JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 37, No. 4, July–August 2014

Engineering Notes

Predicting Pilot Behavior in Medium-Scale Scenarios Using Game Theory and Reinforcement Learning

Yildiray Yildiz* and Adrian Agogino†
University of California, Santa Cruz, Moffett Field, California 95035
and
Guillaume Brat‡
Carnegie–Mellon University, Moffett Field, California 95035

DOI: 10.2514/1.G000176

I. Introduction

A KEY element to meet the continuing growth in air traffic is the increased use of automation. Decision support systems, computer-based information acquisition, trajectory planning systems, high-level graphic display systems, and all advisory systems are considered to be automation components related to next-generation (NextGen) air space [1]. In the NextGen air system, a larger number of interacting human and automation systems are expected as compared with today. Improved tools and methods are needed to analyze this new situation and predict potential conflicts or unexpected results, if any, due to increased human–human and human–automation interactions. In a recent NASA report [1], among others, human–automation function allocation, methods for transition of authority and responsibility as a function of operational concept, and transition from automation to human control are mentioned as "highest priority research needs" for NextGen air space development.

There have been several methods developed for modeling, optimizing, and making predictions in air space systems. Brahms agent modeling [2] framework has been successfully used to model human behavior but it is not used to predict possible outcomes of large-scale complex systems with human–human and human–automation interactions. For optimization, Tumer and Agogino [3] used agent-based learning to optimize air traffic flow, but they did not model pilot behavior, which is critical for being able to predict system outcomes.

Presented as Paper 2013-4908 at the AIAA Modeling and Simulation Technologies Conference, Boston, MA, 19–22 August 2013; received 24 July 2013; revision received 18 January 2014; accepted for publication 26 January 2014; published online 8 May 2014.

*Associate Scientist, University Affiliated Research Center, NASA Ames Research Center, Mail Stop 269-1; currently Assistant Professor, Mechanical Engineering Department, Bilkent University, Room EA 104, Ankara 06800, Turkey. Senior Member AIAA.

†Scientist, University Affiliated Research Center, NASA Ames Research Center, Mail Stop 269-1.

‡Technical Lead, Silicon Valley Campus, NASA Ames Research Center, Mail Stop 269-1. Senior Member AIAA.

In the proposed approach, the authors first mathematically define pilot goals in a complex system. These goals can constitute, for example, staying on the trajectory, not getting close to other aircraft, or having a smooth landing. The authors then use game theory and machine learning to model the outcomes of the overall system based on these pilot goals, together with other automation and environment variables.

Formally, the authors use of a game-theoretic framework known as semi network-form games (SNFGs) [4], to obtain probable outcomes of a NextGen scenario with interacting humans (pilots) in the presence of advanced NextGen technologies. Our focus is to show how this framework can be scaled to larger problems that will make it applicable to a wide range of air traffic systems. Earlier implementations of this framework [4–7] proved useful for investigating strategic decision making in scenarios with two humans. In this Note, for the first time, the authors investigate a dramatically larger scenario, which includes 50 aircraft corresponding to 50 human decision makers. The method presented in the Note is a step toward predicting the effect of new technologies and procedures on the air space system by investigating pilot reactions to the new medium. These predictions can be used to evaluate the performance vs efficiency tradeoffs.

In Sec. II, the employment of game theory is explained in predicting the complex system behavior. In this section, two components of the approach are also presented: level-K reasoning and reinforcement learning. In Sec. III, the main components of the investigated NextGen scenario are presented. In this section, the air space and aircraft models, pilot goals, and a general description of the scenario are explained. In Sec. IV, simulation setup details are provided. In Sec. V, the simulation results are shown, where four different variations of the NextGen scenario are investigated with different levels of complexity and congestion. Finally, in Sec. VI, the Note is concluded by giving a summary and takeaway notes of this study, together with future research directions.

II. Game-Theory-Based Prediction

Game theory is used to analyze strategic decision making among a group of "players." Typically, players represent human decision makers, though the concept of a player can be expanded to other decision makers including animals in evolutionary game theory or complex automated decision makers. In this Note, players are pilots. In the context of this Note, the key aspect of players is that they observe the environment, they take actions based on these observations, and the actions they take influence the environment and the other players. Figure 1 presents a graph representation of this process, where S_A^0 represents the initial states of the overall system, such as positions and velocities of aircraft, O_i^0 represents the observation bias and noise for each aircraft i at the initial stage, and P_i^0 represents the decision-making pilot i at the initial stage. When pilots make their observation and act, their aggregate action changes the state of the overall system to S_B^0. In the next stage, this process is repeated. The goal of game theory is to predict the actions of these players based on their goals. These goals are represented as "reward functions," which are some function of the system state. The authors assume that the players are trying (though imperfectly) to maximize their reward functions.

Given a set of goals represented as reward functions, the actions of the players may be predicted. However, several challenges need to be overcome. First, determining how a player can attempt to maximize their reward function can be a difficult inverse problem. Second, players may not be able to perfectly maximize their reward functions. Finally, the best action of a player will depend on the actions of all the

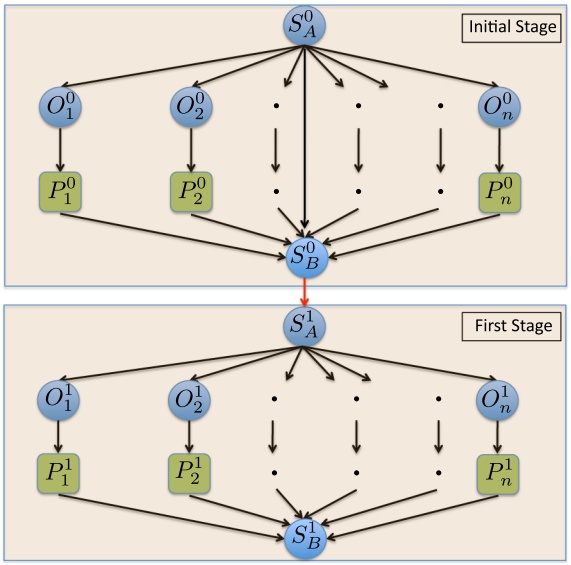


Fig. 1 Schematic representation of the NextGen scenario with *n* number of aircraft, as a multistage game.

other players. Multiple solutions may exist, and many solutions may be unstable.

The best ways of handling these issues heavily depend on the number of players, the size of the state space, the size of the action space, and the complexity of the reward functions. In this Note, the authors use a concept called "level-K" reasoning combined with reinforcement learning.

Our goal is to predict the behavior of a particular player; yet, how this player behaves depends on the behavior of other players. Level-K reasoning helps us address this problem through a hierarchical approach, which begins by assigning basic behaviors to every player. Then, given the reward function of a player, and basic behaviors of other players, player behavior is predicted. Reinforcement learning helps us to make these predictions in an iterative manner for games with multiple stages. This approach is explained in more detail next.

A. Level-K Reasoning

The basic idea in level-K reasoning [8,9] is that humans show different levels of reasoning in games. The lowest level, level-0 reasoning, is nonstrategic, meaning that a level-0 player does not take other players' possible moves into consideration. Level-0 strategies can be random or can be constructed using expert system knowledge. A level-1 player assumes that other players have level-0 reasoning and tries to maximize his/her reward function based on this assumption. Similarly, a level-2 player assumes that other players have level-1 reasoning, and so on. It is noted that, once a player makes a certain level assumption about the other players, other players simply become a part of the environment and the problem reduces to single agent decision making.

B. Reinforcement Learning

SNFG framework [6] extends the standard level-K reasoning to model time-extended scenarios. In a time-extended scenario with N steps, a player makes N action choices. Therefore, the player needs to optimize his/her policy (his map from observations/memory to actions) to maximize the average reward

$$\sum_{i=1}^{N}(r_i/N)$$

where r_i represents the reward at time step i. Reinforcement learning (RL) is a tool that is used to tweak player policies at each time step toward maximizing the reward without knowing the underlying model of the system. The RL algorithm takes system states as inputs

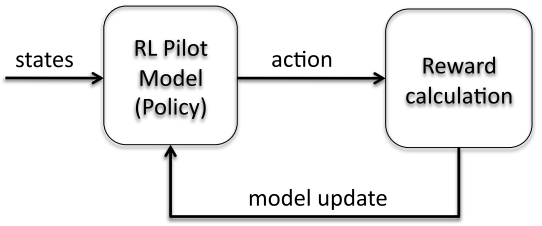


Fig. 2 Reinforcement learning schematic diagram.

and gives an appropriate action (agent move) as the output. When the actions are performed, the system states change. The reward is calculated based on these new states and the RL algorithm uses this reward to update the agent policy. In the next round, the updated policy is used to produce the next action given the new states (see Fig. 2). This process continues until the average reward converges to a certain value.

There are various reinforcement learning methods that can be used for this purpose [10]. In this Note, the authors use a method developed by Jaakkola et al. [11]. The reason for this choice is that the Jaakkola algorithm has local converge guarantees for scenarios where the player cannot observe all of the system states, which is the case for the scenario investigated in this Note. The details of the scenario are explained in the following sections.

III. Next-Generation Scenario Model

The authors tested the game-theoretic approach on an air traffic scenario, where 50 aircraft had to space themselves efficiently using automatic dependent surveillance-broadcast (ADS-B). ADS-B is a satellite-based technology that provides aircraft the ability to receive other aircraft identification, position, and velocity. This technology is expected to support NextGen air space operations, where the volume of operations is projected to be dramatically higher than what it is now. In the scenario, 50 aircraft are approaching to a single sector. (In the existing air space system, sector capacities are much lower, but it is expected that, to achieve NextGen air space goals, sector capacities will need to be increased dramatically.) Thanks to the ADS-B technology, pilots are aware of other aircraft, to a certain degree. Given this ADS-B information, pilots are supposed to continue flying on their assigned trajectory while at the same time protecting separation from other aircraft.

A. Air Space Model

Aircraft are assumed to be at the en route phase of the flight, flying level at the same altitude, throughout the scenario. Accordingly, the air space is approximated as a two-dimensional Cartesian grid.

B. Aircraft Model

Aircraft are assumed to be controlled by an automatic pilot in velocity control mode. This is approximated by allowing aircraft to move to a neighboring intersection in the grid, either diagonally or straight, at every time step.

C. Scenario Description

At time $t = t_0$, aircraft have their initial positions and directions p_0^i and d_0^i, $i = 1, 2, \ldots, 50$, where 50 is the number of aircraft in the scenario. Initial positions p_0^i are either randomly or with a certain structure assigned on the grid with the exclusion of a sector region in the center. Initial directions d_0^i are assigned in such a way that each aircraft aims toward the center of the sector. As an example for random initial position assignment, see Fig. 3. At time $t = t_0$, a goal position gp^i, which is where the aircraft is supposed to reach, is also assigned to each aircraft. This goal position gp is simply where the initial direction arrow intersects an edge of the grid.

At times $t = t_k$, $k = 1, 2, \ldots$, aircraft move toward the center of the sector and toward their goal position gp. Pilots observe surrounding aircraft and try to protect separation while following

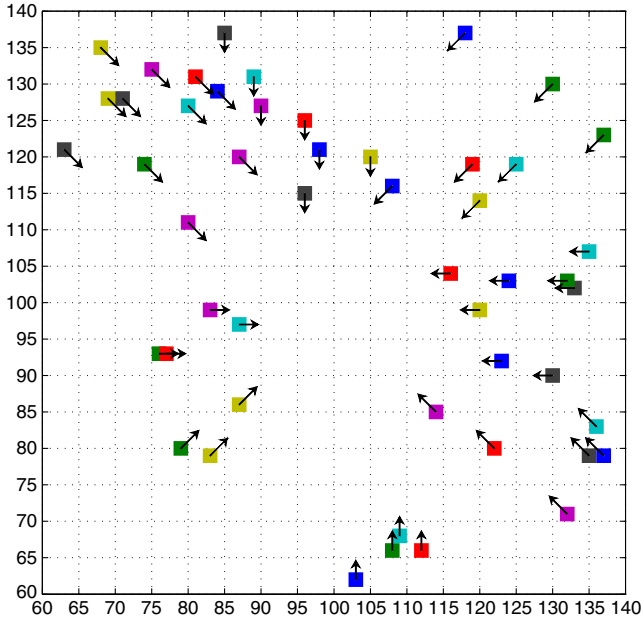


Fig. 3 Initial positions and directions of aircraft for the "randomly distributed" case.

their assigned trajectory. The assigned trajectory is a straight line from the initial position p_0 to the goal position gp.

D. Pilot Reward Function

The pilot's reward function, or "goal function" U_i, is a mathematical representation of the preferences of the pilot about different states of the system. For the investigated scenario, it is assumed that the following factors play a role in pilot decisions.

1. *Preventing a Separation Violation*

The most important task for the pilots is to keep a safe distance from other aircraft. A separation violation is modeled as two or more aircraft sharing the same intersection in the grid. Therefore, the first term of the reward function is formed as

$$u_1 = -N_{\text{violation}} \tag{1}$$

where $N_{\text{violation}}$, the number of separation violations, represents the number of aircraft existing in the same intersection with the considered aircraft. The minus sign reveals that this term needs to be minimized to maximize the overall reward function.

2. *Decreasing the Probability of a Separation Violation*

The pilots' second important task is keeping the aircraft at a safe distance from other aircraft and therefore decreasing the probability of a separation violation. The aircraft that are at the neighboring intersections of the aircraft in consideration are assumed to be at a "nonsafe" distance and hence increase the likelihood of a separation violation. The pilots' goal is to minimize the number of these surrounding aircraft during flight. The second term, modeling this goal, is given as

$$u_2 = -N_{\text{neighbor}} \tag{2}$$

where N_{neighbor} stands for the number of neighboring aircraft.

3. *Staying on the Assigned Trajectory*

The pilots' third task is to stay at their assigned trajectories. This task is divided into two components. The first component is approaching to the final goal point. The second component is staying as close as possible to the assigned path. An aircraft can approach to its final destination without staying very close to the assigned path. Similarly, an aircraft can stay exactly on the assigned path and not approach the final destination, if, for example, it goes on the opposite direction. So, the mutual existence of these two subtasks are necessary.

The first task, getting close to the final destination, is modeled by an indicator function, which gets the value of one or zero, depending on whether after each step they are closer (1) or not (0) to their final destination in the grid. This is expressed as

$$u_{31} = I_{\text{close}} \tag{3}$$

where, I_{close} stands for the indicator function for getting close to the final destination.

The second subtask, staying on the assigned path, is modeled by the negative of the distance of the aircraft to the closest point on the assigned path. This is expressed as

$$u_{32} = D_{\text{path}} \tag{4}$$

where D_{path} stands for the distance to the assigned path.

4. *Minimizing Effort*

As human beings, pilots tend to choose inaction or the action that needs the least effort, if possible. This final term is modeled as

$$u_4 = -I_{\text{effort}} \tag{5}$$

where I_{effort} takes the value of one if pilots change aircraft heading and zero otherwise.

Combining the preceding components, the reward function U for a given pilot can be given as

$$\begin{aligned} U = {} & \omega_1(-N_{\text{violation}}) + \omega_2(-N_{\text{neighbor}}) \\ & + \omega_{31}(I_{\text{close}}) + \omega_{32}(D_{\text{path}}) + \omega_4(-I_{\text{effort}}) \end{aligned} \tag{6}$$

where ω_js are the weighting assigned to each component.

IV. Simulation Setup

To represent the air space, an 80×80 Cartesian grid is used. At time $t = t_0$, 50 aircraft are distributed on this grid, either randomly or with a certain structure, excluding a central region, which represents a sector. Aircraft directions are assigned in such way that all aircraft head toward the sector. See Fig. 3 for a random initial distribution.

A. Pilot Move Space

In this model, pilots are assumed to have three actions: diagonal right, diagonal left, and straight.

B. Pilot Observations and Memory

ADS-B technology can provide pilots the information, position, velocity, etc. of other aircraft. However, a pilot has limited ability to use all this information for his/her decision making. For this scenario, the authors model these pilot limitations by assuming that pilots can observe a limited section of the grid in front of them. Pilot observations on the grid are presented in Fig. 4, where pilot A observes whether or not any aircraft is headed toward the regions that are marked by an "x" sign. In this particular example, another aircraft is heading toward one of these regions that is marked with a larger x sign. Therefore, pilot A will see this section on the grid as "full," whereas the rest of his observation space, the small x signs, will be "empty."

In addition to these ADS-B observations, pilots also know their configuration (i.e. diagonal or straight), their best directional move M_{BD}, and best trajectory move M_{BT}. M_{BD} is the move that would make the aircraft approach to its final destination more than any alternative move would. Similarly, M_{BT} is the move that would make the aircraft approach to its trajectory more than any alternative move. Finally, pilots have a memory of what their actions were at the previous time step.

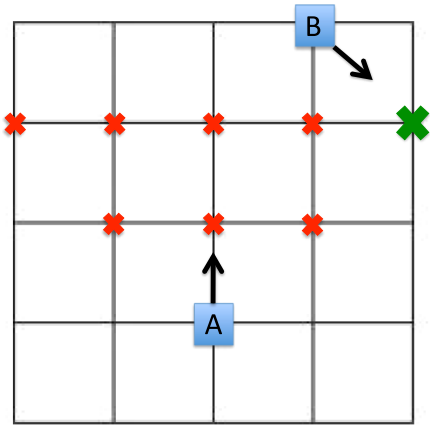


Fig. 4 Pilot observation space.

Eight ADS-B observations, one configuration, one M_{BD}, one M_{BT}, and one previous move make up 12 total inputs for the reinforcement learning algorithm. Observations and the configuration have binary values, one or zero. The previous move, M_{BD} and M_{BT} have three dimensions each: diagonal left, diagonal right, or straight. Therefore, the number of states for which the reinforcement learning algorithm need to assign appropriate actions is $2^9 \times 3^3 = 13,824$. During the RL process, a pilot observes these states, takes action based on his/her current policy (the mapping from observations to actions), observes the new states that are influenced by his/her actions, and updates his/her policy based on the effect of these new states on his/her reward function. This process continues until the policy converges.

Figure 5 shows a schematic diagram of RL pilot model inputs and outputs.

C. Level-0 Pilot

In general, level-0 players are modeled as uniformly random (i.e. they do not have any preference over any moves). However, depending on the application, this selection may vary. One important property of level-0 players is that they need to be nonstrategic: Their actions should be independent of other players' actions. In this scenario, the authors modeled level-0 players as pilots that fly with a predetermined fixed heading, regardless of other pilots' positions or intents.

V. Simulation Results

In this section, four safety-related scenarios are investigated to show the predictive capabilities of the proposed approach. In these scenarios, the authors explore safety issues such as loss of separation and deviations from the assigned trajectories, together with pilot performances via "average rewards" pilots obtain during their flight. The authors also make predictions on how high-density air traffic affect these issues.

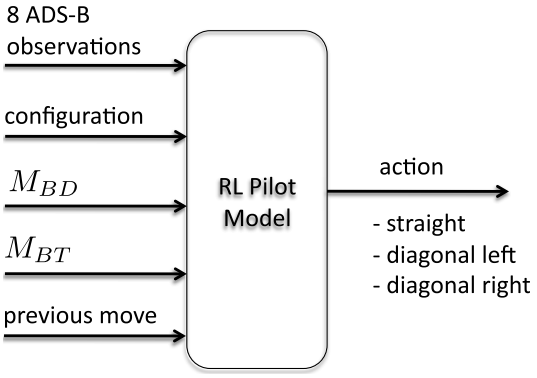


Fig. 5 Reinforcement learning pilot model inputs and output.

The authors first use RL to obtain level-1 and level-2 pilot policies, which are mappings from system states to actions. RL training simulations are conducted by initializing the player's policy with a uniform random distribution over actions and then running the RL algorithm, which tweaks the player policy in certain episodes to increase the average reward. These runs are stopped when the average reward converges to a fixed value. When a level-1 pilot is being trained, level-0 behavior is assigned to the remaining pilots. Similarly, when a level-2 pilot is being trained, level-1 behavior is assigned to the remaining pilots.

After level-1 and level-2 pilot policies are determined, the following scenarios are simulated to make system level predictions.

A. Scenario 1: Introducing Self-Navigating Aircraft to Air Space, Configuration 1

In this scenario, two sets of aircraft are flying in fixed trajectories toward a sector located at the center of the air space grid. Figure 6 shows the initial positions of the aircraft together with their heading. The aircraft are located in such a way that there is no danger of separation violation if aircraft follow the assigned trajectories, which are straight lines, perfectly. It is reminded that a separation violation is modeled as two or more aircraft sharing the same grid intersection. Level-0 pilots are defined as pilots flying with a predetermined fixed heading, regardless of the surrounding aircraft presence. In this scenario, the authors start with assigning level-0 behavior to all pilots and then replace these pilots, in increasing numbers, with level-1 pilots.

Figure 7 presents the evolution of this scenario, when all pilots are level 0. As expected, the aircraft follow perfect fixed trajectories and no separation violation event occurs. This is not an interesting result, because the outcome was already known: The pilots were given trajectories and spaced such that no safety violations would occur. The real question is "what happens if these perfectly spaced pilots are replaced with self-navigating pilots?" It is noted that the outcome of this may be unpredictable because the original solution is somewhat brittle. As explained earlier, self-navigating pilots have ADS-B technology onboard and they can observe their surroundings, as depicted in Fig. 4. In this scenario, self-navigating aircraft pilots are modeled as level-1 strategic thinkers: They assume that other pilots are level 0 and then they try to choose optimal actions that will maximize their reward functions. Pilot reward function was explained in Sec. III.D.

The system is simulated after replacing a various number of fixed trajectory aircraft with self-navigating aircraft. Figure 8 shows the effects of this newly introduced ADS-B-equipped aircraft into the

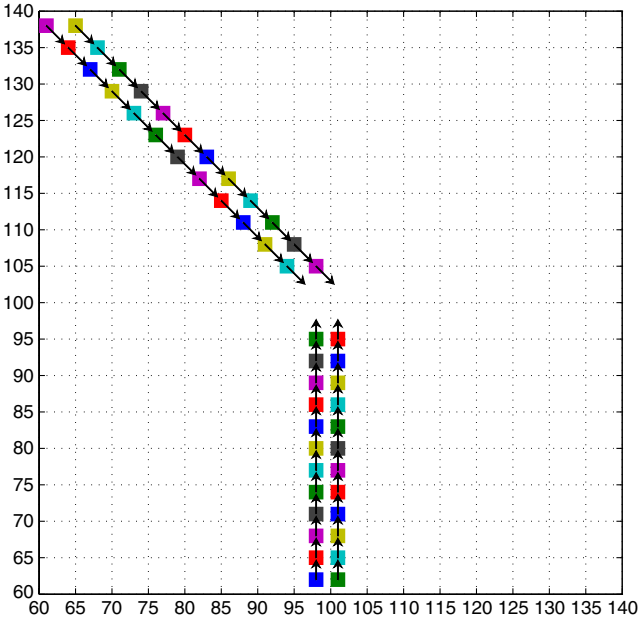
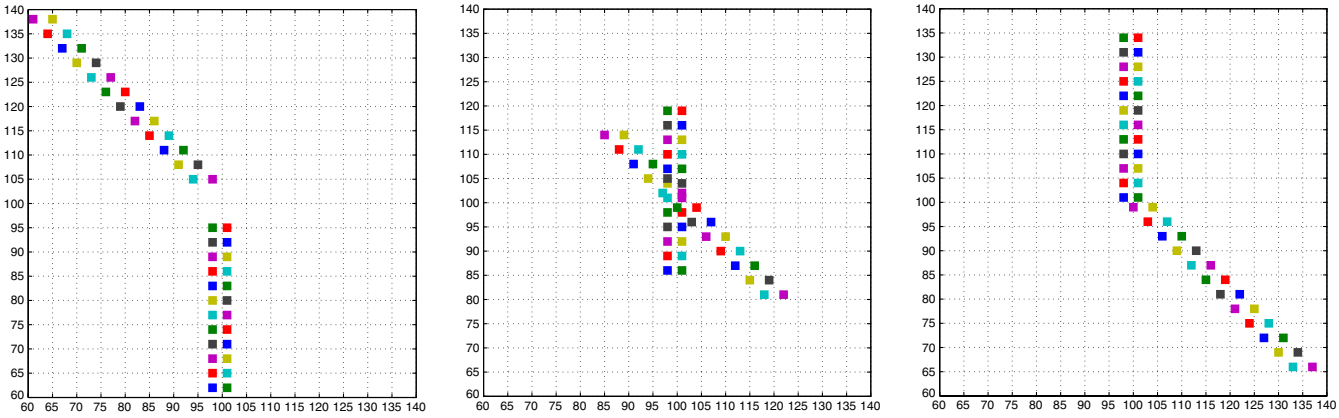
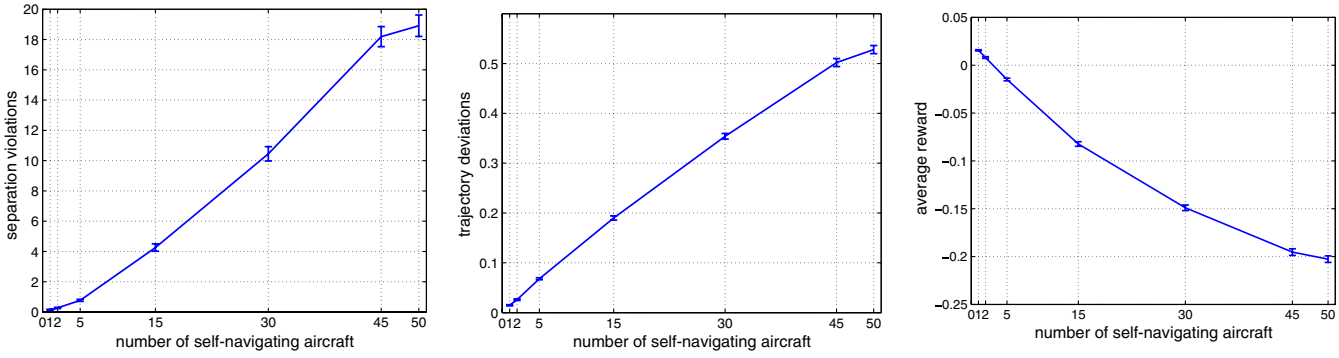


Fig. 6 Initial positions and headings of the aircraft for scenario 1.



a) Initial positions b) Positions at sim step 24 c) Positions at sim step 39

Fig. 7 Evolution of scenario 1 when all aircraft have fixed paths with constant heading.



a) Separation violations b) Trajectory deviations c) Average rewards

Fig. 8 Effect of introducing self-navigating aircraft into a perfectly structured air space (configuration 1).

system, in increasing numbers. Three variables are investigated: 1) separation violations, which are represented by two or more aircraft sharing the same grid; 2) trajectory deviations, which are the average distance, in terms of unit grid length, of the aircraft to their respective assigned trajectories, averaged over all aircraft; and 3) average rewards, which are the average value of the reward functions averaged over all aircraft. It is seen that, as the number of self-navigating aircraft in the system increases, the number of separation violations and trajectory deviations increases, as expected. As a consequence, the average pilot reward decreases. It is noted that the original scenario is a special one where the aircraft trajectories are very close to each other and, therefore, to prevent separation violations, these trajectories are very carefully assigned. Under these circumstances, self-navigating pilots' assignments are very challenging: The system is brittle; there is no room for even small deviations from the trajectories. On the other hand, self-navigating pilots cannot observe the whole air space and they do not get any guidance from the ground. They operate only with the observations they obtain from ADS-B.

The quantitative analysis so far may suggest that the introduction of self-navigating pilots in a tightly spaced air space without ground control holds serious risks. It also shows that, in general, the speed of increase in separation violations increases as the number of self-navigating aircraft increases, whereas the speed of increase in trajectory deviations decreases. This may reveal that unpredictable aircraft behavior may be less of a concern compared with separation violations.

B. Scenario 2: Introducing Self-Navigating Aircraft to Air Space, Configuration 2

This scenario is similar to the first one: Fixed trajectory aircraft are replaced with self-navigating aircraft in an air space scenario and the effects on separation violations, trajectory deviations, and average pilot rewards are observed. However, a different flying configuration is used in this scenario, which is shown in Fig. 9. This configuration is more brittle than the first one because there is less free space around the aircraft, on average. Figure 10 presents the evolution of this scenario when all aircraft have fixed trajectories (level-0 pilot). As in the previous scenario, when there is no self-navigating aircraft, there occurs no separation violations, by design.

Figure 11 presents a comparison between the effects of replacing the fixed trajectory aircraft with self-navigating aircraft in

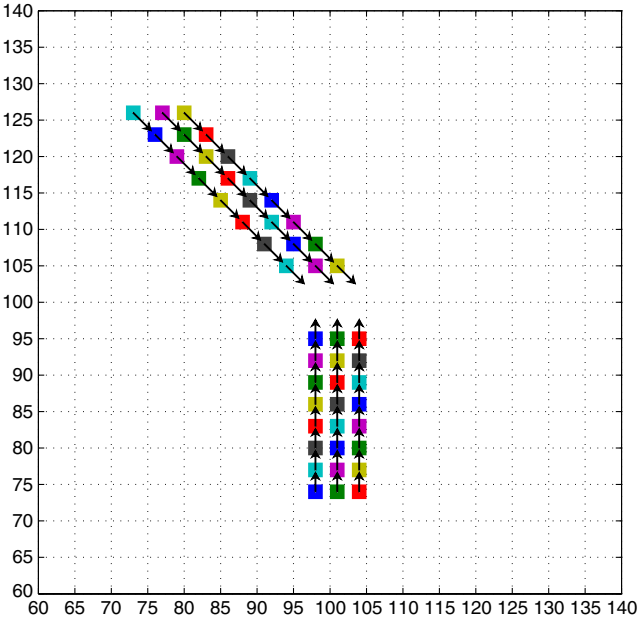
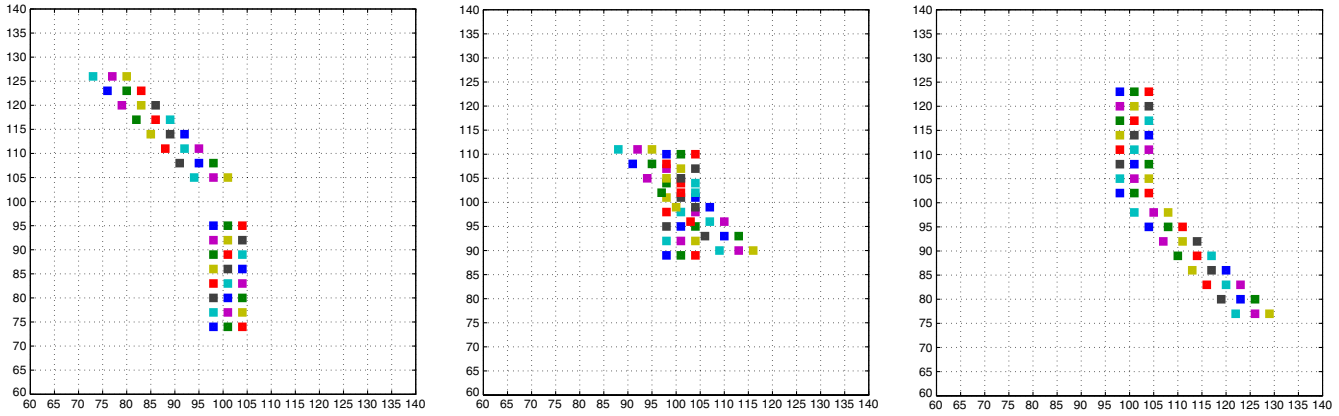


Fig. 9 Initial positions and headings of the aircraft for scenario 2.



a) Initial positions b) Positions at sim step 15 c) Positions at sim step 28

Fig. 10 Evolution of scenario 2 when all aircraft have fixed paths with constant heading.

configurations 1 and 2, in terms of separation violations, trajectory deviations, and average rewards. Because the second configuration is more brittle, the number of separation violations and trajectory deviations are larger, which translates to lower average rewards.

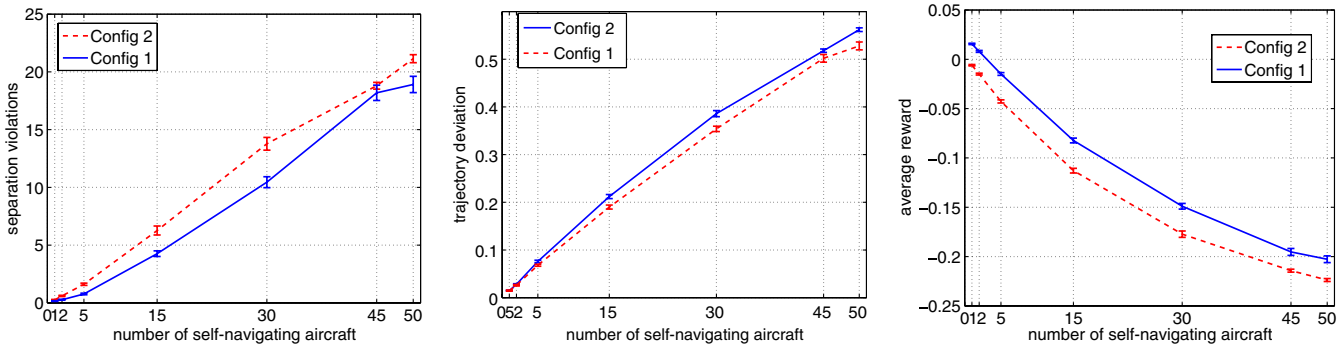
The quantitative analysis of the effect of brittle trajectories on safety and efficiency may be useful for future design of aircraft routes. For example, although configuration 2 causes more separation violations, in general, it may be more efficient to design the trajectories as such due to some other considerations. This quantitative analysis may help find a "sweet spot" or a balance between brittleness of the system and efficiency, which will result in a safe and efficient, in terms of throughput, for example, air space.

C. Scenario 3: Introducing Self-Navigating Aircraft, with a Different ADS-B Setting, to Air Space

In the previous two scenarios, the effect of introducing ADS-B-equipped self-navigating aircraft to air space was investigated. It was assumed that the ADS-B data link provided these pilots the positions of nearby aircraft. Their observation space was given in Fig. 4. In this scenario, it is assumed that the set of observations that a pilot can use is smaller: Self-navigating pilots can only use the observations at three points in front of them (straight ahead, diagonal right, and diagonal left grid points). This may correspond to a different ADS-B setting that gives more limited information to the pilot, or to pilots not being able to handle more information.

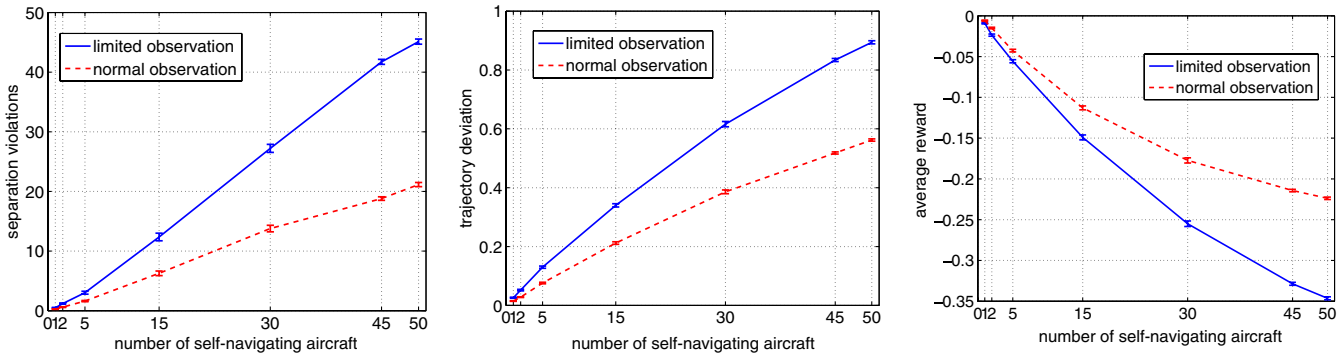
Figure 12 shows the effects of having an ADS-B system that provides less information about surrounding aircraft, by comparing the results with the previous scenario, where the pilots had a larger observation space. Number of separation violations, trajectory deviations, and average rewards are affected negatively, as expected. What is interesting is that the system deterioration is faster than linear with the increase in the number of self-navigating aircraft.

The results of this investigation may give clues on the amount of information that is needed to be provided to the pilots with



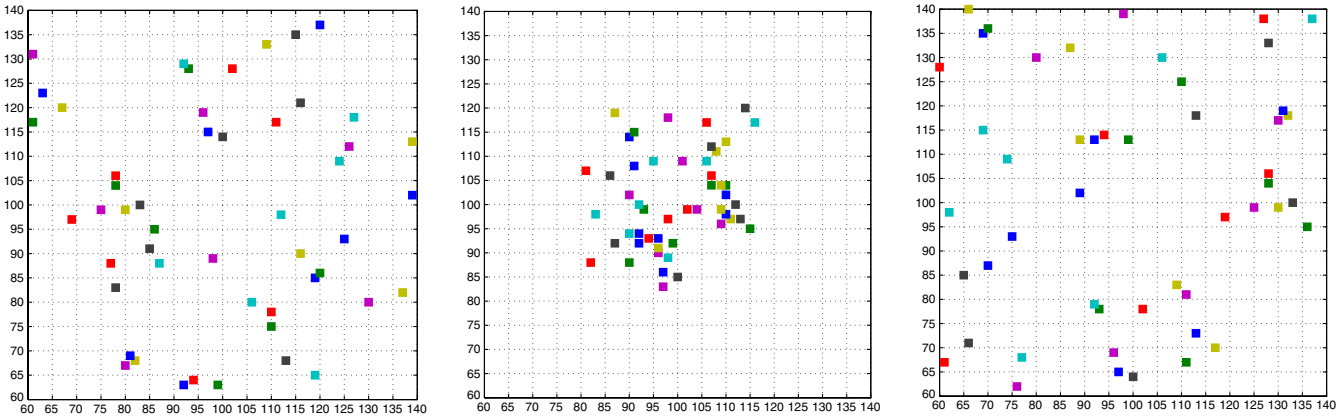
a) Separation violations b) Trajectory deviations c) Average rewards

Fig. 11 Comparison of effects of introducing self-navigating aircraft for two different flying configurations.



a) Separation violations b) Trajectory deviations c) Average rewards

Fig. 12 Comparison of the effects of introducing self-navigating aircraft to air space for two different ADS-B settings.



a) Initial positions b) Positions at sim step 29 c) Positions at sim step 50

Fig. 13 Evolution of scenario 4 when all aircraft have fixed paths with constant heading.

ADS-B-equipped aircraft. It goes without saying that the results presented here are obtained from simulating a simplified scenario to show the capabilities of the game-theoretic approach. In real applications, the assumptions and simplifications should be carefully tailored depending on the complexity of the problem.

D. Scenario 4: Changing Air Space Density

In this scenario, the authors investigate how air space density makes a quantitative effect on system safety. The scenario begins with aircraft randomly initialized in the air space with assigned trajectories that will make them fly on straight lines toward a sector located in the middle of the air space grid and continue flying straight until they reach the boundaries of the grid. Figure 3 shows an example initialization with 50 aircraft. Figure 13 shows the evolution of this scenario when 50 aircraft are used with level-0 pilots. It is reminded that, by design, level-0 pilots never change their initial heading and fly on straight line trajectories.

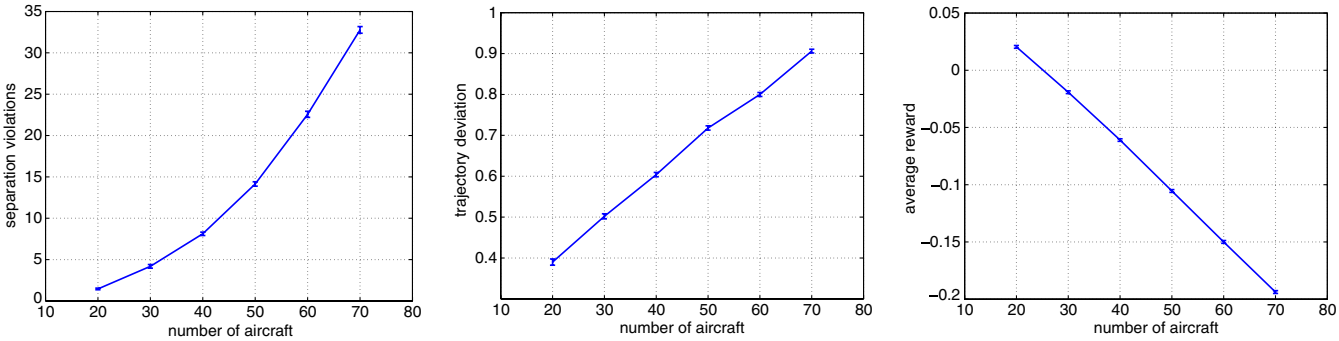
To make the scenario more realistic, the authors used a mixture of pilot types: level-0, level-1, and level-2. Some experimental studies [12] show that, in general, the level-0 type has minimum frequency and level-1 types are more frequent than level-2 types. Level-3 types, which assume that their opponents are level-2, are rare. This is intuitive because the amount of reasoning gets unreasonably taxing for humans as levels increase. These types of distributions are regarded as behavior parameters. Existing data or previous analysis can be used for estimating type distributions. For simulations, the following type distributions are used: 10% level 0, 60% level 1, and 30% level 2.

Figure 14 presents simulation results where the effect of air space density variations on separation violations, trajectory deviations, and average rewards is quantitatively investigated. As expected, as the air density increases, all these variables are negatively affected. An interesting result is that, although trajectory deviation and average reward varies almost linearly with air space density, separation violations shows an almost exponential increase. These quantitative estimation analyses may serve as a useful tool for designing NextGen air space structure, where dramatically increased air space densities are expected.

VI. Conclusions

In this Note, an implementation of a game-theoretic framework for the problem of predicting the effect of newly introduced technologies to the next generation air space, where human–automation interactions will play a crucial role in the performance and safety of the system, is presented. This framework is tested on a scenario where ADS-B information is being integrated into a 50 aircraft system, allowing some of the aircraft to self-navigate. Simulation results are provided that present the quantitative effect of introducing self-navigating aircraft into the air space on separation violations, trajectory deviations, and pilot performances. In addition, results are shown that present the effect of increasing air space density on these variables.

The simulation results show that introducing self-navigating pilots to an air space system where aircraft are initially spaced perfectly negatively affects investigated safety and performance variables. This is an expected result because each self-navigating pilot is a deviation from the original perfect configuration. Furthermore, as the brittleness of the original configuration increases, the negative effect on safety and performance variables increases. In addition, the simulation results also present the negative effects of congestion on these variables. What is more important, however, is that the proposed framework can provide quantitative estimates of how these variables are affected, which can be used to optimize the air space. Because the focus of this work is to show the predictive capabilities of the proposed approach for midscale air space scenarios, using simplified system models, the main conclusion is that the proposed approach scales well with the number of players (pilots). For future research is planned to investigate more complex integration tasks. These tasks



a) Separation violations b) Trajectory deviations c) Average rewards

Fig. 14 Effect of air space density variations.

will likely involve continuous variables, large-scale simulation, and modeling behavior at multiple resolutions of detail.

References

[1] Sheridan, T. B., Corker, K. M., and Nadler, E. D., "Final Report and Recommendations for Research on Human-Automation Interaction in the Next Generation Air Transportation System," U.S. Dept. of Transportation, Research and Innovative Technology Administration TR-DOT-VNTSC-NASA-06-05, 2006.

[2] Acquisti, A., Sierhuis, M., Clancey, W. J., and Bradshaw, J. M., "Agent Based Modeling of Collaboration and Work Practices Onboard the International Space Station," *Proceedings of the 11th Conference on Computer-Generated Forces and Behavior Representation*, Vol. 8, 2002, pp. 315–337.

[3] Tumer, K., and Agogino, A., "Distributed agent-Based Air Traffic Flow Management," *Proceedings Sixth International Joint Conference on Autonomous Agents and Multiagent Systems*, No. 255, 2007, pp. 342–349.

[4] 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. 75–111.

[5] Yildiz, Y., Lee, R., and Brat, G., "Using Game Theoretic Models to Predict Pilot Behavior in NextGen Merging and Landing Scenario," *Proceedings AIAA Modeling and Simulation Technologies Conference*, AIAA Paper 2012-4487, 2012.

[6] Lee, R., Wolpert, D. H., Bono, J., Backhaus, S., Bent, R., and Tracey, B., "Counter-Factual Reinforcement Learning: How to Model Decision-Makers that Anticipate the Future," *Decision Making and Imperfection*, Springer, New York, 2013.

[7] 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.

[8] 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

[9] Costa-Gomes, M., and Crawford, V., "Cognition and Behavior in Two-Person Guessing Games: An Experimental Study," *American Economic Review*, Vol. 96, No. 5, 2006, pp. 1737–1768. doi:10.1257/aer.96.5.1737

[10] Wiering, M., and van Otterlo, M. (eds.), *Reinforcement Learning, State-of-the-Art*, Springer, New York, 2012.

[11] Jaakkola, T., Satinder, P. S., and Jordan, I., "Reinforcement Learning Algorithm for Partially Observable Markov Decision Problems," *Advances in Neural Information Processing Systems 7: Proceedings of the 1994 Conference*, MIT Press, Cambridge, MA, 1994.

[12] Costa-Gomes, M. A., Crawford, V. P., and Iriberri, N., "Comparing Models of Strategic Thinking in Van Huyck, Battalio, and Beil's Coordination Games," *Games and Economic Behavior*, Vol. 7, Nos. 2–3, 1995, pp. 365–376.