Journal of Guidance, Control, and Dynamics
Vol. 40, No. 1, January 2017

Unmanned Aircraft Systems Airspace Integration: A Game Theoretical Framework for Concept Evaluations

Negin Musavi,* Deniz Onural,† Kerem Gunes,† and Yildiray Yildiz‡
Bilkent University, 06800 Ankara, Turkey

DOI: 10.2514/1.G000426

The focus of this paper is to present a game theoretical modeling framework for the integration of unmanned aircraft systems into the National Airspace System. The problem of predicting the outcome of complex scenarios, where manned and unmanned air vehicles coexist, is the research problem of this work. The fundamental gap in the literature is that the models of interaction between manned and unmanned vehicles are insufficient: 1) They assume that pilot behavior is known a priori, and 2) They disregard pilot reaction and the decision-making process. The contribution of this paper is to propose a realistic modeling framework that will fill this gap. The foundations of the proposed method are formed by game theory, which investigates strategic decision making between intelligent agents; bounded rationality concept, which is based on the fact that humans cannot always make perfect decisions; and reinforcement learning, which is shown to be effective in human learning in psychology literature. An analysis of integration is conducted using an example scenario in the presence of manned aircraft and fully autonomous unmanned aircraft systems equipped with sense-and-avoid algorithms.

Nomenclature

m	=	message
P	=	probability
Q	=	value function for state-action pairs
R	=	minimum safety distance, nm
$\boldsymbol{r}$	=	relative position vector between the unmanned aircraft system and the intruder, nm
$\boldsymbol{r}_m$	=	minimum relative position vector between the unmanned aircraft system and the intruder, nm
$\boldsymbol{r}_0$	=	initial relative position vector between the unmanned aircraft system and the intruder, nm
s	=	state
V	=	value function for states
ϵ	=	learning rate
ζ	=	angle between $\boldsymbol{r}$ and $\boldsymbol{v}_{AB}$, rad
$\boldsymbol{v}$	=	velocity vector, m/s
$\boldsymbol{v}_A$	=	velocity vector of the unmanned aircraft system, m/s
$\boldsymbol{v}_B$	=	velocity vector of the intruder, m/s
$\boldsymbol{v}_{AB}$	=	relative velocity vector between the unmanned aircraft system and the intruder, m/s
$\boldsymbol{v}_x$	=	X component of the velocity vector, m/s
$\boldsymbol{v}_y$	=	Y component of the velocity vector, m/s
$\boldsymbol{v}_A^d$	=	velocity adjustment command vector for the unmanned aircraft system, m/s
$\pi(\cdot)$	=	policy
Ψ	=	heading angle, rad
Ψ_d	=	desired heading angle, rad

I. Introduction

DUE to their operational capabilities and cost advantages, the interest in unmanned aircraft systems (UASs) is increasing rapidly. However, the UAS industry has not realized its potential as much as desired (for example, a fully developed civilian UAS market still does not exist), and the biggest reason behind this is thought to be that UASs still do not have routine access to the National Airspace System (NAS) [1]. UASs can fly only in segregated airspace with restricting rules, because technologies, standards, and procedures for a safe integration of UASs into airspace have not matured yet. The aviation industry is very sensitive to risk, and for new vehicles such as UASs to enter into this sector, they need to be proven to be safe and it must be shown that they will not affect the existing airspace system in any negative way [2,3]. This needs to be done before giving UASs unrestricted access to the NAS. Since the routine access of UASs into the NAS is not a reality yet, and thus there is not enough experience accumulated about the issue, it is extremely hard to predict the effects of the technologies and concepts that are developed for the integration. Therefore, employing simulations is currently the only way to understand the effects of UAS integration on the air traffic system [4]. These simulation studies need to be conducted with realistic hybrid airspace system (HAS) models, where manned and unmanned vehicles coexist.

Many existing HAS models in the literature are based on the assumption that the pilots of manned aircraft always behave as they should, without ever deviating from the ideal behavior. However, it is not realistic to expect that the pilot, as a decision maker (DM), will always behave deterministically. It is not always predictable, for example, whether a pilot agrees with a traffic control alert system's (TCAS's) resolution advisory or not [5]. The collision between two aircraft (a DHL Boeing 757 and a Bashkirian Tupolev 154) over Uberlingen, Germany, near the Swiss border at 21:35 (Coordinated Universal Time) on 1 July 2002, is good evidence that pilots may decide not to act parallel to a TCAS advisory or may ignore traffic controller's commands during high-stress situations [5]. Furthermore, in a recent study, it was shown that only 13% of pilot responses ended up matching the deterministic pilot model that was assumed for TCAS development [6,7]. In light of the preceding discussion, it is seen that incorporating a human decision-making process in HAS models may improve the predictive power of these models. It is noted that the rules and procedures that need to be followed by the pilots, such as obeying TCAS commands and its pilot model predictions 100% of the time, can also be incorporated into the game theoretical modeling framework that is proposed in this paper.

One of the primary impediments that hinders the integration of UASs into NAS is the lack of a matured sense-and-avoid (SAA) capability. For a SAA method to be approved, it should be analyzed in order to determine its potential impact on the surrounding air traffic and on the specific UAS mission. To perform an analysis and evaluation of any SAA logic, it is necessary to model the actions that a

Presented as Paper 2016-1001 at the Infotech@Aerospace Conference, San Diego, CA, 04–08 January 2016; received 22 January 2016; revision received 14 July 2016; accepted for publication 15 July 2016; published online 29 September 2016. Copyright © 2016 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved. Copies of this paper may be made for personal and internal use, on condition that the copier pay the per-copy fee to the Copyright Clearance Center (CCC). All requests for copying and permission to reprint should be submitted to CCC at www.copyright.com; employ the ISSN 0731-5090 (print) or 1533-3884 (online) to initiate your request.

*Ph.D. Student, Mechanical Engineering, Main Campus, Cankaya.

†Student, Electrical Engineering, Main Campus, Cankaya.

‡Assistant Professor, Mechanical Engineering, Main Campus, Cankaya. Senior Member AIAA.

pilot would take when facing a conflict [8]. There are various studies in the literature that use HAS models to evaluate the safety of SAA systems. In their work [8], Maki et al. constructed a SAA logic based on the model developed by the Massachusetts Institute of Technology's (MIT) Lincoln Laboratory [9] and Kuchar et al. [10] did a rigorous analysis of the TCAS to implement the SAA algorithm for remotely piloted vehicles. In both of these studies, the SAA system was evaluated through simulations in a platform that used MIT's NAS encounter model. In the evaluations, it was assumed that pilot decisions were known a priori and depended on the relative motion of the adversary during a specific conflict scenario. In their work, Perez-Batlle et al. [11] classified separation conflicts between manned and unmanned aircraft and proposed separation maneuvers for each class. These maneuvers awere tested in a simulation environment, where it was assumed that the pilots would follow these maneuvers 100% of the time. Florent et al. [12] developed a SAA algorithm and tested it via simulations and experiments. In both of these tests, it was assumed that the intruding aircraft did not change its path while the UAS was implementing the SAA algorithm. There were other simulation studies, such as [13], that tested and evaluated different collision avoidance algorithms, where some predefined actions were used as pilot models. There are also substantial studies with remotely piloted aircraft where the effects of SAA systems on the workload and situational awareness of the pilots are investigated via simulations and flight tests [14–16].

In this study, we build upon the aforementioned successful approaches and incorporate human decision making into HAS modeling. Specifically, in the HAS model developed in this work, the pilot behavior is not assumed to be known a priori and decisions are obtained using 1) the bounded rationality concept, which helps model imperfect decisions as opposed to modeling the pilot as a perfect decision maker; and 2) reinforcement learning, which helps model time-extended decisions as opposed to assuming one-shot decision making. To predict pilot reactions in complex scenarios where UASs and manned aircraft coexist, in the presence of automation such as a SAA system, a game theoretical methodology is employed, which is formally known as the semi-network-form games [6]. Using this method, probable outcomes of HAS scenarios are obtained that contain interacting humans (pilots) who also interact with an UAS equipped with a SAA algorithm. The obtained pilot model is used in close encounters, where TCAS and air traffic management instructions can also be incorporated. To obtain realistic pilot reactions, bounded rationality is imposed by using level-K approach [17,18], which is a concept in game theory that models human behavior assuming that humans think in different levels of reasoning. In the proposed framework, pilots optimize their trajectories based on a goal function representing their preferences for system states. During the simulations, UASs fly autonomously based on a preprogrammed flight plan. In these simulations, the effect of certain system variables (such as the horizontal separation requirement and required time to conflict for UASs) and the effect of responsibility assignment for conflict resolutions on the safety and performance of the HAS are analyzed (see [19] for the importance of these variables and responsibility assignment for UAS integration.). To enable the UASs to perform autonomously in the simulations, it is assumed that they employ a SAA algorithm. The simulation results are provided for two different SAA methods; in addition, these two methods are compared quantitatively in terms of safety and performance using the proposed modeling framework.

In prior works, the method proposed in this paper was used to investigate small-scale scenarios (in terms of number of agents): In [20], the dynamics between a smart grid operator and a cyber attacker is modeled, and in [6,21,22], the dynamics between two interacting pilots are modeled. More recently, in [23], a medium-scale scenario with 50 interacting pilots was analyzed. In the study with 50 pilots, the simulation environment used a gridded airspace where aircraft moved from one grid intersection to another to represent movement. In addition, the pilots could only observe grid intersections to see whether or not another aircraft was nearby. All these simplifying assumptions decreased the computational cost but also decreased the fidelity of the simulation. In this study, the following occurs:

1) A dramatically more complex scenario in the presence of manned and unmanned aircraft is investigated.

2) The simulation environment is not discretized and the aircraft movements are simulated in continuous time.

3) Realistic aircraft and UAS physical models are used.

4) Initial states of the aircraft are obtained from real flight data. Hence, a much more representative simulation environment with the inclusion of UASs equipped with a SAA algorithm is used to obtain probabilistic outcomes of HAS scenarios.

The organization of the paper is as follows: In Sec. II, the proposed modeling method is explained. In Sec. III, the HAS with its components is described in detail, together with an investigation of model validation. In Sec. IV, simulation results are provided with detailed discussions. Finally, conclusions are provided in Sec. V.

II. Modeling Methodology

The most challenging problem in the prediction of the outcomes of complex scenarios where manned and unmanned aircraft coexist is obtaining realistic pilot models. A pilot model in this paper refers to a mapping from observations of the pilot to his/her actions. To achieve a realistic human reaction model, certain requirements need to be met. First, the model should not be deterministic because it is known from everyday experience that humans do not always react exactly the same when they are in a given "state." Here, the term state refers to the observations and the memory of the pilot. For instance, observing that an aircraft is approaching from a certain distance is an observation and remembering one's own previous action is memory. Second, pilots should show the characteristics of a strategic decision maker, meaning that the decisions must be influenced by the expected moves of other "agents." Agents can be either the other pilots or the automation logic of UASs. Third, the decisions emanating from the model should not always be the best (or mathematically optimal) decisions because it is known that human actions are less than optimal in many situations. Finally, it should be considered that a human DM's predictions about other human DMs are not always correct. To accomplish all of these requirements, level-k reasoning and reinforcement learning are used together, forming a nonequilibrium game theoretical solution concept. It is noted that, in this study, the UASs are assumed to be fully autonomous.

A. Game Theoretical Modeling of Interactive Decision Making

Level-k reasoning is a game theoretical solution concept for which the main idea is that humans have various levels of reasoning in their decision-making process. Level 0 represents a "nonstrategic" DM who does not take into account other DMs' possible moves when choosing his/her own actions. This behavior can also be named as reflexive because it only reacts to the immediate observations. In this study, given a state, a level-0 pilot flies an aircraft with constant speed and heading starting from its initial position toward its destination. A level-1 DM assumes that the other agents in the scenario are level 0 and takes actions accordingly to maximize his/her rewards. A level-2 DM takes actions as though the other DMs are level 1. In a hierarchical manner, a level-k DM takes actions assuming that the other DMs behave as level-(k-1) DMs.

B. Reinforcement Learning for the Partially Observable Markov Decision Process

Reinforcement learning is a mathematical learning mechanism that mimics the human learning process. An agent receives an observable message of an environment's state and then chooses an action, which changes the environment's state; the environment in return encourages or punishes the agent with a scalar reinforcement signal known as reward. Given a state, when an action increases (decreases) the value of an objective function (reward), which defines the goals of the agent, the probability of taking that action increases (decreases). Reinforcement learning algorithms mostly involve estimating two value functions: the state value function V, and the state-action value function Q [24]. $V(s)$, which is the value of state "s," is the total amount of reward an agent can expect to gather in the

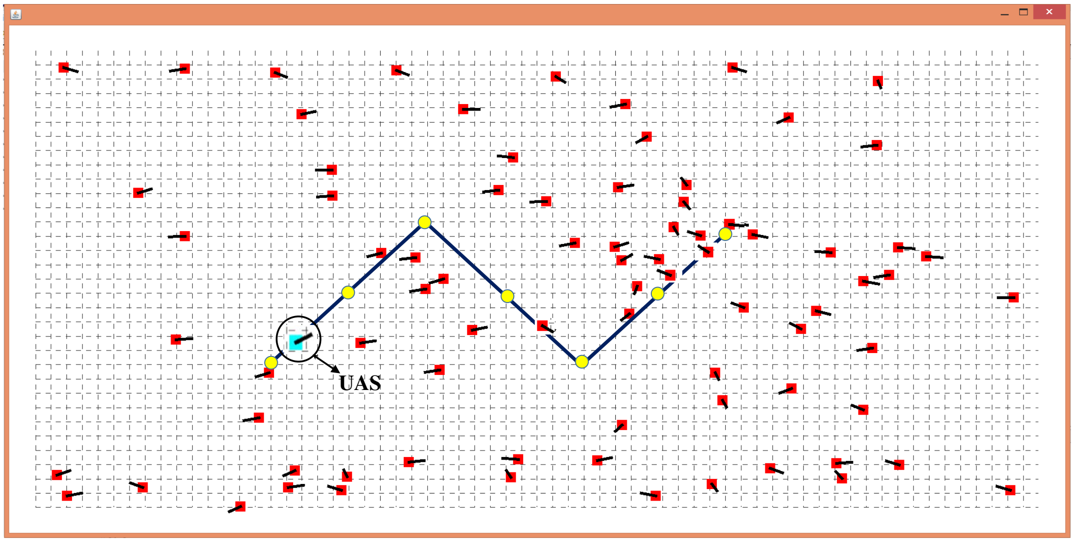


Fig. 1 Snapshot of the hybrid airspace scenario in the simulation platform. Each square stands for a 5 × 5 n mile area.

future, starting from that state. It is an estimate of how good it is for an agent to be in a particular state. $Q(s,a)$, which is the value of taking action "a" in state s under a particular policy, is the total amount of reward an agent can expect to gather in the future, starting from that state and taking that action by following the given policy. It is an estimate of how good it is for an agent to perform a given action in a given state.

Since the human DM as the agent in the reinforcement learning process is not able to observe the whole environment state (i.e., the positions of all of the aircraft in the scenario), the agent receives only partial state information. By assuming that the environment is Markov, and based on the partial observability of the state information, this problem can be generalized to a partially observable Markov decision process (POMDP). In this study, the reinforcement learning algorithm developed by Jaakkola et al. [25] is used to solve this POMDP. In this approach, different from conventional reinforcement learning algorithms, the agent does not need to observe all of the states for the algorithm to converge. A POMDP value function and a Q-value function, in which m and a refer to the observable message of the state s and action, respectively, are given by

$$V(m) = \sum_{s \in m} P(s|m) V(s) \tag{1}$$

$$Q(m,a) = \sum_{s \in m} P(s|m) Q(s,a) \tag{2}$$

A recursive Monte Carlo strategy is used to compute the value function and the Q-value function. It is noted that the pilot model is given in terms of a policy π, which is a mapping from observations or messages m to actions a. During reinforcement learning, the policy update is calculated as follows, where the policy is updated toward π^1 with an ε learning rate after each iteration:

$$\pi(a|m) \to (1-\varepsilon)\pi(a|m) + \varepsilon \pi^1(a|m) \tag{3}$$

where π^1 is chosen such that

$$J^{\pi^1} = \max_a (Q^{\pi}(m,a) - V^{\pi}(m))$$

For any policy $\pi^1(a|m)$, J^{π^1} is defined as

$$J^{\pi^1} = \sum_a \pi^1(a|m)(Q^{\pi}(m,a) - V^{\pi}(m)) \tag{4}$$

It is noted that the pilot model, or the policy, is obtained once the policy converges during this iterative process.

C. Combining Game Theory with Reinforcement Learning

The method employed in this study carefully combines the two concepts explained in the preceding sections: game theory and reinforcement learning (RL). The method consists of two stages: 1) obtaining pilot reaction models with various levels (level k), and 2) simulating a given scenario using these models. In the first stage, which can also be considered as the "training" stage, a level-1-type model is trained by assigning level-0 behavior to all of the agents except the one that is being trained. The trainee learns to react as best as he/she can in this environment using RL. Thus, the resulting behavior becomes a level-1 type. Similarly, a level-2 behavior is trained by assigning level-1 behavior to all of the agents but the trainee. This process continues until the highest desired level is reached. Once all of the desired levels are obtained, the first stage ends and, in the second stage, a given scenario is simulated by assigning certain proportions of these levels to the agents in the scenario. It is noted that, in the scenario investigated in this paper, the pilots of the manned aircraft have level-0, level-1, and level-2 behavior types; whereas the movements of the UAS are commanded via sense-and-avoid algorithms.

III. Components of the Hybrid Airspace Scenario

The investigated scenario consists of 180 manned aircraft with predefined desired trajectories and an UAS that moves based on its preprogrammed flight plan from one waypoint to another. Figure 1 shows a snapshot of this scenario where the small squares correspond to manned aircraft and the large square corresponds to the UAS. The size of the considered airspace is 600 × 300 km. The airspace is gridded, just to make it easier to visually grasp the dimensions (two neighboring grid points are 5 n mile away); nevertheless, all aircraft, manned or unmanned, move in the airspace continuously. Circles show the predetermined waypoints that the UAS is required to pass. The lines passing through the waypoints show the predetermined path of the UAS. It is noted that the UAS does not follow this path exactly because it needs to deviate from its original trajectory to avoid possible conflicts using an onboard SAA algorithm. The initial positions, speeds, and headings of the aircraft were obtained from the Flightradar24 Web site, which provides live air traffic data.** The data were collected from the air traffic volume on Colorado state, U.S. airspace, on 11 March 2015. It is noted that, in the Next Generation

**Data available online at http://www.flightradar24.com [retrieved 11 March 2015].

Airspace System (referred to as NextGen), air travel demand is expected to increase dramatically; thus, traffic density is expected to be much more than it is today. To represent this situation, the number of aircraft in the scenario is increased by projecting various aircraft at different altitudes to a given altitude. To handle the increase in aircraft volume in NextGen, it is expected that new technologies and automation will be introduced, such as the automatic dependent surveillance–broadcast (ADS-B), which is a technology that enables an aircraft to receive other aircraft's identification, position, and velocity information, as well as to send its information to others. In the investigated scenario, it is assumed that each aircraft is equipped with the ADS-B. It is noted that the ADS-B can also provide information about the flight path of an aircraft, which is highly relevant to collision avoidance. In our simulations, we provided this crucial information by answering the following question for each agent: In a given time window, where in my observation space do I expect an intruding aircraft? (See Sec. III.A for details.)

A. Pilot Observations and Memory

Although ADS-B provides the positions and the velocities of other aircraft, with his/her limited cognitive capabilities, a pilot cannot possibly process all this information during his/her decision-making process. In this study, in order to model pilot limitations, including the limitations at visual acuity and perception depth, as well as the limited viewing range of an aircraft, it is assumed that the pilots can observe (or process) the information from a limited portion of the nearby airspace. This limited portion is simulated as equal angular portions of two cocentered circles called the "observation space," which is schematically depicted in Fig. 2. The radius of the inner circle represents the pilot vision range, which is taken as 1 n mile based on a survey executed in [26]. The radius of the outer circle is a variable that depends on the separation requirements. Since the standard separation for manned aviation is 3–5 n miles [11], this radius is taken as 5 n miles. Whenever an intruder aircraft moves toward one of the six regions of the observation space (see Fig. 2), the pilot perceives that region as "full." The pilot, in addition, can roughly distinguish the approach angle of the approaching intruder. A full region is categorized into four cases: with 1) 0 deg < approach angle < 90 deg, 2) 90 deg < approach angle < 180 deg, 3) 180 deg < approach angle < 270 deg, and 4) 270 deg < approach angle < 360 deg. Figure 2 depicts a typical example, where pilot A observes that aircraft B is moving toward one of the six regions. In this particular example, pilot A perceives the region as full with the approach angle in the interval [90 deg,180 deg] and the rest of the regions as "empty." The information about the emptiness and fullness of a region, as well as the approach angle, is fed to the reinforcement learning algorithm simply by assigning 0 to empty regions and 1, 2, 3, and 4 to full regions, based on the approach angle classifications explained previously. Pilots also know the best action that would move the aircraft closest to its trajectory [best trajectory action (BTA)] and the best action that would move the aircraft closest to its final destination [best destination action (BDA)]. Moreover, pilots have a memory of what their actions were at the previous time step. Given an observation, the pilots can choose between three actions: 45 deg left, straight, or 45 deg right, which are coded with numbers 0, 1, and 2. Six ADS-B observations (one BTA, one BDA, and one previous move) make up nine total inputs for the reinforcement learning algorithm. Observations get five values: 0, 1, 2, 3, or 4. The previous move, BTA, and BDA have three dimensions each: 45 deg left, 45 deg right, or straight. Therefore, the number of states for which the reinforcement learning algorithm needs to assign appropriate actions is $5^6 \times 3^3 = 421,875$.

B. Pilot Objective Function

The goal of the reinforcement learning algorithm is to find the optimum probability distribution among possible action choices for each state. As explained previously, reinforcement learning achieves this goal by evaluating actions based on their return, which is calculated via a reward/objective function. A reward function can be considered as a happiness function, goal function, or utility function that represents, mathematically, the preferences of the pilot among different states. In this paper, the pilot reward function is defined as

$$\begin{aligned}\text{reward} = {} & w1 * (-C) + w2 * (-S) + w3 * (-CA) \\ & + w4 * (D) + w5 * (-P) + w6 * (-E)\end{aligned} \tag{5}$$

In Eq. (5), C is the number of aircraft within the collision region. Based on the definition provided by the Federal Aviation Administration, the radius of collision is taken as 500 ft [27]. S is the number of air vehicles within the separation region. The radius of the separation region is 5 n miles [11]. CA represents whether the aircraft is getting closer to the intruder or going away from the intruder, and takes values of 1 for getting closer or 0 for going away. D represents how much the aircraft gets closer to or goes away from its destination normalized by the maximum distance it can fly in a time step. The time step is determined based on the frequency of pilot decisions. The average time step during reinforcement learning is determined to be 20 s. P represents how much the aircraft gets closer to or goes away from its ideal trajectory normalized by the maximum distance it can fly in a time step, and E represents whether or not the

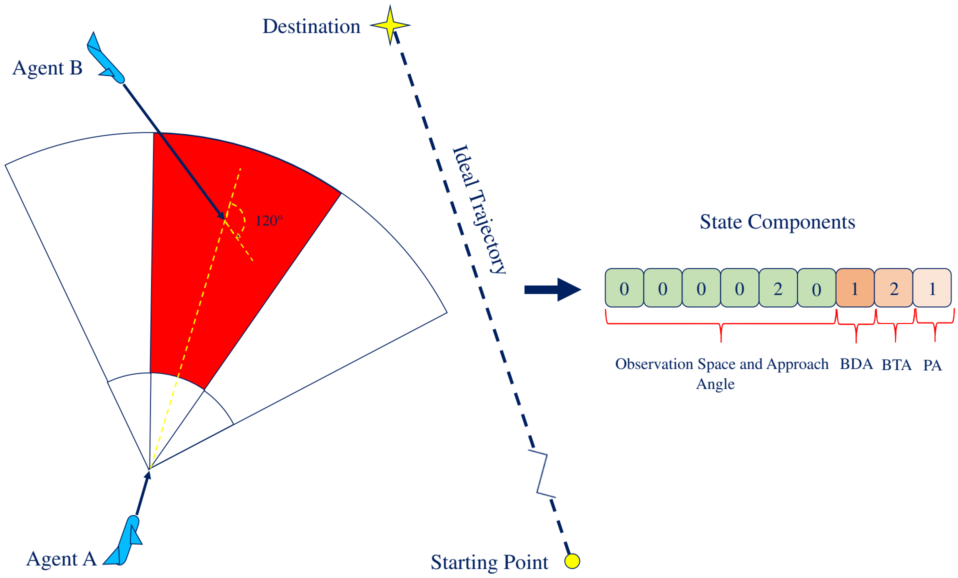


Fig. 2 Pilot observation space (PA, previous action).

pilot makes an effort (move). E gets a value of 1 if the pilot makes a new move and 0 otherwise.

C. Manned Aircraft Model

The initial positions, speeds and heading angles of the manned aircraft were obtained from the Flightradar24 Web site (see footnote **). It is assumed that all aircraft are in their en route phase of travel with constant speed $\|\boldsymbol{v}\|$ in the range of 150–550 kt. Aircraft are controlled by their pilots who may decide to change the heading angle for 45 or −45 deg, or they may decide to keep it unchanged. Once the pilot gives a heading command, the aircraft moves to the desired heading ψ_d in the constant speed mode. The heading change is modeled by first-order dynamics with the standard rate turn: a turn in which an aircraft changes its heading at a rate of 3 deg per second (360 deg in 2 min) [28]. This is modeled as first-order dynamics with a time constant of 10 s $(45 \times (1 - 1/e)/3 \approx 10)$. Therefore, the aircraft heading dynamics can be given as

$$\dot{\psi} = -\frac{1}{10} \times (\psi - \psi_d) \tag{6}$$

and the velocity $[\boldsymbol{v} = (v_x, v_y)]$ is then obtained as

$$v_x = \|\boldsymbol{v}\| \sin\psi \tag{7}$$

$$v_y = \|\boldsymbol{v}\| \cos\psi \tag{8}$$

D. UAS Model

The UAS is assumed to have the dynamics of a RQ-4 Global Hawk with an operation speed of 340 kt [29]. It moves according to its preprogrammed flight plan and is equipped with a SAA system. The SAA system can initiate a maneuver to keep the UAS away from other traffic, if necessary, by commanding a velocity vector change. Otherwise, the UAS will continue moving based on its mission plan. Therefore, the UAS always receives a velocity command either to satisfy its mission plan or to protect its safety. Since the UAS has a finite settling time for velocity vector changes, the desired velocity $\boldsymbol{V}_d$ cannot be reached instantaneously. Therefore, the velocity vector variation dynamics of the UAS is modeled by a first-order dynamics with a time constant of 1 s [30], which is represented as

$$\dot{\boldsymbol{v}} = -(\boldsymbol{v} - \boldsymbol{v}_d) \tag{9}$$

E. Sense-and-Avoid Algorithms

To assure that the UAS can detect probable conflicts and can autonomously perform evasive maneuvers, it should be equipped with a SAA system. In this paper, two different SAA algorithms are investigated. These SAA algorithms were developed by Fasano et al. [31] (referred as SAA1) and Mujumdar and Padhi [30] (referred as SAA2). The algorithms consist of two phases: the conflict detection phase and the conflict resolution phase. In the detection phase, the SAA algorithms project the trajectories of the UAS and the intruder aircraft in time, using a predefined time interval; if the minimum distance between the aircraft during this time is calculated to be less than a minimum required distance R, it is determined that there will be a conflict. The same conflict detection logic is used for both of the SAA algorithms. To prevent the conflict, the UAS starts an evasive maneuver in the conflict resolution phase, which is handled differently for SAA1 and SAA2. In the SAA1 resolution phase algorithm, a velocity adjustment is suggested that guarantees minimum deviation from the trajectory. The velocity adjustment command $\boldsymbol{v}_A^d$ for the UAS is given in the following equation:

$$\boldsymbol{v}_A^d = \left[\frac{v_{AB}\cos(\eta - \zeta)}{\sin(\zeta)}\right]\left[\sin(\eta)\frac{\boldsymbol{v}_{AB}}{v_{AB}} - \sin(\eta - \zeta)\frac{\boldsymbol{r}}{\|\boldsymbol{r}\|}\right] + \boldsymbol{v}_B \tag{10}$$

where $\boldsymbol{v}_A$ and $\boldsymbol{v}_B$ refer to the velocity vectors of the UAS and the intruder. The relative position and velocity between the UAS and the intruder are denoted as $\boldsymbol{r}$ and $\boldsymbol{v}_{AB}$, respectively. Also, ζ is the angle between $\boldsymbol{r}$ and $\boldsymbol{v}_{AB}$; and η is calculated as

$$\eta = \sin^{-1}\frac{R}{\|\boldsymbol{r}\|}$$

In the case of multiple conflict detections, the UAS will start an evasive maneuver to resolve the conflict that is predicted to happen earliest. In the SAA2 algorithm, the velocity adjustment vector is determined as given in the following equation, in which $\boldsymbol{r}_m$ stands for the minimum relative position vector between the UAS and the intruder during the conflict:

$$\boldsymbol{v}_A^d = \frac{-\boldsymbol{v}_A(\boldsymbol{r}_0 \cdot \boldsymbol{v}_{AB}/\|\boldsymbol{v}_{AB}\|) - (R - \|\boldsymbol{r}_m\|)(\boldsymbol{r}_m/\|\boldsymbol{r}_m\|)}{\|-\boldsymbol{v}_A(\boldsymbol{r}_0 \cdot \boldsymbol{v}_{AB}/\|\boldsymbol{v}_{AB}\|) - (R - \|\boldsymbol{r}_m\|)(\boldsymbol{r}_m/\|\boldsymbol{r}_m\|)\|} \tag{11}$$

where $\boldsymbol{r}_0$ refers to the initial relative position vector between the UAS and the intruder. In this solution strategy, the UAS moves to resolve the conflict until it retains the minimum safe distance with the intruder. Similar to the SAA1 algorithm, in the case of multiple intruders, the UAS will start an evasive maneuver to resolve the conflict that is predicted to happen earliest.

F. Model Validation

As noted earlier, since the routine access of UASs into NAS is not a reality yet, and thus there is not enough experience accumulated about the issue, it is extremely hard to predict the effects of the technologies and concepts that are developed for the integration. Therefore, employing a simulation is currently the only way to understand the effects of UAS integration on the air traffic system [4]. However, regardless of whether the modeled system exists or whether it is expected to be a reality in the future, the representative model should be validated [32].

In the following, we break down the validation task into two steps. In the first step, we explain that the underlying hierarchical game theoretical modeling approach is a useful and valid approach to model complex human interactions, based on earlier experimental studies. In the second step, we investigate the validity of the proposed approach for the UAS integration implementations. Since UAS integration data are not available yet, we take a different approach in this step: We first provide a validation methodology that can be used to validate the proposed approach when the data for UAS integration become available. We also explain that the proposed model has enough degrees of freedom that can be used to obtain a predictive model using these data. Then, we proceed to show that the trajectories created by the proposed model are similar to that of a validated encounter model for manned aircraft created using real radar data. Finally, we explain a validation method called “face validation,” which is commonly used when the modeled system is expected to be a reality in the future, and we argue that our simulation results in the next section can be used to apply this method.

1. *Validation of the Game Theoretical Modeling Approach*

In this study, we used a well-known game theoretical modeling approach called “level-k” reasoning. The advantage of this approach is its computational simplicity, where the intelligent agent makes behavioral assumptions about others and then produces the best response accordingly based on a reward function. Because of this simplicity, in multimove scenarios, such as the ones treated in this research, level-k reasoning provides computationally tractable solutions. This approach not only provides a computationally efficient solution but is also shown to be able to model complex human interactions in experimental settings: In [17], several experimental results were conducted using various sizes of subject pools that were made to play different games. Using the data from these games, models of strategic thinking were evaluated and compared; although some performed better than others, depending on the game type, the level-k approach was found to be “behaviorally more plausible.” It is noted that these experimental results are cited here, not to prove that the proposed approach is better than other game theoretical approaches but, to provide real-world data showing that the level-k modeling approach can represent real-world behavior in complex decision-making scenarios. In the scenarios studied in this work, the intelligent agents (pilots) are also strategic decision makers, as explained in earlier sections, and they need to make decisions in a complex environment to maximize their rewards. Therefore, the underlying game theoretical approach can also be considered as a good fit to the problem studied in this work.

2. *Validation of the Proposed Modeling Approach for UAS Integration Concepts*

As presented in the previous section, the game theoretical modeling approach is shown to model real-world behavior in earlier experimental studies. In this section, we address the following question: Can this approach be reliably used to model UAS integration scenarios?

a. Validation Methodology. There exist several model validation methods such as face validity, “historical data validation,” “parameter sensitivity analysis,” and “predictive validation” [32]. Among these methods, the most definitive technique is predictive validation, where the model is used to predict the system’s behavior and then the outputs of the model and the real system are compared. Here, we explain two main aspects of predictive validation [9] that can be used to validate the proposed model in this study when the UAS integration data become available: First, relevant statistics between the model and the real data should have reasonable agreement. For example, for UAS integration, the average deviation of the UAS from their intended trajectory against the type of the SAA algorithm should be similar between the model and the data. Similarly, the average number of separation violations between manned and unmanned aircraft for different kinds of SAA algorithms should match. Second, individual encounters should show similar characteristics. For example, the minimum separation distance between UAS and manned aircraft, as well as pilot decisions during encounters with similar geometry (approach angle, heading, etc.), should be able to be predicted with reasonable accuracy by the model.

b. Comparison with a Validated Model. Since UAS integration data are not available yet, we compared the results of the proposed model with a manned-aircraft-only encounter model created and validated by the Lincoln Laboratory using real radar data [9]. Sample trajectories are given in two text files that are open to the public: `cor_ac1.txt` and `cor_ac2.txt`. Among the encounters provided, five of them (listed as 3rd, 16th, 23rd, 34th, and 45th encounters) did not employ altitude or speed variations for conflict resolution, and thus can be used for our purposes. The objective of these comparisons is to show that the actions taken by the pilots in the proposed game theoretical model and the validated model are similar. In addition, minimum separation distances experienced between the aircraft and the times that the minimum separation occurs are shown to be reasonably close to each other for the compared models, such that the status of separation violation remains the same.

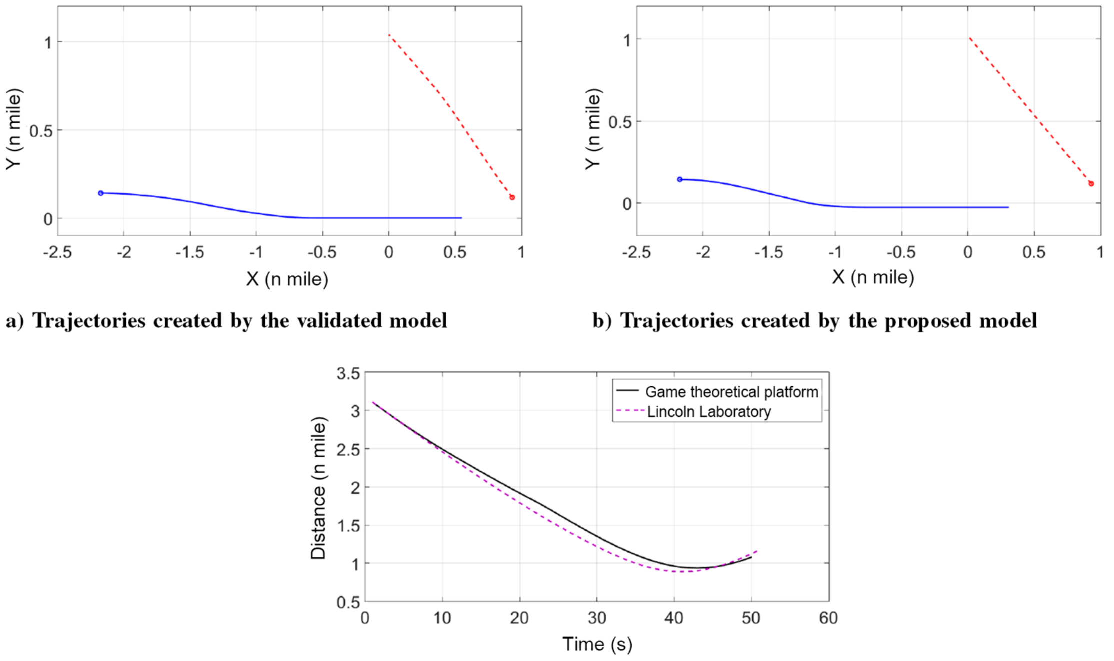
Figure 3a demonstrates aircraft trajectories during encounter number 3, listed in `cor_ac1.txt`. In Fig. 3b, the same encounter is regenerated using the proposed game theoretical model by assigning the same initial positions, initial heading angles, and initial speeds to two aircraft. It is seen that, although the trajectories are not exactly the same, pilot decisions determined by the game theoretical model are similar to the decisions provided by the validated model. In addition, according to Fig. 3c, the minimum separation distance is experienced after about 40 and 42 s for the validated model and the game theoretical model, respectively; and the difference between the minimum distances is about 0.05 n miles. In this example, the pilots represented by the solid and dashed curves are modeled as level-1 and level-0 pilots, respectively. Figures 4–7 also show similar characteristics where the encounter trajectories and pilot decisions are similar for the validated and proposed models.

c. Face Validation. Face validation is a validation method used for models that are developed for systems that are expected to be a reality in the future, such as UAS integration models [32]. In this method, two aspects of the model are evaluated:

1) Is the logic in the conceptual model correct?

2) Are the input–output relationships of the model reasonable?

The core ideas of the proposed framework (such as the level-k game theoretical concept, reinforcement learning, and bounded rationality) are supported by several references earlier in this study. In addition, the logic of the objective function used during the modeling process is detailed, where it is seen that the choice of the terms is logical. Finally, in the Simulation Results and Discussion section of this work (Sec. IV), input–output relationships of the model are discussed at length to show that they represent reasonable system

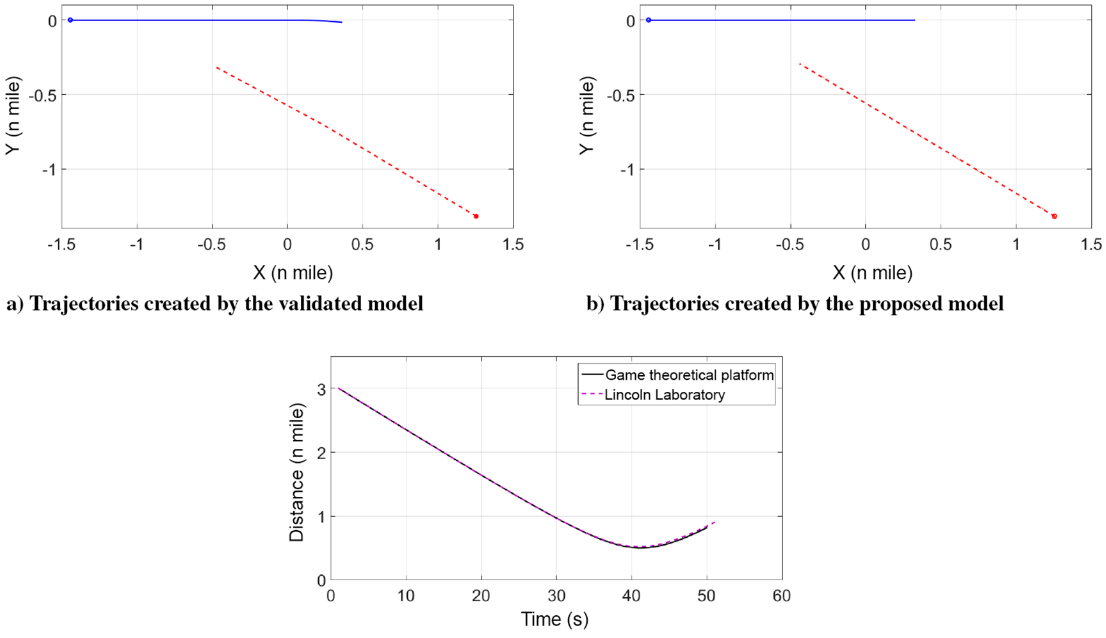


a) Trajectories created by the validated model

b) Trajectories created by the proposed model

c) Separation distances for each model

Fig. 3 Comparison of the trajectories created by the validated model and the game theoretical modeling approach for sample encounter number 3.

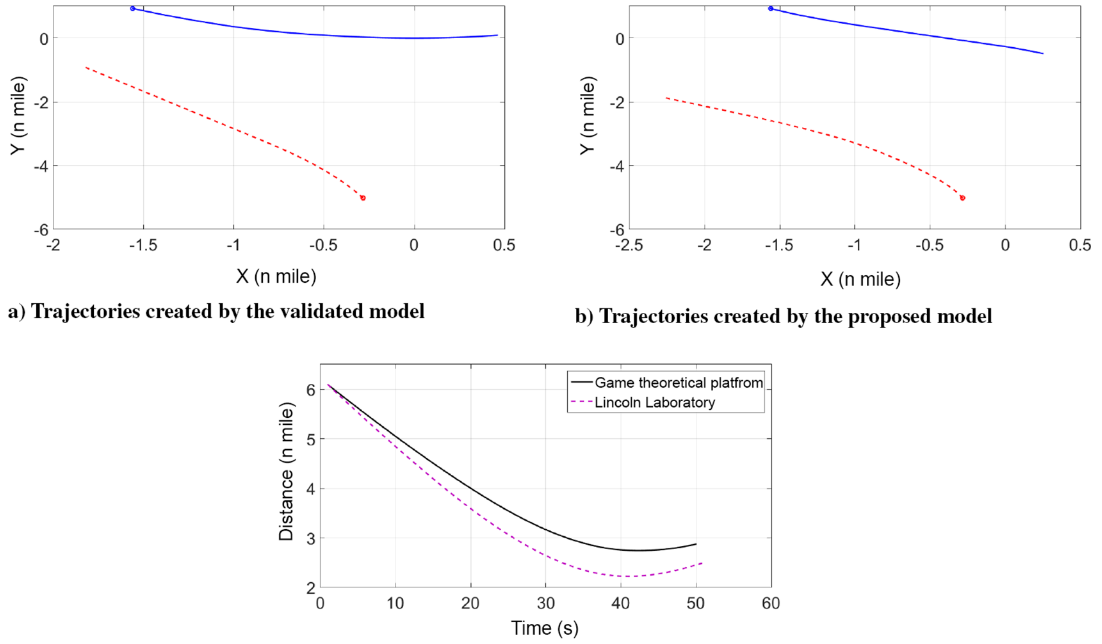


a) Trajectories created by the validated model

b) Trajectories created by the proposed model

c) Separation distances for each model

Fig. 4 Comparison of the trajectories created by the validated model and the game theoretical modeling approach for sample encounter number 16.



a) Trajectories created by the validated model

b) Trajectories created by the proposed model

c) Separation distances for each model

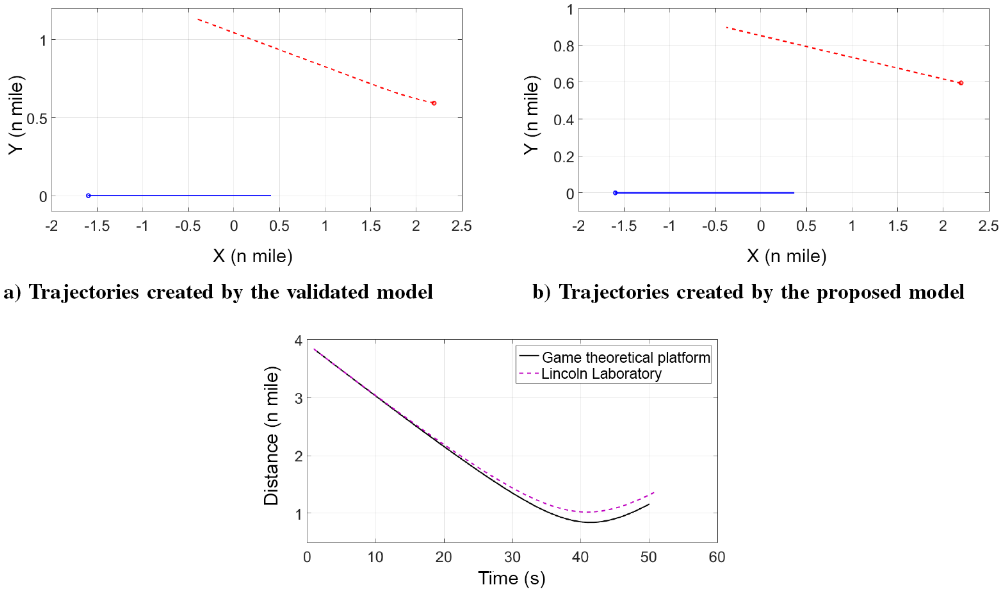
Fig. 5 Comparison of the trajectories created by the validated model and the game theoretical modeling approach for sample encounter number 23.

behavior. Therefore, the steps needed for the face validation of the proposed method are completed.

d. Remark. It is noted that, without collecting, processing, and analyzing real HAS data (which may be available in the near future), as well as carefully comparing the outputs with the model using available statistical validation tools (see [33]), the validation of the model cannot be accepted as completed. It is important for a model to have enough degrees of freedom so that, when discrepancies with the real data are detected, the model can be modified accordingly to obtain a match with the data with reasonable accuracy [33]. In this regard, the proposed game theoretical framework is a strong candidate for a successful UAS integration model because it contains several degrees of freedom, such as the objective function weights representing the importance of each term. In addition, the modular structure of the objective function allows the designers to add/subtract terms to achieve an agreement with the data.

IV. Simulation Results and Discussion

In this section, the results of a quantitative analysis of a simulation for UAS integration scenarios are presented. Before showing the results for the scenario explained in the previous section, single-encounter scenarios, where a single UAS and a single manned aircraft are in a collision path, are investigated. Later, the results for the scenario with multiple encounters in a crowded airspace are shown. A quantitative comparison between the two SAA algorithms in terms of their performance and safety is also presented.

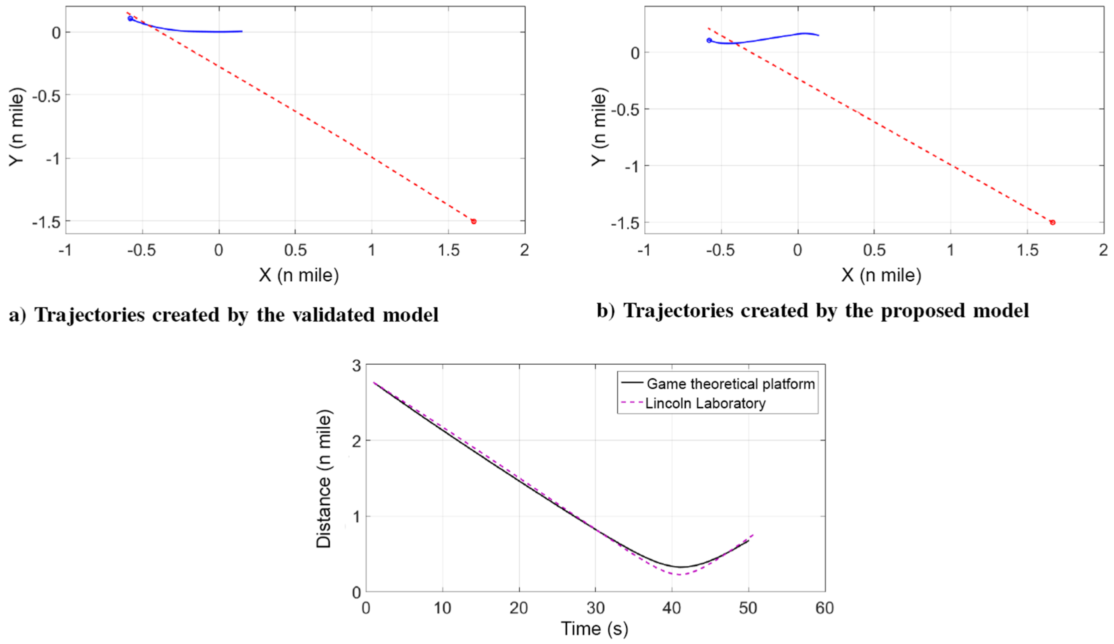


a) Trajectories created by the validated model

b) Trajectories created by the proposed model

c) Separation distances for each model

Fig. 6 Comparison of the trajectories created by the validated model and the game theoretical modeling approach for sample encounter number 34.



a) Trajectories created by the validated model

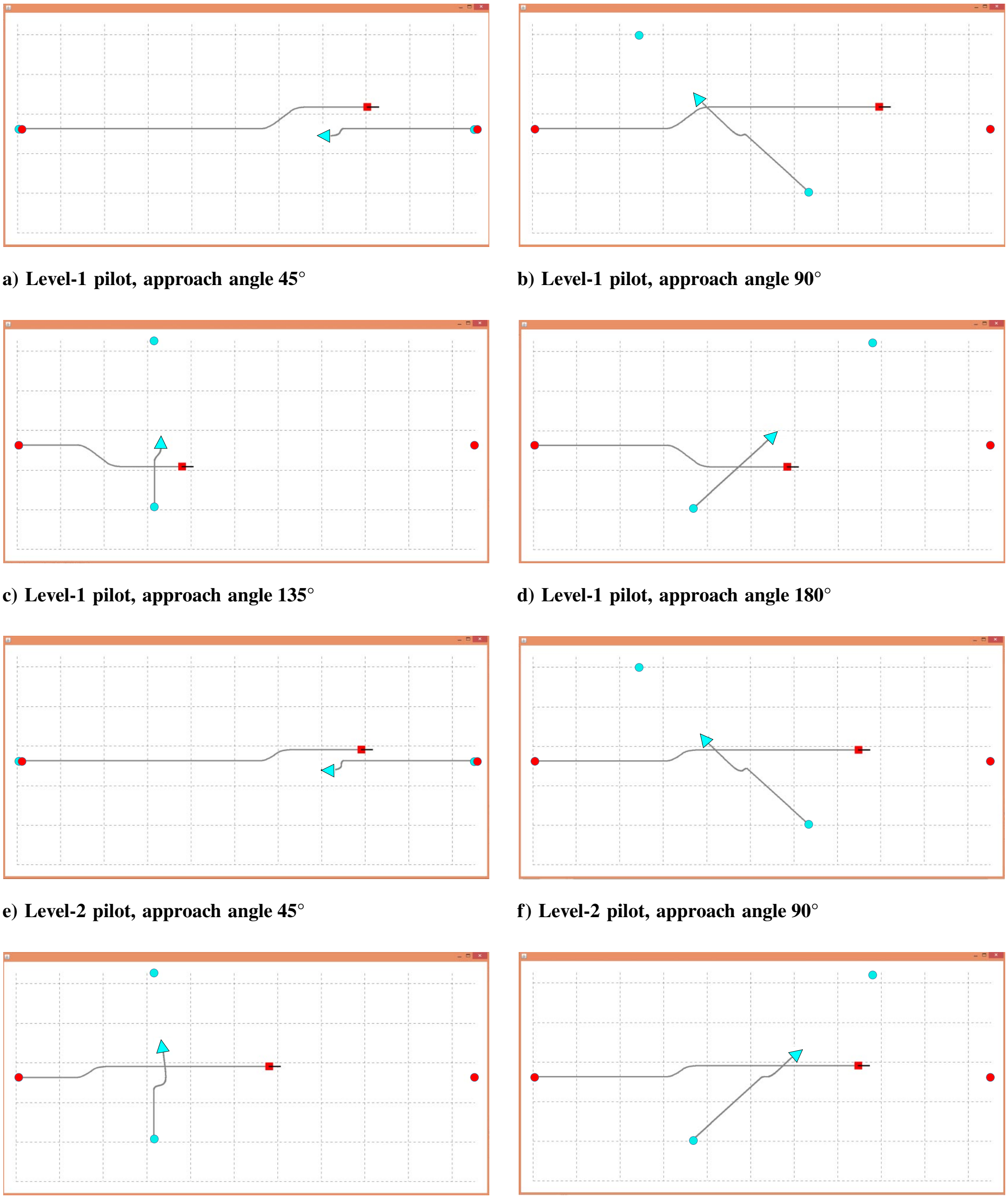
b) Trajectories created by the proposed model

c) Separation distances for each model

Fig. 7 Comparison of the trajectories created by the validated model and the game theoretical modeling approach for sample encounter number 45.

A. Hybrid Airspace Scenarios with a Single Encounter

To investigate the reactions of a level-k pilot during a conflict with an UAS, four single-encounter scenarios are designed. In these four scenarios, level-1 and level-2 policies are used for the manned aircraft pilots and the UAS follows the guidelines of the SAA1 algorithm, which may command velocity adjustments in order for the UAS to avoid the conflict. Apart from pilot levels, the effect of different approach angles, which take the values of 45, 90, 135, and 180 deg, are also investigated. Figure 8 depicts the snapshots of four cases, where the square corresponds to the manned aircraft and the triangle corresponds to the UAS. The track lines right behind the manned aircraft and the UAS represent their traveled path from their initial positions to where they stand in the snapshot. Circles show the initial positions and destinations. The geometric size of the scenarios is 100×50 km. In all cases, the manned aircraft and the UAS are heading toward a conflict that is detected by both the UAS, via its SAA system, and the pilot 20 s before a probable miss separation. A miss separation is declared when the relative distance becomes less than 5 n miles. The pilot then starts an evasive maneuver based on the level-1 and level-2 reasoning policies, and the UAS implements its own evasive maneuver based on the SAA1 system, for which the working principles are explained in Sec. III.E. Figure 9 depicts the separation distance and trajectory deviations during these single-encounter scenarios. Comparing the performances of the level-1 and level-2 pilots, it can be seen that the level-1 pilot maneuvers in a way that he/she provides more separation distance than a level-2 pilot, except for the case with a 90 deg approach angle. This, in general, is expected because the level-1 pilot assumes that the intruder is a level-0 decision maker who will continue his/her given path without changing his/her direction; therefore, the level-1 DM takes the



a) Level-1 pilot, approach angle 45°

b) Level-1 pilot, approach angle 90°

c) Level-1 pilot, approach angle 135°

d) Level-1 pilot, approach angle 180°

e) Level-2 pilot, approach angle 45°

f) Level-2 pilot, approach angle 90°

g) Level-2 pilot, approach angle 135°

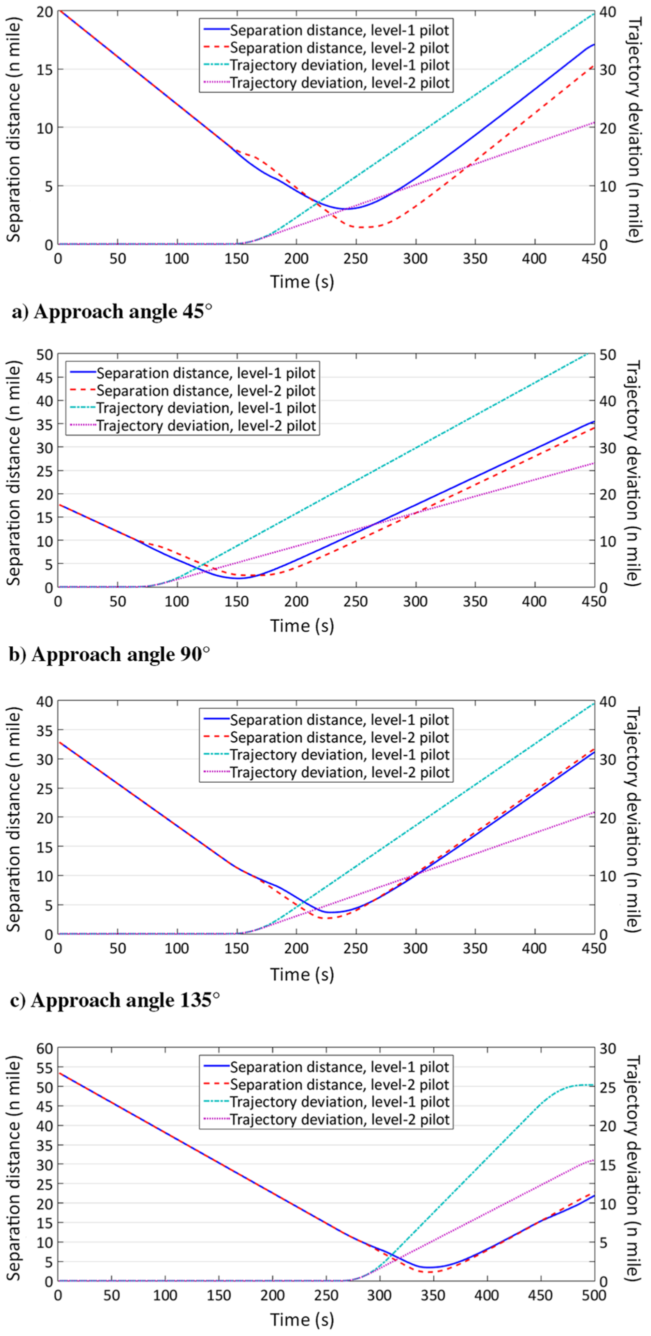
h) Level-2 pilot, approach angle 180°

Fig. 8 Level-1 and level-2 pilots interacting with UASs.

responsibility of the conflict resolution himself/herself. On the other hand, the level-2 pilot considers the intruder as a level-1 DM who will make maneuvers to avoid the conflict; therefore, the conflict resolution responsibility will be shared. That is why the level-2 pilot, in comparison with the level-1 pilot, avoids the UAS with less separation distance. It is noted, however, that the level-1 pilot deviates from its ideal trajectory significantly more than the level-2 pilot. Even in the case of a 90 deg approach angle, where the minimum separation distance between the level-1 pilot and the UAS is slightly less than the case with the level-2 pilot, the trajectory deviation of the level-2 pilot is less than that of the level-1 pilot. These analyses show that the type of pilot reaction during a conflict scenario makes a significant impact on the results when evaluating the performances of the SAA algorithms. The same conclusions are derived when the UAS maneuvers based on the SAA2 logic; however, these results are omitted to save space.

B. Hybrid Airspace Scenario with Multiple Encounters

The details of this scenario were explained in Sec. III. In this section, the scenario is simulated to investigate 1) the effect of the variations in the objective function parameters, 2) the effect of the distance and the time horizons, and 3) the effect of responsibility assignment for conflict resolution on safety and performance. Since the loss of separation is the most serious issue, the safety metric is taken as the number of separation violations between the UAS and the manned aircraft. Performance metrics, on the other hand, include 1) averaged manned aircraft trajectory deviations, 2) UAS trajectory deviation, and 3) total flight time of the UAS. In all of the simulations, level-0, level-1, and level-2 pilot policies are randomly distributed over the manned aircraft in such a way that 10% of the pilots fly based on level-0 policies, 60% of the pilots act based on level-1 policies, and 30% use level-2 policies. This distribution is based on the experimental results discussed in [17]. It is noted, although the



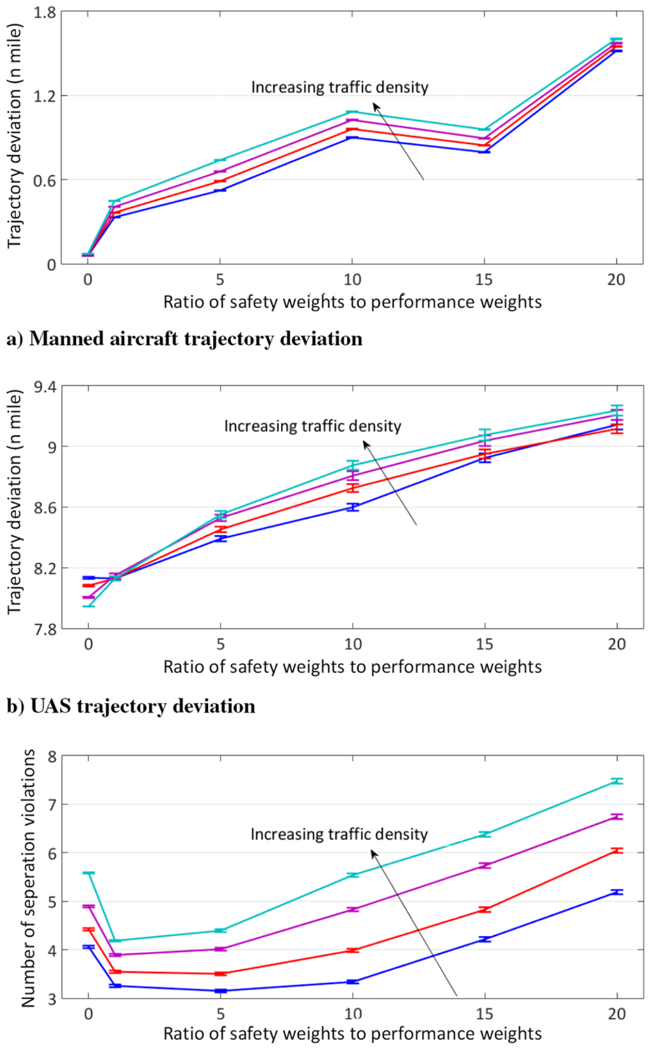
a) Approach angle 45°

b) Approach angle 90°

c) Approach angle 135°

d) Approach angle 180°

Fig. 9 Safety vs performance of level-k pilot interacting with UAS.



a) Manned aircraft trajectory deviation

b) UAS trajectory deviation

c) Number of separation violations

Fig. 10 Pilot model sensitivity analysis.

given distribution is obtained from human experimental studies, the studies did not necessarily include pilots, and therefore may not be fully representative but can easily be adapted to other distributional data for this framework.

1. Sensitivity Analyses of the Weighting Parameters in the Objective Function

In this section, the sensitivity of the pilot model to its parameters, which are the weight vector components of the objective function in Eq. (5), is investigated. Specifically, the effect of the ratio of the sum of the weights of the safety components of the objective function over the sum of the weights of the performance components

$$r = \frac{w1 + w2 + w3}{w4 + w5 + w6}$$

is investigated for various traffic densities. The results of this analysis for various traffic densities in the HAS are depicted in Fig. 10. It is seen that, as r increases, the trajectory deviations of both the manned aircraft and the UAS increase, regardless of the traffic density. Cooperation of the manned aircraft and the UAS to resolve the conflict reduces the number of separation violations up to a certain value of r. However, the number of violations starts increasing with a further increase in r. What this means is that, as pilots become more sensitive about their safety and start to overreact to probable conflicts with extreme deviations from their trajectories, the traffic is affected in a negative way.

Figure 11 presents the effect of increasing the ratio r in a single-encounter scenario where surrounding traffic does not exist. The percentage values provided in the figure is obtained for 5000 encounters. For each r value, these 5000 encounter “episodes” are repeated 1000 times to obtain reliable statistics. As expected, in the absence of surrounding traffic, the increase in the ratio r decreases the number of separation violations.

Remark: In this study, it is emphasized that humans are not expected to behave optimally in complex situations in the presence of multiple decision makers and automation due to 1) limited observation space, 2) limited processing power, and 3) limited information about other decision makers. The exploited modeling framework captures this suboptimal pilot response using several tools explained in earlier sections, including the RL algorithm, which provides convergence guarantees to a local maximum. The pilot behavior we observe in Fig. 10 is an example of suboptimal behavior where, in the presence of surrounding traffic, increased safety parameter weights (after a certain point) can cause extreme trajectory deviations and increased separation violations. Figure 11, on the

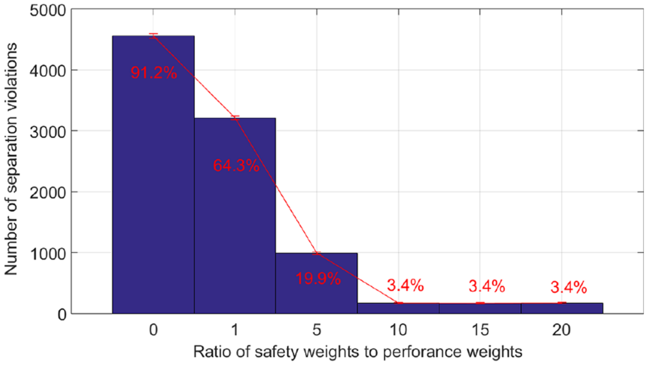


Fig. 11 Pilot model sensitivity analysis for the single-encounter scenario.

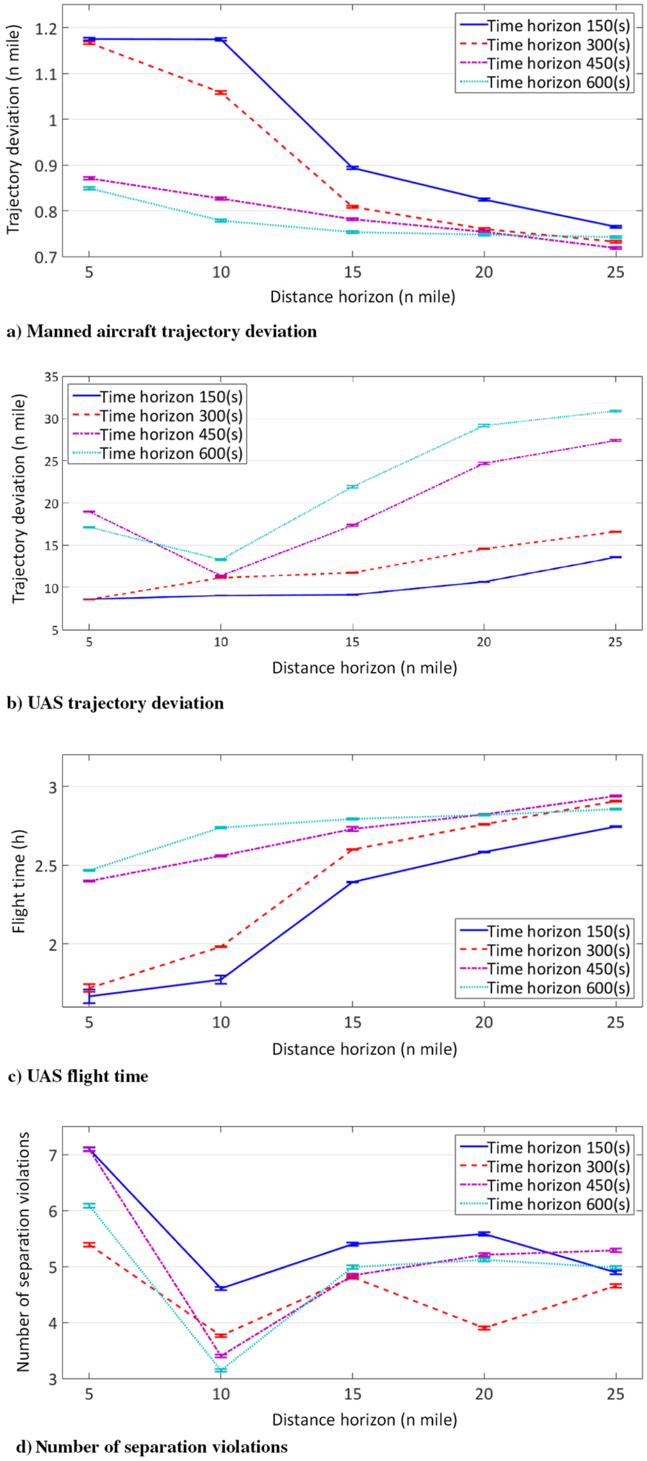
other hand, presents the expected behavior of decreasing violations with increased safety weights, when the scenario is much simpler.

2. Effect of Distance and Time Horizons on Performance and Safety

Although the standard separation distance for manned aviation is 3–5 n miles [11], UASs might require wider separation requirements compared to manned aircraft. In the following analysis, horizontal separation requirement for the UAS is called the distance horizon, and the effect of it is reflected into the simulation by defining this value as the “scan radius” for the SAA algorithm: The SAA algorithm considers an intruding aircraft as a possible threat only if the aircraft is within the scan radius. Another variable for which the effect is investigated is defined as the time to separation violation and is called the time horizon. In the simulation, the time horizon is used as the time interval, within which the UAS predicts a probable conflict. Figures 12 and 13 show the effects of the time horizon and the distance horizon on the safety and performance of the system, when SAA1 and SAA2 algorithms are used, respectively. When the SAA1 algorithm is employed, it is seen in Fig. 12 that increasing the distance horizon of the UAS makes the SAA1 system detect probable conflicts from a larger distance, which in turn increases the UAS trajectory deviation. Increasing the time horizon makes a similar effect on trajectory deviation. A high UAS trajectory deviation results in higher flight times for the UAS to complete its mission. In addition, higher distance and time horizons reduce the trajectory deviations of the manned aircraft because conflicts are resolved mostly by the UAS. When the UAS foresees the probable conflicts earlier (with increased time and distance horizons), the number of separation violations generally decreases. Increasing the distance and time horizons after a certain point does not improve the safety (number of separation violations), because the UAS starts to disturb the traffic unnecessarily due to the overreactions of the SAA system.

The first observation that strikes the eye in the case of SAA2 system utilization (see Fig. 13) is that the time horizon variations do not affect the results as much as the case of SAA1 system utilization. The second important difference of the SAA2 algorithm is that increasing the distance horizon consistently improves the safety (Fig. 13d), unlike the case of SAA1 algorithm utilization, where larger distance horizon values do not make a major effect on safety. The reason for this difference can be explained by comparing Figs. 12b and 13b: SAA1 causes the UAS deviate from its trajectory significantly more than the SAA2, and thus separation violation numbers for the SAA1 do not improve further after a point due to a significant impact on the surrounding traffic. However, it should be noted that, in general, the violation numbers of the SAA1 are lower than that of the SAA2 (see Figs. 12d–13d). After this quantitative analysis, it can be said that the SAA1 system results in a safer flight (less number of violations), whereas the SAA2 system provides a higher-performance flight (lower deviations from the trajectory).

It is important to note that, when the technologies and procedures mature enough to enable full integration of UAS into the NAS, it would not be unrealistic to expect that the ratio of unmanned to manned aircraft will increase dramatically. Since the HAS is a complex system where several intelligent agents move simultaneously, it is impossible to predict the effects of increased UAS presence. Therefore, it is important and useful to investigate the response of the overall system to the increased number of UASs. Figure 14 shows the effect of increasing the number of UASs in HAS. It is seen that, as the number of UASs increases, trajectory deviations, flight times, and separation violations increase. It is noted that no mode/phase changes are observed in the system.



a) Manned aircraft trajectory deviation

b) UAS trajectory deviation

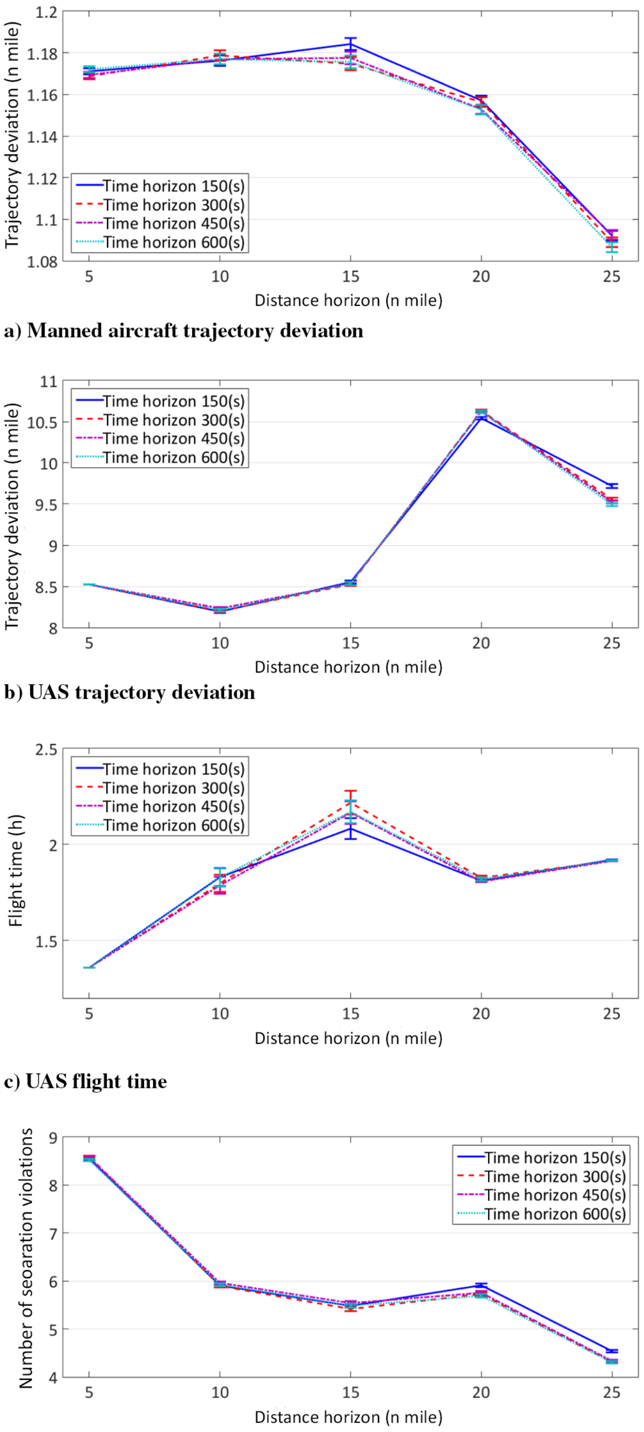
c) UAS flight time

d) Number of separation violations

Fig. 12 Safety vs performance in HAS when the SAA1 is employed.

3. Separation Responsibility Analysis

Another issue to be addressed that is important in studying the integration of UAS into NAS is the separation responsibility [19]: it is crucial to determine which of the agents (manned aircraft or UAS) will take the responsibility of conflict resolution. Figure 15 depicts a



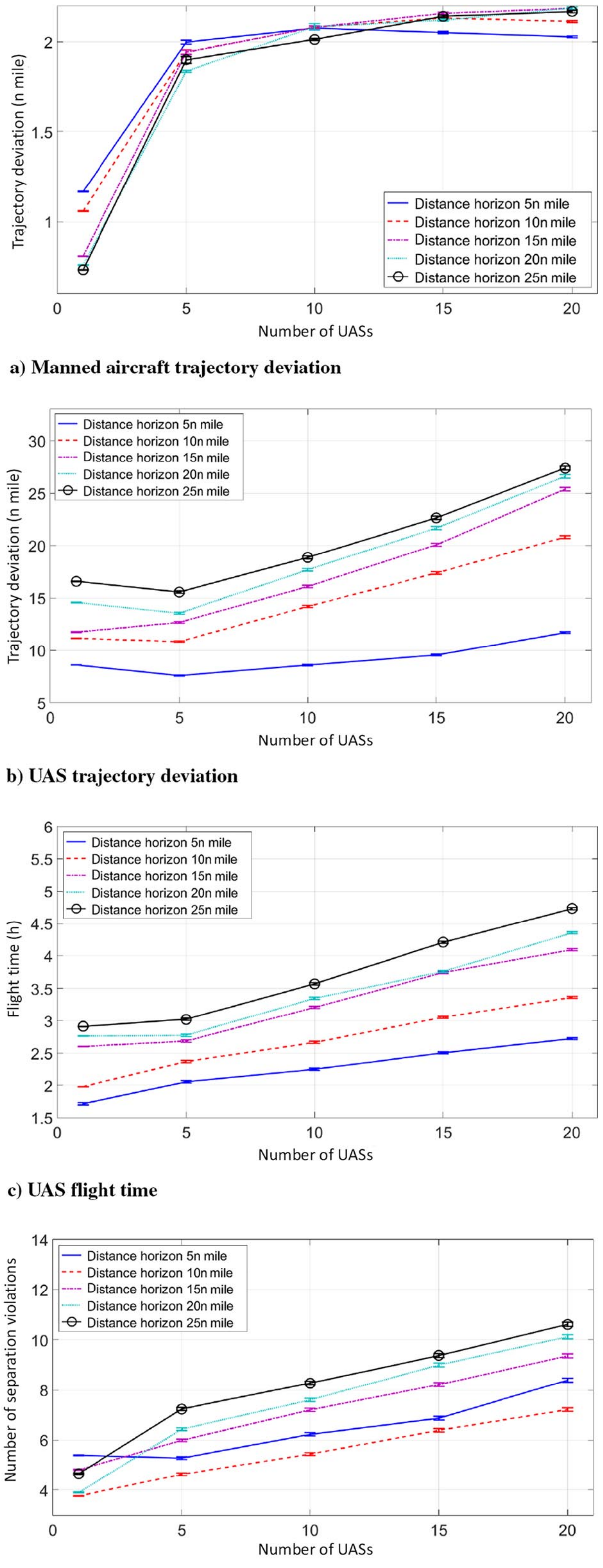
a) Manned aircraft trajectory deviation

b) UAS trajectory deviation

c) UAS flight time

d) Number of separation violations

Fig. 13 Safety vs performance in HAS when the SAA2 is employed.



a) Manned aircraft trajectory deviation

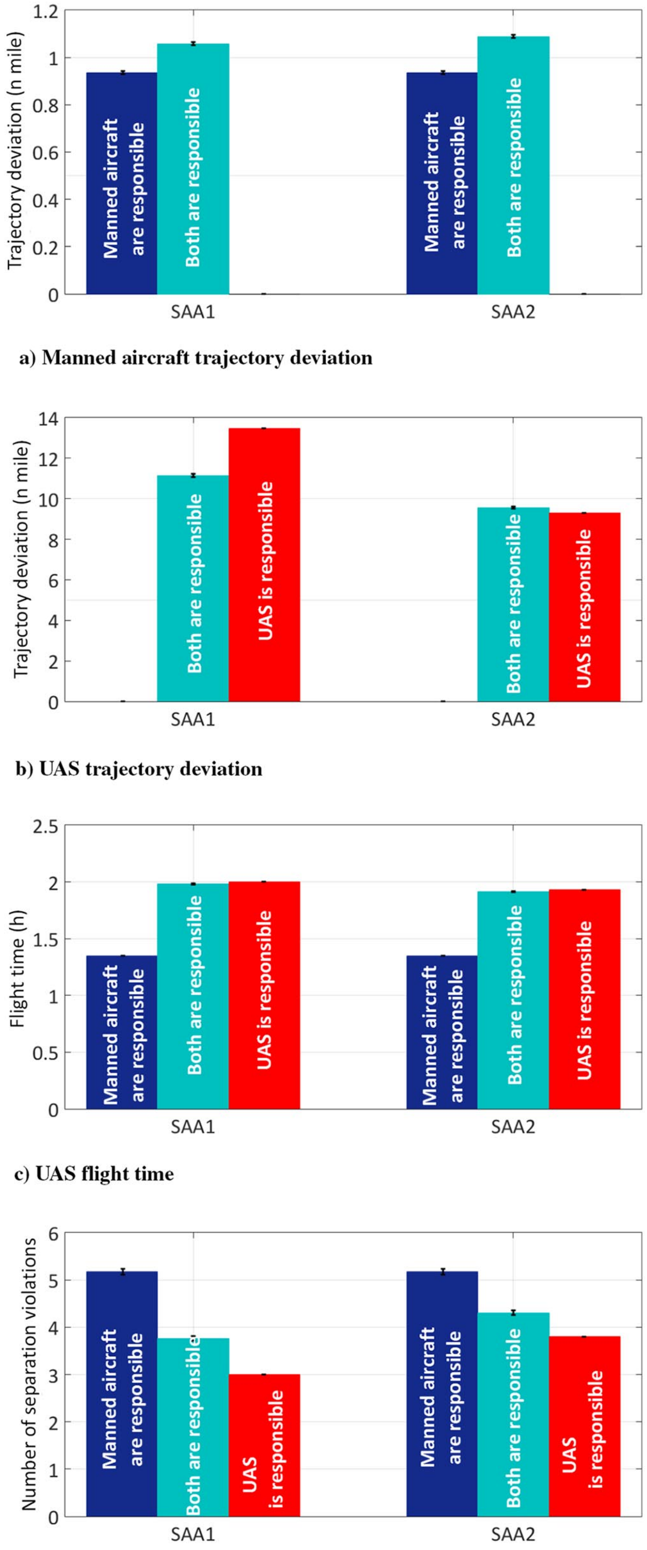
b) UAS trajectory deviation

c) UAS flight time

d) Number of separation violations

Fig. 14 Safety vs performance in HAS when the SAA1 is employed in a multi-UAS scenario.

comparison of different resolution responsibility cases: manned aircraft are responsible, both manned aircraft and the UAS are responsible, and only the UAS is responsible. In the case when only manned aircraft are responsible for conflict resolution, the UAS is forced to continue its path without executing the SAA system and the manned aircraft act as level-1 and level-2 DMs. In the case when the UAS is responsible for the conflict resolution, the manned aircraft are forced to continue their path without changing their heading, and the UAS executes its SAA system. In the case when both the manned aircraft and the UAS are responsible for the conflict resolution, they both execute their evasive maneuvers. Figure 15a shows that manned aircraft deviate more from their trajectory when both the UAS and the manned aircraft share resolution responsibility, as compared to the case when only the manned aircraft are responsible. This is true for both the SAA1 (the results on the left) and the SAA2 (the results on the right) algorithms. The reason for increased trajectory deviation for the manned aircraft in the case of shared responsibility is that the pilots' assumptions about possible UAS actions are not always correct, which forces the pilots to make additional adjustments in their



a) Manned aircraft trajectory deviation

b) UAS trajectory deviation

c) UAS flight time

d) Number of separation violations

Fig. 15 SAA1, SAA2, and safety vs performance in HAS.

trajectory, which in turn increases manned aircraft trajectory deviations.

On the other hand, Fig. 15b shows that the UAS deviates from its trajectory more when it is responsible for the resolution, as compared to the case when the responsibility is shared, when the SAA1 is used. For the case of SAA2 utilization, deviations are less and do not change much based on the responsibility assignments. Figure 15c shows, as expected, that for both the SAA1 and SAA2, the UAS flight times are the shortest when only the manned aircraft become responsible for the resolution. Perhaps the most important result is given in Fig. 15d, where it is shown that, for both SAA1 and SAA2 utilizations, the safest case is when the resolution responsibility is given to the UAS.

V. Conclusions

In this paper, a game theoretical modeling framework is proposed for use in the integration of unmanned aircraft systems into the National Airspace System as a means for concept evaluations. The method provides probabilistic outcomes of complex scenarios where both manned and unmanned aircraft coexist. Thus, by providing quantitative analyses, the proposed framework proves itself to be useful in investigating the effect of various system variables, such as separation distances and the utilization of different sense-and-avoid algorithms, on the safety and performance of the airspace system. The method can also be used to analyze the effect of responsibility assignment for conflict resolution, between manned and unmanned aircraft. The proposed framework is flexible so that any rules and procedures that the pilots are required to follow (for example, traffic control alert systems advisories) can be incorporated into the model.

Acknowledgments

This effort was sponsored by the Scientific and Technological Research Council of Turkey under grant number 11E282. The authors would like to thank to Kalmanje Krishnakumar of NASA Ames Research Center and Adrian Agogino of the University of California, Santa Cruz, whose valuable feedback helped to improve the quality of the paper.

References

[1] Dalamagkidis, V. K. P., and Piegl, D. L. A., "On Unmanned Aircraft Systems Issues, Challenges and Operational Restrictions Preventing Integration into National Airspace System," *Progress in Aerospace Sciences*, Vol. 44, No. 7, 2008, pp. 503–519. doi:10.1016/j.paerosci.2008.08.001

[2] "Integration of Civil Unmanned Aircraft System (UAS) in the National Airspace System (NAS) Roadmap," U.S. Dept. of Transportation, Federal Aviation Administration, Washington, D.C., 2013, https://www.faa.gov/uas/media/UAS_Roadmap_2013.pdf [retrieved 07 July 2016].

[3] "Roadmap for the Integration of Civil Remotely-Piloted Aircraft Systems into the European Aviation System," European Commission, European RPAS Steering Group Final, Brussels, Belgium, 2013, http://ec.europa.eu/DocsRoom/documents/10484/attachments/3/translations/en/renditions/native [retrieved 07 July 2016].

[4] DeGarmo, M. T., "Issues Concerning Integration of Unmanned Aerial Vehicles in Civil Airspace," MITRE Corp., Center for Advanced Aviation System Development TR MP 04W000323, McLean VA, 2004, https://www.mitre.org/sites/default/files/pdf/04_1232.pdf [retrieved 07 July 2016].

[5] Salas, E., and Maurino, D., *Human Factors in Aviation*, 2nd ed., Elsevier, New York, 2010, pp. 75–79.

[6] Lee, R., and Wolpert, D., "Chapter: Game Theoretic Modeling of Pilot Behavior During Mid-Air Encounters," *Decision Making with Multiple Imperfect Decision Makers*, Intelligent Systems Reference Library Series, Springer, New York, 2011, pp. 88–89. doi:10.1007/978-3-642-24647-0-4

[7] Kuchar, J. E., and Drumm, A. C., "The Traffic Alert and Collision Avoidance System," *Lincoln Laboratory Journal*, Vol. 16, No. 2, 2007, pp. 277–296.

[8] Maki, D., Parry, C., Noth, K., Molinario, M., and Miraflor, R., "Dynamic Protection Zone Alerting and Pilot Maneuver Logic for Ground Based Sense and Avoid of Unmanned Aircraft Systems," *Infotech@Aerospace*, AIAA Paper 2012-2505, 2012, pp. 1–10. doi:10.2514/6.2012-2505

[9] Kochenderfer, M. J., Espindle, L. P., Kuchar, J. K., and Griffith, J. D., "Correlated Encounter Model for Cooperative Aircraft in the National Airspace System Version 1.0," Lincoln Lab., Massachusetts Inst. of Technology ATC-344, Cambridge, MA, Oct. 2008.

[10] Kuchar, J. K., Andrews, J., Drumm, T. H., Heinz, V., Thompson, S., and Welch, J., "A Safety Analysis Process for the Traffic Alert and Collision Avoidance System (TCAS) and See-and-Avoid Systems on Remotely Piloted Vehicles," *AIAA 3rd Unmanned Unlimited Technical*

Conference, Workshop and Exhibit, AIAA Paper 2004-6423, 2004, pp. 1–13.
doi:10.2514/6.2004-6423

[11] Perez-Batlle, M., Pastor, E., Royo, P., Prats, X., and Barrado, C., "A Taxonomy of UAS Separation Maneuvers and Their Automated Execution," *Proceedings of the 2nd International Conference on Application and Theory of Automation in Command and Control Systems*, IRIT Press, London, May 2012, pp. 1–11.

[12] Florent, M., Schultz, R. R., and Wang, Z., "Unmanned Aircraft Systems Sense and Avoid Flight Testing Utilizing ads-b Transceiver," *Infotech@ Aerospace*, AIAA Paper 2010-3441, 2010, pp. 1–8.
doi:10.2514/6.2010-3441

[13] Billingsley, T. B., "Safety Analysis of TCAS on Global Hawk Using Airspace Encounter Models," Ph.D. Dissertation, Massachusetts Inst. of Technology, Cambridge, MA, 2006.

[14] "Remotely Piloted Aircraft Systems," European Defense Agency Fact Sheet, Brussels, Belgium, Jan. 2015, https://www.eda.europa.eu/docs/default-source/eda-factsheets/2015-01-30-factsheet_rpas_high, [retrieved 07 July 2016].

[15] Alfredson, J., Hagstrom, P., and Sundqvist, B. G., "Situation Awareness for Mid-Air Detect-and-Avoid System for Remotely Piloted Aircraft," *Procedia Manufacturing*, Vol. 3, Dec. 2015, pp. 1014–1021.
doi:10.1016/j.promfg.2015.07.161

[16] Alfredson, J., Hagstrom, P., Sundqvist, B. G., and Pellebergs, J., "Workload of a Collision Avoidance System for Remotely Piloted Aircraft," *Nordic Ergonomics Society Annual Conferrence*, Copenhagen, Denmark, Aug. 2014, pp. 819–824.
doi:10.4122/dtu:2496

[17] Costa-Gomez, M. A., Craford, V. P., and Irriberri, N., "Comparing Models of Strategic Thinking in Van Huyck, Battalio, and Beil's Coordination Games," *Games and Economic Behavior*, Vol. 7, Nos. 2–3, 2009, pp. 365–376.
doi:10.1162/JEEA.2009.7.2-3.365

[18] Stahl, D., and Wilson, P., "On Players Models of Other Players: Theory and Experimental Evidence," *Games and Economic Behavior*, Vol. 10, No. 1, 1995, pp. 218–254.
doi:10.1006/game.1995.1031

[19] "Unmanned Aircraft Systems (UAS) Integration in the National Airspace System (NAS) Project," NASA Advisory Council Aeronautics Committee, UAS Subcommittee DFRC-E-DAA-TN5477, June 2012.

[20] Backhaus, S., Bent, R., Bono, J., Lee, R., Tracey, B., Wolpert, D., Xie, D., and Yildiz, Y., "Cyber-Physical Security: A Game Theory Model of Humans Interacting over Control Systems," *IEEE Transactions on Smart Grid*, Vol. 4, No. 4, 2013, pp. 2320–2327.

[21] Yildiz, Y., Lee, R., and Brat, G., "Using Game Theoretic Models to Predict Pilot Behavior in Nextgen Merging and Landing Scenario," *AIAA Modeling and Simulation Technologies Conference*, AIAA Paper 2012-4487, 2012, pp. 1–8.
doi:10.2514/6.2012-4487

[22] Lee, R., Wolpert, D., Bono, J., Backhaus, S., Bent, R., and Tracy, B., "Counter-Factual Reinforcement Learning: How to Model Decision-Makers and Anticipate the Future," *Decision Making and Imperfection*, Vol. 474, No. 4, 2013, pp. 101–128.
doi:10.1007/978-3-642-36406-8-4

[23] Yildiz, Y., Agogino, A., and Brat, G., "Predicting Pilot Behavior in Medium-Scale Scenarios Using Game Theory and Reinforcement Learning," *Journal of Guidance, Control, and Dynamics*, Vol. 37, No. 4, 2014, pp. 1335–1343.
doi:10.2514/1.G000176

[24] Sutton, R. S., and Barto, A. G., *Reinforcement Learning: An Introduction*, MIT Press, Cambridge, MA, 1998, p. 61.

[25] Jaakkola, T., Satinder, P. S., and Jordan, I., "Reinforcement Learning Algorithm for Partially Observable Markov Decision Problems," *Proceedings of the Advances in Neural Information Processing Systems*, Denver, Colorado, Nov. 1995, pp. 345–352.

[26] Wolfe, R., "NASA ERAST Non-Cooperative DSA Flight Test," *Proceedings of the AUVSI Unmanned Systems Conference*, Baltimore, MD, July 2003, pp. 1–11.

[27] Concept of Operations for the Next Generation Air Transportation System, Version 2.0, Federal Aviation Administration, Joint Planning and Development Office, 13 June 2007, https://info.aiaa.org/tac/AASG/ACOTC/Shared%20Documents/NextGen_v2.0.pdf [retrieved 07 July 2016].

[28] "Pilot's Handbook of Aeronautical Knowledge," U.S. Dept. of Transportation, Federal Aviation Administration STD " FAA-H-8083-25B," 2016, http://www.faa.gov/regulations_policies/handbooks_manuals/aviation/phak/media/pilot_handbook.pdf [retrieved 07 July 2016].

[29] Dalamagkidis, K., Valavanis, K. P., and Piegl, D. L. A., *On Integrating Unmanned Aircraft Systems into the National Airspace System*, Vol. 54, Springer Science and Business Media, New York, 2011, p. 125.
doi:10.1007/978-94-007-2479-2

[30] Mujumdar, A., and Padhi, R., "Reactive Collision Avoidance Using Nonlinear Geometric and Differential Geometric Guidance," *Journal of Guidance, Control, and Dynamics*, Vol. 34, No. 1, 2011, pp. 303–310.
doi:10.2514/1.50923

[31] Fassano, G., Accardo, D., and Moccia, A., "Multi-Sensor-Based Fully Autonomous Non-Cooperative Collision Avoidance System for Unmanned Air Vehicles," *Journal of Aerospace Computing, Information, and Communication*, Vol. 5, No. 10, Oct. 2008, pp. 338–360.
doi:10.2514/1.35145

[32] "Systems Engineering Guide: Verification and Validation of Simulation Models," MITRE Corp., McLean VA, https://www.mitre.org/publications/systems-engineering-guide/se-lifecycle-building-blocks/other-se-lifecycle-building-blocks-articles/verification-and-validation-of-simulation-models [retrieved 07 July 2016].

[33] Law, A. M., "How To Build Valid And Credible Simulation Models," *Proceedings of the Winter Simulation Conference*, Miami, Florida, Dec. 2008, pp. 39–47.