Proceedings of the 2017 Winter Simulation Conference
W. K. V. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, eds.

BUILDING AN INTEGRATED SIMULATION ENVIRONMENT FOR MODELING TRAFFIC MANAGEMENT INTERACTIONS

Shin-Lai (Alex) Tien
David Bodoh
Huina Gao
James DeArmon

Center for Advanced Aviation System Development (CAASD)
The MITRE Corporation
7515 Colshire Drive
McLean, VA 22102, U.S.A.

ABSTRACT

The Federal Aviation Administration (FAA) is funding and encouraging new concepts for improving air traffic flow management (TFM) decision-making. The resulting automation capabilities will need to be operationally integrated into the existing air traffic management (ATM) system. Understanding how a new capability will interact with existing system components is challenging because of the range of possible real-world situations, which must be handled by the ATM system. Although there are fast-time traffic simulation tools available for modeling the impact of TFM actions, they are often developed as stand-alone tools, which are not extensible or flexible to work in concert with other advanced TFM capabilities for conducting integration studies or quantifying benefits. To address this gap, we have built a fast-time, distributed simulation platform integrating state-of-the-art traffic simulator and allows the plug-in of advanced TFM prototypes – or other experimental capabilities that already exist – so their interactions can be studied. In this paper, we discuss the requirements and the necessary components for building this platform, which requires an architecture that is flexible enough to support many different configurations of modeling tools and applications. We then use a TFM integration case study to demonstrate the utility of the platform. We show, with only minor effort, a proposed TFM prototype can be plugged into the platform and its benefits can be evaluated.

1 INTRODUCTION

As part of continuing improvement to the National Airspace System (NAS) automation, new concepts and capabilities for enhancing traffic flow management (TFM) decision-making are being proposed or developed for future operating environments (FAA 2017). Integrating these TFM capabilities in the current ATM system and understanding optimal strategies for using them effectively are critical to achieve associated benefits. Due to the scale of investment and complexity of systems integration, thoroughly analyzing operational integration issues is necessary.

Fast-time simulation is commonly used to analyze the interactions between new and existing TFM capabilities due to favorable low cost and ease of setup, compared with Human-in-the-loop (HITL) experiments. However, many commercial and proprietary air traffic simulation tools are rarely equipped to account for the various analysis and modeling needs for analyzing the interactions with new TFM concepts. Most of the time, a new TFM concept already has a working prototype with sophisticated algorithmic or procedural design, which is impractical to embed inside an air traffic simulation tool. Therefore, there is a need for creating a simulation environment that is amenable to these extensions.

The problem of lacking extensibility of simulation models exists not only in air traffic simulations, but also in other application areas. Kewley et al. (2008) illustrates the need in military application to have a federated simulation that can integrate the subsystems developed under different domains and fidelities to interact with each other. Developing a large scale stand-alone model to support systems integration [at all levels] is a time-consuming process that is often not possible. Ni (2006) also illustrates in surface transportation the challenges of building a single simulation that intends to address all the modeling needs in the same level of detail and in the same scale of resolution.

Recognizing the need for providing access points for working with external models or simulations, commercial air traffic simulation tools, such as Total Airspace and Airport Modeler (TAAM) and AirTOp, recently provided Application Program Interfaces (API) for exposing internal functionalities (Jeppesen 2016 and Airtopsoft 2016). The enhanced extensibility of these tools presented a new opportunity of using simulation in conducting integration studies, comparing alternatives, or quantifying benefits of a future TFM concept.

In this paper, we propose to build an extensible, distributed fast-time simulation platform for facilitating TFM integration studies allowing realistic prototype or fielded TFM capabilities to engage and influence simulated air traffic, leveraging the introduction of these traffic simulation APIs. On the proposed platform, a dedicated air traffic simulator is employed not only for simulating air traffic movement but also for executing the directives from individual TFM capabilities. It is expected that integrating a traffic simulator with the TFM capabilities in this way will enable a better understanding of overall system dynamics, failure modes, and strategies for using new TFM capabilities effectively. Moreover, it will encourage the interoperability and reusability of the simulation modeling elements, since the plug-and-play nature of the platform eliminates the need for ad hoc design of each new TFM interaction study.

To build this simulation platform, the traffic simulator and the TFM capabilities should each satisfy the requirements of exposing a run-time application interface, maintaining a logical simulation clock, having access to a shared communications network, and automating the human decision components. There exists a variety of available distributed simulation modeling systems which provide the necessary services of time management, data exchange, and subscription management, and they can expedite the development of the proposed platform (Fujimoto 2015).

This paper describes our approach to build the proposed platform, summarizes our implementation experience, and then documents how the platform is applied to addressing a TFM integration research need. One of the objectives of this paper is to aid other research institutions by describing the requirements and architecture for the fast-time simulation integration platform for them to build similar capabilities when conducting TFM integration studies. Section 2 of this paper specifies the requirements and the architecture design of the simulation platform. Section 3 discusses the background of a TFM integration study and the interactions of two capabilities to be modeled in the simulation. Our approach to building the simulation platform for facilitating the integration study is summarized in Section 4, and the initial study results are summarized in Section 5.

2 SIMULATION ARCHITECURE DESIGN

The proposed fast-time simulation platform requires an architecture that can support a variety of air traffic management experiments, exercising a variety of configurations of modeling tools and applications. Generally, many usable applications already exist in some form but with varying degrees of fidelity and functionality. Without targeting specific applications, the architecture needs to be flexible and extensible to meet the current and anticipated analysis needs. In most cases, different combinations of diverse capabilities must be configured so the researchers can exercise them in concert to address specific concerns about current and future TFM policies and technology. This section describes the technical simulation-level requirements that would be used to build the proposed platform.

2.1 Requirement Overview

The architecture design is inspired by the High-Level Architecture (HLA), which is a federated simulation standard widely used in the defense industry (IEEE 2010) and applied in other domains (Bodoh 2003 & Wall, 2015). The research described in this paper applied the HLA concept for building a distributed simulation experiment using disparate capabilities. As a first consideration, all modeling components must be capable of either "time-stepped" fixed-interval time advancement or next event time advancement. Time-stepped advancement, which nominally advances time much faster than wall-clock time, precludes the use of HITL experiments. The applications that are typically designed for real-time decision support need to be modified for operations in a fast-time environment, which requires the changes in time advancement method and the replacement of human decisions/interactions with automated or scripted actions. Any decision points that would normally pause and defer to a person must instead be configured using some heuristic to auto-select an action from a set of options. Alternative policies for selection criteria can be treated as one of the experimental effects and analyzed via a Monte Carlo (randomized sampling) approach.

All components must have access to a common network. The architecture must be flexible enough to engage installations that only run on specific operating systems (e.g., Linux or Windows) or in specific locations. For timely data exchange among these applications, there must some shared protocol that allows expedient conveyance of events and messages.

The architecture must support the ability to interchange components, so that as tools with different fidelity or capabilities become available, they can be readily plugged-in; this offers analysts a selection of options best suited to their research. A loosely-coupled functional interface to implemented capabilities helps shield other applications from lower-level details.

All involved components must expose a data layer runtime Application Program Interface (API) for reporting current flight positions, issuing Traffic Management Initiatives (TMIs), signaling acceptance of reroutes, and exchanging other messages. Any local event of relevance to another component must be exposed and reflected so both applications have a common perspective of the system state.

Finally, for simulation repeatability and troubleshooting, the system and each involved component must be deterministic, so repeated runs using the same configuration can produce the same results. Any stochastic elements must be controlled via random number seed variables.

2.2 Necessary Services and Modeling Components

To satisfy the requirements, the necessary services and modeling components are discussed below:

2.2.1 Run-Time Infrastructure

This is the software protocol which all the simulation modeling components on this platform need to follow for designing their APIs, so they can coordinate their logical processes, synchronize simulation time advancement, and exchange data during runtime.

2.2.2 Time Management Function

This controls the advance of logical time in the simulation run at a fixed step size (e.g., 1 second). All the fast-time simulation modeling components need to finish processing their events at the current time step before the time management function moves the simulation clock.

2.2.3 Data Distribution Service

This governs how data is communicated among the simulation modeling components during runtime. For the efficiency of data exchange, publication/subscription type services can be used – each application publishes the data that are meaningful to others and subscribes to the data it needs as input.

2.2.4 Simulation Management Function

This is the function that controls the state of each simulation modeling component. It starts, pauses, continues, and ends the simulation model execution.

2.2.5 TFM Modeling Component

These are the modules which are plugged-in to the platform for integration study. Regardless of their TFM purposes, they each need to have an API which allows the Simulation Management Function to control the state of the simulation. Also, they should have the data distribution interfaces that can exchange data and comply with the protocol defined in the Run-Time Infrastructure (RTI). They receive system status (e.g., flight positions, trajectories, airspace constraints) and output TFM actions (e.g., reroutes, departure delays)

2.2.6 TFM Integration Policy Module

This governs how TFM actions from multiple simulation modeling components should interact. For example, if an airborne flight has already taken an action from Component A, the Integration Policy Module may dictate that it cannot take another action from Component B in the next 15 minutes due to procedural or operational considerations. Such treatment of the precedence relations of TFM actions is typical in a TFM integration study, either from the functional or procedural perspective.

2.2.7 Traffic Simulator

Our idea is to employ a dedicated air traffic simulator to evaluate the actions generated by individual TFM components, e.g., revised departure times, flight reroutes, and requested times of arrival (RTAs) at fixes. This traffic simulator will perform the following functions:

- Advance flights through airspace from origin to destination airports.
- Provide the current status of system state (e.g., flight positions, flight trajectories, sector traffic counts).
- Ingest and execute the actions levied by the TFM Integration Policy Module.
- Implement fundamental air traffic control (ATC) rules and constraints (e.g., aircraft separation, departure/arrival procedures, facility letter of agreement (LOA) restrictions, sector capacities, and special use airspace (SUA) transits).

Figure 1 illustrates the proposed architecture of the RTI. To initialize a run, all the simulation modeling components, the Integration Policy Module, and the traffic simulator need to register with the RTI and start their publication/subscription services. During a simulation run, the simulation clock only moves from one time step to the next when all the components have finished their processes from the prior step. At each step, the TFM modeling components receive data of interest provided by the traffic simulator and then publish the TFM actions when some triggering events are activated. Subsequently, the Integration Policy Module, which subscribes to TFM actions, determines whether to publish those actions to the traffic simulator based on the current operative integration policies. The traffic simulator applies

TFM actions, if any, and then updates system status. At the end of the simulation time, all the modeling components are halted.

If there are stochastic elements in any of these modeling components, they shall be parameterized to enable a Monte Carlo analysis.

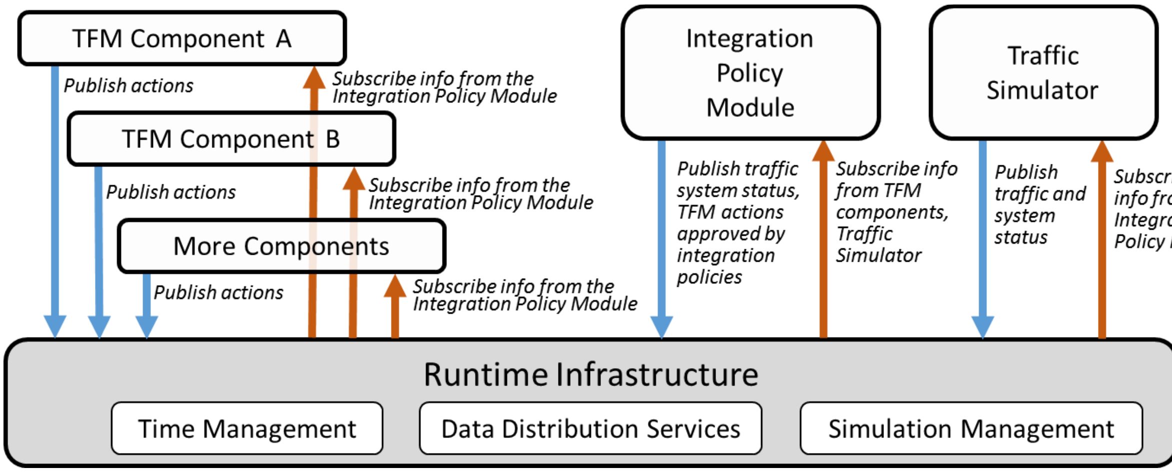


Figure 1: Architecture of the Proposed TFM Integrated Simulation Platform

3 CASE STUDY: ENABLING ARRIVAL METERING DURING EN ROUTE WEATHER VIA AN ADVANCED REROUTE CAPABILITY

This section describes the background of an operational problem and how a new TFM capability is expected to mitigate that problem. This problem illustrates the need for the proposed platform championed in this paper.

3.1 Arrival Metering during En route Weather

The Time-Based Flow Management (TBFM) system is a means of modulating ("metering") air traffic flows using time sequencing. In the case of arrival flows, as flights progress toward a destination airport, the TBFM system constructs a trajectory including predictions of future crossing times at specific meter reference points calculated using winds, aircraft performance characteristics, flight plan information, etc. When a flight crosses the "Freeze Horizon" (FH: a specified distance from the meter reference point where assigned times are "frozen"), a scheduled time of arrival (STA) at the meter reference point is assigned for the flight, based on competing demand and scheduling constraints. One type of meter reference point is a meter fix (MF) which is typically on or near the Terminal Radar Approach Control (TRACON) boundary, about 40 to 60 nautical miles (NMs) from the destination airport. A simplified diagram of a typical metering design is shown in Figure 2.

Delivering the flights per their STAs to the MFs will ensure steady, manageable traffic flows into the TRACON airspace, consistent with the runway arrival capacity. During periods of high demand, controller and pilot actions, such as speed adjustments or vectoring, may be necessary to meet the scheduled times (Shresta, 2014). However, a serious challenge for arrival TBFM is maintaining time-based schedules in conditions of severe en route weather. Severe weather affects routing predictability. Route changes or vectors implemented inside the FH alter trajectories from the flights plan, making flying time to the meter fix less predictable and STAs difficult to achieve. Often, TBFM is turned-off in these cases, and a less efficient flow management technique is employed, namely, miles-in-trail (MIT) spacing at the MF and further upstream. MIT spacing is less efficient because each flow is managed independently, without consideration for the relative demand on each flow and the merging of flows as they approach the airport.

In sum, lacking a dynamic reroute capability, flights passing through severe en route weather inside the FH will likely cause a path modification altering flying time, making trajectory predictions unstable, and STAs potentially unachievable.

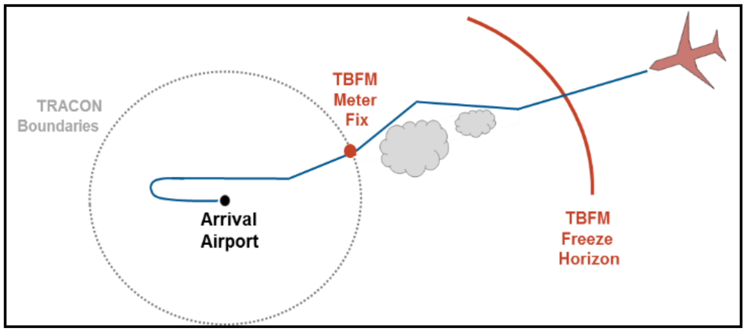


Figure 2: Notional flight path, TBFM FH, and MF

3.2 Benefit of Integrating A Reroute Capability

One of the promising solutions to the challenge of metering flights during weather impact is to incorporate a dynamic route planning capability for the flights, to identify reroute opportunities before their crossing the FH, and making the revised trajectory available to TBFM (Gong 2015).

In the current traffic management tool set, automated identification of airborne flights affected by severe weather is not available. In light of this need, MITRE CAASD and NASA, in partnership with the FAA, have developed a concept and prototype capability to identify such problems and help to resolve them. This capability is referred to as Advanced Flight-Specific Trajectories (AFST). See Stewart et al. (2012) wherein the capability was called En route Flow Planning Tool.

The AFST concept is based on the premise that as planning horizons decrease (shorter "look-ahead" times), predictions about constraints and affected flight trajectories become more certain. With this increased certainty, flow management decisions can become more precise. AFST will provide more efficient and flexible flight trajectory options which are consistent with prevailing en route constraints, e.g., weather blockage, sector congestion, or special use airspace activation.

For the problem of TBFM use during severe weather, AFST will formulate, prior to a flight's crossing of the FH, reroutes around or through weather. The adjusted route can then be provided to TBFM and because it has been de-conflicted with the weather, the new path can be maintained without minor or no further modification. This means that dependable schedule times can be assigned and achieved – TBFM can remain active, thereby enabling greater efficiencies as compared to turning it off and reverting to MIT restrictions.

4 BUILDING THE SIMULATION PLATFORM

This section describes the setup for the AFST/TBFM integration study.

4.1 RTI, Data Distribution, Time Management, and Simulation Management

As described in Section 2, the proposed platform is similar to the HLA in that there is a runtime architecture which manages several distributed simulation services. To meet the requirements, MITRE's Center for Advanced Aviation System Development (CAASD) developed the Simulation Data

Distribution Framework (SDDF), a cross-platform architecture which supports several types of simulation experiments, connecting a variety of applications. Additionally, SimBuilder, developed by MITRE CAASD, is a deployment utility with a user interface for managing a set of applications which operate on a simulation platform. These two technologies work in unison to satisfy the requirements.

SDDF/SimBuilder allows applications to share a service host and communications port for joining a common simulation modeling experiment. This distinguishes the applications from those joining to other concurrent simulations, avoiding interference. It also allows independent applications to join a simulation ad-hoc, even after the simulation clock has started. Applications that subscribe to services via SDDF must provide handling routines for events received from publishing applications. SDDF will buffer those events until the subscriber posts a "READY" status, which triggers those routines – such transactions happen regularly within the main processing cycle.

The SDDF/SimBuilder centrally scripts the execution parameters of all applications, including the hosts, displays, and command-line parameters used for each one. This feature is convenient for simulation repeatability, start-up, and shut-down.

SDDF/SimBuilder serve the role of time management by offering multiple modes of fixed-interval time advancement. An HITL experiment typically runs on par with wall-clock time, i.e., one second of simulation time takes one second. SDDF also supports a "time-stepped" protocol wherein the simulation is running as fast as it can, without any applications lagging. (The other major time-advance mechanism, "next event time advance," is not available in SDDF.)

Under SDDF, applications have a means to advertise the event contents they publish and, they can discover the services advertised by other applications in the simulation.

Note that SDDF/SimBuilder is one of several MITRE CAASD implementations for distributed simulation technologies based on published standards, such as HLA and Distributed Interactive Simulation (DIS) in Fujimoto (2015) and IEEE (2010). Other research institutions may follow similar standards and build this architecture for individual simulation needs.

4.2 A Dedicated Traffic Simulator

The requirements of a traffic simulator for the proposed platform were discussed in Section 2.2. While there are many traffic simulator options, the Total Airport and Airspace Model (TAAM), a highly reputable air traffic simulation package owned by Jeppesen (2017), was selected for this case study for its modeling capabilities in terminal airspace and its well-documented API.

Through its APIs, TAAM allows external programs during runtime to control simulation progress (start, pause, end), to obtain traffic status, and to input TFM actions, i.e. add/delete flights, amend flight plan, implement RTAs for flights at fixes (Jeppesen 2016). An SDDF-compatible interface software module called TAAM SDDF Gateway was developed so the runtime communication with TAAM can be effected.

4.3 Advanced Flight-Specific Trajectories (AFST)

MITRE CAASD has developed a working prototype of AFST and it has been evaluated in several HITL experiments using its sophisticated route evaluation/generation algorithms. With only minimal changes to the software, the AFST prototype was configured to plug into our TFM integration platform.

In operational use, AFST will require user interactions and facilitates decision-making. A user would follow these steps:

- Identify constraints (e.g., severe weather): use AFST's trajectory evaluation algorithms to search for opportunities for rerouting affected flights.
- Consider alternatives: construct and rank, via quantitative measures, various feasible routes.
- Collaborate on proposed route revisions: negotiate and seek consensus with dispatchers, and controllers using AFST's computer-human interface.

- Execute revisions: once a new route is determined, distribute route update to air traffic control personnel for review and issuance to pilots via AFST's interface.

Since this case study is focused on evaluating AFST-TBFM integration using fast-time simulation, not on human interaction with the tools, the human decisions to be made in AFST are replaced with an automated process. For example, the route selection of AFST, which was designed to be decided by human is now programmed by selecting the top-ranked route from the feasible ones generated by AFST's algorithms. Automating human decisions this way allows AFST to run in a fast-time simulation mode.

The AFST prototype software is compatible with SDDF/SimBuilder, so there was no additional interface to be developed. The only change required was to subscribe to flight track data from TAAM and publish its reroute messages during runtime.

4.4 TBFM Scheduler

The functionality of TBFM in this case study is to provide the STAs at the MFs. To simulate this functionality without access to the actual (proprietary) TBFM software, a scheduling software to mimic TBFM functionality was developed. This software application takes into account site adaptation data (e.g., airways, fixes, and airport runways), aircraft separation matrix, and flying time estimates to generate a schedule and assign flights' STAs.

During simulation runtime, the scheduler subscribes to the TAAM-published estimated times of arrival (ETA) at the MFs, then calculates and publishes the schedule of STAs.

4.5 Integration Policy Module

The Integration Policy Module governs how the AFST prototype, TBFM scheduler, and TAAM communicate, and the module implements a set of policies/rules to model the interactions between simulation components. The following defines how AFST and TBFM should interact:

- AFST sends flight reroutes which avoid en route weather.
- TAAM implements AFST reroutes and updates the ETAs to MFs.
- The TBFM scheduler receives the updated ETAs and revises the metering schedule when needed.
- TAAM receives the updated metering schedule and delivers the flights pursuant to STAs at the MFs, meeting the schedule times to the extent possible: with speed changes (subject to the airframe performance characteristics), or by executing vectors or holding.
- No change in STA is allowed after flights cross the FH.

4.6 Summary

Figure 3 is a screenshot of SimBuilder's user interface, illustrating the models/services in the proposed platform running in the Linux environment. Not shown on the screenshot is the TAAM application, which is running on an external Windows machine and connecting to the simulation platform via the TAAM SDDF Gateway.

In the center of Figure 3 is the Integration Policy Module. During the execution of a simulation, it passes flight positions to AFST and MF ETAs to the TBFM modeling component. For example, a pre-specified condition might be once a flight crosses the FH, its ETA update will not be sent to TBFM (since we want the original STA to be achieved). Likewise, the Integration Policy Module passes STAs and reroute messages to TAAM. TAAM will model the reroutes and apply these to the appropriate flights (as flight plan amendments). TAAM will then execute the STAs as RTAs to deliver the flights to MFs as close to the STAs as possible.

5 SIMULATION EXPERIMENT

5.1 Scenario Setup

An experiment was designed to simulate an operational scenario with arrival flows in severe weather, using the AFST prototype and TBFM scheduler components for TFM functionality.

The terminal airspace of George Bush Intercontinental Airport (IAH) at Houston, Texas was selected for the study area. The weather data was taken from July 25, 2016, wherein convective activity and thunderstorms were present near IAH from 18:00Z to 22:00Z. The weather did not block the MF completely nor did it significantly impact arrival runway capacity, meaning TBFM could potentially still operate.

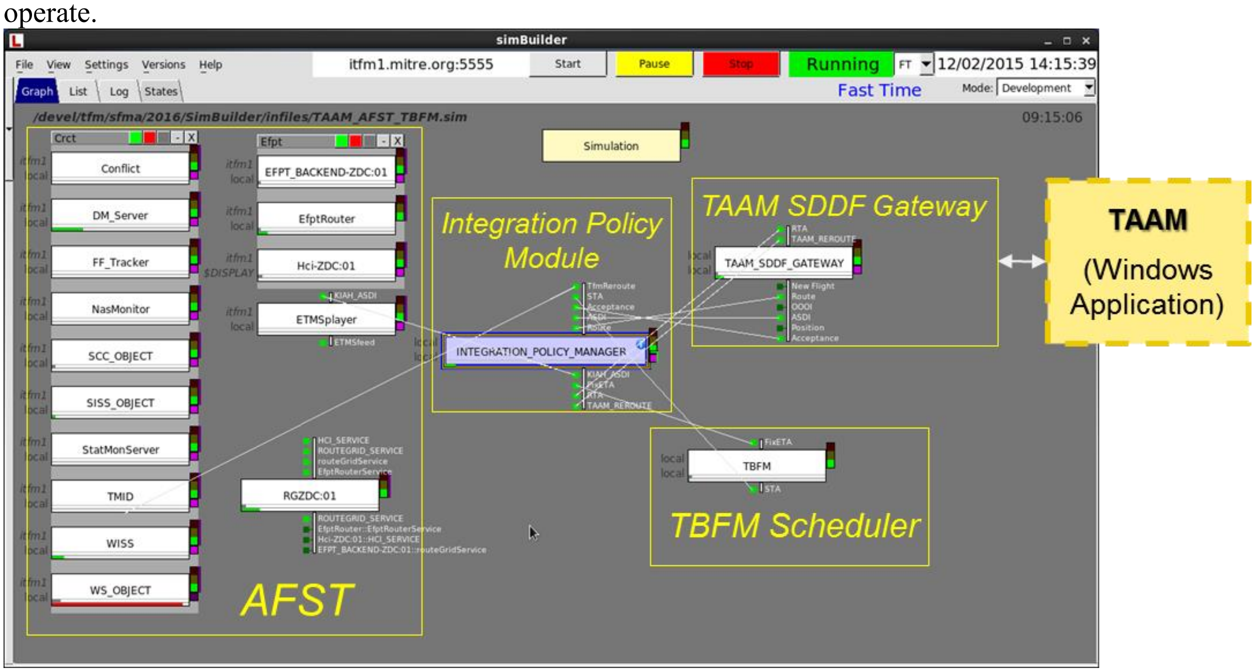


Figure 3: Connecting Simulation Modeling Components in a Time-Synchronized, Discrete Event Simulation Platform for TBFM/AFST Integration Study

The traffic data was taken from December 2, 2015, which was a clear weather day for IAH. There were 198 arrivals and 200 departures scheduled for the period of 18:00Z to 22:00Z. By using clear-weather traffic, the simulation modeling elements were able to modify flight paths for weather when needed. Although it is a relatively small, regional traffic scenario, it is sufficient for demonstrating the usage of the proposed simulation platform. Future research could study how the computational performance and scalability are determined by individual components plugged into this platform.

The scenario assumes that TBFM arrival metering will remain active, and AFST will generate weather-avoidance reroutes for individual flights based on forecast weather before the flight crosses the FH. Figure 4 shows the seven MFs (in green) used in this scenario, (i.e., RIICE, ZEEKK, MPORT, GMANN, LINKK, DOOBI, and SUUNR). The FHs for jet aircraft are about 190 to 250 NM away from the MFs. The movement of the weather was from southeast to northwest.

5.2 Experiment Results

The simulation experiment started by launching all the modeling components (i.e., SimBuilder, TAAM, AFST, TBFM, and the Integration Policy Module). Once SimBuilder confirmed positive connection statuses, it advanced the simulation clock in the step size of one second. When all the modeling components finished their respective computations for this time, SimBuilder advanced another one-second step. This process was repeated until the end of the simulation time. SDDF provided the "publish and subscribe" services so each modeling component could disseminate and receive data during runtime execution.

The speed of this simulation was 1.2 to 4 times faster than wall-clock time, depending on the number of active flights in the simulation, (i.e. the more the active flights, the more the weather avoidance problems the AFST prototype solves simultaneously). After the end of the simulation, the results were compiled from the log files of the modeling components.

As the simulation progressed, there were 22 weather-avoidance reroutes identified and published by AFST, and TAAM applied these to the affected flights. The total extra distance due to AFST reroutes was 361 NM, an average of about 16 NM per flight. Regarding meter time conformance, the total deviation between the STAs and the actual arrival times at the MFs was 586 seconds for the 198 arrivals, an average of 3 seconds per flight. The source of the deviation is mainly from the flight merging and maneuvering activities in TAAM.

Table 1 illustrates for a particular flight (UAL21) the event messages exchanged among simulation components in the runtime execution. This flight departed per TAAM at 17:22:00 and was scheduled to cross MF DOOBI at 19:42:28. Its original planned route is shown (in light blue) in Figure 5(a). Before its path reached the FH, AFST detected a weather blockage risk, so AFST issued a new route at 18:24:03.

This new route, shown in Figure 5(b), avoided the weather and was routed to an alternate MF, MPORT. The ETA to the MF then became 19:58:41, which was estimated by TAAM using the new route. TBFM used this ETA to calculate an STA with the same time (indicating no additional delay was needed for this flight). At time 19:35:08, TBFM froze the STA of UAL21. At 19:58:42, TAAM delivered the flight to the MF.

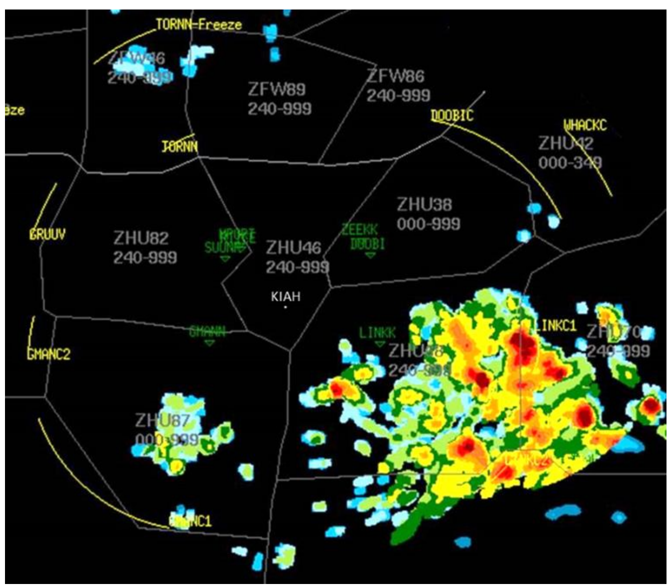


Figure 4: IAH Airspace, Meter Fixes, and Weather at 19:00Z on July 25, 2016

Table 1: Timeline of Significant Events For Flight UAL21

Sim Time (hh:mm:ss)	Acting Module	Event or Action
00:00:00	TAAM	Creates a new flight UAL21, scheduled to depart 17:22 on a flight plan [KBUF..EWC..FLM..BWG..SQS..AEX..JERNY..BEATL..DOOBI..SKNRD..KIAH]
17:21:59	TAAM	Creates a route event of UAL21
17:22:00	**TAAM**	**Departs UAL21 from the origin airport**

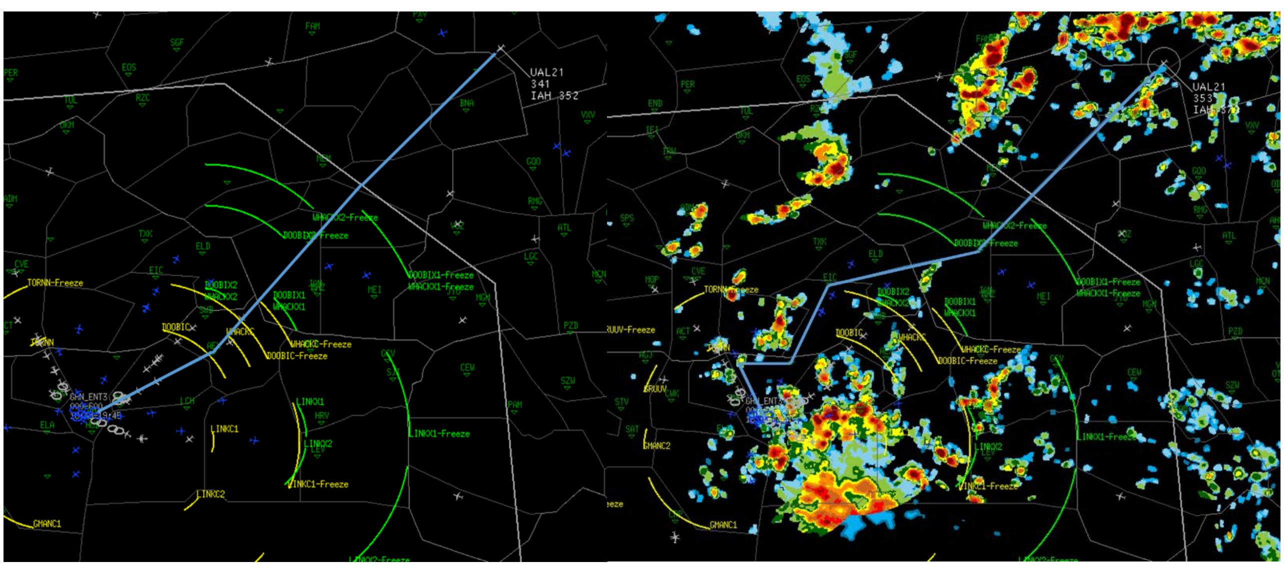
18:13:58	TAAM	Publishes the ETA to MF (DOOBI) of UAL21 as 19:41:35
18:14:00	**TBFM**	**Schedules UAL21 to MF (DOOBI) at 19:42:28**
18:14:00	TAAM	Accepts the request to deliver UAL21 to MF (DOOBI) at 19:42:28
18:24:03	**AFST**	**Publishes a reroute event of UAL21**
18:25:11	**TAAM**	**Accepts and amends the flight plan of UAL21 [BWG.SQS.EIC.LFK.LOA..MPORT..KIAH]**
18:23:58	TAAM	Publishes the ETA to MF (MPORT) of UAL21 as 19:58:41
18:25:01	**TBFM**	**Schedules UAL21 to MF (MPORT) at 19:58:41**
18:25:01	TAAM	Accepts the request to deliver UAL21 to MF (MPORT) at 19:58:41
19:35:08	**TBFM**	**Freezes the STA of UAL21 to MF (MPORT)**
19:58:42	**TAAM**	**Delivers UAL21 to MF (MPORT) at 19:58:42**

5.3 Summary

In this experiment, we simulated an integrated operation of TBFM and AFST using the proposed platform. The operational scenario demonstrated that following the AFST rerouting of flights meant weather had little impact on the stability of the metering schedule, i.e., most STAs to the MFs could be met within an acceptable level of conformance.

In follow-on research we plan to use this simulation platform and expand the analysis to additional operational scenarios to better understand the ability, and limitation of AFST finding usable reroutes around/through weather. If too few reroutes are found, then flights may make unplanned deviations around weather, TBFM and TAAM ETAs will be out of sync, assigned STAs will not be unachievable, and metering will be ineffective and turned off. On the other hand, if a sufficient proportion – say 90% of weather-blocked flights receive viable reroutes – then benefits of keeping TBFM active can be achieved. Of course, in real-world application, a traffic manager may see reroute opportunities which the AFST algorithm does not identify.

In the posited scenario, positive benefits will accrue via the improved efficiency of metering, compared to MITs. Also of positive benefit, reroutes may likely reduce tactical ATC maneuvers, implying reduced controller workload in the adverse conditions of severe en route weather.



(a) Original Route (b) Weather-Free Route by AFST

Figure 5: UAL21 Original Route vs Weather-Free Route by AFST

6 CONCLUDING REMARKS

We have built an extensible, distributed fast-time simulation platform for facilitating TFM integration studies. The platform allows realistic prototypes of proposed or fielded TFM capabilities to interact with each other and influence simulated air traffic. A dedicated air traffic simulator played the role of advancing flights in time and space, and evaluating the actions derived from the TFM capabilities. This addressed the need of using advanced TFM capabilities for conducting integration studies or quantifying benefits.

Such a simulation platform can not only support simulating TFM interactions but also promote the reuse and interoperability of the existing prototypes or modeling capabilities. The plug-and-play facility of the platform eliminates the use of the ad hoc design for each new study and avoid the need to build a stand-alone simulation tool that is expected to account for all of the TFM system's complexities.

An initial experiment simulated AFST and TBFM under conditions of severe en route weather. AFST demonstrated an ability to construct viable weather-avoidance reroutes. When these reroutes were implemented prior to the FH, metering generated achievable STAs. Future work will investigate the success rate of AFST constructing reroutes.

NOTICE

REFERENCES

FAA, "Decision Support Systems (DSS) Status," CDM Spring 2016 General Meeting, Retrieved January 1, 2017, from https://cdm.fly.faa.gov/wp-content/uploads/7.-DSS-CDM-DFW-3T-Status_041320016.pdf.

Kewley R., Cook, J., Goerger, N., Henderson, D., and Teague, E., "Federated Simulations for Systems of Systems Integration," In Proceedings of the 2008 Winter Simulation Conference, Austin, TX. IEEE, 2008.

Ni D., "A Framework for New Generation Transportation Simulation. In Proceedings of the 2006 Winter Simulation Conference, Monterey, CA. IEEE, 2006.

Jeppessen, TAAM Reference Manual and Model Changes Summary, December 2016.

Airtopsoft, User Manual of AirTOp V2.3.25, April 2016.

Fujimoto, R., "Parallel and distributed simulation," In Proceedings of the 2015 Winter Simulation Conference, December 2015.

IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)-- Framework and Rules," in IEEE Std 1516-2010 (Revision of IEEE Std 1516-2000) , vol., no., pp.1-38, August 18 2010.

Bodoh, D, Wieland, F., "Performance Experiments with the High Level Architecture and the Total Airport and Airspace Model (TAAM)," Proceedings of the Seventeenth Workshop on Parallel and Distributed Simulation, 2003.

Wall, T., Rodgers, M., Fujimoto, R., Hunter, M., "A Federated Simulation Method for Multi-modal Transportation Systems: Combining a Discrete Event-based Logistics Simulator and a Discrete Time-step-based Traffic Microsimulator," Simulation: Transactions of the Society for Modeling and Simulation International, Vol. 91(2) 148–163, 2015.

Shresta, S. et al., "Comparison of Strategies for Continuing Time-based Metering during Inclement Weather", ATC Quarterly, Vol. 22, No. 1, Air Traffic Control Association Institute (publisher), 2014.

Gong, C., McNally, D., Lee, C., "Dynamic Arrival Routes: A Trajectory-Based Weather Avoidance System for Merging Arrivals and Metering," 15th AIAA Aviation Technology, Integration, and Operations Conference, , Dallas, TX, 22-26 June 2015.

Stewart, T., Askey, L., Hokit, M., "A Concept for Tactical Reroute Generation, Evaluation and Coordination", The MITRE Corporation Center for Advanced Aviation System Development, DOI: 10.2514/6.2012-5586, 12th American Institute of Aeronautics and Astronautics (AIAA) Aviation Technology, Integration, and Operations (ATIO) Conference, Indianapolis, IN, 17-19 September 2012.

Jeppesen webpage describing TAAM: http://ww1.jeppesen.com/industry-solutions/aviation/government/total-airspace-airport-modeler.jsp, accessed October 2016.

AUTHOR BIOGRAPHIES

Dr. Alex Tien is a Lead Analyst at The MIRE Corporation. His research interest includes air traffic flow management, aviation system performance analysis, and weather impact modeling. In recent years, he has been involved in developing concepts and decision support capabilities to improve TFM strategic planning process and operational awareness. He received his doctorate in Civil Engineering from University of Maryland, College Park. sltien@mitre.org

David Bodoh is a Lead Systems Engineer at The MITRE Corporation. He has been designing and building air traffic management simulation models for over twenty years. He is experienced in developing real-time and fast-time models, especially those using distributed systems. dbodoh@mitre.org

Dr. Huina Gao is a Lead Simulation Engineer at The MITRE Corporation. She has worked in modeling and simulation of air traffic management for over 10 years. She holds a Ph.D. degree in Decision Science and a master degree in Applied Statistics from University of Maryland, College Park. hgao@mitre.org

James DeArmon is a Principal Engineer with The MITRE Corporation. He has worked for the last several years in research and modeling of air traffic flow dynamics. He holds a Master's degree in Operations Research, and has written numerous papers in aviation and applied math. jdearmon@mitre.org