

Initial Demonstration of the Real-Time Safety Monitoring Framework for the National Airspace System Using Flight Data

Indranil Roychoudhury*

SGT Inc., NASA Ames Research Center, Moffett Field, CA 94035, USA

Matthew Daigle* and Kai Goebel† and Lilly Spirkovska‡

NASA Ames Research Center, Moffett Field, CA 94035, USA

Shankar Sankararaman§ and John Ossenfort‡ and Chetan Kulkarni*

SGT Inc., NASA Ames Research Center, Moffett Field, CA 94035, USA

William McDermott‡ and Scott Poll‡

NASA Ames Research Center, Moffett Field, CA 94035, USA

As new operational paradigms and additional aircraft are being introduced into the National Airspace System (NAS), maintaining safety in such a rapidly growing environment becomes more challenging. It is therefore desirable to have an automated framework to provide an overview of the current safety of the airspace at different levels of granularity, as well an understanding of how the state of the safety will evolve into the future given the anticipated flight plans, weather forecast, predicted health of assets in the airspace, and so on. Towards this end, as part of our earlier work, we formulated the Real-Time Safety Monitoring (RTSM) framework for monitoring and predicting the state of safety and to predict unsafe events. In our previous work, the RTSM framework was demonstrated in simulation on three different constructed scenarios. In this paper, we further develop the framework and demonstrate it on real flight data from multiple data sources. Specifically, the flight data is obtained through the Shadow Mode Assessment using Realistic Technologies for the National Airspace System (SMART-NAS) Testbed that serves as a central point of collection, integration, and access of information from these different data sources. By testing and evaluating using real-world scenarios, we may accelerate the acceptance of the RTSM framework towards deployment. In this paper we demonstrate the framework's capability to not only estimate the state of safety in the NAS, but predict the time and location of unsafe events such as a loss of separation between two aircraft, or an aircraft encountering convective weather. The experimental results highlight the capability of the approach, and the kind of information that can be provided to operators to improve their situational awareness in the context of safety.

I. Introduction

As new operational paradigms (e.g., dynamic Traffic Flow Management) and additional aircraft (e.g., Unmanned Aerial Systems) are being introduced into the National Airspace System (NAS), maintaining safety in such a rapidly growing environment becomes very challenging. Moreover, with the traffic in the NAS projected to increase in the near future, advanced tools will be needed just to maintain the current level

*Research Scientist, Intelligent Systems Division, Discovery and Systems Health Area, AIAA Member.

†Senior Scientist, Intelligent Systems Division, Discovery and System Health Area, AIAA Senior Member.

‡Research Scientist, Intelligent Systems Division, Discovery and Systems Health Area.

§Research Scientist, Intelligent Systems Division, Discovery and Systems Health Area, AIAA Senior Member.

of NAS safety. NAS operators must consolidate operations-related information from various sources, which may be imprecise, inaccurate, incomplete, and inconsistent, and apply domain knowledge to interpret the current NAS state and forecast future NAS state, which is both time- and workload-intensive. It is therefore desirable to have an automated framework to provide an overview of the current safety of the airspace at different levels of granularity, as well an understanding of how the state of the safety will evolve into the future given the anticipated flight plans, weather forecast, predicted health of assets in the airspace, and so on, in order to provide operators with improved situational awareness in increasingly complex airspaces.

Aviation safety is of paramount importance and a number of approaches can be found in literature that aim to improve safety of aircraft in the NAS. One such approach looks for accident and incident precursors via data mining, using data from such flight data recording programs as the Flight Operations Quality Assurance (FOQA).¹ A comprehensive archive of safety data is available through the FAA's Aviation Safety Information Analysis and Sharing (ASIAS)² repository, which is being mined by researchers to identify anomalous situations for further analysis to detect emerging issues, potentially leading to policy or procedure changes.^{3,4,5} Once discovered, these precursors can also be probed in real-time as a component of real-time safety assessment. Similarly, the FAA uses data mining techniques to monitor operational parameters that define Key Performance Indicators (KPI) throughout the NAS to identify issues and modify operations before issues become hazards;⁶ EUROCONTROL's Automatic Safety Monitoring Tool (ASMT) serves a comparable purpose in Europe.^{7,8} Finally, an even broader, off-line, approach to finding accident and incident precursors uses the integrated FAA Integrated Safety Assessment Model (ISAM)⁹ and EUROCONTROL Incident Risk Assessment Model¹⁰ to link known precursors and hazards involved with every identified cause of accidents and incidents worldwide. The integrated model can be used for risk-based rule-making.

In contrast, we desire a framework that can, in real-time, not only monitor the current state of safety of the NAS, but predict the future evolution of the state of safety and the occurrence of unsafe events. These research goals are aligned with the NASA Aeronautics Research Mission Directorate's Strategic Thrust 5 of Real-Time System Wide Safety Assurance. Towards this end, in previous work, we developed the Real-Time Safety Monitoring (RTSM) framework.^{11,12} This proposed framework monitors and predicts the state of safety and determines whether any unsafe event will occur within a predetermined time frame of interest. The RTSM framework is model-based, and may include any hazard that can be modeled quantitatively. Further, it handles uncertainty in a systematic manner. To this end, one major contribution of this paper is the use of Latin Hypercube sampling method by the RTSM framework for making predictions about the safety of the NAS. The Latin Hypercube sampling method improves coverage of uncertain variables and provides better accuracy than the Monte Carlo sampling method we had adopted in our earlier implementation of the RTSM framework.¹¹

In previous work,^{11,12} we demonstrated our framework in simulation on three different constructed scenarios. In this paper, we demonstrate this framework on real flight data. The data is obtained from an air traffic management simulation and testing system called the Shadow Mode Assessment using Realistic Technologies for the National Airspace System (SMART-NAS) Testbed.¹³ One of the main benefits for using the SMART-NAS testbed is the collection and integration of data from a variety of sources (e.g., FAA's System Wide Information Management, Aircraft Situation Display for Industry, Airport Surface Detection System Model-X, etc.), and the SMART-NAS testbed serves as a central point of access for all this information from the different sources.

Furthermore, the SMART-NAS testbed allows the testing and validation of new technology concepts in a realistic environment. It employs actual air traffic data, systems and procedures used in Air Traffic Management (ATM) operations, and integrates high-fidelity human-in-the-loop aircraft, tower and air traffic control (ATC) simulators. It also provides the capability to assess near-real-time what-if scenarios for new technology solutions and procedures for particular cases relevant to air traffic management and airline decision support, and will be very useful in further developing requirements for the RTSM framework. Hence, the integration of the RTSM framework in the SMART-NAS Testbed will enable the testing and evaluation of this framework in real-world scenarios, and perhaps even accelerate the acceptance of the RTSM framework to be fielded in the real world. Therefore, this demonstration serves as a first step towards this integration. In this paper, using this testbed, we demonstrate our capability to not only estimate the state of safety in the NAS, but predict the time and location of unsafe events, such as loss of separation between two aircraft, or an aircraft encountering convective weather. By providing such information to operators or decision-support tools, unsafe events or diminished NAS safety can be preemptively avoided, in contrast to reactively mitigating them.

This paper is organized as follows. Section II presents the safety modeling and analysis work that forms the basis of the real-time safety monitoring framework. The monitoring and prediction modules of the RTSM framework, as well as its architecture, are presented in Section III. The experimental setup and results are presented in Section IV. Finally, Section V concludes the paper.

II. Safety Modeling and Analysis

The goal of the Real-Time Safety Monitoring (RTSM) framework is the quantification of the current and future state of safety in the NAS. We are interested in how the state of safety will evolve in the future, and whether any unsafe events will occur, along with their probability, time, and location of occurrence. The framework is model-based, i.e., we require models of the safety space, unsafe events, safety metrics and thresholds, and the dynamic behavior of NAS components. The estimation and prediction steps rely on these models to estimate and predict safety. So, as the first step, we have to model and analyze safety, and determine what qualifies as an unsafe event. In the remainder of the section we summarize the models, and refer the reader to our previous publications for additional details.^{11,12}

A. Modeling the NAS

The first step is to develop a model describing the dynamics of the NAS, i.e., how the state \mathbf{x} evolves in time:

$$\mathbf{x}(k+1) = \mathbf{f}(k, \mathbf{x}(k), \mathbf{u}(k), \mathbf{v}(k)), \quad (1)$$

where \mathbf{f} is the state function, \mathbf{u} is the input vector (exogenous inputs to the system, such as the aircraft's intended flight routes and wind velocity at various altitudes), and \mathbf{v} is the process noise vector.

In order to make a prediction at time k using \mathbf{f} , we require $\mathbf{x}(k)$, which, in general, is not known. Instead, we have available an output vector \mathbf{y} , defined through an output equation:

$$\mathbf{y}(k) = \mathbf{h}(\mathbf{x}(k), \mathbf{u}(k), \mathbf{n}(k)), \quad (2)$$

where \mathbf{h} is the output function, and \mathbf{n} is the sensor noise vector.

To model the NAS at a system-level, we require models of aircraft, pilots, controllers, weather phenomena, restricted airspace, etc. Some of the NAS models used by the RTSM framework are listed below.

- **Aircraft Modeling:** Aircraft models need to at least describe how the positions and velocities of aircraft change in time. Each of these variables change in time based on the actions of a pilot or flight management system (FMS) and with environmental conditions (e.g., wind, temperature, pressure, etc.). Such models already exist in the literature^{14,15,16} that typically assume point-mass models, describing how the aircraft position changes given an intended flight plan.
- **Weather Modeling:** Weather systems are very complex to model, and we take advantage of existing systems for weather prediction.^{17,18} Weather models must describe how a weather region moves in time and changes shape, and is usually described by a dynamic region specified as a polygon, e.g., as in the Convective Weather Avoidance Polygon (CWAP),¹⁸ which is constructed based on edges in the echo top field and winds at the flight altitude that indicate the storm boundaries that the pilots will tend to follow to avoid convective weather. In this paper, we use polygons defined by convective Significant Meteorological Information (SIGMET),¹⁹ which also includes an estimate of the speed and heading of the weather region.

B. Determining Safety Metrics and Thresholds

A set of *safety metrics*, which define the safety space, can then be evaluated based on available data and are predictive of unsafe events. A safety metric, ϕ , can be quantitatively computed as an algebraic function \mathbf{F} of the states, \mathbf{x} , that quantify these hazards. Each safety metric is associated with a *threshold* that specifies when the state of the NAS transitions from safe to unsafe. The RTSM framework makes predictions about the occurrence of different unsafe events by predicting when different safety metrics transition from an acceptable region into an unacceptable region. The boundaries between the acceptable and unacceptable

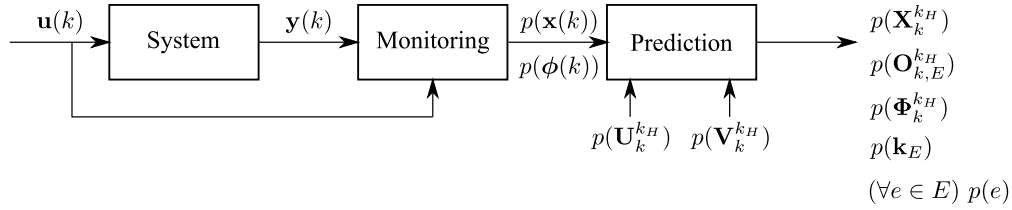


Figure 1: The Real-time Safety Monitoring Framework Architecture.

spaces for each safety metric, ϕ , are defined through threshold functions, $\mathbf{T}_E(\phi(k))$. Threshold functions can take any general form. Selecting an appropriate threshold can be challenging. Thresholds are determined through a mix of techniques, including consultation with subject matter experts (SMEs) and extraction from historical NAS data using data mining techniques, among others. The violation of these thresholds would result in unsafe events such as loss of separation. Some examples of safety metrics and corresponding thresholds relevant to our demonstration experiments presented in Section IV are listed below:

- **Loss of Separation:** The safety metric corresponding to the *loss of separation event* is the separation distance between two aircraft. The corresponding threshold for this separation distance is 1000 feet vertically and 5 nmi laterally as per the en-route ATC separation standards. A loss of separation event occurs anytime two aircraft are closer than the vertical and/or lateral separation thresholds.
- **Convective Weather Encounter:** Similarly, for the *convective weather encounter event*, the corresponding safety metric is the distance between an aircraft and the nearest edge of a convective weather avoidance polygon. In this paper, we define a convective weather encounter event to occur anytime an aircraft reaches the edge of a convective weather avoidance polygon.

III. The Real-Time Safety Monitoring Framework

Given mathematical definitions and models for safety metrics related to different unsafe events, we can monitor and predict safety of the NAS. Monitoring and prediction is performed in an integrated manner in since the outputs of the monitoring step are inputs to the prediction step. Fig. 1 presents the overall computational architecture of the RTSM framework. In the *monitoring* step, given the inputs \mathbf{u} , the outputs $\mathbf{y}(k)$, the state equation \mathbf{f} , the output equation \mathbf{h} , and the set of unsafe events E , we must (i) estimate the current state $p(\mathbf{x})$, (ii) compute from $p(\mathbf{x})$ using the safety metric equation \mathbf{F} , $p(\phi(k))$, and (iii) determine using the threshold function \mathbf{T}_E whether any of the events in E have occurred. In the *prediction* step, at time k , given $p(\mathbf{x}(k))$, the state function \mathbf{f} , the safety metrics ϕ , the safety metric function \mathbf{F} , the set of events E and threshold function \mathbf{T}_E , the future input probability density function $p(\mathbf{U}_k^{k_H})$, and the future process noise probability density function $p(\mathbf{V}_k^{k_H})$, we must compute probability density functions for $\mathbf{X}_k^{k_H}$, $\Phi_k^{k_H}$, $\mathbf{O}_{E,k}^{k_H}$, \mathbf{k}_E , and $P(e)$ for each $e \in E$.

A. Monitoring Safety in the NAS

Monitoring the NAS can be viewed as a Bayesian inference problem, where the goal is to infer, i.e., estimate, the state of the NAS, $\mathbf{x}(k_P)$ at any desired time k_P based on the known inputs to the NAS, $\mathbf{u}(k_P)$, and the available observations, $\mathbf{y}(k_P)$. In other words, the monitoring problem is defined as computing $p(\mathbf{x}(k_P)|\mathbf{y}(k_P))$. Various sensors measuring the current state of the NAS, such as GPS, radar, aircraft instrumentation, and weather observations constitute the available observations, and intended flight routes, scheduled departure times, etc. constitute the inputs.

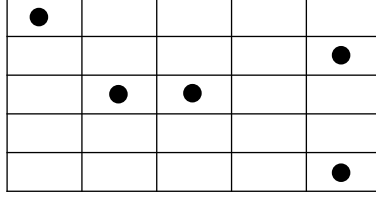
Many algorithms are available to solve this general problem. Since our models are nonlinear, they require a nonlinear filter (such as unscented Kalman filters or particle filters^{20,21,22,23}). These algorithms all have the same basic structure. For a new time step, we first predict the new state, based on the previous state and the inputs. Then, we correct the state based on the available observations and the likelihood functions. Given some assumption of uncertainty in the model and some assumption of uncertainty in the observations, the algorithms find the most likely state. Given an estimate of the current state, the corresponding values

Algorithm 1 $\Phi_{k_P}^{k_H} \leftarrow \text{Predict}(\mathbf{x}(k_P), \mathbf{U}_{k_P}^{k_H}, \mathbf{V}_{k_P}^{k_H}, k_H)$

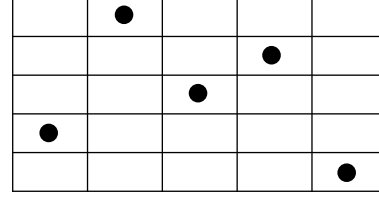
```

1: for  $k = k_P$  to  $k_H$  do
2:    $\mathbf{x}(k+1) \leftarrow \mathbf{f}(k, \mathbf{x}(k), \mathbf{U}_{k_P}^{k_H}(k), \mathbf{V}_{k_P}^{k_H}(k))$ 
3:    $\phi(k+1) \leftarrow \mathbf{F}(\mathbf{x}(k+1))$ 
4: end for

```



(a) Random 2-D Sampling



(b) 2-D Latin Hypercube Sampling

Figure 2: Illustrating Random 2-D Sampling and 2-D Latin Hypercube Sampling

of the safety metrics can be computed, and the event thresholds can be checked to determine if any unsafe events have been detected.

B. Predicting Safety in the NAS

The goal of the prediction problem is to compute the future evolution of the NAS state \mathbf{x} , the safety metrics ϕ , and the occurrence of safety events E from prediction time k_P up through the prediction horizon k_H . In loss of separation prediction, for example, $k_H - k_P = 20$ minutes is used in practice. Ultimately, we want to find the probability density functions of the state trajectory, $p(\mathbf{X}_{k_P}^{k_H})$, the safety metric trajectory, $p(\Phi_{k_P}^{k_H})$, and the event times $p(\mathbf{k}_E)$, as well as the probability of the different events occurring within the prediction horizon.

The underlying function to any prediction algorithm is given as Algorithm 1. At a given time of prediction k_P , with realizations of the state $\mathbf{x}(k_P)$, the future inputs $\mathbf{U}_{k_P}^{k_H}$, the future process noise $\mathbf{V}_{k_P}^{k_H}$, and a prediction horizon $k_H > k_P$, the algorithm simulates the NAS state forward up to k_H , and computing the corresponding safety metrics. From the safety metrics, we can compute whether any of the unsafe events have occurred and, if so, when they first occur. Note that the performance of this prediction algorithm is directly dependent on the accuracy of the information fed in the form of inputs into this algorithm.

In the past,¹¹ we used Monte Carlo sampling for prediction. A standard Monte Carlo approach generates many realizations or samples of the prediction inputs, and calls Algorithm 1 for each realization. However, Monte Carlo sampling may require using a large number of samples, and hence an equal number of evaluations of Algorithm 1. In particular, exhaustive sampling may be necessary to accurately capture information regarding the tails of probability distributions where unsafe events are captured (since they have lower likelihood of occurrence). Hence, in many scenarios, such exhaustive Monte Carlo sampling could be cost-prohibitive or not suitable for real-time operations.

Therefore, in this paper, in order to overcome this challenge, we use Latin Hypercube Sampling that focuses on generating samples consistently over the entire domain of all random variables. Consider N_d random variables, and say, it is desired to generate N_s samples. First, the range of each variable is divided into N_s equally probable intervals, thereby forming a hypercube. In order to facilitate such division into equally probable intervals, the original probability distributions of the variables are not considered at first, and instead, the samples are first generated from the standard uniform random variable $[0, 1]$. Then, sample positions are chosen such that there is exactly one sample in each row and exactly one sample in each column of this grid (similar to placing rooks on a chess board so that no rook may attack another). The resulting hypercube can be denoted as U_{ij} where ‘ i ’ denotes the sample number (varies from 1 to N_d) while ‘ j ’ denotes the random-variable number (varies from 1 to N_d). The difference between two-dimensional random Monte Carlo sampling and two-dimensional Latin hypercube sampling is illustrated in Fig. 2a (Monte Carlo samples) and Fig. 2b (Latin Hypercube samples).

Each generated Latin Hypercube sample is then passed through Algorithm 1, and corresponding samples of $\Phi_{k_P}^{k_H}$ are generated. This set of samples establishes a probability distribution (in terms of density) for the safety metrics and the related variables. For example, we can compute the probability of some event e occurring by k_H simply as the number of samples with finite k_E divided by N . Algorithm 1 can also be easily extended to provide the probability of the event occurring at each time point within $[k_P, k_H]$. Over a set of events, we can easily also compute the probability of any safety event occurring within the prediction horizon, and the probability of one occurring at each time point, and related probability distributions.

An additional advantage of the Latin hypercube sampling approach is that it is a variance reduction technique. Traditional Monte Carlo sampling is highly non-deterministic and may produce significantly different results when repeated (especially when the number of samples is small). On the other hand, Latin hypercube sampling reduces such variation effectively. This in turn further improves the prediction of probabilities and event times in the context of safety assessment.

C. Computing Overall Safety of NAS

Once the uncertainty in the prediction of the different safety metrics is computed, the likelihood of unsafe events can be computed. Consider a generic unsafe or undesirable event e , and the above prediction framework can be used to compute the probability of occurrence of e ; let $P(e)$ denote the corresponding probability.

While it is important to compute the likelihood of each unsafe event, it is also important to combine the likelihoods across multiple events. If any unsafe event were to occur, then it means that the airspace is unsafe. Consider events e_i ($i = 1$ to n) that correspond to multiple unsafe events. If e denotes the event that the NAS is unsafe, then, e is said to have occurred when *any* of the events e_i has occurred. We denote the probability that the entire airspace is unsafe as:

$$P(e) = P(\cup_{\forall i} e_i) \quad (3)$$

Note that $P(e_i)$ ($\forall i$, i.e., for each hazard) and $P(e)$ can be calculated for all future time instants based on information available at the present time instant. As explained earlier, time keeps evolving and these probabilities (for occurrence of unsafe-incidents in the future) keep evolving too.

IV. Demonstration Experiments

In this section, we demonstrate the approach using real flight data obtained from the SMART-NAS Testbed. The goal here is to show how the RTSM framework performs when estimating and predicting the state of safety of the NAS using real flight data.

A. Experimental Setup

We first obtained real flight data from the SMART-NAS Testbed for the day of December 29, 2015, for flights in the San Francisco International Airport terminal area. Note that this date was chosen arbitrarily. As mentioned earlier, one of the main benefits for using the SMART-NAS testbed is the collection and integration of data from a variety of sources. Based on the data specifications expected by the RTSM application, the SMART-NAS testbed generates the data in the correct format by calculating and transforming the input parameters to the testbed.

As an example of this, the prediction of the unsafe events in this demonstration involves inputs from both the Aircraft Situation Display to Industry (ASDI) data feed^a and the FAA's System Wide Information Management (SWIM) data feed.²⁴ Surface data is obtained from the Airport Surface Detection System Model-X (ASDE-X)^b. Flight plan data was resolved to latitude and longitude coordinates using the En Route Automation Modernization (ERAM)^c system, and is used as input to the aircraft models for trajectory prediction. Finally, Rapid Refresh weather data was obtained from the THREDDS Data Server (TDS)^d. As convective weather patterns are incorporated into the calculations, additional sources will need to be incorporated as well. The SMART-NAS Testbed serves as a central point of access for all of this information.

^a<https://www.fly.faa.gov/ASDI/asdi.html>

^bhttps://www.faa.gov/air_traffic/technology/asde-x/

^chttps://www.faa.gov/air_traffic/technology/eram/

^d<http://www.unidata.ucar.edu/software/thredds/current/tds/>

For the purpose of this paper, all input data were retrieved or calculated using real archived flight data and logged in CSV-formatted files by the testbed. Input for the RTSM calculations at every second include the following: (1) aircraft type, (2) call sign, (3) departure or arrival flag, (4) latitude, (5) longitude, (6) altitude, (7) heading, (8) speed, (9) wind direction, (10) wind magnitude, (11) and wind gust speed. Also included are the latest flight plan information reported for a given flight, resolved to latitude and longitude coordinates and updated asynchronously.

To overcome lack of an interesting weather phenomenon at the selected date and to illustrate the capability of the framework for other safety metrics, we superimposed a convective weather that occurred elsewhere in the airspace by artificially moving the location of the convective weather to the Oakland International Airport (OAK) area. These weather polygons were obtained from the SIGMETs.¹⁹

In our experiments, we examine the results of the RTSM framework during certain time periods throughout the day on selected safety metrics and unsafe events. The results are presented in the remainder of this section.

B. Experimental Results

We first consider a loss of separation (i.e., conflict) event between two aircraft - one landing and the other taking off. We use a separation standard of 4 nautical miles to demonstrate the approach. The framework is able to predict and detect the loss of separation event, giving operators an opportunity to continue as planned or change operations to avoid the loss of separation. Fig. 3 shows the actual separation distance between the two flights as a function of time. We can see that at around 0.6 minutes, the loss of separation event occurs, and these two aircraft remain in conflict till about 2 minutes. Fig. 4 shows the predicted time remaining until a loss of separation event for the two aircraft, plotted as a function of the time of prediction (offset so that the first prediction occurs at time 0). The true time until the conflict is shown as the dashed line, and the predicted mean values as points marked by an \times . The predictions generated for this scenario is an underestimate. This is because the actual trajectory of an aircraft has a lot of uncertainty like the speed of descent or altitude that our models do not capture to a high accuracy. Improving these models will result in better predictions.

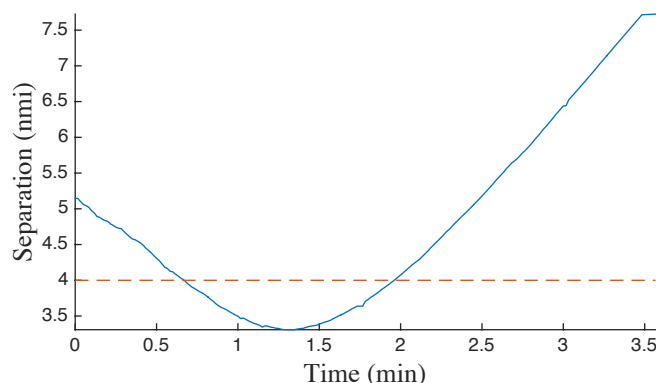


Figure 3: Actual loss of separation as a function of time.

Fig. 5a shows detected locations of loss of separation events (where the aircraft is when it loses separation with another aircraft) during a portion of the day's operations, computed using a separation requirement of 3 nautical miles. These locations are indicated using red dots. As observed, there are some conflicts due to takeoffs on parallel runways. Fig. 5b shows detected locations for a separation requirement of 5 nautical miles, and with this more strict requirement, many more conflicts are detected during the same time period, including aircraft on approach and starting their routes.

A convective weather region is assumed to exist centered at Oakland International Airport, and so we predict when aircraft will encounter the region. For one of the aircraft headed east, Fig. 6a shows the predicted times until the weather encounter event. Originally, due to the uncertainty both in the weather region movement and the aircraft speed, predictions are overly optimistic, but then quickly converge and are fairly accurate after convergence. Due to the uncertainty associated with how the region will move in

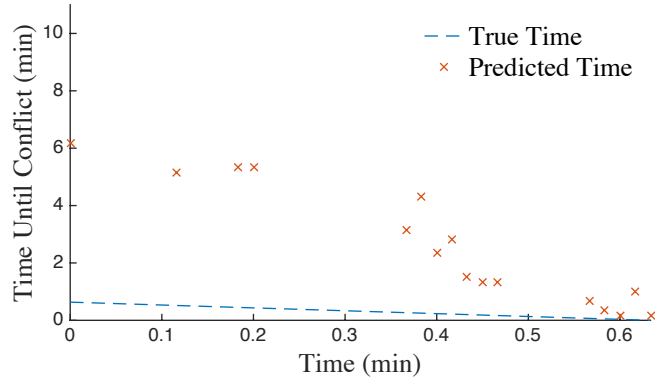
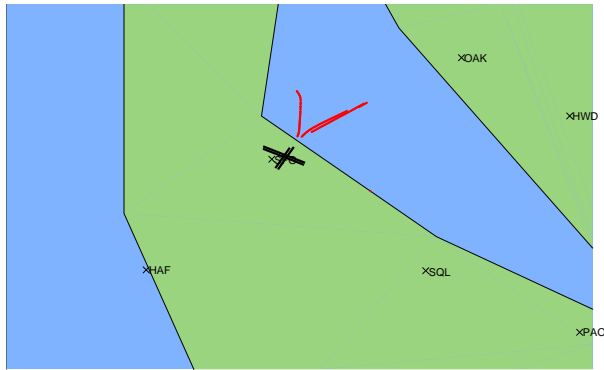
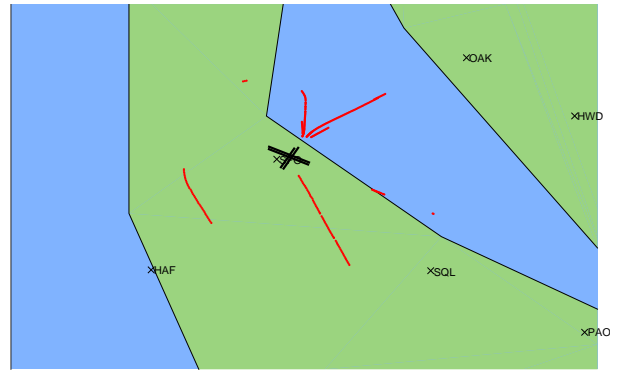


Figure 4: Predicted time until loss of separation as a function of time.



(a) Detected locations of loss of separation for 3 nmi threshold.



(b) Mean predicted locations of loss of separation for 5 nmi threshold.

Figure 5: Detected and predicted locations of loss of separation for different separation thresholds.

time, there is uncertainty in the predicted times, as indicated by the additional markers indicating the two standard deviations from the mean. Fig. 6b shows (using white dots) where the aircraft is predicted to be when it enters the convective weather region. With this information, an operator can determine where to divert an aircraft to avoid those locations. Although the time scale here is pretty short, predictions are made up to five minutes in advance in this demonstration. For this particular aircraft, it is taking off from SFO, and once airborne, the algorithm can quickly determine that it will encounter the weather within less than one minute of takeoff. For a weather region further away, the time scale would be longer and operators would know within five minutes whether the weather region will be encountered. The prediction horizon can also be extended (it is an input parameter to the algorithm) so that predictions can be available for longer time horizons, e.g., 20 minutes or longer.

V. Conclusions

This paper described the results from employing the RTSM framework on real flight data. It was shown that the framework accurately predicts the time to unsafe events as well as locations where the unsafe events are predicted to happen (if applicable). The prediction algorithms made use of Latin Hypercube sampling.

In this paper, we limited our approach to flight activity on the day of December 29, 2015 within a 50 nautical miles radius around the San Francisco International Airport (SFO). The application of the RTSM framework to the entire NAS and running NAS-wide experiments will be tackled as part of future research, which would enable us to analyze the scalability of the RTSM Framework. For NAS-wide implementation, a distributed approach will be developed. Open questions for future work center around a validation strategy,

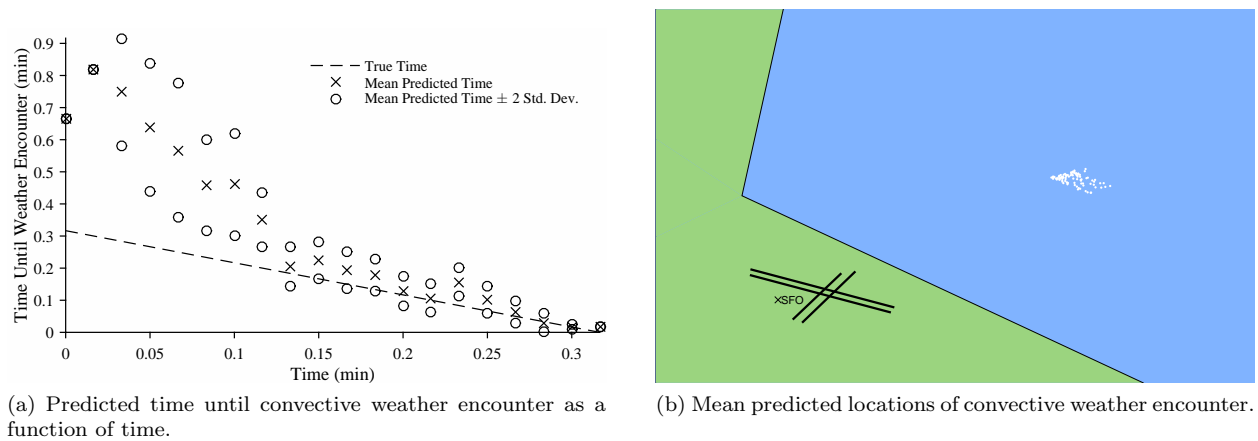


Figure 6: Predictions about convective weather encounter event.

proper display of the information to maximize the information utility for operators, and additional scenario development that scales the investigation from the local level to the regional and NAS levels.

Currently, work is in progress to fully integrate the RTSM Framework with the SMART-NAS testbed. Once integrated into the SMART-NAS testbed, it will be possible to test and validate the RTSM framework in realistic environments that employ actual procedures used in ATM operations, and integrates high-fidelity human-in-the-loop aircraft, tower, and ATC simulators to be able to assess near-real-time what-if scenarios. Finally, we would also want to research and develop new safety metrics and include these in the computation of the safety of the NAS.

Acknowledgements

This work was supported by the SMART-NAS project under the Airspace Operations and Safety (AOSP) program within the NASA Aeronautics Research Mission Directorate (ARMD). We would like to extend our sincere gratitude to Chok Fung Lai and Gano Chatterji for their continuous help and guidance.

References

- ¹Federal Aviation Administration, "Flight Operations Quality Assurance (DOT Advisory Circular No. 120-82)," 2004.
- ²Federal Aviation Administration, "Aviation Safety Information Analysis and Sharing (ASIAS)," .
- ³Gorinevsky, D., Matthews, B., and Martin, R., "Aircraft anomaly detection using performance models trained on fleet data," *Intelligent Data Understanding (CIDU), 2012 Conference on*, IEEE, 2012, pp. 17–23.
- ⁴Das, S., Matthews, B. L., Srivastava, A. N., and Oza, N. C., "Multiple kernel learning for heterogeneous anomaly detection: algorithm and aviation safety case study," *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 2010, pp. 47–56.
- ⁵Matthews, B., Das, S., Bhaduri, K., Das, K., Martin, R., and Oza, N., "Discovering anomalous aviation safety events using scalable data mining algorithms," *Journal of Aerospace Information Systems*, Vol. 10, No. 10, 2013, pp. 467–475.
- ⁶FAA Air Traffic Organization, "2014 Safety Report," .
- ⁷EUROCONTROL, "Automatic Safety Monitoring Tool," .
- ⁸Pozzi, S., Valbonesi, C., Beato, V., Volpini, R., Giustizieri, F. M., Lieutaud, F., and Licu, A., "Safety monitoring in the age of big data," *Ninth USA/Europe air traffic management research and development seminar (ATM2011)*, 2011.
- ⁹Balakrishna, P., Borener, S. S., Crook, I., Durston, A., and Robinson, M. J., "Applying Probabilistic Risk Assessment to Safety Risk Analysis in Aviation," *Analyzing Risk through Probabilistic Modeling in Operations Research*, 2015, pp. 321.
- ¹⁰Perrin, E., Kirwan, B., and Stroup, R., "A systemic model of ATM safety: the Integrated Risk Picture," *Proc. 7th USA/Europe ATM R&D Seminar, Barcelona*, 2007.
- ¹¹Roychoudhury, I., Spirkovska, L., Daigle, M., Balaban, E., Sankararaman, S., Kulkarni, C., Poll, S., and Goebel, K., "Predicting Real-Time Safety of the National Airspace System," *AIAA SciTech Forum*, Jan. 2016.
- ¹²Roychoudhury, I., Spirkovska, L., Daigle, M., Balaban, E., Sankararaman, S., Kulkarni, C., Poll, S., and Goebel, K., "Real-Time Monitoring and Prediction of Airspace Safety," Tech. Rep. NASA/TM-2015-218928, NASA Ames Research Center, Moffett Field, CA, USA, Nov. 2015.
- ¹³Palopo, K., Chatterji, G., Guminsky, M., and Glaab, P., "Shadow Mode Assessment using Realistic Technologies for the

National Airspace System (SMART NAS) Test Bed Development,” *AIAA Modeling and Simulation Technologies Conference*, June 2015.

¹⁴Chatterji, G., Sridhar, B., and Bilimoria, K., “En-route flight trajectory prediction for conflict avoidance and traffic management,” *AIAA Guidance, Navigation, and Control Conference*, July 1996.

¹⁵Tandale, M. D., Wiraatmadja, S., Menon, P. K., and Rios, J. ., “High-Speed Prediction of Air Traffic for Real-Time Decision Support,” *AIAA Guidance Navigation and Control Conference, Portland OR*, 2011, pp. 8–11.

¹⁶Slattery, R. and Zhao, Y., “Trajectory synthesis for air traffic automation,” *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 2, 1997, pp. 232–238.

¹⁷Russell, C., “Predicting Airspace Capacity Impacts Using the Consolidated Storm Prediction for Aviation,” *10th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, September 2010.

¹⁸Rubnich, M., Matthews, M., and DeLaura, R., “Use of the Convective Weather Avoidance Polygon (CWAP) to Identify Temporally Coherent Convective Storm Boundaries,” *13th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, August 2013.

¹⁹Federal Aviation Administration, “Pilot’s Handbook of Aeronautical Knowledge,” 2008.

²⁰Julier, S. J. and Uhlmann, J. K., “A new extension of the Kalman filter to nonlinear systems,” *Proceedings of the 11th International Symposium on Aerospace/Defense Sensing, Simulation and Controls*, 1997, pp. 182–193.

²¹Julier, S. J. and Uhlmann, J. K., “Unscented filtering and nonlinear estimation,” *Proceedings of the IEEE*, Vol. 92, No. 3, March 2004, pp. 401–422.

²²Daigle, M., Saha, B., and Goebel, K., “A Comparison of Filter-based Approaches for Model-based Prognostics,” *2012 IEEE Aerospace Conference*, March 2012.

²³Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T., “A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking,” *IEEE Transactions on Signal Processing*, Vol. 50, No. 2, 2002, pp. 174–188.

²⁴Crescenzo, D. D., Strano, A., and Trausmuth, G., “SWIM: A Next Generation ATM Information Bus-The SWIM-SUIT Prototype,” *14th IEEE International Enterprise Distributed Object Computing Conference Workshops*, 2010, pp. 41–46.