first and resolved conflicts sequentially.

III. Experiment Set-up

To conduct generalized comparisons, random and generic scenarios with increasing number of operations were generated. Since UAM vehicles are mostly still under development, vehicle models and separation minimums are not well-defined. In this study, the target ground speeds of flights ranged from 15 to 30 knots and minimum separation was assumed to be 50 feet. The airspace in test scenarios was a simple two-dimensional region (1.5 nmi by 1.5 nmi) designed to be traversed in approximately three minutes of flight time. Although the flight speeds in the experiments were lower than expected for UAM vehicles (e.g., 100 knots) and the minimum separation might be small, the general trends and principles found through the experiments should be able to hold. In the scenarios, flights were required to go through the predefined region with origin and destination outside of the region. To increase the operational density and complexity, all flights in each scenario were set to depart within a three-minute window. Fig. 4 shows a sample scenario with 50 vehicles, where circle and cross markers represent origins and destinations, respectively.

An early study [29] showed that the intrinsic complexity of operations for a given scenario doesn't only depend on

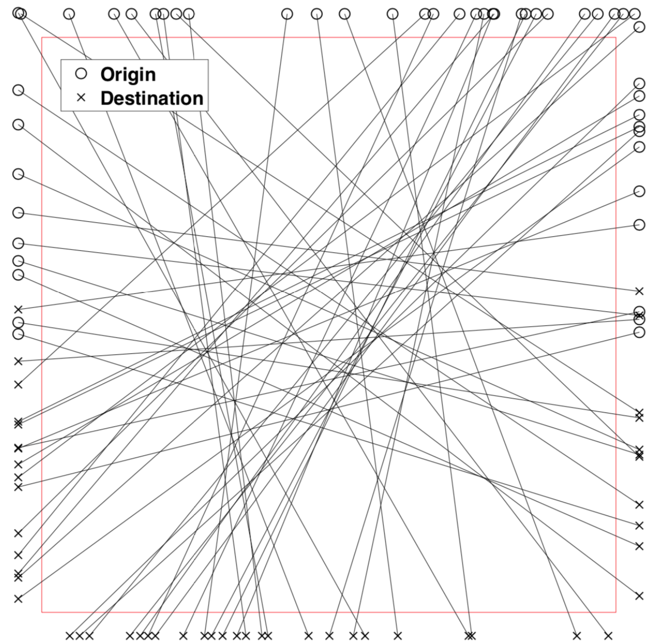


Fig. 4 A Sample Scenario with 50 vehicles

the number of aircraft. For instance, a scenario with 50 aircraft may not pose any losses of separation without a conflict resolution service, whereas a scenario with 5 aircraft may pose multiple losses of separation if there is no conflict resolution service. However, statistically, a large number of randomly generated scenarios with 50 aircraft should pose more potential conflicts–and hence be more complex for the conflict resolution function–than randomly generated scenarios with 5 aircraft. Therefore, in this study, more than 200 scenarios were randomly generated for each "number of aircraft" test point to ensure that, statistically, scenarios with more aircraft were more complex than scenarios with fewer aircraft. In the experiments, over 3,800 scenarios were created for 19 different levels of density with increased number of aircraft from 3 to 70 aircraft. In the experiments, besides the aforementioned three architectures, the input error and the reception probability of the output(i.e., conflict resolution maneuvers) are also used as independent variables. Table 2 shows the test metrics used in the experiments. Two metrics, loss of separation and extra flight distance caused by resolution maneuvers, are used to quantify safety and efficiency, respectively.

Table 2 Test metrics

Independent variables	Values
Number of aircraft	3, 5, 7, 9, 11, 13, 16, 19, 22, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70
Number of scenarios	200 scenarios randomly generated at each density level
Architectures	centralized, decentralized (uniform rule based), decentralized (mixed rule based)
Output reception probability	100%, 99.9%, 99.5%, 98%, 95%
Input state error (ft)	0, 0.5

IV. Results

This section will analyze performance differences among three architectures based on simulation results. The tolerance of the centralized architecture to input error and output reception probability will also be examined.

A. Baseline case: Centralized vs. decentralized in the absence of input or output errors

In the absence of communication and information errors, the difference between centralization and decentralization is in their objectives. Centralized conflict resolution cross-checks and tries to deconflict resolution maneuvers with all other proximate aircraft before transmitting the maneuvers to individual aircraft. Conversely, with the decentralized architecture (when there is no real-time negotiation), the conflict resolution function onboard the aircraft relies on the predefined rules and computes and executes its own conflict resolution maneuvers without knowing others' conflict resolution maneuvers.

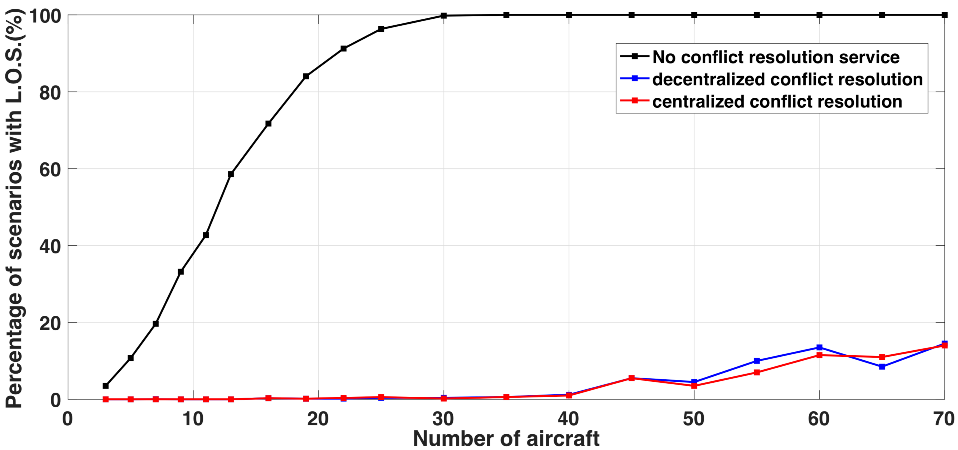


Fig. 5 Safety comparison between centralized and decentralized

Fig. 5 presents the safety metric comparison between centralized and decentralized conflict resolution in terms of the percentage of scenarios with one or more losses of separation. The black curve indicates the percentage of scenarios with losses of separation when conflict resolution services were disabled. As the number of aircraft was increased, the likelihood that a scenario would have a loss of separation also increased. The blue curve represents the percentage of scenarios with one or more losses of separation when conflict resolution services were decentralized with predefined uniform rules, whereas the red curve corresponds to the percentage of such scenarios when conflict resolution services were centralized. It is noted that both architectures were able to resolve 100% of the potential conflicts for the large majority of scenarios. At the test points with the higher aircraft counts, the centralized architecture performed slightly better than the decentralized architecture, albeit only marginal. The centralized conflict resolution produced a total of 3.2% of scenarios with losses of separation (out of 3,800 scenarios), whereas the decentralized architecture produced a total of 3.4% of scenarios with losses of separation.

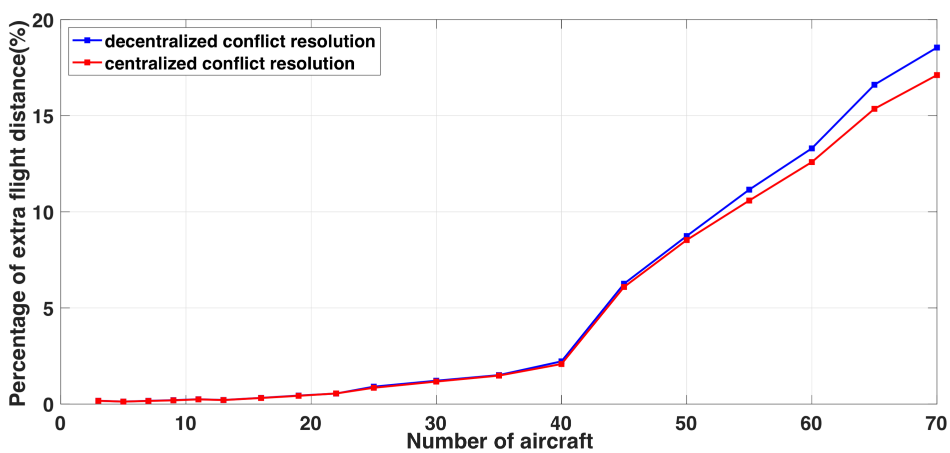


Fig. 6 Efficiency comparison between centralized and decentralized

Fig. 6 presents the efficiency comparison between centralized and decentralized conflict resolution in terms of the percentage of extra flight distance. The blue and red curves represent the decentralized and centralized architectures, respectively. It is not surprising that the centralized outperforms the decentralized with 1.4% improvement at the level of 70 aircraft. In both the safety and efficiency comparisons, the performance advantage of the centralized architecture mostly happened when the number of aircraft exceeded 40 or 50, when the chance of multiple simultaneous conflicts was high. When the number of aircraft was low, the likelihood of multiple conflicts in close proximity was also low and there was no difference between solving conflicts sequentially versus in parallel.

B. Uniform rules vs. mixed rules in decentralized architectures

As introduced before, another decentralized architecture referred to as the decentralized with mixed rules (shown in Fig. 3(a)) allows mixed conflict resolution rules in the same airspace. To simulate the operations with mixed rules in the same airspace, two different algorithms (named A and B for convenience) with different sets of rules and parameters were applied. In every scenario, 50% of the aircraft were randomly chosen to use algorithm A, and the other 50% of aircraft used algorithm B.

Fig. 7 compares safety metrics among three different cases. The blue curve shows the percentage of scenarios with one or more losses of separation when only the algorithm A (or rule set A) was used for conflict resolution. Algorithm A was used to represent the decentralized architecture in the previous results section. The magenta curve shows the percentage of scenarios with one or more losses of separation when only algorithm B was used for confliction resolution. It is noted that algorithm A clearly outperformed algorithm B. However, when both services were deployed in the same area, with half of the aircraft using A and the other half using B, the resulting percentage of scenarios with one or more losses of separation (shown as the black curve) was almost the same as when only algorithm B was deployed. The safety performance was undermined and dominated by the poorer performer, which was algorithm B in this experiment. This result shows that a uniform set of rules is necessary for operations in the same area, otherwise, the performance might be negatively impacted and dominated by the poorer performer.

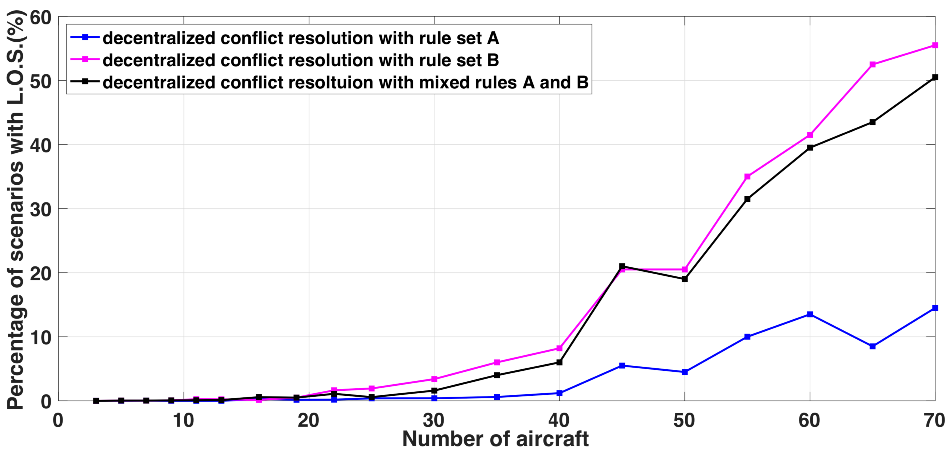


Fig. 7 Comparison between uniform rules and mixed rules in decentralized architectures

C. Sensitivity to input errors ($\tilde{\vec{x}}$)

Input errors (e.g. aircraft state error) exist in both centralized and decentralized architectures. Though the ground-based centralized service has the advantage of better ground facilities (e.g., advanced radar and powerful computing facility), the information error is still inevitable. Fig. 8 shows the change in the performance advantage of centralized over decentralized when input errors were modeled in both architectures. The blue bars represent the improvement of centralized over decentralized in terms of the percentage of scenarios with losses of separation when there was no input error. When input errors were modeled with a standard deviation of 0.5 feet, the performance advantage quickly reduced from 2% to less than 0.3 % as shown in the brown bars. The possible explanation would be: The centralized service depends on the information collected for all aircraft and makes decisions in sequence to ensure final maneuvers are coordinated, hence, if the aircraft that were involved in the conflicts in the front of the conflict list have imperfect state information, ripple effects will likely pass on to the rest of the conflict list. Since the centralized algorithm will solve the conflicts in the front of the conflict list first, the incorrect resolution maneuvers might affect the following conflicts should there be correlation between these conflicts.

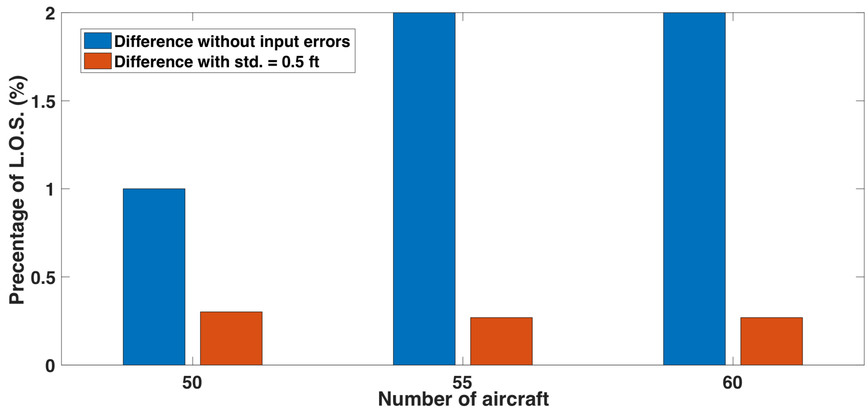


Fig. 8 Sensitivity of performance gain to the aircraft state errors

D. Tolerance to output reception error ($P\{\hat{\vec{y}}(t)|\vec{y}(t)\}$)

Once the centralized service finishes computing resolution maneuvers, it needs to transmit those resolutions to the aircraft through communication links that may experience packet loss or imperfect reception, where the reception probability of these resolutions can be represented as $P\{\hat{\vec{y}}(t)|\vec{y}(t)\}$ as modeled in the previous section. This section investigates the impact of various reception probabilities on the system performance, which is measured by the percentage of scenarios with losses of separation.

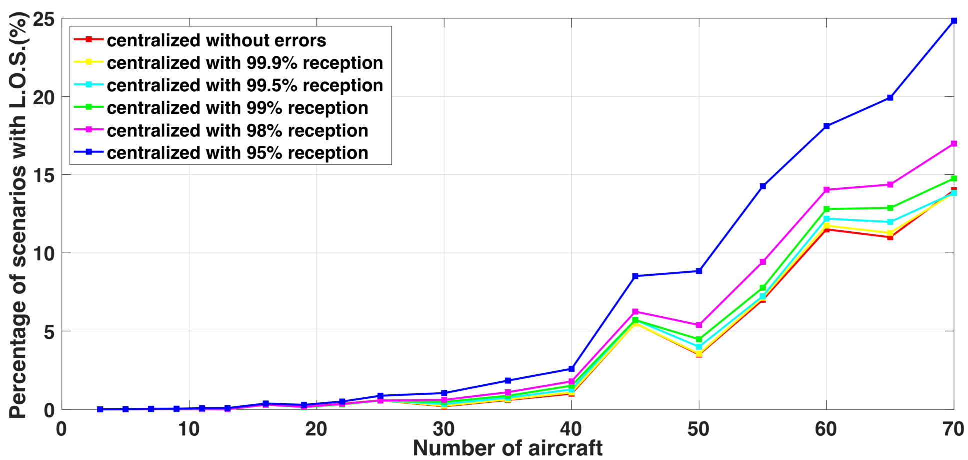


Fig. 9 Centralized architectures with different resolution reception probability

Fig. 9 presents the safety performance metric with $P\{\hat{\vec{y}}(t)|\vec{y}(t)\} = \{99.9\%, 99.5\%, 99\%, 98\%, 95\%\}$. It shows that when the reception probability was decreased, the percentage of scenarios with losses of separations increases. The increase varied from 0% to 2% when the number of aircraft was lower than 40. Above 50 aircraft, a 2% to 5% degradation in reception probability resulted in 2% to 11% more scenarios having losses of separation. At 70 aircraft, when the reception probability drops from 100% to 95%, the percentage of scenarios with losses of separation went from 14% to 25%, a 78% increase.

Converted directly from Fig. 9, Fig. 10 depicts the cumulative percentage of scenarios with losses of separations starting from the lowest number of aircraft to the highest number of aircraft. It can be seen that when errors were not modeled, the centralized architecture (red curve) performed better than the decentralized (shown as a black dashed line). The overall percentages for them are 3.2% and 3.4%, respectively, as discussed in the previous section. As the

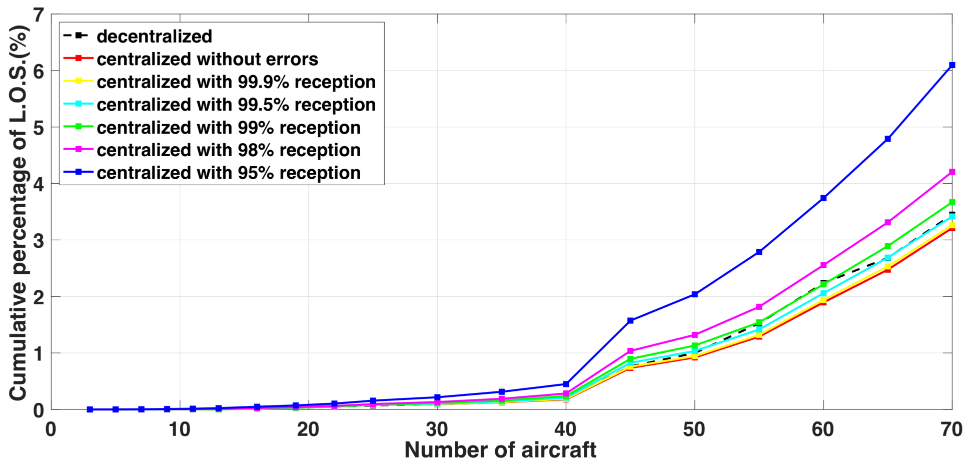


Fig. 10 Cumulative percentage of scenarios with losses of separation

reception probability was reduced, the curve starts to bend upwards and the first crossover point at which the cumulative percentage of scenarios with losses of separation for centralized was worse (higher) than the decentralized occurred with lower and lower aircraft count (i.e., less and less complex scenarios). For instance, with 99.5% of reception probability, centralized became worse than decentralized when the number of aircraft was 22. That point occurred at 20 aircraft when the reception probability was down to 99%.

Collectively, these points form a performance frontier between centralized and decentralized architectures as shown in the Fig. 11. The bottom right side of the frontier represents the area where the decentralized conflict resolution outperformed centralized. Since in the gray area when the number of aircraft is less than 18, the centralized and the decentralized performed the same, that leaves the top and right side of the boundary the only region where the centralized performed better. It is noted that when the reception probability of transmitted resolutions was less than 99.5%, the safety performance of the centralized architecture deteriorated quickly. This suggests that the centralized system may not be suitable for high volume traffic operations when the reception probability is not near perfect.

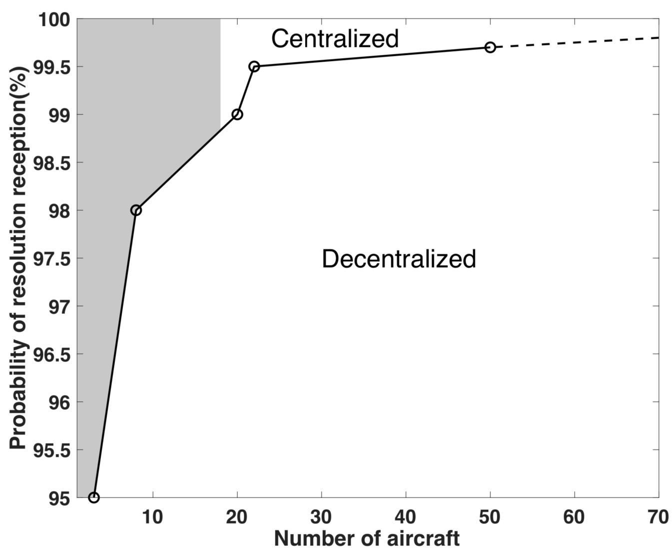


Fig. 11 Performance boundary between centralized and decentralized architectures

Although the experiments in this section focused on the normal situations, the results could apply to the abnormal conditions such as the presence of cyber-attacks and the degradation would depend on the severity of the attack. These results suggest that the performance of the centralized architecture would be more vulnerable than decentralized architectures by virtue of the fact that decentralized architectures do not need to communicate resolution maneuvers, because they are computed and executed onboard. This vulnerability must be considered when designing the centralized architecture and architecture for conflict resolution.

V. Conclusions

While the UAM concept has caught great attention due to its potential to significantly improve the way of moving goods and people, there are still many critical questions needed to be addressed to enable UAM concept development and operations. This work studies one of them: Should the conflict resolution in a UAM traffic system be centralized or decentralized? There are only a few works in comparing centralization and decentralization for air traffic management system, an in-depth study is needed to address not only the impact from the objective perspective but also the sensitivity to information and communication errors. This work modeled and analyzed three conflict resolution architectures: centralized, decentralized with uniform rules, and decentralized with mixed rules. The communication and information errors were also modeled. A high-fidelity Monte Carlo traffic simulator was used to simulate sUAS aircraft in a generic, two-dimensional airspace with increasing traffic complexity.

The results showed that when information and communication were perfect, the centralized conflict resolution slightly outperformed decentralized. However, the centralized architecture was found to be sensitive to the modeled input errors, and the results of this study suggest that the marginal performance advantage relative to the decentralized architecture can quickly diminish. More important, the safety performance of the centralized conflict resolution architecture compromised when the output reception error was increased. Additionally, a comparison between decentralized conflict resolution architectures found that, with mixed rules, the system performance was undermined and dominated by the poorer performer. To examine if the findings in this study will still hold in general, future work will explore vehicles with higher speeds with different separation minimums. And advanced centralized and decentralized aglorithms will also be incorporated.

VI. Acknowledgement

The author would like to thank Dr. Husni Indris (NASA) for the discussion of airspace operations under various architectures and Todd Farley (NASA) for valuable feedback during the review process.

References

[1] Thipphavong, D., Apaza, R., Barmore, B., Battiste, V., Burian, B., Dao, Q., Feary, M., Go, S., Goodrich, K., Homola, J., Idris, H., Kopardekar, P., Lachter, J., Neogi, N., Ng, H., Oseguera-Lohr, R., Patterson, M., and Verma, S., "Urban Air Mobility Airspace Integration Concepts and Considerations," *AIAA Aviation 2018*, Atlanta, Georgia, 2018.

[2] Mueller, P., E. Kopardekar, and Goodrich, K., "Enabling Airspace Integration for High-Density On-Demand Mobility Operations," *AIAA Aviation 2017*, Denver, Colorado, 2017.

[3] Kopardekar, P., Rios, J., Prevot, T., Johnson, M., Jung, J., and Robinson, J. E., "Unmanned Aircraft System Traffic Management (UTM) Concept of Operations," *16th AIAA Aviation Technology, Integration, and Operations Conference*, Washington, D.C., 2016.

[4] *Unmanned Aircraft System (UAS) Traffic Management (UTM)*, FAA, 2018. URL `https://utm.arc.nasa.gov/docs/2018-UTM-ConOps-v1.0.pdf`.

[5] Rios, J., Smith, I. S., Venkatesan, P., Smith, D. R., Baskaran, V., Jurcak, S. M., Strauss, R., Iyer, S. K., and Verma, P., "UTM UAS Service Supplier Development: Sprint 1 Toward Technical Capability Level 4," Tech. Rep. NASA/TM-2018-220024, NASA, 2018.

[6] Rios, J., Smith, I. S., Venkatesan, P., Smith, D. R., Baskaran, V., Jurcak, S. M., Iyer, S. K., and Verma, P., "UTM UAS Service Supplier Development: Sprint 2 Toward Technical Capability Level 4," Tech. Rep. NASA/TM-2018-220050, NASA, 2018.

[7] Wing, D. J., and Cotton, W. B., "For Spacious Skies: Self-separation with Autonomous Flight Rules in US Domestic Airspace," *11th AIAA Aviation Technology, Integration, and Operations Conference*, Virginia Beach, VA, 2011.

[8] Hwang, I., and Tomlin, C., "Protocol-based Conflict Resolution for Finite Information Horizon," *Proceedings of the American Control Conference*, Anchorage, AK, 2002.

[9] Dowek, G., Munoz, C. A., and Carreno, V. A., "Provably Safe Coordinated Strategy for Distributed Conflict Resolution," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, San Francisco, CA, 2005.

[10] Sislak, D., Volf, P., and Pechoucek, M., "Agent-based Cooperative Decentralized Airplane Collision Avoidance," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 12, No. 1, 2011, pp. 36–46.

[11] Yoo, H., Brasil, C. L., Buckley, N., Hodell, G. S., Kalush, S. N., Lee, P. U., and Smith, N. M., "Impact of Different Trajectory Option Set Participation Levels within an Air Traffic Management Collaborative Trajectory Option Program," *AIAA Aviation Technology, Integration, and Operations Conference*, Atlanta, GA, 2018.

[12] Menon, P. K., Sweriduk, G. D., and Sridhar, B., "Optimal Strategies for Free-Flight Air Traffic Conflict Resolution," *Journal of Guidance, Control, and Dynamics*, Vol. 22, No. 2, 1999, pp. 202–211.

[13] Alonso, R., Dessein, W., and Matouschek, N., "When Does Coordination Require Centralization," *American Economic Review*, Vol. 98, No. 1, 2008, pp. 145–179.

[14] Veetil, V. P., "Coordination in Centralized and Decentralized Systems," *International Journal of Microsimulation*, Vol. 10, No. 2, 2017, pp. 86–102.

[15] Bicchi, A., and Pallottino, "On Optimal Cooperative Conflict Resolution for Air Traffic Management Systems," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 1, No. 4, 2000, pp. 221–232.

[16] Krozel, J., Peters, M., Bilimoria, K., Lee, C., and Mitchell, S. B., "System Performance Characteristics of Centralized and Decentralized Air Traffic Separation Strategies," *4th USS/Europe Air Traffic Management R&D Seminar*, Santa Fe, NM, 2001.

[17] Xue, M., Rios, J., Silva, J., Ishihara, A., and Zhu, Z., "Fe^3: An Evaluation Tool for Low-Altitude Air Traffic Operations," *AIAA Aviation Forum*, Atlanta, GA, 2018.

[18] Hansman, R. J., Alcabin, M. S., Ball, M. O., L., C. M., Dunlay, W. J., Elias, A. L., Fearnsides, J. J., Lebacqz, J. V., Powderly, M. J., Smith, P. J., Trani, A., Wall, R., Zacharias, G. L., Menzies, T. R., and Hemel, S. V., "Air Traffic Controller Staffing in the En Route Domain: A Review of the Federal Aviation Administration's Task Load Model," Tech. Rep. Special Report 301, Transportation Research Board, National Research Concil of the National Academies, 2010.

[19] Cook, B., Arnett, T., Macmann, O., and Kumar, M., "Real-Time Radar-Based Tracking and State Estimation of Multiple Non-Conformant Aircraft," *AIAA SciTech Forum*, Grapevine, Texas, 2017.

[20] Park, C., Lee, H., and Musaffar, B., "Radar Data Tracking Using Minimum Spanning Tree-Based Clustering Algorithm," *11th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, Virginia Beach, VA, 2011.

[21] Hwang, I., Balakrishnan, H., Roy, K., and Tomlin, C., "Multiple-Target Tracking and Identity Management with Application to Aircraft Tracking," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 3, 2007, pp. 641–652.

[22] Kuchar, J. K., and Yang, L. C., "A Review of Conflict Detection and Resolution Modeling Methods," *IEEE Transactions On Intelligent Transportation Systems*, Vol. 1, No. 4, 2000.

[23] Bach, C., R. Farrell, and Erzberger, H., "An Algorithm for Level-Aircraft Conflict Resolution," Tech. Rep. Technical Report, AR-2009-214573, NASA, 2009.

[24] Krozel, J., and Peters, M., "Conflict Detection and Resolution for Free Flight," *Air Traffic Control Quarterly*, Vol. 5, No. 3, 1997.

[25] Sahawneh, L. R., and Beard, R. W., "Chain-based Collision Avoidance for UAS Sense-and-Avoid Systems," *AIAA Guidance, Navigation, and Control Conference*, Boston, MA, 2013.

[26] Cook, B., Cohen, K., and Kivelevitch, E., "A Fuzzy Logic Approach For Low Altitude UAS Traffic Management," *AIAA Science and Technology Forum and Exposition 2016*, San Diego, California, 2016.

[27] Hwang, I., and Tomlin, C., "Protocol-based Conflict Resolution for Finite Information Horizon," *Proceedings of the American Control Conference*, Anchorage, AK, 2002.

[28] Mueller, E., and Kochenderfer, M. J., "Multi-Rotor Aircraft Collision Avoidance using Partially Observable Markov Decision Processes," *AIAA Modeling and Simulation Technologies Conference*, Washington, D.C., 2016.

[29] Xue, M., and Do, M., "Scenario Complexity for Unmanned Aircraft System Traffic," *AIAA Aviation Forum*, Dallas, TX, 2019.