Development and Validation of an Automated Simulation Capability in Support of Integrated Demand Management

Heather Arneson, Ph.D.
NASA Ames Research Center, Moffett Field, CA, 94035

Antony D. Evans, Ph.D.
Crown Consulting Inc., NASA Ames Research Center, Moffett Field, CA, 94035

Jinhua Li, Ph.D.
Universities Space Research Association, NASA Ames Research Center, Moffett Field, CA, 94035

Mei Yueh Wei
NASA Ames Research Center, Moffett Field, CA, 94035

Integrated Demand Management (IDM) is a near- to mid-term NASA concept that proposes to address mismatches in air traffic system demand and capacity by using strategic flow management capabilities to pre-condition demand into the more tactical Time-Based Flow Management System (TBFM). This paper describes an automated simulation capability to support IDM concept development. The capability closely mimics existing human-in-the-loop (HITL) capabilities, automating both the human components and collaboration between operational systems, and speeding up the real-time aircraft simulations. Such a capability allows for parametric studies that will inform the HITL simulations, identifying breaking points and parameter values at which significant changes in system behavior occur. This paper also describes the initial validation of individual components of the automated simulation capability, and an example application comparing the performance of the IDM concept under two TBFM scheduling paradigms. The results and conclusions from this simulation compare closely to those from previous HITL simulations using similar scenarios, providing an initial validation of the automated simulation capability.

I. Introduction

Integrated Demand Management (IDM) is a near- to mid-term NASA concept that is intended to contribute to future user-negotiated routing of aircraft to enable the flexible management of aircraft and National Airspace System (NAS) resources based on reliable predictions of future states.[1] IDM will develop more powerful, integrated operations and tools for managing flight trajectory constraints by leveraging existing systems as well as new automation tools and methods. A central component of IDM is the user-negotiated routing of aircraft, which plays an increasingly important and effective role in the flexible management of aircraft and NAS resources based on reliable predictions of future states. Flight delays result when the capacity of critical resources, such as airspace or airports, is insufficient to meet demand. Better matching between demand and available capacity can lead to efficiency improvements, with increased throughput, reduced delays and more efficient flight trajectories. However, because of uncertainties in both demand and capacity (e.g., due to weather), such matching can be difficult. IDM proposes to address mismatches in demand and capacity by using strategic flow management capabilities with the Traffic Flow Management System (TFMS) toolset to pre-condition demand into the more tactical Time-Based Flow Management System (TBFM), which would enable TBFM to better manage delivery of demand to capacity constrained resources.

A number of human-in-the-loop (HITL) simulation studies have been carried out with experienced air traffic manager and controller participants in the NASA Ames Airspace Operations Laboratory (AOL) to verify and develop the IDM concept.[1,2] While these experiments have produced important results, the resources and time required to conduct real-time studies with human participants limits the number of experimental conditions that can be tested as well as the overall number of HITL simulations that can be conducted. IDM concept development would benefit from the evaluation of a wider variety of capabilities and concepts across a broader range of possible conditions. An automated simulation capability that can closely mimic the HITL simulation capabilities, automate both the human

components and the collaboration between operational systems, and speed up the real-time aircraft simulations could thus provide real benefit. Such a capability would allow for parametric studies to be conducted that complement the HITL simulations, for example, by identifying breaking points and parameter values at which significant changes in system behavior occur. Such a capability could also be used to simulate background traffic during HITL simulations, improving realism.

NASA is also developing the Shadow Mode Assessment using Realistic Technologies for the NAS (SMART-NAS) Test Bed (SNTB) – a platform that is designed to enable high-fidelity HITL and automation-in-the-loop simulations by allowing operational systems to be integrated with high-fidelity models.[3,4] The SNTB is envisioned to leverage cloud-based services to provide cost-effective scalability for large, multi-facility, multi-organization simulations, and will ultimately enable complex, gate-to-gate HITL experiments for concepts such as IDM. Therefore, the automated simulation capability will be built to fit within the SNTB framework. This paper describes the development of such a capability, its initial validation, and an example application, the results of which are compared to results from similar HITL simulations.

II. Background

In Traffic Flow Management (TFM), demand for capacity-constrained airspace or airports is controlled by a series of Traffic Management Initiatives (TMIs), which use departure delays and airborne delays, as well as pre-departure and airborne reroutes, to manage access to the constrained resources. Two key TFM systems that exist in both current and planned future operations are the TFMS and TBFM systems, both of which are described below. Both systems address imbalances between demand and capacity, but in different operational contexts and different timeframes: whereas TFMS is more strategic in nature, with a planning horizon of hours and geographical reach that spans across the NAS, TBFM is more locally targeted to terminal airspace surrounding airports, and has a shorter, more tactical time horizon. The capabilities comprising these two systems are very powerful, but address different parts of the capacity management problem without close coordination. IDM aims to integrate and coordinate demand management across these systems to work together more effectively.[1] Within the IDM concept framework, TFMS may redistribute demand from an overloaded resource to one with reserve capacity, or allocate delay so that it is calibrated to variations in demand, and better matched to the demand that TBFM can accommodate. This should allow TBFM to manage delivery to the capacity-constrained destination more efficiently. A short overview of TFMS, TBFM and IDM is given below. A more detailed description of these concepts is available in Ref. 1.

A. Traffic Flow Management System (TFMS)

TFMS is a set of planning tools and capabilities that support NAS-wide traffic flow management and monitoring. In current operations, this is achieved primarily through Ground Delay Programs (GDPs) and Airspace Flow Programs (AFPs). GDPs and AFPs are TMIs put in place by the Air Traffic Control System Command Center (ATCSCC) when the capacity at a particular airport or airspace, respectively, is expected to fall well below the expected demand. These programs delay the departure of flights destined for the affected airport or airspace to reduce demand to manageable levels. Departure delay is assigned through TFMS at least 45 minutes before departure and often many hours before the flight will arrive at the constrained airport or airspace. Operators may choose to re-route flights around the constraints associated with an AFP, but their options are often restricted to a re-route advisory specified by the ATCSCC.

The IDM concept utilizes a new TMI called the Collaborative Trajectory Options Program (CTOP). New CTOP-specific capabilities within TFMS support a more complex characterization of the reduced capacity problem using multiple constraints, called Flow Constrained Areas (FCAs). Like GDPs and AFPs, CTOP is initiated by the ATCSCC when the capacity at a particular airport or airspace is expected to fall well below expected demand. When airspace is constrained, CTOP allows for greater flexibility than an AFP by allowing airlines to route their flights out of the associated FCA, or across a different FCA that has capacity available. CTOP allows flight operators to submit a preference-weighted set of alternative routes called a Trajectory Option Set (TOS) from which the program can select. CTOP distributes traffic to minimize congestion and reduce demand to manageable levels by delaying departures and allocating trajectories from the TOS to flights scheduled to pass through the specified FCAs. Departure delay is assigned via the TFMS's Expect Departure Clearance Time (EDCT). Like GDPs and AFPs, CTOP provides operators with advance notice of constraints and some opportunity to "swap slots" (exchange assigned arrival times) among their flights to reduce the impact on their operations.

Although the objective of these TFMS tools is to set departure times and routes such that maximum rates at capacity constrained resources are not exceeded, several factors impact a flight's actual arrival time at the resource. For example, EDCTs may not be precisely met. A flight is considered to be EDCT compliant if it departs within +/-

5 minutes of the EDCT, but in today's operations, many flights depart outside of EDCT compliance limits.[6] Additionally, winds and changes in trajectory due to weather conditions will affect flight time. Thus, using only TFMS tools, there is no guarantee that resource constraints will be satisfied.

B. Time-Based Flow Management (TBFM)

As described by Refs. 1 and 5, TBFM is an arrival scheduling tool that was developed to improve traffic delivery into capacity-constrained airports or airspace resources. TBFM creates a schedule for all inbound flights which provides a Scheduled Time of Arrival (STA) for each flight at specific constraint satisfaction points (CSPs), such as the runway threshold, final approach fix, or meter fix located near the boundary between en-route airspace which is controlled by an Air Route Traffic Control Center (referred to as a Center) and the terminal area around the airport which is controlled by the Terminal Radar Approach Control (TRACON). Arrivals are sequenced by a "first-come first-served" scheduling algorithm according to the Estimated Time of Arrival (ETA) of each flight at the relevant CSP. An STA is then assigned to each flight to satisfy the required inter-arrival spacing for a sequence of fixes through to the final CSP (typically the runway threshold or final approach fix). The arrival sequence and assigned STA for each flight continues to change as the flight ETA changes, up until the flight crosses a specified TBFM freeze horizon. This is defined at a particular time horizon or geographical distance from the CSP. At that point, the flight's STA is "frozen" and the target arrival time at the CSP is fixed. Air traffic controllers are then responsible for managing aircraft using delay vectors and speed clearances to meet these assigned STAs.

Flights that depart from airports within the TBFM freeze horizon are referred to as "TBFM-internal" departures. These flights can receive delay on the ground in order to ensure that they fit into the overhead stream being controlled by TBFM. Their departure times are obtained using a TBFM departure scheduling function that assigns a departure time corresponding to an available arrival slot. Upon take off, flights can receive further airborne delay from TBFM, but this is typically small because the departure scheduling process ensures that these flights are departing into gaps in the overhead stream.

Coupled scheduling and extended metering are two new TBFM features that have been introduced to improve performance. Without these features, TBFM arrival metering uses a single freeze horizon and schedule for each arrival flow. Coupled scheduling and extended metering include multiple coupled or uncoupled schedules for each flow, to a series of CSPs, each with its own freeze horizon. This delivers a more consistent traffic flow to the TRACON boundary by smoothing out errors in trajectory predictions.

C. Integrated Demand Management (IDM)

As stated above, IDM research is focused on the integration of the strategic TFMS and tactical TBFM concepts to manage delivery of demand to constrained resources. HITL (e.g., Refs. 1, 2) and human-out-of-the-loop experiments, in which procedures are simplified and subject matter experts are not used to control traffic (e.g., Ref. 6), have been completed at NASA Ames by researchers in the AOL to investigate the operational feasibility of the IDM concept under realistic conditions. In these experiments, air traffic controller participants solve simulated flow management problems using CTOP and TBFM. Air traffic is simulated using the Multi-Aircraft Control System (MACS), which is a high-fidelity air traffic control simulation environment for prototyping scheduling systems and simulating air traffic.[7] During a HITL simulation, MACS is used to run various workstation types for pseudo-pilots and air traffic controllers. In conjunction with MACS, an emulation of CTOP, called nCTOP (NASA CTOP),[1] was constructed to emulate the key functions of the TFMS version of CTOP, with some added functionality from the original CTOP concept that may be introduced in the operational system in the future. The CTOP functions modeled include setting capacity constraints at an FCA, automatically allocating trajectories from airline provided Trajectory Options Sets (TOSs) and assigning departure delay to pre-departure flights in order to balance the predicted demand at the FCA according to its capacity limit.[1] Inputs to nCTOP include FCA capacities, departure schedule, and TOSs. Based on the calculated EDCTs and TOS allocation, nCTOP outputs updated routes and departure times to all MACS stations through a simulation manager. MACS stations for each pilot and controller communicate with all other MACS stations in the simulation, updating aircraft positions. TBFM is used for arrival scheduling, and allows HITL simulation participants to schedule departure times for flights that depart from inside their destination's TBFM freeze horizon, called internal departures, to fit into the overhead stream based on calculated Scheduled Times of Arrival (STAs) at the meter fix and runway threshold. This is referred to as internal departure scheduling. The operational TBFM system (versions 4.2.3 and 4.4.1) is used, with NASA modifications that implement Terminal Sequencing and Spacing (TSAS) (in version 4.2.3 only), extended metering, and a NASA developed airport adaptation. This system, including the stated modifications, will henceforth referred to as hTBFM.

III.   Automation Functionality and Simulation Implementation

Whereas a wealth of information and insight is gained through HITL simulations, these are expensive and time-consuming activities, especially for the IDM concept, which requires five hours or more to simulate flights from pre-departure at origin airports to arrival at the destination airport. As such, few HITL simulations have been completed and thus only a small subset of potentially interesting problems have been studied. The ability to run additional scenarios in an automated and fast-time fashion would be immensely useful. Such a simulation capability could be used to test various scenarios before using them in a HITL simulation, to perform parameter studies to quantify system performance of the IDM concept under various circumstances, and to simulate background traffic in HITL simulations.

Automated simulation of the IDM experimental setup described in Section II.C requires the integration of a CTOP emulator, an air traffic simulator, and a TBFM emulator. These components are shown in the software flow diagram in Figure 1. Although the HITL simulation capabilities include these three components, several factors make automating and speeding up these simulation tools difficult. The hTBFM tool only runs in real time, and has challenges associated with its integration with other software. A number of processes within hTBFM are also designed to be performed manually through the hTBFM interface, including the scheduling of internal departures. The nCTOP tool was designed as a decision support tool for air traffic managers and thus also requires human input.
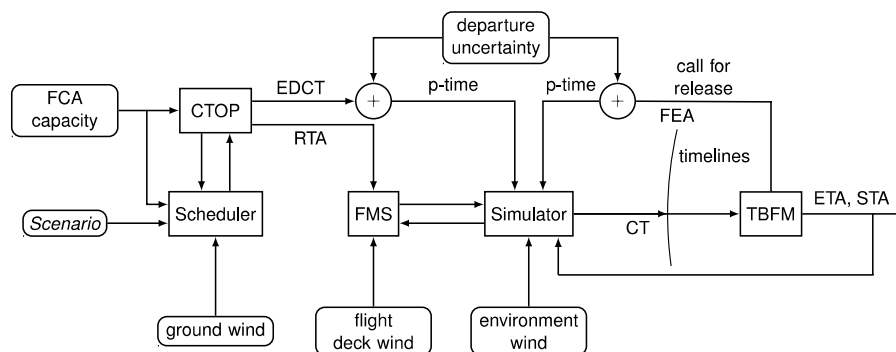

Figure 1. Flow diagram indicating information flow between components of the automated simulation capability.

The automated simulation capability under development utilizes a number of simulation tools already developed at NASA, including MACS, an emulation of CTOP called nCTOP, and an emulation of TBFM developed for the Optimized Route Capability (ORC) tool.[10] MACS and nCTOP are also used in the IDM HITL simulations. Although all of these tools exist, none can be used as-is and some amount of modification or enhancement is required. These tools and the associated changes made to them are described in more detail below.

A. CTOP Emulator

The CTOP emulator used in the IDM HITL simulations, nCTOP,[1] was designed for interactive use by simulation participants. Through the GUI, participants can visualize demand and set capacities. Nominal flight information is sent from MACS to nCTOP and TOS information is specified in a file. Given this information, nCTOP assigns trajectories and EDCTs which can then be sent to MACS to update the simulation in progress. The development of nCTOP has been closely tied to MACS with various Application Programming Interfaces (API) developed to enable communication between the two tools. Given that the IDM concept is under development, nCTOP is continually updated to add functionality and respond to user requests. The automated simulation capability described in this paper must be able to keep up with these frequent changes to remain relevant. The use of MACS is essential to the rapid integration of new versions of nCTOP into the automated simulation capability.

Given that nCTOP was designed as a decision support tool for HITL participants, various functions within the tool require automation to be able to run without participant intervention. Communication between MACS and nCTOP is performed through a TCP/IP connection. During a HITL, the TCP/IP server is started by MACS and an nCTOP operator uses the nCTOP GUI to connect to the server. A modification was made to nCTOP to automate the connection to the TCP/IP server. Additionally, nCTOP was modified to input a configuration file to automate the execution of various commands, including: set program start time, set FCA capacities in increments of 15 minutes, run balance algorithm, run TOS selection algorithm, send TOS selection and EDCTs to MACS, and exit nCTOP. These modifications enable fully automated execution of nCTOP commands and transfer of flight plans and EDCTs output from nCTOP to MACS in order to modify a traffic simulation in progress. Development of nCTOP has been focused

on real-time simulations. Additional work needs to be done to synchronize nCTOP and MACS when the MACS simulation is run faster than real-time. This will be addressed in future work.

B. Air Traffic Simulator

A variety of air traffic simulation tools have been developed at NASA Ames, each designed with different objectives. Most notably, the Future Air Traffic Management Concepts Evaluation Tool (FACET)[8] and Airspace Concept Evaluation System (ACES)[9] were developed to enable the development and evaluation of advanced ATM concepts. Although either of these tools could be used to simulate air traffic for the automated simulation capability, MACS was chosen for a variety of reasons. First, MACS is the air traffic simulation tool used in IDM HITL simulations at NASA. Given that a primary objective of the automated simulation capability is to automate and speed up the HITL simulations, using MACS provides the easiest comparison between the automated simulation results and HITL results. Furthermore, the generation of scenarios is currently a lengthy and labor-intensive process. By using MACS, the automated simulation capability can make use of the scenario information of HITL simulations, as well as the scenario generation capabilities of the SNTB, which has been developed for MACS scenario generation. Using the same scenario format and files reduces workload and potential errors in translating a HITL scenario to an automated simulation capability scenario. Finally, as discussed above, nCTOP software is closely tied to MACS.

In order to automate the HITL process, several user-initiated actions must be automated, specifically: set simulation speed, set simulation run time, open TCP/IP socket for communication with nCTOP, and exit MACS. A batch mode functionality had already been implemented in a command line version of the MACS software and was easily added to the MACS distribution used in the IDM HITL simulations. This capability allows the user to set simulation speed and duration in a configuration file. MACS was also modified to automatically connect to the TCP/IP server required for communication between MACS and nCTOP. In the automated simulation capability, the server is started by a simulation manager, which is described in more detail in Section D.

C. TBFM Emulator

A number of TBFM emulators exist, including one developed for the Optimized Route Capability (ORC),[10] and schedulers in the Center TRACON Automation System (CTAS) and MACS. None of these emulators include the TBFM capability to schedule internal departures. However, the ORC scheduler runs very fast, provides a high-fidelity model of the operational TBFM scheduling system, and is easily accessible for further development. For these reasons, it was chosen as the basis for the TBFM emulator developed in this paper.

A number of features of the operational TBFM system are not modeled by the ORC scheduler. The most relevant of these include freezing flight STAs based on a specified freeze horizon, scheduling internal departures pre-departure, extended metering and coupled scheduling. The scheduling of internal departures is a manual process in the operational TBFM system.

The underlying ORC scheduler was further developed and incorporated into the TBFM emulator for the automated simulation capability described in this paper (henceforth referred to as eTBFM), with an adaptation for Newark Liberty International Airport (EWR) modified from the EWR TBFM adaptation used in the HITL simulations for use by the ORC scheduler. The ORC scheduler is integrated into the eTBFM wrapper function that takes as an input timelines generated by MACS that provide ETAs at the meter fix, and an indication of which flights have crossed the freeze horizon. This wrapper function automatically schedules internal departures a set time before their expected departure time (typically EDCT), also input from MACS. This allows modeling of TBFM internal departure scheduling in one of two ways. In the first scheduling paradigm, internal departures are fitted into the overhead stream with no priority given to airborne flights (sometimes referred to as 'checkbox on', referring to a checkbox on the TBFM departure scheduling window that enables this functionality). In the second, priority is given to airborne flights, meaning internal departures are held on the ground until a suitable gap exists in the overhead stream (sometimes referred to as 'checkbox off'). This is consistent with the scheduling performed manually in the operational TBFM system. eTBFM is also capable of accepting ETAs that change dynamically, in the form of timelines that update meter fix ETAs. Once a flight crosses the freeze horizon, eTBFM considers the calculated STA frozen, and does not recalculate it as its ETAs change. STAs for internal departures are frozen the moment they take off. A flow diagram showing the algorithm used for scheduling internal and external flights, using eTBFM, is shown in detail in Figure A.1 in Appendix A.

In the current version of eTBFM, neither extended metering nor coupled scheduling are modeled. This functionality will be incorporated into eTBFM in future work.

D. Automated Simulation Manager

In order to run the simulation fully automated and in batch mode, a simulation manager is required to synchronize the components discussed above. One of the simulation manager's functions is to set up the server required for

communication between MACS and nCTOP. In the current configuration, such external communications need to be set up manually by specifying an IP address and a port number on both the server side (MACS) and client side (nCTOP and simulation manager) as shown in **Figure 2**.

As shown in **Figure 2**, the simulation manager is connected directly to eTBFM. MACS generates ETA timelines at specified locations, notably the TBFM freeze horizon, and the arrival meter fixes for the airport of interest. These timelines are passed to eTBFM, which uses them directly as meter fix ETAs, and to indicate whether or not flights have crossed the TBFM freeze horizon. As shown in the flow diagram of the eTBFM scheduling algorithm in Figure A.1 in Appendix A, STAs of external flights are frozen once they pass the TBFM freeze horizon. In contrast, STAs of internal flights are frozen as they depart from their respective origin airports.

As a development tool, a graphical user interface (GUI) was built to facilitate processes such as setting up communication with MACS and controlling MACS simulation. This is shown in **Figure 3**, and allows the user to perform various tasks, such as: specify the scenario file to be run by MACS; start MACS and run the simulation; start nCTOP and connect with MACS; set MACS simulation run speed (from real-time (1x) to up to 20 times faster than real time (20x)); and start data collection.

As mentioned previously, a key feature of the automated simulation capability is the ability to run tens or even hundreds of simulations with varying scenarios and parameters without requiring any manual interruption. To achieve this, the GUI functionality was automated with a command line interface. During command line execution, the TCP/IP sockets are set up and connected with external tools automatically during launch. Next, a number of simulation parameters, such as simulation speed and simulation duration, are automatically loaded into MACS from user-specified batch executable files. The command line interface allows for the sequential and automatic execution of multiple scenarios.

A detailed flowchart of the automated simulation software is given in Figure A.2 in Appendix A. As mentioned in the previous section, the software takes various scenario files and simulation settings as the input. The software then launches MACS, nCTOP and eTBFM sequentially and automatically to generate dynamic arrival flight schedules.
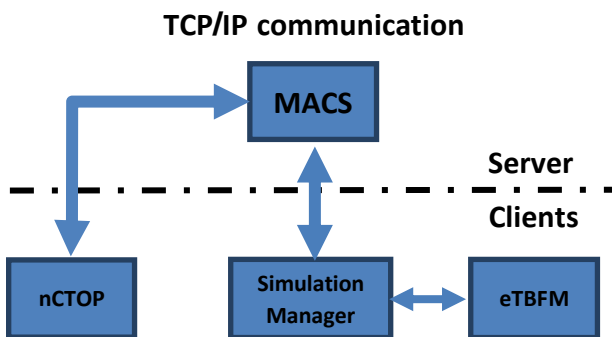

Figure 2. MACS external communication

Figure 3. A Graphical User Interface (GUI) to set up communication between MACS and nCTOP and set MACS parameters.

IV.   Verification and Validation

A. Air Traffic Simulator

In order to conduct fast-time simulations, MACS software that was used in the HITL simulations was run at a faster speed. Although the functionality already existed in MACS, the trajectory integrity was never validated in the fast-time mode. Therefore, flight tracks produced by MACS running at five, ten and twenty-five times faster than real time (5x, 10x, 25x) were compared to flight tracks generated by MACS running in real time (1x). The traffic scenario used is identical to that used in Ref. 6 consisting of flights departing various airports and with destination EWR. The scenario was run both with and without environmental winds. The wind data used are identical to the "mild" and "heavy" wind conditions of the experiments presented in Ref. 6 with average wind speeds at an altitude of 30,000 ft of 45 knots and 60 knots and up to 130 knots and 160 knots maximum, respectively. MACS was run at 1x, generating reference real-time data, and then again at 1x and at 5x, 10x and 25x. During the simulation, flights are allowed to depart for 4 hours. Beyond this time, trajectories of airborne flights continue to be simulated until the final flight lands at roughly 5 hours and 30 minutes into the simulation. A detailed comparison was made between the fast-time flight tracks and the corresponding real-time flight tracks. Track data were generated at 12-second intervals. Each track was

resampled to obtain latitude, longitude and altitude information at 60-second multiples of simulation time. For the given set of 196 flights, over 22,000 such points were generated. For each flight and each simulation speed, comparisons between each of these resampled points and the corresponding resampled points in the reference real-time track were made. Tracks were compared to the reference real-time tracks based on lateral and vertical distance between corresponding track points and total flight time. Lateral distance between corresponding track points is decomposed into components parallel to the real-time track ("along-track") and perpendicular to the real-time track ("cross-track"). Box and whisker plots are used to present the comparison results. Edges of the whiskers are at the 10th and 90th percentile of the data, lower and upper boundaries of the box are at the 25th and 75th percentile, and the 50th percentile is indicated by a red line within the box.

Results of these comparisons are shown in **Figure 4** through **Figure 6**. Results are shown for the scenario without wind only as results for the scenario with wind are similar. Along-track differences at each simulation speed compared to the corresponding reference real-time tracks are shown in **Figure 4**. Note that the data for 1x are a measure of repeatability of the simulation. The differences between the 1x tracks and the reference 1x tracks are small, but nonzero, indicating that the simulation cannot be repeated precisely. It can be seen that at all simulation speeds, the absolute values of the 25th and 75th percentiles of along-track distances from 1x tracks are all below 0.25 nmi. Cross-track distances were also analyzed and found to be quite small with absolute values of the 25th and 75th percentiles of the cross-track distances from 1x tracks below 0.01 nmi.
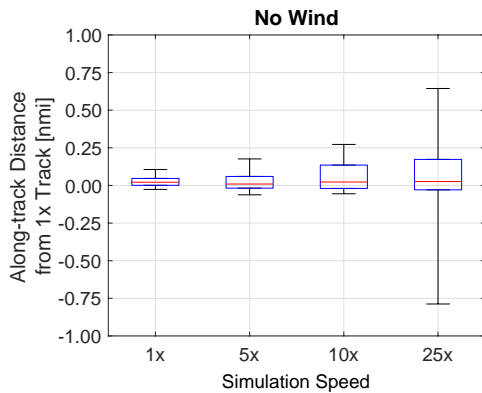

Figure 4. Component of distance from real-time track points to associated fast-time track points along the real-time track. A positive value indicates that the fast-time track is ahead of the real-time track.
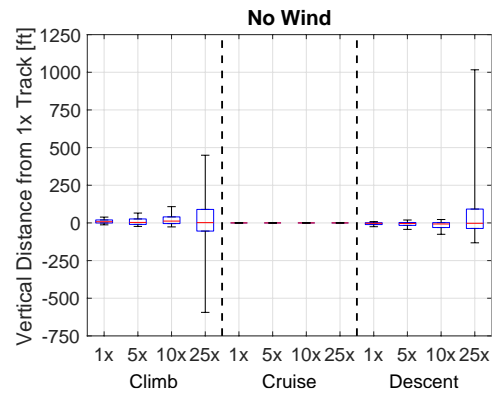

Figure 5. Vertical distance from real-time track points to associated fast-time track points along the real-time track. A positive value indicates that the fast-time track is above the real-time.

Vertical differences in tracks at each simulation speed are compared to the corresponding reference real-time tracks and shown in **Figure 5**. It was found that most of the differences in vertical position occurred during climb and descent. For this reason, results are presented for the climb, cruise and descent phases of flight separately. These phases were found using the real-time flight tracks with the climb phase extending from departure to the time at which the flight track reaches cruise altitude and descent extending from initial descent from the cruise altitude to the landing. A total of about 1,500, 14,400 and 4,400 resampled track points make up the data sets for the climb, cruise and descent phases of all flights, respectively. During the cruise phase, over 90% of resampled track points are within one foot of the corresponding reference track point (except for the 25x tracks with wind, for which 89% are within one foot).

Total flight times were also compared, with results shown in **Figure 6**. Note that results are not shown for the 25x simulations. This is due to an error in the MACS simulation code which caused many flights in the 25x simulations to not land. Track data up to the landing point are valid, so the position comparisons could be made,
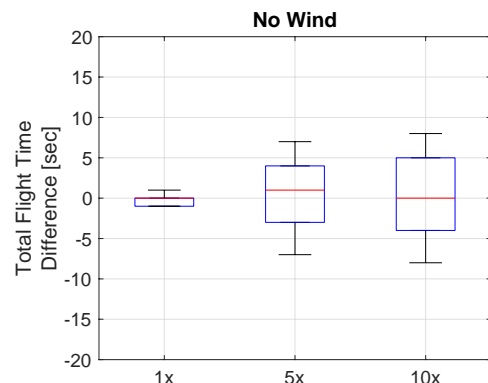

Figure 6. Difference in total flight time between the real-time track and fast-time tracks. A positive value indicates that the flight time of the fast-time track is longer than the real-time.

but an accurate simulation landing time could not be obtained for the 25x flights and thus this data is not shown in **Figure 6**.

Whether or not these differences in location and flight time are acceptable depends on the application. For IDM development, the timing of arrivals at constraint satisfaction points (CSPs) is of utmost importance. For the problem discussed in Section IV.C in which flights arriving into EWR are scheduled by CTOP and then TBFM as the flight enters the freeze horizon, maximum arrival capacity at the EWR runway is 44 aircraft per hour. At this rate, the time between arriving flights is about 80 seconds. Spacing at the various CSPs leading into the EWR runway will have inter-arrival spacing of roughly 80 seconds or more as flights on different flows must be spaced properly so that the appropriate rate is achieved when the flows merge. Clearly, errors in CSP arrival time of +/- 10 seconds is significant and would affect the resulting TBFM schedule. However, the IDM study takes into account various uncertainties. EDCTs are met with some error that is obtained by sampling from a distribution found in GDP data. Errors of +/-10 seconds are 3.33% of the EDCT compliance window of +/-300 seconds (+/- 5 minutes). Although small, this is still noteworthy and could be offset by adjusting the departure error sampling distribution. Given these results, MACS simulation speeds of up to 10x are considered acceptable for IDM studies.

B. TBFM Emulator

An initial validation of eTBFM was completed by comparing outputs of the emulator and hTBFM (TBFM version 4.4.1, as run in Ref. 2), given dynamic timelines at the freeze horizon and runway threshold, and internal departure EDCTs. One hour of arrival traffic from the distributed traffic scenario described by Ref. 6 was simulated, which captures the arrival of 39 flights to EWR from a variety of origins (domestic and international), operating a variety of aircraft types. No wind is simulated, and inter-arrival spacing of 0.4 nmi is imposed at the final approach fix. Given the demand level, this results in an airport arrival rate of 50 aircraft per hour – a high rate for EWR. All three primary arrival fixes at EWR are used: PENNS, SHAFF and DYLIN.

Inputs to eTBFM include a series of timelines generated by MACS, which consist of meter fix ETAs that can change over time and freeze horizon crossing times, and internal departure EDCTs. Using hTBFM, internal departures are scheduled manually using the paradigm in which internal departures are fitted into the overhead stream with no priority given to airborne flights. A traffic manager monitors the hTBFM internal departure timeline and schedules departures at a suitable time, as determined by the traffic manager, in advance of their EDCTs. In this experiment internal departure scheduling using hTBFM was done (manually) as close to 20 minutes before the flight EDCTs as possible. In contrast, eTBFM schedules internal departures automatically 20 minutes before their EDCT.

hTBFM applies extended metering, where each flow includes two sequential uncoupled schedules, first to a metering point upstream of the meter fix, and secondly to the meter fix. Each schedule has its own freeze horizon, defining an extended metering region and a meter fix region. eTBFM does not apply extended metering, including only a schedule to the meter fix. For this experiment the freeze horizon used for eTBFM is that of the meter fix region in hTBFM. Delays, ETAs and STAs are therefore compared for the meter fix region only.

Results are presented in Figure 7 and Table 1 below. Figure 7 shows fairly close correlation between the ETAs and STAs calculated by eTBFM and hTBFM at the meter fix (Figure 7a) and runway threshold (Figure 7b). Although eTBFM uses the same airport adaptation as hTBFM, results are not expected to match precisely given that some simplifying assumptions are built into the eTBFM scheduler, and the scheduling of internal departures in hTBFM is manual. Average metering delays calculated by eTBFM and hTBFM (in the meter fix region only), and average differences in ETAs and STAs, are shown in Table 1, with standard deviations included in brackets. Average delays calculated by eTBFM are higher than those of hTBFM at the meter fix and at the threshold (by 63 and 44 seconds respectively). The ETAs generated by eTBFM are, on average, earlier than those generated by hTBFM. The STAs calculated by eTBFM at the threshold are, accordingly, also slightly earlier than those calculated by hTBFM. However, because of the higher meter fix delays calculated by eTBFM, eTBFM calculated STAs at the meter fix are later than those calculated by hTBFM. Standard deviations in metering delay, and in the differences between ETAs and STAs, are large (between 75 and 135 seconds). This is because TBFM is very sensitive to flight sequencing, and especially to when internal departures are scheduled. When TBFM is used operationally, controllers are shown and expected to meter to ETA and STA values that are truncated to the nearest minute, thus errors within a minute are considered acceptable. However, future research will work to reduce the differences between eTBFM and hTBFM shown here.

For the 10 internal departures simulated, Figure 7c compares the scheduled times of departure calculated by eTBFM and hTBFM. There are noteworthy differences, primarily because of differences in flight sequencing. Sequencing is greatly affected by the exact timing of when the flight is scheduled, which is a manual process in hTBFM, and was therefore not possible to control precisely.

Average scheduling delays calculated by eTBFM and hTBFM are shown in Table 2. Despite the differences in sequencing, eTBFM and hTBFM show very similar results, with an average difference of only 4 seconds. This is a

key output of eTBFM, but needs to be confirmed with a validation simulating a larger number of internal departures. The high standard deviations are a direct consequence of the differences in flight sequencing.
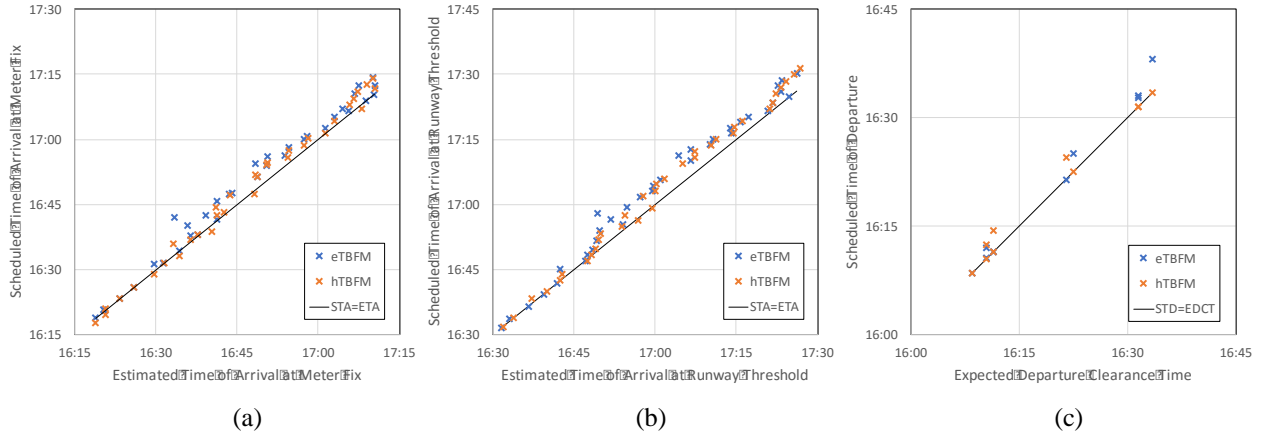

(a)   (b)   (c)

Figure 7. Comparison of ETAs and STAs between eTBFM and hTBFM, at (a) the arrival metering fix, and (b) the runway threshold. (c) Comparison of internal departure scheduled departure times calculated by eTBFM and hTBFM.

Table 1. TBFM emulator validation results, comparing metering outputs from hTBFM to those of eTBFM. Standard deviations shown in brackets.

	Meter Fix		Threshold	
	[Seconds]			
Avg. hTBFM metering delay (standard deviation)	80	(104)	136	(106)
Avg. eTBFM emulator metering delay (standard deviation)	143	(131)	180	(135)
Avg. ETA Error: hTBFM-eTBFM (standard deviation)	19	(75)	52	(77)
Avg. STA Error: hTBFM-eTBFM (standard deviation)	-43	(104)	7	(100)

Table 2. TBFM emulator validation results, comparing internal departure scheduling outputs from hTBFM to those of eTBFM. Standard deviations shown in brackets.

	Seconds	
Avg. hTBFM internal departure scheduling delay (standard deviation)	66	(72)
Avg. eTBFM internal departure scheduling delay (standard deviation)	70	(90)
Avg. hTBFM calculated scheduled departure time - eTBFM calculated scheduled departure time (standard deviation)	-4	(129)

V. Example Application

The motivation for the development of the automated simulation capability is to be able to test IDM concepts in an automated fast-time manner. In order for the automated simulation capability to be useful, results must be qualitatively similar to those of HITL simulations. That is, quantitative results should show similar trends, allowing researchers to draw similar conclusions about the effectiveness and impact of IDM concepts whether using the automated simulation capability or HITL simulation. Two experiments were conducted using the automated simulation capability described in this paper to show an example application, and to allow some of the results to be compared to results from similar HITL simulations. Note that because the scenarios used in these experiments are not exactly the same as those used in the previous HITL simulations and HITL simulation participants take actions that are not repeatable, individual flight delays and flight sequences will differ, but average delays should be similar.

The experiments include all components of the simulation capability developed and were run automatically at real-time speed. Both experiments simulate a typical 5-hour traffic scenario including only flights destined for EWR, as was done in Ref. 1, 2 and 6. Weather and wind impacts are not included in the simulation, which is one key difference compared to previous HITL simulations. A single FCA is applied at EWR, set at a rate of 44 aircraft per hour. Inter-arrival spacing of 0.4 nmi is set at the final approach fix in eTBFM, which results in a 44 aircraft per hour rate at the airport in the absence of wind given the demand supplied by nCTOP. In contrast to the eTBFM validation experiment in Section IV.B, for these experiments the freeze horizon used for eTBFM is that of the extended metering region in

previous HITL simulations running a similar scenario[1,2,6]. This means that TBFM airborne delays are considered across the region covering both the extended metering and meter fix regions from previous HITL simulations. Departure errors used in these experiments are identical to those used in previous HITL simulations,[1,2,6] which were drawn from historical data. Using these departure errors, 70% of external flights are EDCT compliant (i.e., flights depart within +/- 5 minutes of their assigned EDCT) and 100% of internal departures are compliant with their TBFM scheduled departure time, or "call for release" time (i.e., flights depart within +2/-1 minute of their assigned time).

The two experiments differ only in the paradigm used for the scheduling of internal departures – prioritizing airborne flights (as is typically done operationally today), and not prioritizing airborne flights. These experiments should confirm the conclusions of the HITL simulations[1,2,6] suggesting that, under the IDM concept, when airborne flights are not prioritized over internal departures, internal departure scheduling delays are reduced, while the level of airborne delay that must be absorbed by TBFM is still operationally feasible.

Results are shown below. Figure 8 shows ground delays assigned by nCTOP by flight, as a function of their runway threshold STA. Ground delays assigned by nCTOP are independent of the paradigm used for the scheduling of internal departures. Only one set of delays is therefore plotted. The magnitude of ground delays found in these experiments (with a maximum of 39 minutes) are similar to those reported in Ref. 2 (with a maximum of 46 minutes), which simulates a similar scenario.
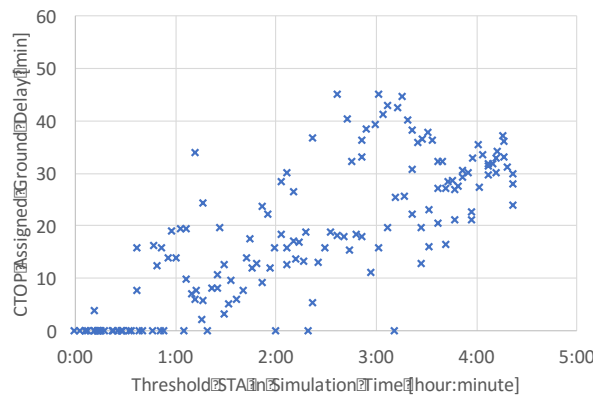

Figure 8. Ground delay minutes assigned by nCTOP by flight.

Figure 9 shows eTBFM internal departure scheduling delay minutes by flight for the two experiments, as a function of their runway threshold STA. As expected, by not prioritizing airborne flights over internal departures, internal departure scheduling delays are reduced. The average internal departure scheduling delay decreases from just over 18 minutes to just under 3 minutes. The average internal departure scheduling delay with no priority given to airborne flights is consistent with results from Ref. 2, which simulates a similar scenario and reports average internal departure scheduling delays of 2 minutes 10 seconds.

(a)   (b)
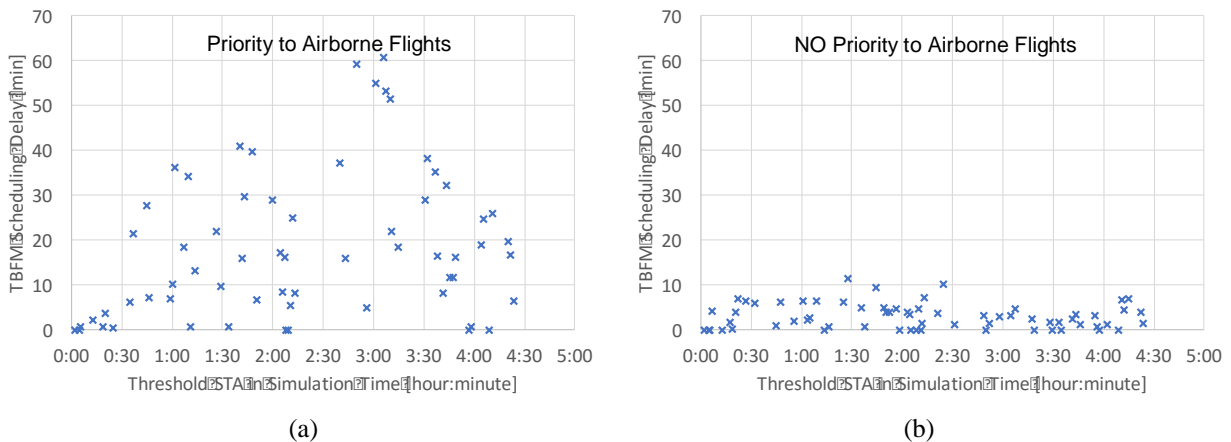
Figure 9. eTBFM internal departure scheduling delay minutes by flight with (a) priority given to airborne flights, and (b) NO priority given to airborne flights.

10

Table 3 shows the distribution of TBFM airborne delay for the two simulations. Given the airspace over which TBFM metering is simulated, airborne delay is considered operationally acceptable if it is below 7 minutes, marginal if it is between 7 and 14 minutes, and unacceptable if it is greater than 14 minutes.[*] As can be seen, in both cases very few flights have operationally unacceptable airborne delay (between 0% and 1%). However, with no priority given to airborne flights, the percentage of marginal airborne delays increases (from 5% to 13%). However, the vast majority of flights still maintain acceptable levels of airborne delay (87%). The results with no priority given to airborne flights are consistent with results from Ref. 1 and Ref. 2, which simulate similar scenarios. Ref. 1 reports 1.5% unacceptable airborne delay, 17% marginal, and 82% acceptable, while Ref. 2 reports 7% unacceptable airborne delay, 28% marginal, and 65% acceptable.[†] The results in Table 3 suggest that the IDM concept simulated may enable TBFM to be run with no priority given to airborne flights, yielding the improvements in internal departure scheduling delay shown in Figure 9. These conclusions are consistent with the conclusions from previous HITL simulations.[1,2]

Table 3. Distribution of TBFM Airborne Delay

Simulation	TBFM Scheduling Paradigm	% TBFM Airborne Delay Acceptable (<7 min)	% TBFM Airborne Delay Marginal (7-14 min)	% TBFM Airborne Delay Unacceptable (>14 min)
1	Priority to Airborne Flights	94%	5%	1%
2	NO priority to Airborne Flights	87%	13%	0%

The average throughput at the runway, over the final four hours of simulation, with priority given to airborne flights, is 43.5 aircraft per hour. With no priority given to airborne flights, this increases slightly to 44.3 aircraft per hour, potentially indicating better utilization of the constrained resource. The throughput results also confirm that the target throughput of 44 aircraft per hour was achieved, as in Ref. 1 and Ref. 2.

VI.   Conclusions and Future Work

In this work, an initial automated simulation capability was developed to support IDM research and development. This simulation capability can be used to test various scenarios and parameters in order to identify candidate scenarios for HITL simulations. Although most of the software tools used in the HITL simulations were adopted for use in the automated simulation capability (with the notable exception of the version of TBFM used in the HITL simulations), all had to undergo some amount of modification to enable the automation of the simulation. As mentioned, the version of TBFM used in the HITL simulations is not suitable for use in an automated manner for a variety of reasons. Thus, an alternative TBFM emulator was built for the automation simulation capability starting from an arrival scheduler developed for the Optimized Route Capability (ORC) tool. Functionality of the newly developed TBFM emulator includes the implementation of two scheduling paradigms (prioritizing airborne flights and not prioritizing airborne flights), and the automation of internal departure scheduling in accordance with TBFM procedures. MACS and nCTOP were used to provide the same functionality in the automated simulation capability as in the HITL simulations. A simulation manager was built to coordinate this set of software. The automated simulation capability can be run in batch mode from the command line.

Verification and validation was performed on each component of the automated simulation capability separately. From MACS simulations, it was found that 80% of the differences between the real-time and the 5x trajectory points are within 0.25 nmi along-track, 0.02 nmi cross-track, 1 ft vertically in the cruise phase of flight, and 15 seconds in total flight time. Results from the eTBFM scheduler were found to match corresponding results found using hTBFM with average difference between STAs and ETAs at the threshold of 60 seconds and 30 seconds, respectively. Finally,

---

[*] In Ref. 1 and 2, airborne delays in the extended metering region are considered to be acceptable if they are below 5 min, marginal if between 5 and 10 minutes, and unacceptable if over 10 minutes. Airborne delays in the meter fix region are considered to be acceptable if they are below 2 min, marginal if between 2 and 4 minutes, and unacceptable if over 4 minutes. In this experiment, airborne delays are considered across the region covering both the extended metering and meter fix regions. Hence the thresholds for acceptable, marginal and unacceptable delays used for this work are calculated by adding the thresholds over the extended and meter fix regions in Ref. 1 and Ref. 2.

[†] Results from Ref. 1 and 2 quoted apply to the meter fix region only. In both studies the distribution of airborne delay is more acceptable in the extended metering region. Differences between the results from Ref. 1 and 2 are because of differences in the scenarios simulated, airport adaptation used, and procedures used for human-operator inputs.

all components of the automated simulation capability were used together to execute two simulations as a proof-of-concept. Results for TBFM required airborne delay and internal departure scheduling delay are comparable to those from previous HITL simulations using similar scenarios.[1,2]

Future work in the development of the automated simulation capability includes incorporating additional functionality and increasing simulation speed to faster than real-time. Increasing simulation speed requires modifications to nCTOP in order to synchronize times between nCTOP and MACS. The TBFM emulator will be further developed to include extended metering and coupled scheduling. In the current automated simulation capability, STAs and TBFM-scheduled internal departure times output from the TBFM emulator are not input to MACS to update the simulation. Future work will enable this communication and MACS speed control functionality will be used to simulate traffic manager actions to delay flights within the TBFM freeze horizon to meet STAs.

Ultimately, this automated simulation capability will be integrated into the SNTB. The SNTB is currently in development with plans to be a platform for research, development, and analysis of air traffic management concepts. MACS is used as the primary traffic simulation tool, which contributed to the choice of MACS for use in the automated simulation capabilities. Work done to automate the communication between MACS and other components of the automated simulation capability and run the simulations in batch mode will be useful additions to the SNTB platform. Scenarios involving more than a couple hundred flights require multiple instances of MACS to perform traffic simulation. SNTB has the required architecture to run and coordinate multiple instances of MACS. Once integrated into SNTB, this automated simulation capability will be scalable to NAS wide scenarios.

Acknowledgements

References

[1] SMITH, N., BRASIL, C., LEE, P.U., BUCKLEY, N., GABRIEL, C., MOHLENBRINK., C, OMAR, F., PARKE., B., SPERIDAKOS, C., and YOO, H., Integrated Demand Management: Coordinating Strategic and Tactical Flow Scheduling Operations, 16th AIAA Aviation Technology, Integration, and Operations Conference, 2016, AIAA 2016-4221.

[2] YOO, H.S., BRASIL, C., BUCKLEY, N., MOHLENBRINK, C., SPERIDAKOS, C., PARKE, B., HODELL, G., LEE, P.U., and SMITH, N.M., Integrated Demand Management: Minimizing Unanticipated Excessive Departure Delay while Ensuring Fairness from a Traffic Management Initiative, 17th AIAA Aviation Technology, Integration, and Operations Conference, 2017, AIAA 2017-4100.

[3] PALOPO, K., CHATTERJI, G.B., GUMINSKY, M.D. and GLAAB, P.C., Shadow Mode Assessment using Realistic Technologies for the National Airspace System (SMART NAS) Test Bed Development. 15th AIAA Aviation Technology, Integration, and Operations Conference, 2015, AIAA 2015-2794.

[4] ROBINSON, III, J. E., LEE, A., and LAI, C., "Development of a High-Fidelity Simulation Environment for Shadow-Mode Assessments of Air Traffic Concepts," RAeS and AIAA Modelling and Simulation in Air Traffic Management Conference, London, 14-15 November, 2017.

[5] TYO, R., and PRESSLER, C., Industry Day Briefing: Time-Based Flow Management (TBFM), Federal Aviation Administration Industry Day, April, 2014. from:
http://www.faa.gov/about/office_org/headquarters_offices/ato/service_units/pmo/industry_day/media/day_1_-_tbfm_bob_tyo_-_chris_pressler_-_final.pdf
[Retrieved June 12, 2017].

[6] YOO, H.S., MOHLENBRINK, C., BRASIL, C., BUCKLEY, N., GLOBUS, A., SMITH, N.M. AND LEE, P.U., Required time of arrival as a control mechanism to mitigate uncertainty in arrival traffic demand management, 35th IEEE/AIAA Digital Avionics Systems Conference, 2016, IEEE.

[7] PREVOT, T., Exploring the many perspectives of distributed air traffic management: The Multi Aircraft Control System MACS, In Proceedings of the HCI-Aero, 2002, pp. 149-154.

[8] BILIMORIA, K., SRIDHAR, B., CHATTERJI, G.B., SHETH, K. AND GRABBE, S. FACET: Future ATM concepts evaluation tool. *Air Traffic Control Quarterly*, January 2001, **9**(1), pp.1-20.

[9] SWEET, D.N., MANIKONDA, V., ARONSON, J.S., ROTH, K. AND BLAKE, M. Fast-time simulation system for analysis of advanced air transportation concepts. In *Proceedings of the AIAA modeling and simulation technologies conference* August 2002 (pp. 5-8).

[10] ZELINSKI, S., XUE, M., and BASSETT, P., Optimized Route Capability (ORC) Intelligent Offloading of Congested Arrival Routes, 16th AIAA Aviation Technology, Integration, and Operations Conference, 2016, AIAA 2016-4357.

Appendix A: Algorithm flow diagrams.

Figure A.1 shows the algorithm used by eTBFM developed for the scheduling of both internal and external arrival flights to the constrained airport. This includes both scheduling modes, that is, the mode in which internal departures are fitted into the overhead stream with no priority given to airborne flights, and the mode in which priority it given to airborne flights, meaning internal departures are held on the ground until a suitable gap exists in the overhead stream.
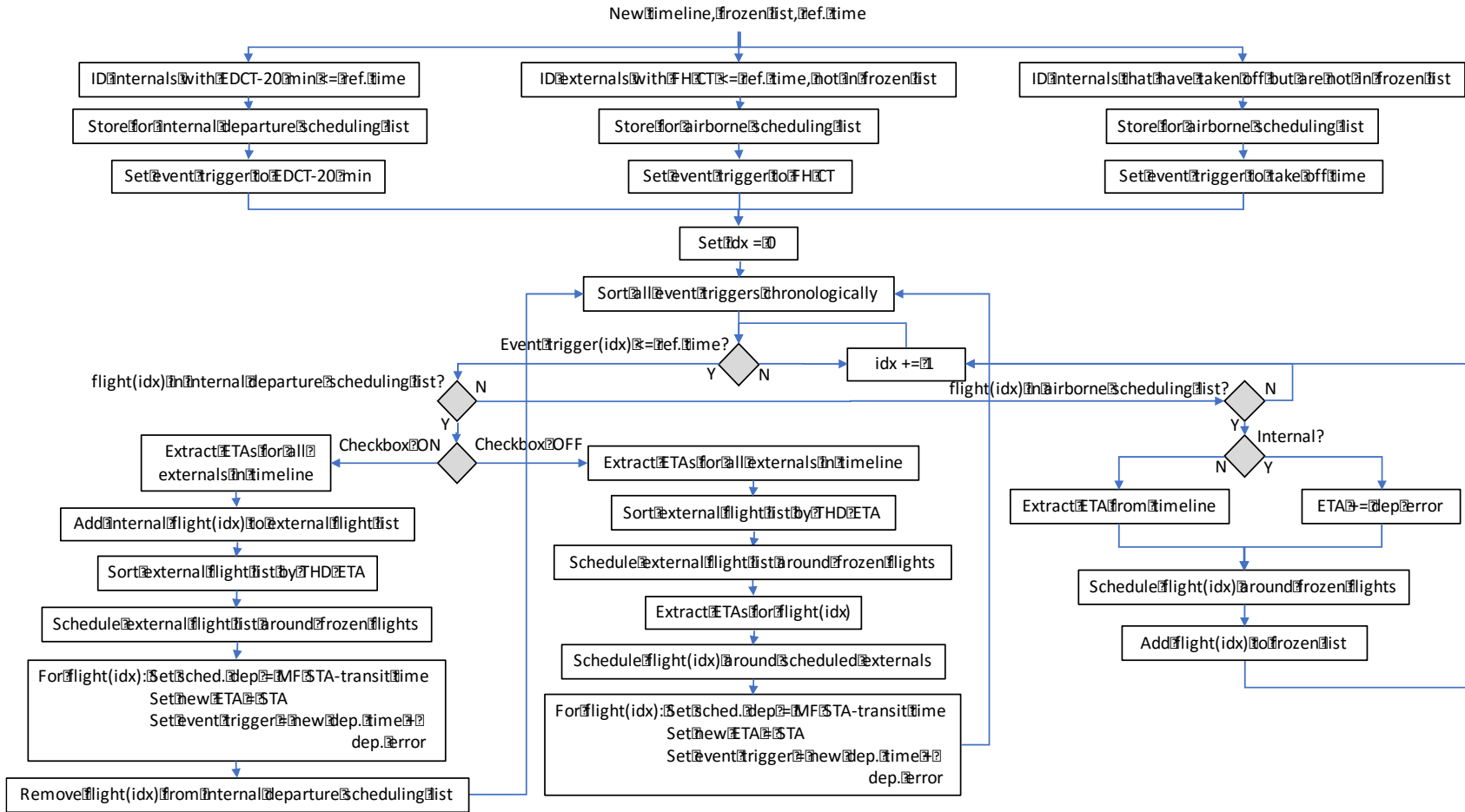

Figure A.1 eTBFM scheduling algorithm for internal and external flights.

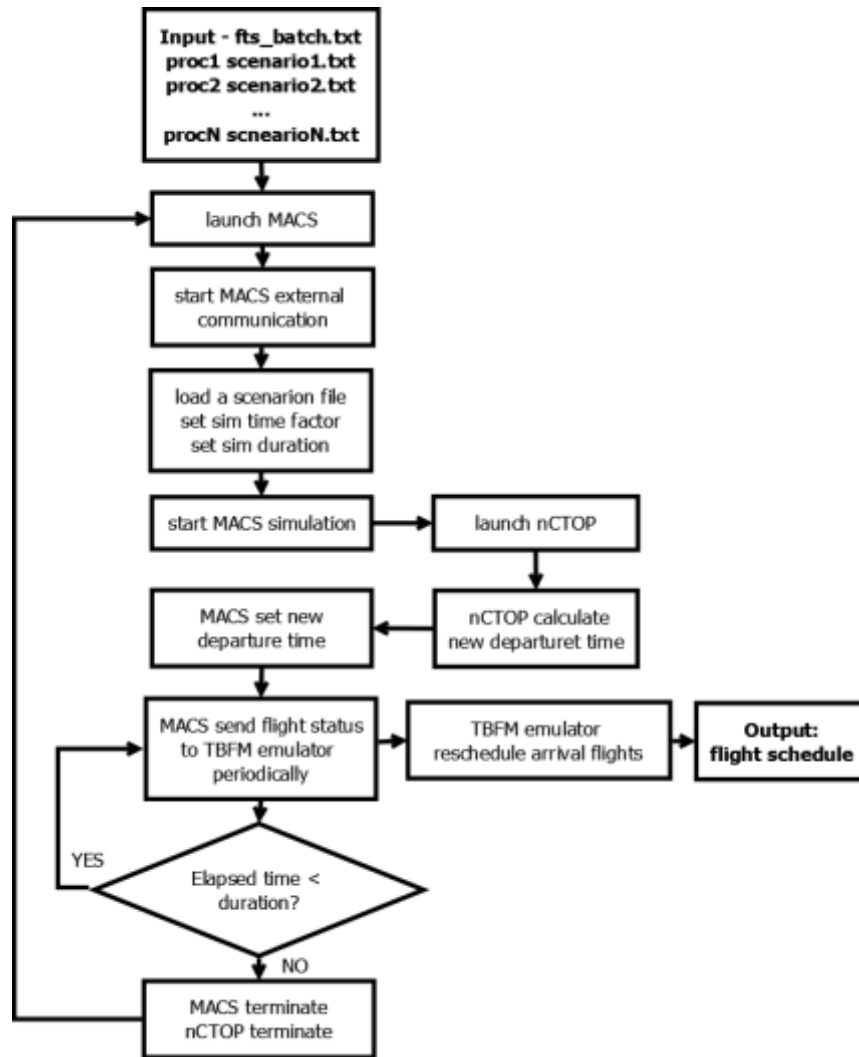Figure A.2 shows the overall flowchart of the automated simulation software.

Figure A.2 Process flow chart of the automated simulation capability.