Simplifying Semi Automated Forces Operations in the Virtual Training Environment

Brian D. Saute
Army Aviation Systems Engineer
Orlando, Florida
Brian.Saute@peostri.army.mil

Darryl L. High
Senior Software Engineer
Binghamton, New York
DLHigh@RockwellCollins.com

ABSTRACT

Imagine being in the role of an instructor who is controlling the virtual environment for a Black Hawk or Chinook flight simulator. Your responsibility is to closely observe the student crew, analyze the performance of critical tasks, and provide feedback during flight and in after-action review. Because this is a high fidelity, full motion, flight simulator, you are likely to experience G-force accelerations in the roll, pitch, and yaw axis as the student crew executes a training mission. With advances in Semi-Automated Forces (SAF), and their integration with flight simulators, complex and elaborate scenarios can now be used to further enhance the realism of a training mission. While the realism can greatly benefit the student(s), it can easily create a challenge for an instructor who needs to seamlessly coordinate and control all the set-up, virtual environment settings, instructions, observations, and feedback required for successful training. Because of this, the need for simplification has become paramount, as well as a focal point, for programs like the *Lift Simulator Modernization Program* (LSMP) and the *Additional Black Hawk Flight Simulator* (ABHFS) program.

This paper will discuss the steps taken, by LSMP and ABHFS, to simplify the integrated training environment. As the fidelity of flight simulation continues to climb, the effort to maintain simplicity needs to keep pace. Consequently, this paper will also discuss areas of enhancement that could further improve the simplicity, and ease of operation, of the virtual environment in the high-fidelity, virtual, flight simulator domain.

ABOUT THE AUTHORS

Brian D. Saute is a Senior Engineer at the US Army Program Executive Office, Simulation, Training and Instrumentation Office (PEOSTRI). Mr. Saute has worked in the Army Aviation virtual simulation domain extensively with primary emphasis on high fidelity full motion Army Aviation flight simulators. He is currently working as the Lead Systems Engineer for the *Lift Simulator Modernization Program* (LSMP) and *Additional Black Hawk Flight Simulator* (ABHFS) flight simulator programs. Mr. Saute received his BSEE from Old Dominion University in 1985. His research interests include visual systems, computer generated forces, and computer graphics.

Darryl L. High is a Senior Software Engineer with Rockwell Collins Simulation and Training Solutions, in Binghamton, NY, with over eight years of experience in Computer Generated Forces and Flight Simulation. During this time, Mr. High has been involved with various threat environments and their integration with real-time flight simulation. Currently, he is integrating the OneSAF Testbed Baseline (OTB), using Distributed Interactive Simulation (DIS) and High Level Architecture (HLA) technology, with flight simulators on the *Lift Simulator Modernization Program* (LSMP) and the *Additional Black Hawk Flight Simulator* (ABHFS) program.

Simplifying Semi Automated Forces Operations in the Virtual Training Environment

Brian D. Saute
Army Aviation Systems Engineer
Orlando, Florida
Brian.Saute@peostri.army.mil

Darryl L. High
Senior Software Engineer
Binghamton, New York
DLHigh@RockwellCollins.com

LIFT SIMULATOR MODERNIZATION PROGRAM

The Lift Simulator Modernization Program (LSMP) is an Army effort focused on the upgrade of eighteen UH-60 (Black Hawk) and six CH-47 (Chinook) flight simulators. The simulators are high fidelity, full motion, man-in-the-loop training devices that can be used to conduct exercises that range from basic flight operations to emergency procedures to full mission exercises. A number of missions involve combat support activities, and achieving success in this area of training demands an extremely realistic virtual environment. Today, new video technology affords imagery that rivals the integrity of human perception. Simulators that are concurrent with their respective aircraft lend credibility to what is conducted in battle. Up-to-date geographical data yield visual and terrain databases that allow for geo-specific mission training in any area of the world. Computer Generated Forces (CGF) allow pilots to train against a true-to-life enemy and train with supporting friendly forces. In essence, a realistic virtual environment is a fusion of many key components and technologies. On LSMP, these same components were united to achieve successful training. One component was the OneSAF Testbed Baseline.

OneSAF Testbed Baseline (OTB)

OTB provides Computer Generated Forces and can be used to generate complex and elaborate scenarios for the virtual battlefield. It can operate stand-alone or it can interact with other simulations over a Distributed Interactive Simulation (DIS) network. It is a very capable simulation package that includes friendly and enemy forces, air and ground vehicles, air defense units, artillery units, tanks, utility vehicles, helicopters, and fixed wing aircraft. It includes the capability for slingload operations, formation flight, and escort operations. Many of its entities are semi-automated. For example, when tasked to follow an assigned route, a tank will self-navigate the terrain in order to complete the assignment. Entities take weather conditions into account as they perform their various simulated functions. For example, Line Of Sight (LOS) and visibility are important factors for threat engagement and wind is an important factor for helicopter flight performance. To supply this weather, OTB provides an environment editor. When OTB is interacting with other simulations, this editor is important in mitigating fair fight issues (e.g. the weather at this OTB should be the same as the weather at another simulation). OTB even provides an interface to allow the user to spontaneously introduce and detonate various armaments in the virtual battlefield. OTB has a lot to offer. However, the purpose of the above description is not to supply an anthology of OTB capabilities. Rather, the description is meant to remind one of the many ways a user may need to get involved when using OTB in a simulation exercise. This realization led to a design effort on LSMP to make OTB easier to use in a real-time simulator's training environment.

It should be noted that there are many versions of OTB (official and un-official), and it is not the intent of this paper to delineate the multiple versions. LSMP integrated a version of OTB and, to avoid complicating discussions with version references, LSMP's version of OTB (throughout this paper unless otherwise mentioned) will be generically referred to as the SAF (Semi-Automated Forces).

Semi-Automated Forces (SAF) Simplification

It is not that the SAF is extremely difficult to use. It is a simulation package that is vast and robust and, like any other large application, proficiency requires time and practice. Simulator instructors do not always have the time to dedicate to SAF proficiency, and the SAF is not always involved in everything an instructor needs to teach. When the SAF needs to be involved in a training exercise, an instructor usually has more important things to focus on: the student(s). As a result, LSMP sought to assist the instructor by minimizing his or her involvement with the SAF (during times when software and automation could help) and by finding ways to support the instructor when SAF involvement was necessary. Overall, LSMP's SAF simplification approach can be summarized accordingly: *SAF Architecture*, *SAF Automation*, and *SAF Help*.

SAF Architecture

The first step towards making the SAF easier to use, with a real-time simulator, was to devise an architecture that would make the SAF conveniently available, when needed, and "transparent" when not needed. On LSMP each simulator is configured with two SAF Personal Computers (PCs): a real-time SAF and an off-line SAF (see Figure 1).

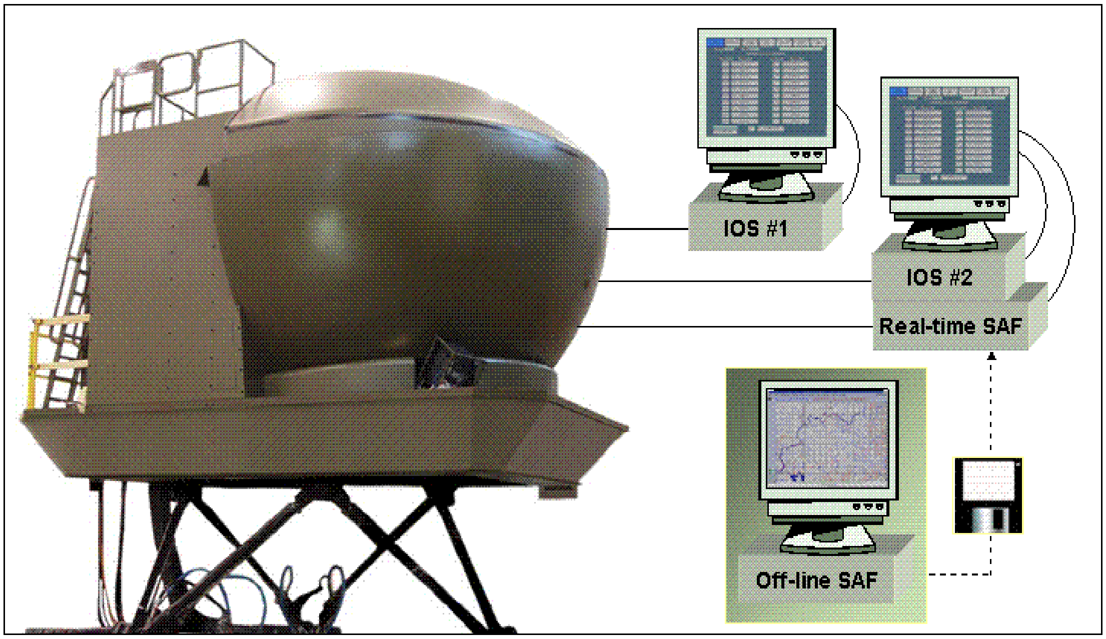

Figure 1. Overview of LSMP's SAF Architecture

The first SAF (off-line SAF) resides away from the simulator and is used to create battlefield scenarios for a simulator's training exercise. Creation of SAF scenarios can be a time-consuming task: one that does not need to interfere with the training priorities of the simulator. Its off-line residence (e.g. the instructor's office) allows an instructor to participate in the self-paced training material that is available on each LSMP SAF PC, develop a SAF proficiency at his or her pace, create SAF scenarios at a location that is more conducive for strategizing, and have a SAF readily available to facilitate scenario sharing with other training sites. Once a scenario is created, it can be brought to the real-time SAF for use with the simulator.

The second SAF (real-time SAF) is networked to the simulator, via a DIS network, and is used to manage the simulator's battlefield scenario during a training exercise. This configuration allows SAF entities to interact with the simulator. SAF entities and their radar emissions will stimulate the simulator's cockpit components (e.g. APR-39) and air defense systems will engage the simulator. This configuration also allows the simulator to interact with SAF entities. The simulator can perform evasive maneuvers, emergency procedures, and deploy its countermeasures to thwart SAF threats. For the most part, the real-time SAF is virtually "transparent" to the instructor. The capability to use the SAF and make changes to a scenario has not been taken away. The real-time SAF, with the same look and feel as the off-line SAF, is always accessible through the Instructor Operator Station (IOS). An IOS button allows the instructor to access the SAF's Plan View Display (PVD) through one of the IOS monitors. Again, "transparency" is only supposed to exist until there is a need to interact with the SAF.

Once a SAF scenario is brought to the simulator, via the real-time SAF, it becomes selectable at the IOS for simulator initialization. At this point, the instructor can now focus on training with little, if any, interaction with the SAF. This reduction in further SAF interaction (i.e. making the SAF even more "transparent") is due to the automation that was done on LSMP.

SAF Automation

The second step towards making the SAF easier to use was one that focused on automating as many of the real-time SAF interactions as possible. The SAF already comes with the innate ability to interact with other simulations over a DIS network. But, it does not come with the ability to synchronize itself with the events, functions, and modes of a real-time simulator. For example, when an instructor implements Initial Conditions (IC) at the simulator's IOS (i.e. simulator initialization for a training exercise), the SAF would need to be set up with the correct terrain database and the proper battlefield scenario. Prior to LSMP, this would require manual interaction with the SAF: exit the SAF, start the SAF with the correct terrain database, and load the appropriate scenario. Another example relates to environment conditions in a training exercise. To minimize fair fight issues, the SAF and the simulator should both operate under the same weather conditions. Prior to LSMP, the instructor would need to set the weather (for the simulator) at the IOS and then set the weather at the SAF. These are only two examples but the issue is the same: the simulator's instructor does not need to be unnecessarily burdened with SAF tasks when the focus should be on the student(s).

To alleviate this, LSMP added an interface to both the simulator and the real-time SAF. The interface takes advantage of *Simulation Management* DIS Protocol Data Units (PDUs) and performs the hand-shaking and processing required to make SAF and simulator synchronization as seamless as possible. With this interface, the instructor can continue to control the simulator's exercise through the IOS as he or she did before. Behind the scenes, the interfaces coordinate the SAF during simulator activities: ICs, Freeze, Resume, Record/Playback, Playback Fly-outs, Demonstration Record/Playback, and changes in natural environment settings. For example; if the instructor were to initiate an IC, the SAF will automatically load the correct scenario based on the instructor's selection at the IOS. If, by chance, the instructor chose a different terrain database, as part of that IC, the SAF will automatically shut itself down, restart itself (using the correct terrain database), and then load an instructor selected scenario. When the instructor places the simulator in freeze, at the IOS, the SAF will enter freeze. When the simulator is taken out of freeze, the SAF will come out of freeze. If the simulator's environment settings change (e.g. instructor makes a change at the IOS), the SAF's environment will automatically update. Even a UH-60 simulator's release of a Volcano minefield will cause an automatic creation of that minefield in the SAF, and the mines will detonate when SAF vehicles cross over the minefield.

Again, once a SAF scenario is brought to the simulator, the instructor can continue to operate the simulator in the same familiar manner: via the IOS with little, if any, need to interact with the SAF. However, there will always come a time when interaction is necessary or desired. LSMP took steps to simplify, and provide help with, this interaction too.

SAF Help

The third step towards making the SAF easier to use was one that focused on supporting the user during SAF usage. For example, the SAF typically runs on a PC with a Linux operating system. Prior to LSMP, starting the SAF required the user to open a terminal window, use Linux commands (at a command line prompt) to navigate to the SAF's executable directory, and enter the name of the SAF's executable (followed by certain command line options). LSMP made this effort easier by adding a desktop button (i.e. desktop shortcut) that launches a Graphical User Interface (GUI) for starting the SAF (see Figure 2).

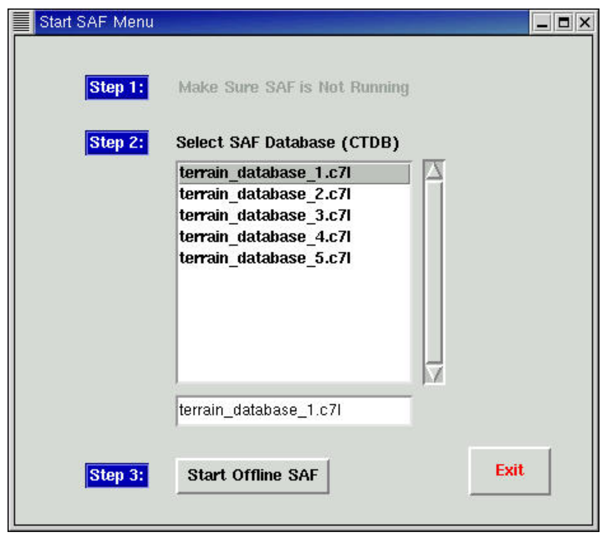

Figure 2. Start SAF GUI

In this manner, the user simply selects the desired terrain database and clicks on a button to start the SAF application for LSMP. The same desktop button and GUI exist on both the off-line SAF and the real-time SAF. This way, what is learned at the off-line SAF is always applicable at the real-time SAF.

Continuing in this fashion, another button was added to each SAF PC desktop. This button also eliminates typing at a command line prompt by launching GUIs to facilitate the transfer of scenarios (see Figure 3).

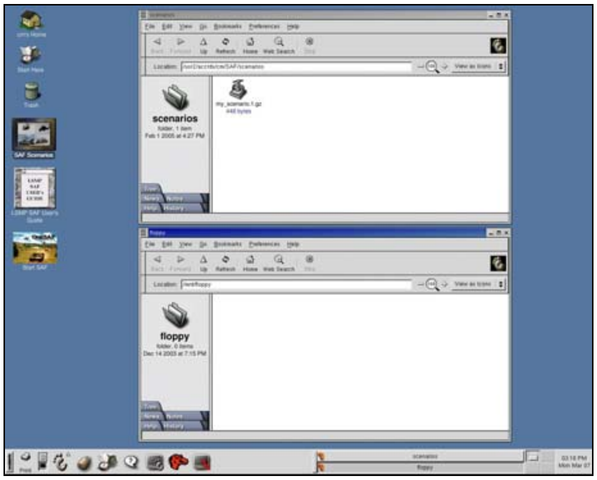

Figure 3. SAF Scenario Transfer GUIs

The GUIs are similar to Microsoft's *Windows Explorer*. One GUI places the user in the appropriate SAF scenarios directory and the other GUI places the user at the floppy disk. This allows the user to drag and drop a SAF scenario to and from disk so a scenario can be transferred from the off-line SAF to the simulator's real-time SAF. Again, the same desktop button and GUIs exist on both the off-line SAF and the real-time SAF to maintain familiarity.

A third desktop button was added to each LSMP SAF PC. This button launches a web browser that lets the user view the LSMP SAF User's Guide (see Figure 4).

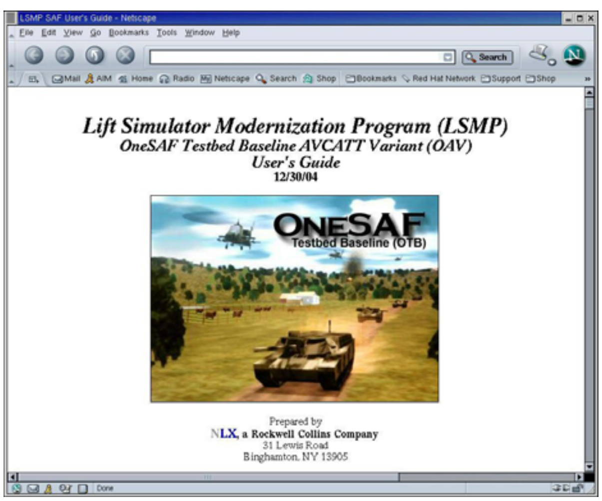

Figure 4. LSMP SAF User's Guide

The SAF takes time to master. For the casual user, there is a lot of information to absorb. The user's guide tries to help with this by making sure that SAF help is always accessible; its desktop button resides on both the off-line SAF and the real-time SAF (i.e. at the simulator). The user's guide is full of illustrations, examples, and hyperlinks (for quick reference). It contains a self-paced tutorial, with exercises, and even provides a cheat sheet with reminders about how to perform common tasks.

In an effort to further bridge the gap between unfamiliarity and SAF proficiency, LSMP offers SAF training courses at each of its simulator installation sites. With the LSMP SAF User's Guide as the foundation, the courses teach future SAF users how to use the SAF, how to create scenarios at an off-line SAF, how to transfer those scenarios from an off-line SAF to the simulator, how to use the real-time SAF with the simulator, and where to get help.

ADDITIONAL BLACK HAWK FLIGHT SIMULATORS (ABHFS) PROGRAM

In the spirit of "making the SAF easier to use," the ABHFS program (a follow-on program to LSMP) is adding the capability to use Falcon View and Portable Flight Planning Software (PFPS) to generate SAF scenarios. This is not a replacement of the Off-line SAF by any means. It does, however, allow instructors to utilize tools they are already familiar with to quickly generate simple SAF scenarios. For this to happen, ABHFS added multiple interfaces. One interface was in the form of a GUI that was added to PFPS. The GUI allows a user to select and position threats that are SAF compatible and save them as a scenario onto a memory stick. Another interface was added to the IOS so that the memory stick could then be brought to the simulator's IOS for insertion into one of the simulator's IC sets. A third interface was added to the SAF so that a subsequent IOS IC, based on the data just acquired from the memory stick, would cause the SAF to load the PFPS scenario.

SAF RELATED ENHANCEMENTS

As synthetic force applications make their advances into the realm of higher fidelity, their level of complexity is bound to increase. Equally important is the need to keep pace with the improvements and keep the SAF easy to use. In some cases, the SAF itself needs to improve in order to provide simplification and ease of use. In other cases, the way the SAF is integrated with real-time flight simulators needs to improve. The following discussions will describe a few areas for SAF related enhancement.

SAF Enhancements for Realistic Airfield Training

Current generation of realistic airfield traffic, using existing SAF behaviors, is a very time consuming task. Occasionally, it is not possible at all. There is a significant amount of loading on today's aviators to fly approaches to large populated airfields. Since much is known about the typical routes aircraft take at these airfields (see Figure 5), it does not seem impractical that this knowledge could be applied toward realistic airfield behaviors in the SAF, that is, without subscribing to an arduous task in advanced cognitive modeling.

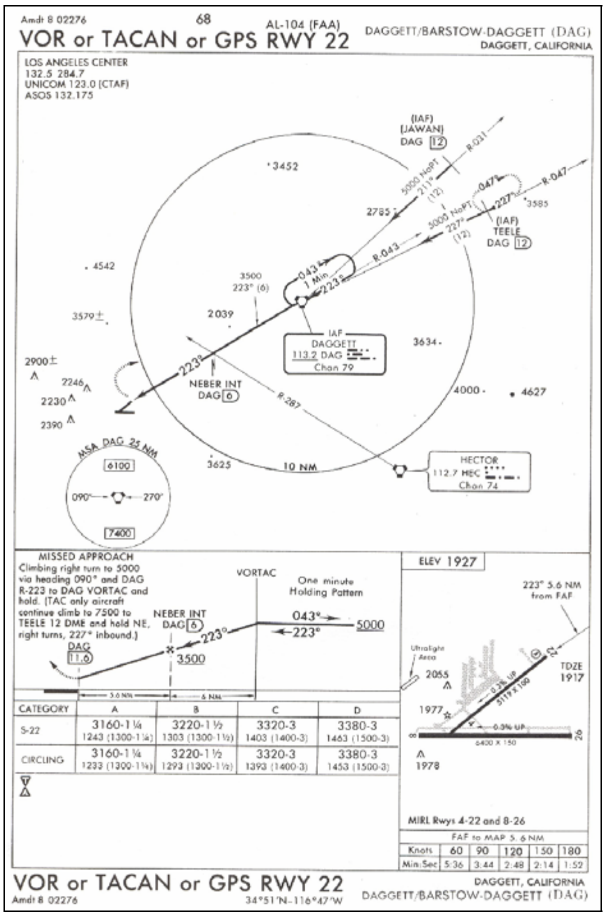

Figure 5. Airfield Approach Plate

To date, it does not appear that this area of exploitation has been thoroughly examined. Digital Aeronautical Flight Information File (DAFIF™) data are being used on LSMP and ABHFS IOS's to track and critique Army aviator approaches to many of the airfields modeled in LSMP and ABHFS databases. The following web site contains detailed information on DAFIF™ databases[1]. The LSMP and ABHFS instrument approach critique capability at the IOS has not been endorsed or otherwise approved by the NGA. The United States Army Department of Simulations (DOS) located at Fort Rucker, Alabama, has accepted this capability via UH-60 and CH-47 Subject Matter Experts (SME). Airfields modeled for the virtual environment, via LSMP and ABHFS databases, include those found in Alaska, California, Texas, Alabama, Georgia, Kentucky, Tennessee, Illinois, North Carolina, Pennsylvania, and Indiana. The eventual plan is to have many databases throughout the continental United States (CONUS) as well as Hawaii, Korea, and Germany. Use of this data on the IOS is very constructive for enhanced IFR training in the flight simulator. However, lack of easy SAF control of other aircraft near the airfields creates a surrealistic environment where the aviator is more or less free to dismiss the loading presented when in the vicinity of many other aircraft. This loading is caused by the need to monitor and respond to radio traffic and, in some cases, monitor other aircraft in the Out-The-Window (OTW) display.

The wide availability of DAFIF™ data raises several questions. Can these data be incorporated and used by the SAF to greatly enhance the realism of aircraft behavior near an airfield? Could this realism include automated messages to the instructor for use in simulating air traffic? Better yet, could the radio traffic be completely automated to include approach control as well as other aircraft contacts?

If it is possible to add realistic airfield behavior to the SAF, what user interface would be needed? Ease of use must not be forgotten. Could airfield activity be set up as a behavior and added to the SAF's execution matrix? If so, how many fields would require editing to "define" the behavior/task? Could airfields have pertinent characteristics associated with them? Would it be possible to click on an aircraft entity and select "perform airfield approach" from a task list? The user would then be required to select an airfield (see Figure 6) and an approach. This should be sufficient information for an aircraft to fly and descend on the approach in a realistic manner and keep the entire process at an "easy to control" level. The SAF already comes with an initial capability for automated messages. While they are currently oriented towards helicopter engagements, it should be relatively straightforward to tailor them for non-combat pilot tasks (e.g. conducting airfield approaches).

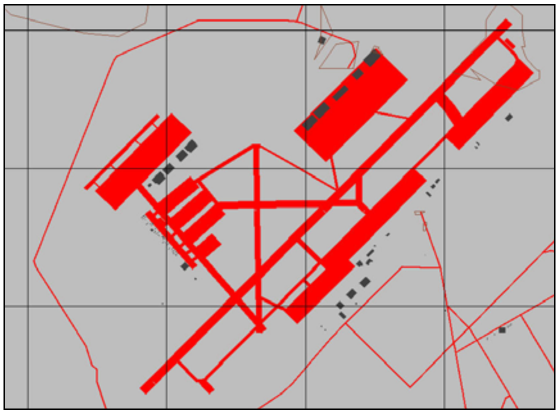

Figure 6. SAF Airfield

SAF Enhancements for Realistic Landing Helicopter Deck (LHD) Operations

The LHD-2 is the largest of all amphibious warfare ships. It resembles a small aircraft carrier; capable of supporting Vertical/Short Take Off and Landing (V/STOL), Short Take Off Vertical Landing (STOVL), Vertical Take Off and Landing (VTOL), tilt rotor and Rotary Wing (RW) aircraft operations, and use of the Landing Craft Air Cushion (LCAC) and other watercraft via a welldeck (see Figure 7).

Figure 7. LSMP and ABHFS LHD

Currently there has been a lot of interest in modeling shipboard operations for helicopters in a more realistic manner. The emphasis to date has been on 3-D sea states, modeling of the landing signal officer, and accurate wind modeling. A prototype was developed to improve this modeling for Black Hawk simulators and was called the J-Ship program. Currently, additional effort is on-going to incorporate a blade element model into the ABHFS program and enhance the wind modeling on approaches to an LHD and Frigate. Cooperation with the Navy has been critical and the program has received detailed information on both kinds of ships, dimensions, lighting, example wind corridors (measured by a fully instrumented test aircraft), etc. However, little has been done to simulate other Rotary Wing Aircraft (RWA) in the vicinity of an LHD and it is currently impossible to precisely specify a realistic SAF RWA landing on an LHD. As is the case with airfields, much is known about correct maneuvers in the vicinity of an LHD. This same information could be used to create new SAF RWA behaviors that approximate other aircraft landings and takeoffs from an LHD.

To simplify the discussion and for illustration purposes, the explanation of approach patterns, for helicopters performing shipboard operations, will be limited to the delta pattern. The delta approach pattern is performed in the overhead, port, or starboard manner as depicted in Figure 8.

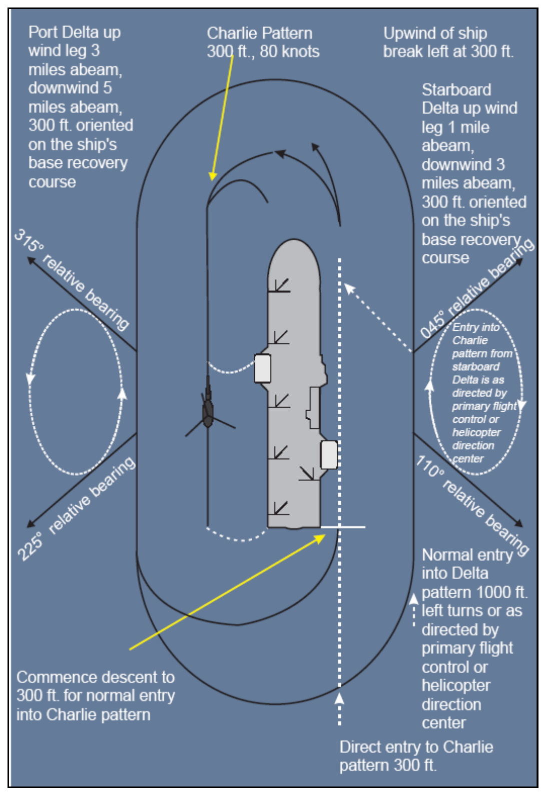

Figure 8. Delta and Charlie Patterns for Helicopters[2]

The overhead delta pattern is a Visual Flight Rule (VFR) left-hand racetrack pattern established in the vicinity of the ship. It is oriented on the Base Recovery Course (BRC) and close aboard the starboard side at an optimum airspeed. During heavy traffic periods, additional delta patterns may be utilized as assigned by Primary Flight Control (PRIFLY). The starboard delta pattern is a holding pattern and is established on the starboard side of the ship's 045° to 110° relative bearing between one and three miles. It is a right-hand racetrack flown at 300 feet and 80 knots. The port delta pattern is a holding pattern established between the ship's 225° and 315° relative bearing between three and five miles. It is a left-handed track flown at 300 feet and 80 knots.

As can be determined from Figure 8, much of the information necessary to realistically model helicopter traffic in the vicinity of an LHD is known and could be applied to a SAF helicopter behavior. When tasking a SAF helicopter to execute an LHD approach, the user should only have to select an LHD and an approach pattern. With the additional ability to precisely land SAF helicopters, this would go a long way towards creating realistic loading on an aviator training for shipboard operations.

SAF Enhancements in General Aviation Behaviors

Army helicopters have different landing techniques. Landings can be roll-on landings to a runway or a point landing such as one that occurs on a helipad. There are also differences in how takeoffs are performed. One is similar to the way an airplane takes off in that a helicopter accelerates down the runway until sufficient power and speed are obtained to cause the aircraft to become airborne. The other type of takeoff is the one that occurs at a location that does not have a runway (e.g. at a helipad). In this case, a pilot uses enough collective to lift off the ground before moving in a forward direction.

Currently, there is very little capability, within the SAF, to effectively simulate these types of landings and takeoffs. Helicopters will take off but they will immediately begin flying their assigned route without any lift-off protocol. When they reach their assigned destination, they land (i.e. they do not understand how to taxi down a runway). Trying to get SAF helicopters to perform these protocols, through various tricks and workarounds, only results in very unrealistic flight characteristics. This is a case where "making the SAF easier to use" requires the modeling of new behaviors. To support these protocols of flight, one behavior would model roll-on landings. The behavior would at least require selection of a runway, a touch down point, and possibly a stopping distance or a deceleration rate. Another behavior would accurately represent takeoffs down runways. It would at least require selection of a runway, an initial acceleration point, possibly a default pre-taxi speed, and the point where the helicopter is to transition from taxiing down a runway to flying its assigned route. Since the SAF supplies a line editor (i.e. an ability to create lines), the editor could be used to create lines to better define the paths involved in a takeoff or landing.

A couple of other basic capabilities that should be added to the SAF, for realistic aircraft operations, are settings and behaviors for aircraft lights and aircraft engines. Engine control is important in that aircraft should not continue to appear ready to fly (e.g. rotors turning) once they have been parked at an airfield. The need for accurate light controls is extremely important for plausible night scenes and Night Vision Goggles (NVG) operations.

SAF Related Enhancements for Better Formation Flight Training

One capability with most real-time flight simulators is the ability of an instructor to fly a mission in the simulator, record the flight, and play the flight back as a lead-ship for a formation flight training exercise. While a legacy feature, it remains a critical tool for many training instructors. A way to embellish this training feature would be to add SAF entities and their ability to fly in formation. By modifying a simulator's recording functionality, external entities (i.e. entities external to the simulator such as SAF entities flying in formation) could be recorded and played back as part of a formation flight training exercise. In a reverse role, the SAF's tether feature could be used to cause SAF helicopters to fly in formation with the simulator. SAF entities would then fly in formation with a real pilot and allow the simulator pilot to gain experience flying a chalk one mission.

SAF helicopters already come with the ability to fly in formation and tether in formation to another helicopter (e.g. a real-time flight simulator). The area where enhancements are needed resides in the fidelity of helicopter flight models and flight behaviors. When the simulator's legacy feature is used to record and playback a lead-ship, the fidelity (flight and behavior) emulates that of a real pilot since it was a real pilot that flew the simulator and made the recording. It is this fidelity that has made it such a valuable training tool. SAF helicopters do not have this fidelity. SAF helicopters do not have real pilots; they are semi-automated. In flight, they tend to over react to the terrain as they self navigate an assigned route. SAF entities do not fully understand protocol. They do not taxi down runways and they do not always correctly space themselves when landing as part of a formation.

Once the fidelity of SAF helicopter flight is improved, training exercises could be constructed to help prevent a recurrence of the deadly scenario in Hawaii (see Figure 9). In this accident, multiple Black Hawks were carrying sling-loads into a Forward Area Refueling Point (FARP). Due to poor environment conditions, use of Night Vision Goggles, and reduced aircraft aerodynamic performance (from carrying sling-loads), a trailing aircraft acquired too much speed and crashed into the aircraft in front of it.

Figure 9. Formation Flight Accident

SAF User Interface Enhancements to Support Full Motion Simulators

Controlling the SAF with a mouse in a six Degree-Of-Freedom (DOF) motion system is less than ideal. If an Army aviation full motion flight simulator (particularly a Black Hawk simulator) were observed throughout a pilot's training exercise, one could easily witness significant motion extrusions on the motion system. One can only speculate about the possible training objectives that result in such violent movement of the motion system. Perhaps the instructor forced autorotation maneuvers, perhaps brownout conditions, or perhaps training with threats triggered the APR-39 Radar Warning Receiver which then resulted in extreme evasive maneuvers. Whatever the cause, there is an obvious physical strain on all who are present in the simulator. Fine and meticulous mouse manipulation under these circumstances is out of the question.

On LSMP and ABHFS, IOS control is based on touch screen technology. Would not a touch screen interface, for the SAF, prove to be a more effective interface for pilot instructors in this environment? Would it be a big impact to implement such an interface throughout the many SAF libraries? Will the OneSAF Objective System (OOS) be more extensible and allow for the provision of this interface without wide spread impacts throughout the entire software code base? These are questions worth examining in order to better customize the Army SAF for simulator instructors who are preparing helicopter pilots for combat.

CONCLUSION

The SAF is a very capable tool that can greatly enhance the reality of flight simulation. As with any "wiz-bang" gadget, a lot of features can make the gadget difficult and complicated to use. On LSMP and ABHFS, a lot of the complications were removed by focusing on ways to minimize unnecessary work: add GUIs to simplify using the SAF, provide help via training courses and accessible on-line documentation, automate SAF interaction with the simulator, and integrate the SAF so that instructors can continue to use tools they are already familiar with (e.g. PFPS, the simulator's IOS, etc).

In an effort to continue making the SAF easier to use, other enhancements should be considered. Sometimes, the enhancements involve the SAF itself. Other times, the SAF can make life easier via the manner it is integrated with a simulator. This paper explored a few SAF related enhancements in the following areas: airfield behaviors, shipboard landing behaviors, landing and takeoff protocols, formation flight, and user interface modifications. The effort should not stop here. CFG applications are being used with real-time flight simulators more and more. As CGF applications mature, developers should keep this in mind as interfaces (user interfaces and simulation interfaces) are designed.

NOTES

[1]DAFIF™ is a product of the National Geospatial-Intelligence Agency (NGA) of the United States Department of Defense (DOD). Detailed information on DAFIF™ databases can be found at https://164.214.2.62/products/digitalaero/index.cfm

[2]Joint Pub 3-04.1, *Joint Tactics, Techniques, and Procedures for Shipboard Helicopter Operations* (10 December 1997). Retrieved June 15, 2005 from http://www.dtic.mil/doctrine/jel/new_pubs/jp3_04_1.pdf