Proceedings of the 1995 Winter Simulation Conference
ed. C. Alexopoulos, K. Kang, W. R. Lilegdon, and D. Goldsman

AN OBJECT-ORIENTED SIMULATION OF AIR FORCE SUPPORT EQUIPMENT USAGE

Eric A. Zahn
Nicholas J. Stute
Patrick K. Clark

TASC, Inc.
2555 University Blvd.
Fairborn, OH 45324, U.S.A.

ABSTRACT

This paper summarizes a joint study by the U.S. Air Force and TASC, Inc. that models aerospace support equipment usage during a deployment of fighter aircraft. The model explains the present role of the equipment, and introduces a possible alternative. The main objective of the study, to replace the current support equipment system while maintaining efficiency, is illustrated. The implementation of the model in IMDE, an object-oriented discrete-event simulation package, is then described. The two options are compared; results are displayed showing the number of sorties aborted with each equipment arrangement.

1 INTRODUCTION

The present military engagement scenario is vastly different from that expected five years ago. In the past, during the Cold War era, locations of military threats were somewhat predictable. The United States had stockpiled reserves of support material, pre-positioned at key locations, to be used when a crisis occurred. Deployment of fighters to those spots became much easier since the planes could utilize the support already present without taking every needed piece with them.

However, the current situation has changed considerably because of the dissolution of the Soviet Union. Many different deployment scenarios can be envisioned, spanning the globe from Iraq to Haiti and from Korea to Somalia. No longer can the Air Force rely on pre-positioned equipment to support their deployed fighters; the support assets must now be moved when they are needed. Research efforts have been targeted to reduce the amount of material needed to support a deployed unit; one study is detailed here, investigating a possible reduction in the number of Aerospace Ground Equipment (AGE) units used.

Each piece of AGE is a single-function motorized cart weighing over one thousand pounds. It has been determined that 25% of the deployed weight and 20% of the deployed volume constitute this type of support equipment. The current configuration is a leading mobility problem and limits deployment capability (Boyle 1994). One possible solution is to develop multi-function aerospace equipment that would replace the multitude of AGE units. This proposed unit, termed a MASS (Multi-function Aerospace Support System), combines power, air, hydraulic, and other utilities into one unit, thus potentially reducing deployment weight and volume. The purpose of this simulation study, undertaken by TASC, Inc., was to determine if a reduced number of MASS units can still support the same number of sorties as the current AGE units.

The simulation software chosen for the study is an object-oriented discrete-event package known as IMDE (Integrated Model Development Environment), developed by TASC, Inc., in conjunction with Armstrong Laboratory at Wright-Patterson AFB. The structure of the system to be studied lends itself well to the object-oriented methodology of IMDE, and model classes of parts, men, equipment, and aircraft can be stored separately in IMDE for use in further studies.

This paper is organized as follows. Section 2 describes the deployment model in depth, concentrating on the maintenance of a squadron of F-16 fighter aircraft with AGE and/or MASS units. Section 3 investigates IMDE and describes the implementation of the model in IMDE. Section 4 presents the results of the simulation study, comparing the different alternatives. Section 5 then concludes the paper by summarizing the results.

2 MODEL DEFINITION

Deployed squadrons of aircraft must fly missions throughout the duration of the deployment according to a specified schedule. These missions ultimately determine the success of the deployment. This model will not determine the military success of each mission flown; the goal is to get as many missions flying as possible and not to cancel any due to manpower or equipment restraints. Each deployment has certain parameters that define the specific military scenario: for instance, the deployment to be modeled will have 18 F-16s per squadron; missions would consist of two sorties; the deployment lasts 30 days; and the mission schedule would generate 2.0 sorties per aircraft per day (Carrico 1995). These missions would be set to fly at pre-defined times throughout the day.

2.1 Mission Process Overview

All missions to be flown for a particular day have a specified lead time, takeoff time, and cancel time. The lead time signifies the earliest time when planes can be assigned to the mission; the cancel time is the amount of time after the scheduled takeoff time when the mission must be aborted if it is not flown.

The airbase is notified of the day's various missions, and then attempts to assign available aircraft from the squadron to a mission at the beginning of the mission's lead time. If there are no aircraft available at this time, the mission is queued.

After being assigned to a mission, an aircraft undergoes pre-flight maintenance, which takes an average of thirty minutes. Afterwards, the aircraft is ready to fly, and the mission is flown at its scheduled time if all aircraft in the mission are ready. If a mission is not ready to be flown at the cancel time, the mission is aborted, and all assigned aircraft are returned to the available pool.

The mission flight is modeled by a randomly drawn time delay, as the intent of this model is to evaluate the effectiveness of support systems on the ground. After the mission has completed and the planes have landed, post-flight maintenance for each aircraft is done by checking for failed parts and subsystems. If failures are found, unscheduled maintenance is performed. This process is detailed below. After the completion of any necessary unscheduled maintenance, the aircraft returns to the available pool of F-16s ready to be assigned to another mission. Figure 1 displays a simple flow chart detailing the sortie process (Carrico, 1995).

2.2 Pre- and Post-Flight Maintenance

For this particular study, 614 different unscheduled maintenance tasks are modeled for an F-16. Each task corresponds to a particular aircraft subsystem, such as the flight control, landing gear, and radar subsystems. These subsystems fail at random times throughout the simulation run, drawn from mean field data values for each subsystem read in at run time. Each subsystem in the model has a failure clock as an attribute, where a breach of the clock represents a failure. When a failure occurs, a variety of tasks are performed, based on discrete probabilities. These tasks might require manpower, parts, support equipment units, or a combination of all three. Scheduled maintenance must be completed on each aircraft as well; men and support equipment are needed to perform a quick check on an aircraft each time an aircraft finishes a sortie. Delays in these processes might result in missions being aborted.

2.3 Existing Model Available

A computer model of the basic scenario of deploying aircraft and supporting squadrons has already been developed by the Air Force. The Logistics Composite Model (LCOM) was created in the late 1960's to model airbase logistics and calculate sortie generation capability. It has evolved since then to become the USAF standard modeling system for manpower and spares allocation. LCOM is a process-based discrete-event simulation package, where the main model is described in a centralized network of nodes. LCOM forms, or tem-

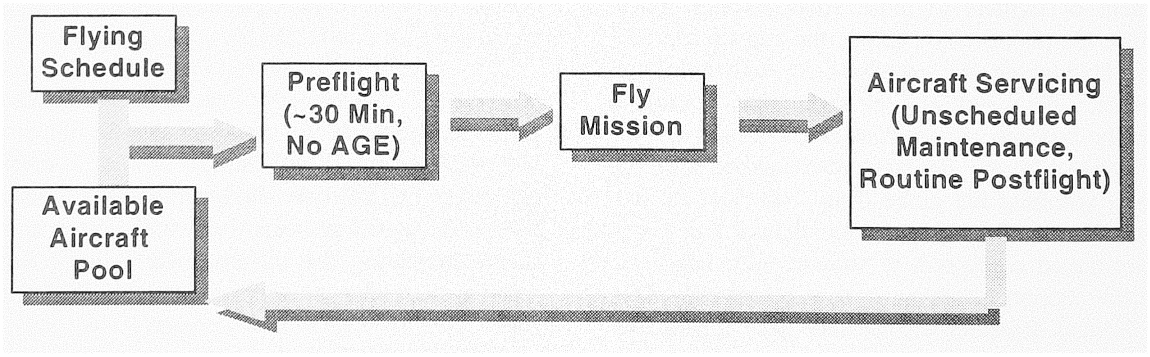

Figure 1: The Process of Flying a Sortie

plates, are used to describe the parameters of the model for a specific run (such as resource quantities, clock decrements, and sortie scheduling). There exists an LCOM database for each weapon system in USAF inventories, detailing all resources, tasks, and sequencing involved in a deployment (Boyle, 1990). However, all available simulation databases do not model support equipment units, such as AGE, and their effects.

2.4 Traditional AGE vs. Proposed MASS

A deployed airbase utilizes many different types of AGE units to support its aircraft. Seven different units to be studied are listed in Table 1. A defined mix of these resources, called a Table of Allowances (TOA), is taken along when deploying a squadron in order to ensure proper and timely service to the aircraft. One TOA used by Hill AFB in Utah is defined below.

A grand total of 43 AGE units are deployed with the squadron. The respective weights and volumes of each piece are also outlined in Table 1. Table 1 displays that this Table of Allowances calls for over 101,000 pounds of AGE units, occupying 11,500 cubic feet of space, to be deployed along with the 18 aircraft. It is no wonder that Air Force personnel and logistics study groups have termed the AGE situation as a leading mobility problem.

A MASS unit that combines multiple functions of the various AGE units into one could alleviate the weight and volume problems in deploying AGE. Simultaneous support services can be provided to an aircraft with one MASS unit instead of two or more AGE units. An initial blueprint for a MASS unit shows that the size and volume would not be much greater than that of any existing AGE unit; any deployments with a reduced number of MASS units would be lighter and smaller in terms of cargo weight and volume.

2.5 Criteria for Evaluation

The feasibility of MASS usage should be evaluated by two statistics. First, the number of sorties that are aborted over the length of the simulation should not increase with the MASS units in place. Secondly, the weight and volume of the required support equipment units should be reduced.

3 IMPLEMENTATION IN IMDE

IMDE is a domain-independent CASE tool for the development of object-oriented discrete event simulation models. The USAF Armstrong Laboratories Logistics Research Division has funded its development at TASC, Inc. since 1990. It is a single X Windows application which directly interfaces to the Versant Object-oriented Database Management System (OODBMS). IMDE was chosen as the simulation package for this study due to its flexibility in modeling special situations, its object-oriented, modular nature, and its graphical descriptions of the behavior of objects.

3.1 Properties of IMDE

IMDE is strongly based in the emerging arena of object-oriented technology. All model parts in the simulation are defined as "objects" that have both variables that describe the state of the object, called attributes, and functions that specify its behavior, termed methods. By storing all objects in its OODBMS, these simulation objects can be re-used in other simulations without recoding. An object's methods are completely defined by a network of graphical "click-and-drag" nodes, which allows both the analyst and customer to visualize the functionality of the method (Clark 1994).

Objects from any field of study can be defined in IMDE. Models from the aerospace, business, and manufacturing fields have all been simulated and

Table 1: Seven Types of AGE and Their Deployed Weights and Volumes

AGE Unit	Qty. Taken	Weight (lbs.)	Volume (cu. ft.)	Total Wt.	Total Vol.
High-pack compressor	2	2,000	179	4,000	358
Low-pack compressor	4	890	109	3,560	436
Nitrogen cart	3	3,340	252	10,020	756
Cooling air	9	1,290	302	11,610	2,718
Power generator	9	3,340	286	30,060	2,574
Hydraulic pressure	2	5,100	438	10,200	876
Lights	14	2,280	269	31,920	3,766
Totals	**43**			**101,370**	**11,484**

analyzed with equal success. The construction of simulation projects in IMDE can be done in a group environment; this allows multiple developers to create objects simultaneously and link them together. Finally, IMDE contains comprehensive data analysis tools, where simple statistics, confidence intervals, hypothesis testing, and sensitivity analysis can be conducted. Time trace plots of desired values can be shown quickly and easily.

3.2 The Model Building Process in IMDE

The construction of models within IMDE is based on building the model entities, or objects, and hooking them together to form complete simulation models. Templates for the design of these objects are called classes, a term borrowed from object-oriented methodology. In the model described above, one aircraft class, a template for the F-16s, would be created, but there are 18 separate instances, or objects, of that class in the model. The definition of a class in IMDE includes detailing its attributes and methods. A graphical editor is employed to visually define all the methods of any object. The editor allows a developer to define different behaviors of a single class with a separate flow chart that sequentially details the logic. Flow charts are created by dropping nodes that represent simulation actions onto a canvas and connecting them in the correct order.

Once the classes for all simulation objects have been defined, they can be linked together in the project editor. This editor allows the user to specify which entities will be present in the model and at what quantities. The project diagram, illustrating the interaction of model parts, is similar to a Coad-Yourdon OOA diagram without the inheritance relationships (Coad 1991). The diagram represents the aggregation relationships of the model. With this framework in place, IMDE allows the user to further configure the model. Class substitution can be performed, replacing original objects in the model with other objects sharing the same inheritance hierarchy. For instance, if a modeler wants to investigate the MASS feasibility with F-15s, a new simulation project can be formed by creating an F-15 class and substituting that class for the F-16 class. Also, lists of objects can be shortened or expanded. One of the attributes in the above model for a squadron is a list of aircraft. The initial number of aircraft per squadron can be changed at this level, and a new simulation could be run without recompilation. Each component of the project can have input sets and statistics collection sets attached, detailing which attributes are parameters to be specified in an experiment file, and which are noted for statistics collection.

At the conclusion of the simulation run(s), the data generated by the statistics collector is loaded into the database and can be analyzed within IMDE. As stated above, IMDE provides a complete data analysis package, calculating standard statistics as means, deviations, skews, and medians, as well as generating histograms, time traces, and scatterplots from the data.

3.3 Detailing the Model's Object Classes

In the support equipment model, four main classes are present that define most of the logic in the simulation. All are detailed below with a quick synopsis of their attributes and methods as defined in IMDE. A project diagram of the model is shown in Figure 2.

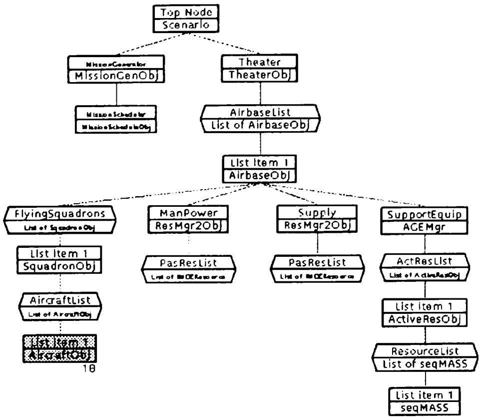

Figure 2: The Project Diagram of the Model in IMDE

First, instances of a Mission class are generated as determined by a external file, which describes the flying schedule for this deployment. Each Mission object has, as attributes, a list of aircraft assigned to the mission; the mission type, lead time, cancel time, takeoff time, and the minimum/maximum number of sorties flying in the mission. The main method of the class is the FlyMission method, which is invoked whenever the Mission gets enough aircraft to start preparing. Pre-flight maintenance is completed on all the aircraft assigned to it, all subsystem clocks are decremented and failures are fixed. The mission is flown by telling each assigned aircraft to fly this mission. Post-flight maintenance is performed, and the aircraft are returned to the Squadron for future assignments.

Instances of the Squadron class are responsible for the assignment of aircraft to waiting missions. Each Squadron object has a list of the aircraft it governs and a list of the current waiting missions. When a Mission is generated by the simulation, the Squadron is notified and its AssignAircraft method is called. This method views all available aircraft and assigns them to the Mission, based on aircraft configuration matches. Similarly, when a Mission is completed, the Squadron invokes its TakebackAircraft method, which removes the aircraft from the mission list, tries to assign them to any available mission or adds them to the available aircraft list.

The Aircraft object has attributes defining its current state (idle, assigned or in maintenance), a list of all 614 subsystems on the Aircraft that could fail, and a matrix detailing what support equipment is present at the aircraft during maintenance. When told to fly a mission, the Aircraft looks at the mission's parameters and then delays until the mission time. Statistical values are also updated. During post-flight maintenance, a method of the Aircraft class named AssessDamage is called by the mission. This method checks each subsystem belonging to the Aircraft, updates its countdown clock, and checks to see if it has failed. If one or more of its clocks has breached, the method FixPart of the subsystem is called and the simulation for this aircraft is suspended until all failed subsystems have completed this method.

Although each subsystem has its own, unique FixPart method which determines what processes define the repair, all 614 subsystem classes share a common parent. The parent class has attributes which define the countdown clock, failure and repair time means, and methods to update the clock. Each subsystem object shares the parent template but has its own values for the attributes.

3.4 Resource Management in IMDE

A special class, the Resource Manager Object class, is pre-defined in IMDE to handle requests for different resources. Attributes of this class are instances of resources that can be allocated to requesting objects. In this model, the Airbase class has three Manager Objects among its attributes, one each for men, parts, and support equipment. All requests for these resources are fielded by the corresponding Manager Object, which determines if the requests are granted or are blocked based on the number currently available. The objects and quantities of managed resources are defined in the Project editor by adding items to the list of available resources.

Due to the complex logic of allocating support equipment to an aircraft needing maintenance, changes were needed in the Resource Manager Object. If there were multiple requests for one type of AGE from the same aircraft, only one request needed to be granted since the AGE could be used on more than one repair simultaneously. A child class of the Resource Manager Object was created to manage support equipment resources. This child class kept most of the logic of its parent but overrode some allocation logic in order to correctly model the system. This is another advantage of IMDE: its object-oriented design allows a developer to make small, specialized changes in logic within a child class and still keep most of the properties of the parent.

This child Manager Object was designed to quickly substitute the proposed MASS unit for any AGE piece when the simulation warrants. The child object reads an input file which defines, for a given run, what functions a MASS unit contains. The support equipment Manager substitutes any requests for those functions with a MASS unit. Since the information describing what is contained in a MASS unit is an input to the manager, no recompilation is necessary. A new simulation model with different functionality included in the MASS unit can be run immediately.

4 RESULTS OF THE SIMULATION

Four different configurations of the model were run in IMDE. As stated above, the main metric in determining the advantages of one model over another is the number of sorties aborted during a thirty-day period. Other statistics of interest include the average utilization of the support units and the average number waiting for the AGE/MASS units.

The four model configurations are summarized below in Table 2. The first and second experiments are simulation runs with existing AGE units, with quantities defined by the TOA, and the reliability of the units varied. There is no reliable source for information regarding breakdowns of these units. Therefore, two extreme cases were tested, one in which the mean time to failure (MTTF) of the AGE units is 10,000 hours and the other with each AGE unit failing with a mean of 100 hours. It is assumed that the true MTTF of AGE units is located between the values used in these two experiments. The third and fourth experiments replace all AGE with 8 and 6 MASS units, respectively. Each MASS unit contains full functionality and each MASS can provide every service of the seven AGE units of interest.

Table 2: Definitions of the Four Experiments.

Exp #	AGE or MASS?	Qty.	MTTF(hrs.)
1	AGE	TOA	10,000
2	AGE	TOA	100
3	MASS	8	100
4	MASS	6	100

Ten runs of each experiment were performed; the average number of sorties aborted over those runs for each experiment is seen in Table 3. Confidence intervals (with $\alpha = 0.05$) were created and are also depicted in Table 3. These intervals show no statistical difference (with 95% confidence) between the four different experiments. If the analyst is solely looking at support equipment usage with AGE and MASS, replacing all existing AGE units with either 8 or 6 MASS units would not affect the number of aborted sorties in a thirty-day deployment.

Table 3: 95% Confidence Intervals for Aborted Sorties

Exp #	Lower Bound	Mean	Upper Bound
1	54.50	68.40	82.30
2	64.98	78.60	92.22
3	55.67	64.80	73.93
4	66.22	77.40	88.58

However, AGE is used for other purposes in a deployment situation as well. For instance, generators are used to provide electricity to offices, facilities, and other needed places. The utilization of the MASS unit for the experiments should not differ much from existing values for a generator since the MASS unit must also fulfill these other needs as well.

Figure 3 compares the utilization of a generator in experiments 1 and 2 (deployments using existing AGE units with different reliabilities) with the utilization of the proposed MASS unit in experiments 3, with 8 MASS units present, and experiment 4, with only 6 MASS available. The values presented are averages over the 10 runs. The utilization of 8 MASS units in experiment 3 maintains the value near the present value, while experiment 4 with only 6 MASS units in place shows a 13% increase in utilization.

One last statistic of interest is the average number of requests waiting for the AGE/MASS unit during the deployment. Figure 4 compares average values for each experiment; experiments 1 and 2 show waiting requests for the most-wanted AGE unit, the generators. When 8 MASS units replace the AGE units in experiment 3, the number waiting does not increase significantly. However, with only 6 MASS units present in experiment 4, the average value increases by a factor of eight. This must be pointed out when making recommendations.

5 CONCLUSIONS

These initial results have shown that by replacing all existing AGE units with 8 MASS units, no additional aborted sorties would occur. Also, the utilization of the MASS unit is relatively close to current levels, and the number of pending requests throughout the deployment is similar to levels found currently. A scenario deploying only 6 MASS units is also satisfactory in generating the needed sorties. However, utilization levels and pending requests increase significantly. If preliminary engineering figures are correct, and the weight of a proposed MASS unit is close to the heaviest AGE unit, the total weight of support equipment during a typical deployment would be 60% less than the current configuration used. The necessary volume for support equipment would be diminished even further with MASS usage.

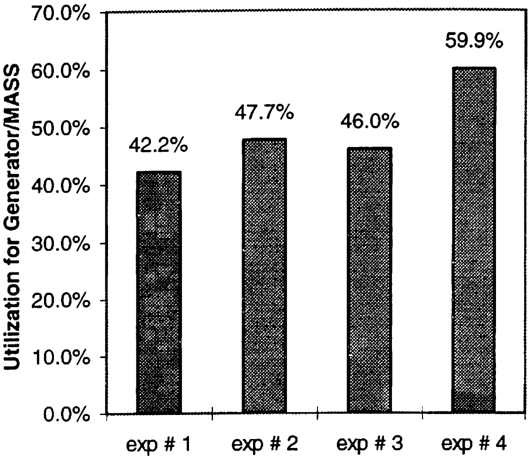

Figure 3: Average Utilization Values

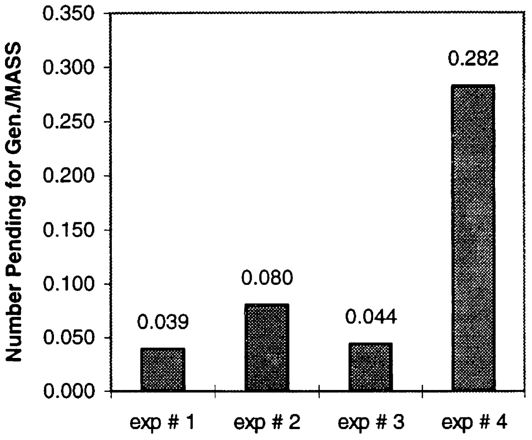

Figure 4: Average Requests Pending Values

Further research on this topic would include numerous other factors that would enhance this model and make it more flexible. First, the importance of the non-support uses of AGE units should be quantified and developed. Different schedules for generating missions can be used to see what effect they would have on the number of aborted sorties. The travel times of the support units, traveling from the shop to the plane, can be varied. More information should be obtained concerning the reliability of current AGE units and proposed MASS units. Finally, it may not be feasible from an engineering standpoint to have seven unique functions in a MASS unit; different functionalities of the MASS unit must be studied and analyzed.

ACKNOWLEDGMENTS

The authors would like to thank the USAF Armstrong Laboratory, Logistics Research Division at Wright-Patterson AFB for their support of this effort under the SIDAC (Supportability Investment Decision Analysis Center) contract. The authors also thank the following people for their assistance in obtaining key information: Capt Pat Vincent, Capt Todd Carrico and Mr. Ed Boyle at Wright-Patterson AFB; CMSgt Orin Grossjean at Springfield Air National Guard; and Mr. Jeff Sumner at TASC, Inc.

REFERENCES

Boyle, E. 1990. LCOM Explained. Logistics and Human Factors Division, Wright-Patterson AFB, Ohio.

CACI Products Company. 1991. *MODSIM II Reference Manual.*

Carrico, T. et al. 1995. Integrated Model Development Environment (IMDE) Multi-Function Aerospace Support System (MASS) Study. Technical Report. AL/HRG, Wright-Patterson AFB, Ohio.

Clark, P. K. et al. 1994. Object-Oriented Simulation with IMDE. In *Western Simulation Multiconference - Object-oriented Simulation Conference '94 Proceedings*, 131-136.

Coad, P. and Yourdon, E. 1991. *Object-Oriented Analysis.* New York: Prentice Hall, Inc.

AUTHOR BIOGRAPHIES

ERIC A. ZAHN is a Member of the Technical Staff at TASC, Inc. He received his B.S. in Systems and Control Engineering in 1993 and his M.S. in Systems, Control and Industrial Engineering in 1994, both from Case Western Reserve University. His interests include discrete-event and object-oriented modeling and simulation.

NICHOLAS J. STUTE is a Member of the Technical Staff at TASC, Inc. He received his B.S. in Computer Science in 1990 and his M.A. in Applied Mathematics in 1992, both from Bowling Green State University. He has worked on the IMDE modeling tool for the past 2 1/2 years.

PATRICK K. CLARK is Manager of the Advanced Modeling and Simulation Section at TASC, Inc. He received his B.S. in Chemistry from the USAF Academy in 1980, and his M.S.E.E. from the University of Colorado in 1984. He served in the U.S. Air Force for nine years in a variety of positions ranging from laboratory research in laser weapons development to flight software and simulation software development for the B-1B strategic bomber. His current research interests include object-oriented modeling and simulation, airbase/airport logistics, computer network simulation, and business process reengineering.