

DARPA STOW SYNTHETIC FORCES*

CDR Peggy Feldmann¹
Defense Advanced Research Projects Agency
Arlington, Virginia 22203

Peter Muller
Advanced Telecommunications, Inc
Alexandria, Virginia

ABSTRACT

The Synthetic Forces Program is an integral part of the the Synthetic Theater of War 97 Advanced Concept Technology Demonstration (STOW-97 ACTD). It has produced, over the last three years, the forces needed to train in a platform-based, seamless, joint synthetic battlespace. Building off Modular Semi-Automated Forces (ModSAF), Synthetic Forces now exists for Army, Navy, Marine Corps, and Air Force platforms, as well as, Opposing Forces and United Kingdom Forces. The construct of these forces include finite state semi-automated forces, rule based multi-echelon command forces, and artificial intelligence pilots. Building decision-makers who plan, replan, and execute higher orders has been a primary goal of the program. Providing tools to efficiently laydown, control, and communicate with the Computer Generated Forces (CGFs) has also been a priority.

This paper provides a description of the Synthetic Forces (SF) used for STOW-97 ACTD. The STOW-97 ACTD will be an integral part of United Endeavor 98-1 (UE 98-1), a United States Atlantic Command (USACOM) sponsored exercise held in late October and early November 1997. Included is a broad overview of how forces are simulated and describes how they move, shoot, communicate, think, and interact. Since this paper is being submitted prior to the UE 98-1 exercise, we will report exercise results in the presentation.

ABOUT THE AUTHORS

CDR Peggy Feldmann, USN, is the program manager of the Synthetic Forces program of the Defense Advanced Research Projects Agency. She has a BS in Oceanography from the US Naval Academy and a MS in Acoustical Engineering from the Naval Postgraduate School, Monterey, California.

Peter Muller is a Senior Systems Engineer at ATI and is a Systems Engineering and Technical Assistance contractor on the Synthetic Forces program. He has a BS from Syracuse University and an MS in Systems Management from USC.

* OSD approved for public release

† Significant contributions to this paper were made by members of the Synthetic Forces team: Dr. Andrew Ceranowicz (Advanced Technologies Inc.), Ms. Susie Hartzog (NRaD), Dr. David Tseng (Hughes Research Labs), Dr. Paul Rosenbloom (USC), Dr. John Laird (U of M), Mr. James Calpin(MITRE), and Mr. Steve Kasputis (Techmatics, Inc.)

DARPA STOW SYNTHETIC FORCES*

CDR Peggy Feldmann[†]
Defense Advanced Research Projects Agency
Arlington, Virginia 22203

Peter Muller
Advanced Telecommunications, Inc
Alexandria, Virginia

OVERVIEW

Background

STOW technologies have been evolving with DARPA sponsorship for a number of years. In fiscal year 1995, STOW-97 was one of the ten Advanced Concept Technology Demonstrations (ACTD's) initiated by the Office of the Secretary of Defense. STOW technologies will play an active part in United Endeavor 98-1, a United States Atlantic Command (USACOM) sponsored exercise held in late October and early November 1997. STOW technologies will continue to evolve and be supported by DARPA at USACOM for an additional two years. The role of the ACTD is often misunderstood both in industry and within the government.

...The ACTD process was initiated in 1994 to permit the early and inexpensive evaluation of mature advanced technologies. The evaluation is accomplished by the warfighter to determine military utility and to develop the concept of operations that will optimize effectiveness. ACTDs are structured and executed so that, when successful, we are able to proceed rapidly into formal acquisition.

By introducing new technologies in the field prior to the initiation of formal acquisition, we allow our operators, who

have experience in combat, to evaluate and to assess the military utility and develop the tactics to ensure that we can realize the full potential of our substantial technology base that's available to us - both Defense and commercial. ACTDs are not a means by which to circumvent the formal acquisition process, but rather a means to enter that process based on a user assessment of the value of the new capability which reduces the user acceptance risk. This process will help us make more informed acquisition decisions and improve our acquisition cycle time.

- Jack Bachkosky, Deputy Under Secretary of Defense (Advanced Technology) in ACTD Times (October 1996)

The Advanced Distributed Simulation (ADS) Program within DARPA consists of the STOW-97 ACTD, the Advanced Simulation Technology Thrust (ASTT), and Simulation in Training for Advanced Readiness (SIMITAR). The STOW-97 ACTD compliments the Joint Simulation System (JSIMS) by bringing advanced distributed simulation technology to USACOM for evaluation and feedback. Originally part of the STOW-97 program, ASTT was split off from the ACTD last year to directly feed advanced simulation technologies to JSIMS.

For the Synthetic Forces development, the initial requirements for the STOW-97 ACTD were

* OSD approved for public release

[†] Significant contributions to this paper were made by members of the Synthetic Forces team: Dr. Andrew Ceranowicz (Advanced Technologies Inc.), Ms. Susie Hartzog (NRaD), Dr. David Tseng (Hughes Research Labs), Dr. Paul Rosenbloom (USC), Dr. John Laird (U of M), Mr. James Calpin (MITRE), and Mr. Steve Kasputis (Techmatics, Inc.)

extremely ambitious. DARPA had to develop entity level Synthetic Forces for all of the Services suitable for Joint Task Force (JTF) Tier III (JTF Staff) training. To accomplish this within the time and funding available, DARPA chose to build upon the basic architecture of Modular Semi-Automated Forces (ModSAF) to develop separate Synthetic Forces for the Army, Navy, Marine Corps, Opposing Force (OPFOR), and Air Forces. This proved extremely challenging since we had to modify the underlying architecture while simultaneously developing new forces and capabilities.

During development, three additional architectural requirements were added which greatly increased the challenge over the last year. First, the underlying description of space was changed from a projection onto a single three dimensional Cartesian space to the Global Coordinate System (GCS) which can represent arbitrarily large regions with a true curved earth representation. Second, we switched from Distributed Interactive Simulation (DIS) to the HLA (High Level Architecture) using DDM (Data Distribution Management) to limit the network load. Third, we chose to make JointSAF independent of processor architecture, allowing us to use "big endian" workstations interchangeably with "little endian" PC's. This has allowed us to switch the majority of our simulation to lower cost Intel PentiumPro computers and increase the number of entities that we can simulate. Additionally, we added the United Kingdom as a partner to the ACTD further expanding the range of entities and systems that needed to be supported by the basic architecture.

Synthetic Forces

JointSAF is the term used to describe all of the Computer Generated Forces (CGF) or Semi-Automated Forces (SAF) for STOW-97. JointSAF consists of a common "core" to which the Services have added extensions to represent their unique physical and behavioral models and control and reporting interfaces. JointSAF includes ArmySAF, MCSAF, NavySAF, AirSAF, OPFORSAF, UKLandSAF, UKSeaSAF, and UKAirSAF. Synthetic Forces were developed by the combined effort of a large number of people in government, industry, and academia in a distributed environment. While this proved to be a management challenge, DARPA felt that this was the best way to push the technology while insuring

that the Services were stakeholders in the program.

In general, the behavior of an individual SAF entity is produced by an integration of individual behaviors, which react to the entity's immediate environment, and unit level behaviors, which coordinate the activities of multiple entities, such as platoons or companies. To increase the fidelity of the representation of the command decision and communication process, STOW developed Command Forces (CFOR). Rather than imposing a specific methodology for representing behaviors, CFOR uses a variety of sophisticated Artificial Intelligence techniques. Underlying these techniques is a common infrastructure which includes utilities for accessing complex terrain information, as well as a common language, CCSIL (Command and Control Simulation Interface Language), to support communication between SAF entities, Command Entities, and C4I systems.

Simulation World and Synthetic Environment

One of the first things that needs to be done in any simulation is to agree to the rules and characteristics of the "simulated world". The STOW Synthetic Environment consists of a realistic digital "world" whose features realistically affect the operations of Synthetic Forces. These features include roadways, buildings, bridges, rivers, ocean, littoral zones (tides), and diurnal effects. It also supports weather effects including rain, snow, dust, fog, and the use of obscurants. A Southwest USA database was built to support initial technology development. It includes high resolution areas around Camp Pendleton, CA and Twenty-Nine Palms, CA. For the ACTD, we have a new database consisting of a 500 x 700 km area in South West Asia. (Lukes, 1997).

In addition to the "dirt" or "water" that the Synthetic Forces move upon, the Synthetic Environment also provides a consistent framework for atmospheric effects that affect visibility. Based on time and location in the world, the simulation automatically calculates ambient light based on both the sun and the moon. There is the ability to fire illumination flares. Wind is used by the simulation primarily for the propagation of obscurants such as dust and smoke. For example, if the soil characteristics warrant it, a vehicle may produce a dust cloud. This cloud would be subject

to wind direction and speed. If the vehicle deploys a smoke grenade, it will also be effected by the wind. Precipitation, visibility and cloud cover are also provided and are used by the simulation primarily to reduce the effective range of optical sensors within the Synthetic Forces.

SYNTHETIC FORCES

How Objects Sense

Since it is important that all forces interpret the synthetic battle space in a consistent manner, we adopted a "Common Model" approach for sensor representation. Each class of sensors such as radar, visual, and ESM have a common model that can be used by any member of the class with appropriate customization being done by data files.

Visual and Infrared Sensors

The visual sensor model uses a standard US Army Night Vision Electronic Sensors Directorate, NVESD, detection model known as ACQUIRE which is described in the AMSAA CCTT compendium. (Topper et. al., 1996) Although classified data can be supported, we are only using unclassified visual sensor data for UE 98-1. The modeling of Infrared sensors in JointSAF is limited to using a constant IR signature. This signature is not affected by how long an engine has been running or whether a vehicle has been in the sun. Improving the IR environment was considered for the ACTD, but will not be done in FY 97.

Line of Sight (LOS) is critical in land combat and a significant computational load is required to constantly perform LOS calculations between objects in the simulation. One way to reduce this computational load is to "ignore" all friendly objects in LOS calculations and in target acquisition. This compromise means that realistic vehicle on vehicle fratricide operations will not be played for ground combat in UE 98-1. This could easily be changed, but significantly reduces the size of the engagement that can be simulated with a given set of hardware. Fratricide is still possible on ground forces using friendly artillery or by aircraft. Since there are far fewer aircraft than ground vehicles, this computational compromise is not required in AirSAF and fratricide is possible.

Radar

There are a number of radar models within

JointSAF. Several different modeling techniques will be demonstrated in UE 98-1.

First, a generic radar model is available which uses range and field of view to determine acquisition of a target. For systems using this model, if the radar is turned on and looking in the right direction, the target will be detected when it reaches a threshold range.

Second, a common radar model is available that characterizes target detection capabilities for, and environmental and interference effects on, a diverse set of complex multi-mode radars. All airborne radars, except for JSTARS, will use this model. DoD-approved and validated intelligence databases were used to derive the radar parameters of the specified systems. The model accounts for environmental effects on radar propagation such as multipath and ducting. Interference effects due to noise jamming can also be included. Sophisticated sensor control features such as waveform or processing mode and Field of View (FOV) selection, sensor look time, dwell time on target, permissible emission duty cycle and report filters based on target attributes are available with this data-driven model.

Third, due to the unique environment (i.e., ground clutter) that the E-8 JSTARS must operate in, a high-fidelity, physics-based model, was used to realistically model the Wide-Area Surveillance (WAS) function of the Moving Target Indicator (MTI) mode of the JSTARS. The model directly interfaces with U.S. Army real-world ground stations, namely the newly developed Common Ground Station (CGS). This radar will be simulated as a separate federate to the simulation.

There is also a simple Radar Warning Receiver model which is used on RWA and provides perfect detection that the RWA is being illuminated by a radar.

Electronic Surveillance Measures (ESM)

A newly developed ESM model will provide detection and classification information for a number of ESM devices. The ESM sensor is closely related to the common radar model described above. It uses sensor parameters of antenna gain, receiver sensitivity, and frequency response and signal and environment factors of source level, pulse repetition rate and interval, propagation loss, and in band noise levels.

How Objects Move

There are basically seven dynamic models within JointSAF which are used to calculate how an object moves: Tracked Vehicle, Wheeled Vehicle, Dismounted Infantry, Rotary Wing Aircraft, Fixed Wing Aircraft, Missile, and Ship.

Tracked and Wheeled Vehicles

Tracked and wheeled vehicle dynamics are modeled with relatively low fidelity three Degrees of Freedom (DOF) models which clamp the vehicle onto the terrain surface. They take into consideration maximum velocity (forward and reverse), maximum acceleration/deceleration, maximum turn rate, soil type, and maximum incline. Sixteen different soil types can be recognized by the models and each of these can have individual performance associated with it. In addition, three levels of dust (small, medium, and large) can be set depending on vehicle speed and soil type. These models also keep track of fuel usage based on vehicle speed in kilometers per hour and consumption rate in liters per hour. Additional limits on speed for Engineer attachments to tracked vehicles such as mine plows are taken into account.

People (Dismounted Infantry, Individual Combatants, Non-Combatants)

People are modeled by having a maximum forward, backward, and energy saving speeds. They can assume standing, kneeling, and prone positions. As with ground vehicles, there are up to sixteen different soil types that can be recognized.

Fixed Wing Aircraft (FWA)

A new FWA dynamics model was developed specifically for the ACTD which is far more accurate in modeling FWA dynamics. The following characteristics are used: coefficient of drag (sub and supersonic), aircraft mass, thrust, lift, angle of attack, air density (as a function of altitude), and fuel usage.

Rotary Wing Aircraft (RWA)

Depending on speed, the helicopter has two basic modes of flight. At low speed, it can face in any direction and maneuvers by tilting the rotor-vector. At high speed, it behaves more like a FWA. The following parameters are considered: Minimum Dive Angle, Rate of Response when not rate limited, maximum G's, cruise velocity, maximum

velocity (up, backwards, and forwards), and fuel consumption at 100% power.

Missiles

JointSAF has three fundamentally different ways to model missiles and these will be discussed in the next section.

Ships

Ship dynamics are modeled taking into consideration maximum velocity (forward and reverse), maximum acceleration/deceleration, maximum turn rate, and sea state.

How Objects Shoot

There are basically three methods used for simulating weapons: Direct Fire, Indirect Fire, and Missiles. Direct Fire weapons include most unguided weapons that fire directly at a target including everything from pistols to tank main guns. Indirect Fire weapons fire at a location, such as howitzers and mortars. Missiles have guidance systems and, unlike direct and indirect fire, can change their flight path.

Direct Fire Weapons

There are a large number of weapons that are modeled as direct fire. All individual rifles, machine guns, and tank main rounds are considered direct fire. Since direct fire engagements are very numerous in land combat, they are modeled differently than missiles. The trajectories of the bullets and rounds are not computed in the simulation and they are not represented as simulation entities. The round is launched, and after an appropriate delay, impacts the target. The effect on the target is determined at the moment the round is fired. For example, if a round is fired and before it reaches the target, the target breaks line of sight by moving behind a building, the outcome will be unaffected. This is generally not an operational concern since the speed of the round is high and the engagement distance is small. In the 3D visualization, you see a muzzle blast indicating fire and a detonation blast indicating detonation, but the rounds themselves are not visualized. Accuracy of direct fire weapons is modeled using the Delivery Accuracy Model described in the Army Materiel Systems Analysis Activity (AMSAA) Close Combat Tactical Trainer (CCTT) Compendium. Delivery Accuracy Tables are provided by AMSAA which are used to calculate biases and dispersions. (Topper et al,

1996)

Missiles

Missiles are simulated in three fundamentally different ways in JointSAF.

First, missile engagements can be simulated using the Direct Fire methodology. This is appropriate for missiles with a short time of flight. All Army anti-tank missiles will use this methodology with the actual position of the missile along its flightpath being modeled by simple flightpath equations for decorative purposes only. Within the simulation, the engagement result is determined at trigger squeeze, but observers see the missile "flying" an approximate trajectory.

Second, specific missiles have their own unique missile models incorporated within JointSAF. Included in this group are Patriot, THAAD, MLRS, and ATACMS.

Third, by a separate simulation federate known as the Ordnance Server (OS). All FWA launched missiles and most OPFOR missiles will be modeled using the Ordnance Server. The Ordnance Server uses the same weapon simulations that are used by the Navy and Air Force Combat Training Simulations (CTS), formally known as Tactical Aircrew Combat Training System (TACTS)/Aircrew Combat Training System (ACTS). The launch command is given from JointSAF for specific missiles and the Ordnance Server provides periodic updates to JointSAF based on its high fidelity calculations. The missile models use a 5-DOF real-time trim aerodynamics model and provides trajectory from launch to intercept/failure. It includes mass and thrust, seeker, guidance, aerodynamics, and autopilot component models. The effects of Chaff and Flares are taken into account by a subset of the missiles simulated. (Ullom, 1996)

Although other services could technically use the Ordnance Server, for program reasons, no Army or Marine Corps missiles will use the OS. Similarly, the surface Navy will only use the OS for simulating the Tomahawk. The Tomahawk simulation is not a TACTS model and was provided by Naval Surface Warfare Center - Dahlgren.

Indirect Fire Weapons

Indirect Fire Weapons, or gun artillery objects, like howitzers and mortars are modeled similarly to

direct fire weapons with a few significant differences. Once a call for fire is given, and after an appropriate delay, the rounds impact an area which is modeled using a bivariate normal distribution with both range and deflection using different sigmas. The artillery and mortar rounds don't actually fly out in the simulation and are not visualized. Since artillery rounds are very useful in disrupting ground maneuver, the JointSAF operator can use the "Bomb Button" to instantly deliver artillery or a 500 pound bomb to any location.

How Objects Are Damaged

JointSAF uses the DIS paradigm that the object that is fired upon is responsible to determine if it was damaged in an encounter. Each object has a look-up table for specific weapons and miss distances. Direct Fire weapons use the Direct Fire Vulnerability model to calculate the damage on the entity. This model uses the munition type, fire-to-detonation range, round dispersion, target orientation, and target exposure to index into a Vulnerability Table containing probability of kills (Pk's) for various kill types.

Indirect Fire weapons use a Carleton Damage function for High Explosive (HE) rounds and a "cookie cutter" equation for Improved Conventional Munitions (ICM). For Indirect Fire, a Vulnerability Table for a given vehicle class being struck with a particular munition class contains Pk's for the various combinations of firer to detonation range, orientation, round dispersions, and target exposures. Pk's are provided for each kill type. The input to the vulnerability model consists of a detonation interaction with data for munition, fuze, location, and trajectory. An internal database in JointSAF is used to classify the munition into one of a number of generic munition classes, such as SMALL_HE, MEDIUM_HE, or LARGE_HE. Each JointSAF land vehicle is classified into a generic target class. Currently JointSAF supports three types of Armored Personnel Carriers (APC 1-3) and three types of tank (tank 1-3). An internal database in JointSAF contains lethality information for each of the generic target classes. A major shortcoming of these models is that they do not take into account differences in cover from intervening terrain or structures when computing damage, rather they assume an equal level of cover for all entities.

How Objects Communicate

Communications within JointSAF occurs either through a radio model or implicitly. Radio communications are supported by a simple radio model which checks frequency, hopset, and encryption to determine if a message was received. It is up to the operator to tune the entity's receivers to the same settings in order for communications to occur. For the ACTD, a data file is used to setup the command nets. Environmental effects are not considered in the communications model. Radio messages are used for communications by the CFOR command entities and for a variety of reports and coordination messages. All CFOR messages are formatted using the Command and Control Simulation Interface Language (CCSIL). Because there are no models of verbal, written, or signal communications most of the lower level control and coordination, including operator control, is accomplished through implicit communications.

SAF Behaviors

How objects "think" within JointSAF is complex and diverse. A human body provides some useful analogies. Just as many bodily functions, such as breathing, are done without conscious thought, many functions in JointSAF are done without operator intervention. Like the body, JointSAF is a complicated system. Just as we can consciously control our breathing, the operator can override operations within the simulation.

The fundamental thinking of an object is embedded in its "behaviors". Most entities in JointSAF have behaviors coded as ModSAF tasks. These tasks are implemented with a software technique known as augmented asynchronous finite state machine (AAFSM). A unit is controlled by a hierarchy of tasks. Higher level unit tasks spawn subtasks that control subordinates. Reactive tasks monitor the situation and override the commanded tasks when necessary.

Most vehicles and weapon systems do not explicitly model crews. A tank in JointSAF for example, doesn't individually model the four man crew, but to an observer it would appear to function as if it did. An operator, using the Graphical User Interface (GUI), can create a tank and order it to move to a new location using simple

point and click motions. The tank does not move in a straight line from Point A to Point B. The vehicles use terrain reasoning to pick a suitable path, just as an actual tank commander would. The tank slows and speeds up based on terrain and avoids obstacles and other vehicles as if it was driven by a competent driver. If the vehicle is attacked enroute, a new task frame overrides the "Move to Point B" it is executing. Depending on the tactical situation, a number of new tasks can be automatically invoked such as deploying smoke, attacking, moving to cover, etc. When the engagement is over, it returns to its original mission.

A vehicle acting by itself may act differently if it's a part of a larger unit. Thus, in addition to individual vehicle behaviors, there are various unit level behaviors. Units keep appropriate tactical formations and react to situations differently than individual vehicles.

SAF Advanced Cognitive Behaviors

While a large number of very sophisticated behaviors can be accomplished using the Task Frame approach, DARPA has invested considerable effort into exploring other ways of representing behaviors. Although sophisticated AI techniques demonstrate the potential to improve behaviors and reduce manpower, they presently require significantly more hardware. For UE 98-1 we will be demonstrating several different software technologies to accomplish some specific missions. These techniques could be expanded in the future and will become much more affordable for large scale exercises as hardware prices continue to fall.

Two different types of Command Entities will use different AI techniques to demonstrate sophisticated Land commander's thought processes. Command Entities (CE's) are complex cognitive models of critical command and control (C²) elements. The CEs process Operational Orders (OPORDs) and plan and execute maneuvers/attacks with appropriate assets according to doctrine and the tactical situation.

The Army Mechanized Company Team Commander uses Constraint Sets which specify how to make a coherent set of decisions based on doctrine traceable to the Army Training and Evaluation Program (ARTEP) for a given unit and

mission. It will be able to accomplish several missions autonomously including Attack, Defend, and Movement to Contact. It plans the mission and chooses the first feasible plan to accomplish the mission.

The Army Battalion Command Entity (Bn CE) in STOW-97 will be able to demonstrate a second level of C2 decision making in software. The Bn CE will be capable of planning and executing an attack mission which will include a Bn Roadmarch, a Bn Occupy Assembly Area, a Bn Tactical Movement, a Bn Breach Obstacle, and a Bn Attack. Also demonstrated will be the Bn CE directing the movement of a Battalion Command Post. In all cases, the Bn CE does its planning based on a Brigade Order generated through a C2 workstation. It then passes its own (Bn) order down to the CFOR Companies below, which perform the detailed maneuver plans for the platoons, as well as the fire support plans for the Company.

The Marine Corps Platoon Leader CE utilizes an expert system for its cognitive decision making. Expert knowledge, captured through an efficient knowledge acquisition process with military subject matter experts (SMEs), is represented as knowledge modules in the derived knowledge base. The knowledge modules are in the form of "fuzzy tables", or sets of rules that cover a specific decision point. The tables are called "fuzzy" because answers to questions that do not precisely match can be interpolated from existing rules. The expert system is queried by the CE in planning a course-of-action and in the execution of the mission. The Marine Corps Platoon Leader will provide the command and control function for infantry units of the notional Marine Expeditionary Force (MEF) (Forward). For the UE 98-1 exercise, a human company commander will task CE-controlled Rifle Platoons via OPORDs input at a C2 Workstation. The CE model for the Rifle Platoon Leader has been developed to be expandable to the Rifle Company Commander level, but not for UE 98-1.

The Rifle Platoon Command Entity (RPCE) can plan and execute a USMC attack (single envelopment, frontal assault, etc.), for a Rifle Platoon including attachments from a Weapons Platoon (M240 MG Squad, 60mm Mortar Section, and Assault Teams) and can call for fire from Weapons Company (HMMWV/HMG and 81mm a

Mortar Section) and Battalion (Artillery Battery) assets. Calls for Naval Gunfire and Close Air Support are also implemented. The RPCE will process OPORDs and will plan and execute a mission which includes Platoon Occupy Attack Position, Platoon Tactical Movements, Platoon Breach Obstacle, Platoon Action on Contact, and Single Envelopment Attack on the designated objective.

All fixed wing and most rotary wing aircraft are controlled by "synthetic pilots" using the Soar/IFOR (/Intelligent Forces) cognitive modeling system. Soar/IFOR entities are completely autonomous, so they do not require any interruption or management by human operators during a run. They are able to manage the variety of goals/objectives that arise in combat situations, reacting to the specific situation in the context of their mission and doctrine. They exhibit goal-oriented behavior -- both reactive and deliberative - - via a hierarchical, real-time, knowledge-intensive decision making process. Much of their behavior results from the simulated communication they have with Soar/IFOR entities, working as teams to achieve their goals, using doctrinally correct communications and organizations. (Tambe, et al, 1995) They can also interact with humans that are participating in the simulation. In contrast to other approaches, they incorporate a large amount of knowledge about their domain, including up to 4800 rules to drive their behavior.

The Soar behaviors are specialized to specific missions using real world practices. As an example, for USAF FWA, the knowledge about the mission originates from the ATO generated by CTAPS. The ATO is then further refined by a human who plans the missions just as a wing operations center would. This means that the people using the system are doing the things they normally would do real world, thus, training can be achieved every step along the way.

In RWA, there is a command entity -- a company commander -- in addition to the individual helicopter IFOR pilots. The command entity is based on Soar, the CFOR infrastructure, and a hierarchical partial-order planner built within Soar. The RWA pilots coordinate through a general model of teamwork.

While this work has been going on in the United States, the United Kingdom has their version of

Command Entities known as Command Agents (CA) to simulate a number of British commanders. These will be demonstrated in UE 98-1 in the United Kingdom.

The goal of all of the advanced cognitive behaviors approaches is to create autonomous entities that can participate in exercises without the need for constant human monitoring. This is a goal that, because of resource constraints, cannot be fully met for STOW-97. For large exercises, such as STOW's participation in UE 98-1, it is necessary to have a layer of operators and role players between the simulation and the training audience. This "response cell" layer must be tailored at both interface sides. On the simulation side, the response cell construct must take full advantage of the strengths of the simulations and compensate for simulation shortcomings. On the training audience side, it must ensure that it can provide the level of information that the audience would normally expect in "go to war" systems. For STOW-97, response cell design has been complicated by the fact that much of this effort was in parallel with software development. As the software product matures, response cell design should become simpler for USACOM's 1998 and 1999 exercises.

Synthetic Forces Initialization and Control

An integral part of any Synthetic Force are the commanders that control and guide the forces. While advanced behavioral models reduce the requirement for constant human monitoring and intervention, human commanders are still needed to setup the order of battle, develop OPORDS, and provide command and control at the higher levels. To make these commanders more effective, DARPA has developed a number of support systems including the ExInit System for initialization and the Command Talk system for voice commands.

Exercise Initialization

To speed up the process of designing and executing a large scale training exercise, DARPA has developed a unique Exercise Initialization (ExInit) system. Based on an Object Oriented, CORBA compliant, architecture using Pentium Pro PCs and the Windows NT operating system, the ExInit system saves hundreds of hours in setting

up an exercise. ExInit accomplishes this by using sophisticated algorithms to place large, complex units in a doctrinally-plausible position and posture relative to the terrain and the enemy situation. In addition, it uses advanced Human Computer Interfaces (HCI) to allow the exercise planner to use voice commands and computer-based pen inputs to accelerate and facilitate the exercise planning process. The ExInit architecture is designed to be simulation independent and has the potential to be used with multiple simulations.

CommandTalk

While today's graphical user interfaces are an enormous advance over previous text interfaces, they still require the operator to spend considerable effort to navigate extensive menu systems. Speech input provides an opportunity to radically speed up the interface and lower the training requirements for SAF operators. To expand the span of control of each operator, DARPA sponsored the CommandTalk program to provide a speech interface to each of the service SAFs. This included the basic speech recognition architecture plus specialized vocabularies for each of the services. As an example, before execution Soar planes can be given all necessary information to fly a specific mission or using CommandTalk, a human Forward Air Controller can provide the nine line brief just prior to the in bound commitment. CommandTalk will be used during the ACTD at each U.S. response cell to control at least one workstation.

C4I Interfaces

CCSIL messages are also used to facilitate information transfer between the simulations and warfighter's C4I systems. STOW will be using several of the products of the Defense Modeling and Simulation Office (DMSO) sponsored Modular Reconfigurable C4I Interface (MRCI) program to demonstrate this. For STOW-97, we will link the Army's MCS/P, the Army/Marine Corps AFATDS, and the Air Force's CTAPS. In addition, the Navy's JMCIS will be interfaced through a separate interface being developed by the STOW program. This will allow the warfighters to use the same equipment for simulation that they use for combat, thus, providing both greater training realism and reducing the training requirements of simulation operators. For example, the Air Force's CTAPS program will generate an Air Tasking Order (ATO) that is automatically translated into CCSIL by the MRCI. This will be picked up by the automated

wing operations center (AWOC) and used to semi-automatically generate all of the missions for simulated aircraft during an exercise. This technology eliminates the need for numerous humans to translate information from the real world into the simulation thus reducing simulation manpower requirements.

Hardware Requirements

The Synthetic Forces software has been modified to allow easy exploitation of processor hardware improvements. Currently the SF software runs on SGI and Sun workstations and Intel Pentium Pro (P6) computers. It will operate under UNIX, SunOS, and Linux operating systems with some applications ported to Windows NT. Within some limitations, exercises of any size can be run with a heterogeneous mix of processors and operating systems.

Characterization of combat entities per processor is difficult for the synthetic forces software since several factors affect this. First is the activity level of the entity in the synthetic battle space. An

entity actively engaged with several others obviously requires more processing power than that same entity in a quiescent state. Second is the level of complexity of the entity. A synthetic Soar/IFOR pilot is more complex and more capable than an individual tank commander and so requires greater processing resources. Finally, the size of the exercise and the additional processing load associated with sending and receiving information objects across a distributed network will require some variable amount of processor resources. While these major factors make the performance of the SF somewhat variable, average and peak performance values achieved during the STOW ACTD exercise, to be conducted in late October, will be presented at the conference.

CONCLUSION

The Synthetic Forces program has been a large scale, distributed, software program developed with numerous government, university and contractor personnel, as shown in Figure 1.

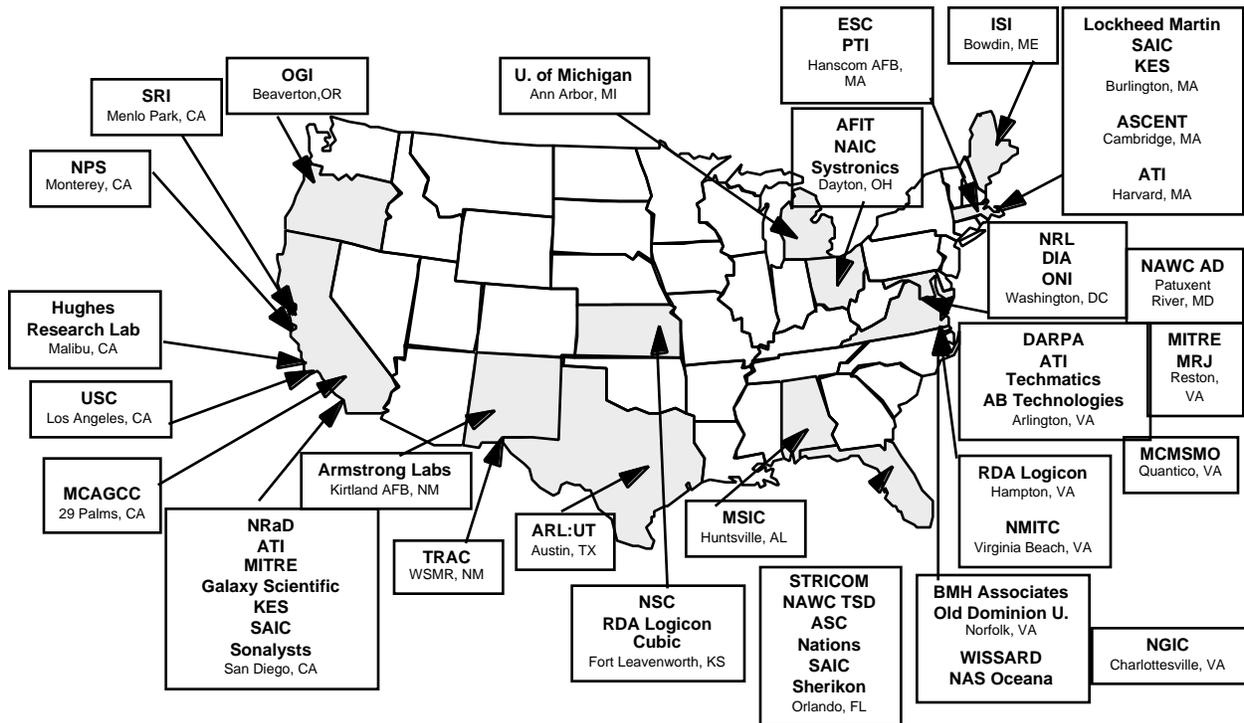


Figure 1. Key Synthetic Forces Participants

Results of the STOW-97 Synthetic Forces capabilities for each Service can be provided by the individual Service managers.

Air Force - Capt Kevin Yentz, USAF Electronic Systems Command, yentzk@hanscom.af.mil

Army - Sandy Veautour, STRICOM,
veautous@stricom.army.mil

Marine Corps - Jeff Clarkson, NCOSSC Research
and Development, clarkson@nosc.mil

Navy Surface - Doug Hardy, NCOSSC Research
and Development, hardydr@nosc.mil

Navy Air - Ed Harvey, BMH, harvey@bmh.com

OPFOR - Pete Wussow, NCOSSC Research and
Development, wussow@nosc.mil

Also, information on the results of the ACTD will be
available on the World Wide Web at
<http://www.stow.com/>

Information on the remaining two years of the
ACTD can be provided by the DARPA STOW
Program Manager, Larry Budge,
lbudge@darpa.mil.

ACKNOWLEDGMENTS

The authors would like to thank all of the Synthetic
Forces developers who's hard work and dedication
made this program a success. Your
professionalism and attitudes are to be admired
and replicated.

REFERENCES

Bachkosky, Jack (1996). *ACTD Times*,
www.acq.osd.mil/at/times.html, October 1996.

Lukes, G., "DARPA STOW Synthetic
Environments". *Proceedings of the 19th
Interservice/Industry Training Systems and
Education Conference*, Orlando, Florida,
December 1997.

Tambe, M.; Johnson, W. L.; Jones, R. M.; Koss,
F.; Laird, J. E.; Rosenbloom, P. S.; and Schwamb,
K. 1995. Intelligent Agents for Interactive
Simulation Environments. *Artificial Intelligence*
70:15-39.

Topper et al (1996). "The Compendium of Close
Combat Tactical Trainer Algorithms, Data, Data
Structures, and Generic System Mappings",
*AMSAA Combat Integration Division Special
Publication No. 74*, June 1996.

Ullom, L. and Fischer, P. (1996). "Using An
Ordnance Server to Provide Validated Weapon
Models to ModSAF". *Proceedings of the Sixth
Conference on Computer Generated Forces and
Behavioral Representation*. Orlando, Florida, July
1996.