AC 2007-1245: EDUCATIONAL TOOLS FOR SYSTEMS SIMULATION AND LABORATORIES LEADING TO THE CAPSTONE DESIGN SEQUENCE IN AEROSPACE ENGINEERING

Karl Siebold, Embry-Riddle Aeronautical University
KARL H. SIEBOLD, Ph.D.

Currently an Assistant Professor of Aerospace Engineering at the Embry Riddle Aeronautical University in Prescott Arizona, where he teaches Spacecraft Systems Engineering and Spacecraft Capstone Design courses. Additionally he teaches Robotics, Thermodynamics, Space Mechanics, Spacecraft Attitude Determination and Control, Control Systems Analysis and Design. He also taught at the Universities of Houston/Clear Lake, Colorado/Colorado Springs at the Johnson Space Center, and Texas A&M in Galveston space related graduate level Engineering and Physics as well as undergraduate level engineering sciences courses. He has more than ten years of experience as an engineer/scientist at the NASA Johnson Space Center in the areas of Space Debris research and Rendezvous Proximity Operations and Capture simulation.

James Helbling, Embry-Riddle Aeronautical University

JAMES F. HELBLING, M.S.A.E. Currently an Assistant Professor of Aerospace Engineering where he teaches structural analysis, computer aided design, and aircraft detail design courses. He has 21 years of industry experience with McDonnell Douglas (now Boeing) and Northrop Grumman Corporation where he specialized in structural fatigue loading and served as manager of F-5/T-38 Engineering.

Darin Marriott, Embry-Riddle Aeronautical University
Darin W. Marriott, Ph. D. Aerospace Engineering

Dr. Marriott is currently an Assistant Professor of Aerospace Engineering at Embry Riddle Aeronautical University. He teaches space propulsion systems, experimental space systems and computer aided design. His graduate research focused on plasma dynamics for space propulsion and his current research involves creation of linear induction catapults for researching high speed launch applications.

Mischa Kim, Embry-Riddle Aeronautical University

Mischa Kim, Ph.D., Dipl.Ing. Currently an Assistant Professor of Aerospace Engineering at Embry-Riddle Aeronautical University in Prescott, Arizona. He teaches Control Systems, Spacecraft Attitude Dynamics & Control, Space Mechanics, and Dynamics. His research interests lie in the field of nonlinear dynamics and control with particular emphasis on spacecraft applications.

Educational Tools for Systems Simulation and Laboratories Leading to the Capstone Design Sequence in Aerospace Engineering

Abstract

During the industrial product development cycle simulation has been playing an increasingly important role, not only during the preliminary design and analysis phases but also through the whole mission operations phase. In a typical university curriculum emphasis during the freshmen, sophomore, and junior years is put on the analysis of engineering problems. In the senior year students are expected to make a switch from analysis based coursework (one answer to an analysis problem) to design based curriculum (multiple answers to a design problem.) Simulation can play an important role to facilitate this transition. A modern curriculum should include teaching the necessary computer tools during early classes, where the student can build course content specific models (for example a thermal model) and save them for later usage in the design classes. At the same time the curriculum should offer a laboratory experience, which validates and fortifies the material. Therefore it is essential to integrate computer-based simulations with hardware interface into the curriculum in a systematic manner. It is clear that computer-based simulation and analysis is indispensable in engineering science and design.

A curriculum is being developed in which analysis methods are synchronized with a core set of software tools. Instruction in these tools will be geared towards teaching students how to use these sophisticated tools. It will also emphasize how to understand and interpret the results using experimental, theoretical and numerical concepts. By combining analysis, simulation, and hardware interfaces students will have a coherent reinforcement of concepts in order to improve their computing skills while at the same time strengthening their grasp of the fundamentals.

Introduction

During the Program/Project Life Cycle of any sophisticated and financially demanding project, simulation plays a dominant role not only in the development, but also in the operations/maintenance phases. However, in order to intelligently make use of the multitude of simulation products available one has to achieve a fundamental understanding of the driving concepts of simulation, which is numerical integration. For this purpose a curriculum timeline has been developed at Embry Riddle Aeronautical University, which tries to parallel NASA's Program/Project Lifecycle /1/. Since the curriculum leads into the capstone design sequence, a schematic displaying the different project phases with its corresponding classes is shown in the following table /Table 1/. It is clear that credit hour constraints make it difficult to take all in depth classes before the actual design sequence starts. The simulation concept understanding and simulation building process is shown in the last row.

	FORMULATION			
Phases	Pre-Phase A Advanced Studies	Phase A Preliminary Analysis	Phase B Definition	
	Mission Feasibility	Mission Definition	System Definition	Preliminary Design
Major Reviews	**MCR**▼ Mission Concept Review	**MDR**▼ Mission Definition Review	**SDR**▼ System Definition Review	**PDR**▼ Preliminary Design Review
Classes	Space Systems Engineering (3) & Experimental Space Systems Engineering Laboratory (3), Space Mechanics (3)		Preliminary Spacecraft Design (4), Control Systems Analysis and Design (3), Spacecraft Attitude Dynamics and Control (3), Space Propulsion (3)	
Simulation	Concepts of Numerical Integration, State Variables and State Derivatives, Object Oriented Programming Techniques (Matlab), Simulation Tools (Simulink)		Spacecraft Simulator Development, State variables and derivatives for Subsystems, Subsystem Requirements and Interface Definition	

IMPLEMENTATION				
Phase C Design	Phase D Development			Phase E Operations
Final Design	Fabrication & Integration	Preparation for Deployment	Deployment & Operational Verification	Mission Ops
CDR▼ Critical Design Review	**SAR**▼ Systems Acceptance Review	**FRR**▼ Flight Readiness Review	**ORR**▼ Operational Readiness Review	**DR**▼ Disposal Review
Spacecraft Detail Design (4), Science Instrumentation Lab (3), Technical Electives (3)				
Replacing simulated Subsystems with Hardware & Testing				

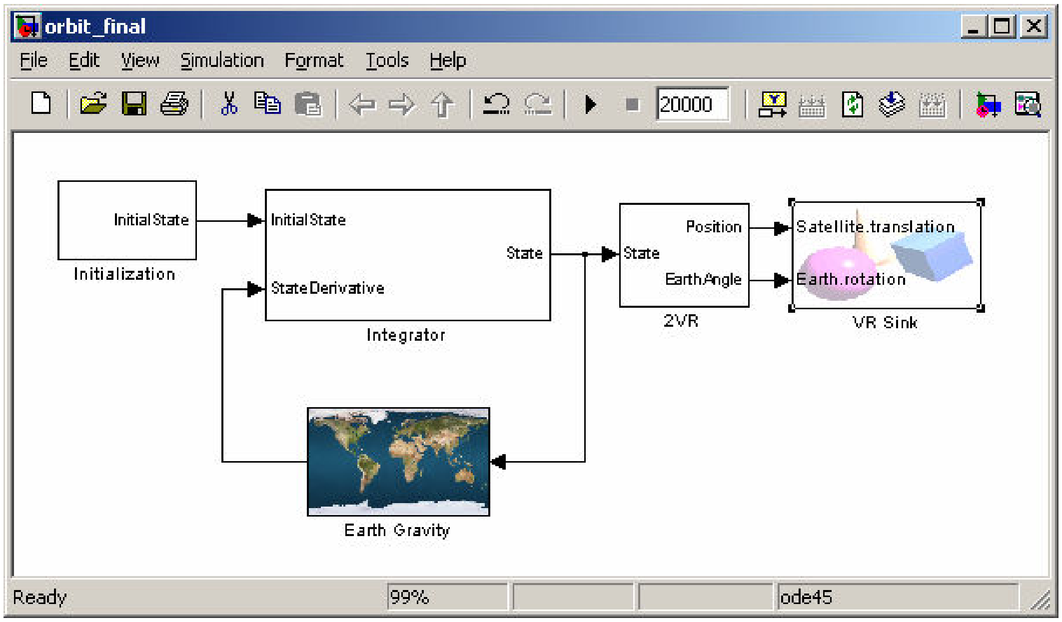
- Table 1 –
NASA Program/Project Life Cycle & Related Classes for the Capstone Design Sequence

Simulation Concepts

In order understand fundamental simulation concepts one needs to have a look at simple numerical integration concepts and their implementation /2/. It is important to visualize that only a first order differential equation of type $\dot{\vec{s}} = f(\vec{s}, t)$ needs to be solved. $\vec{s}$ is the state variable and $\dot{\vec{s}}$ is the state derivative, which must be vectors of the same size. This is accomplished in the space systems engineering course using the single step Euler algorithm with one state variable and its derivative. The regular Matlab programming language is used to accomplish this task. Once the concept of initialization, derivative, and actual numerical integration is understood, the class advances to midpoint methods like the Runge Kutta 2 and Runge Kutta 4 algorithms. The Runge Kutta 4 algorithm with adaptive step size control is the workhorse of the industry and is also the default integrator in Simulink.

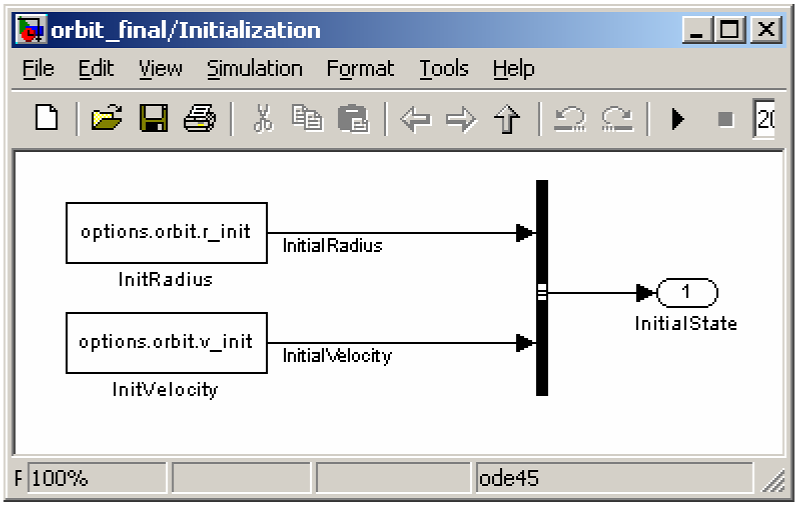
Simulator Development

Once these concepts have been understood the class builds an orbital simulator in the Simulink environment. The first task is to describe the state vector and its derivative and build an integration module with external initialization. The state vector for this problem is simply the radius $\vec{r}$ and velocity vector $\vec{v}$ in the inertial Earth centered reference frame: $\vec{s} = \begin{pmatrix} \vec{r} \\ \vec{v} \end{pmatrix}$ and its derivative $\dot{\vec{s}} = \begin{pmatrix} \vec{v} \\ -\frac{\mu}{r^3}\vec{r} \end{pmatrix}$. It is of utmost importance to understand that the state derivative may only depend on the state itself and time. For this simple case it actually depends only on the first three elements of the state i.e. the radius. /Figure 1/ depicts the overall simulation architecture for this case.

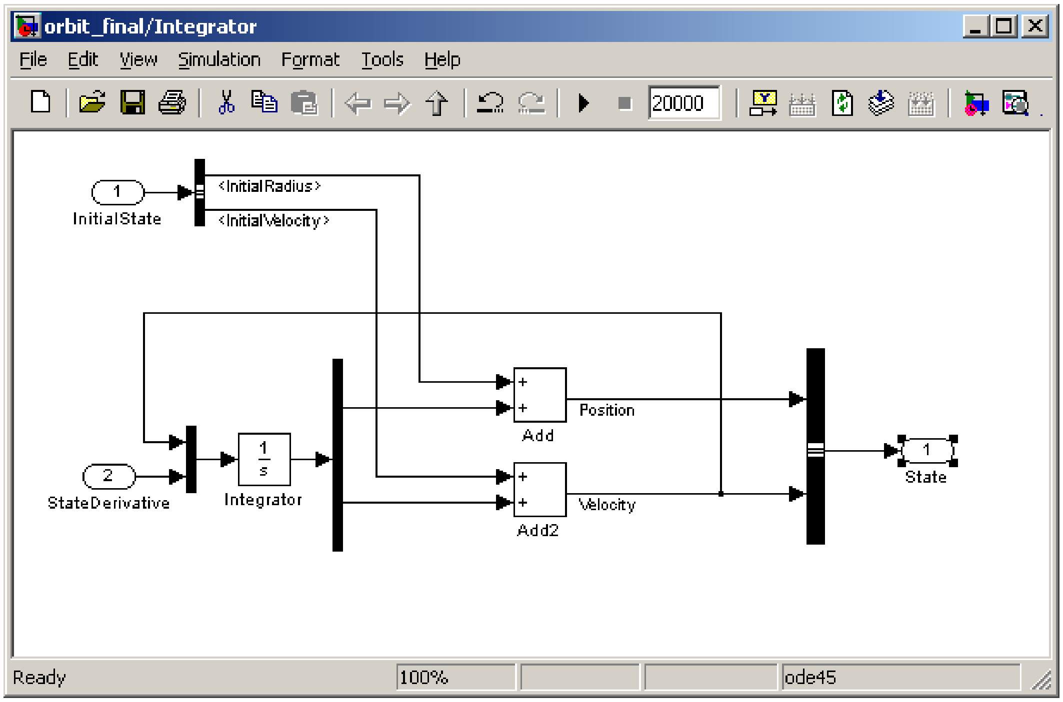


-Figure 1 –
Simulation architecture for two body problem

/Figure 2/ and /Figure 3/ show the initialization and integration mechanism respectively.

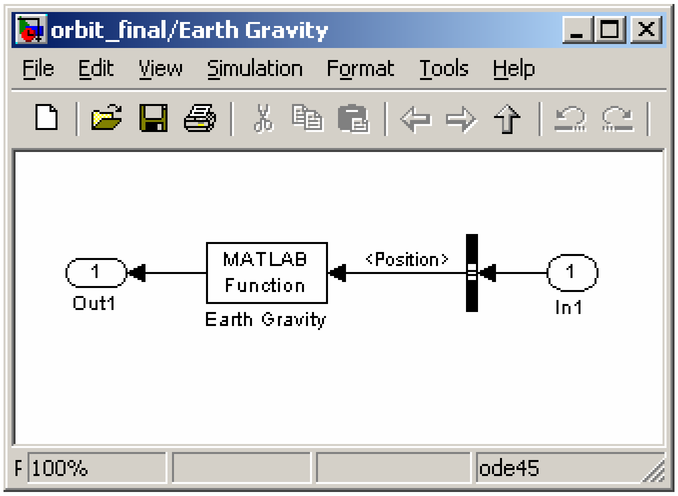


-Figure 2-
Initialization

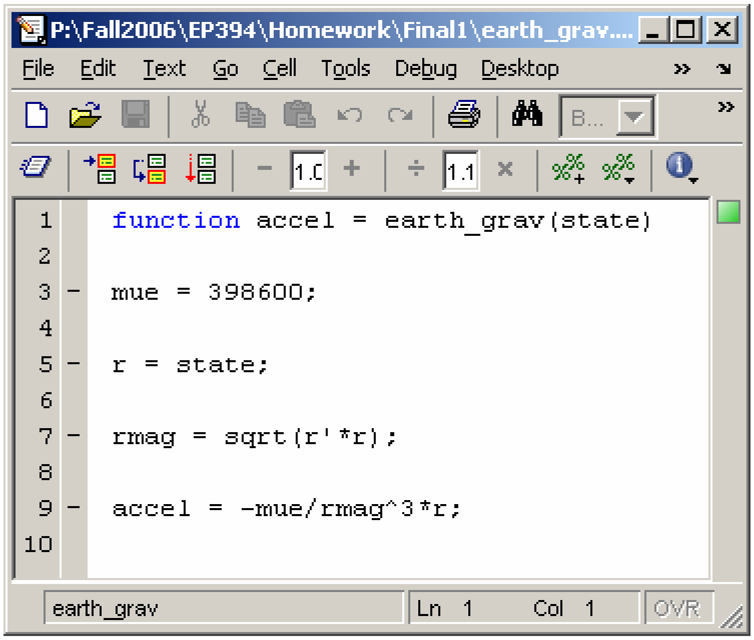


-Figure 3-
Integration Mechanism

As previously explained, the integration requires the state derivative as an input. For the two body problem, the acceleration (derivative of the velocity) is simply the gravitational acceleration caused by the Earth. ($\mu = 398600\frac{km^3}{s^2}$) and depends solely on the position. /Figure 4/ and /Figure 5/ show the subsystem block and the function, which performs the calculation. The derivative function's output is simply the Earth's gravitational acceleration.



-Figure 4-
Earth Gravity Subsystem



```
function accel = earth_grav(state)

mue = 398600;

r = state;

rmag = sqrt(r'*r);

accel = -mue/rmag^3*r;

```

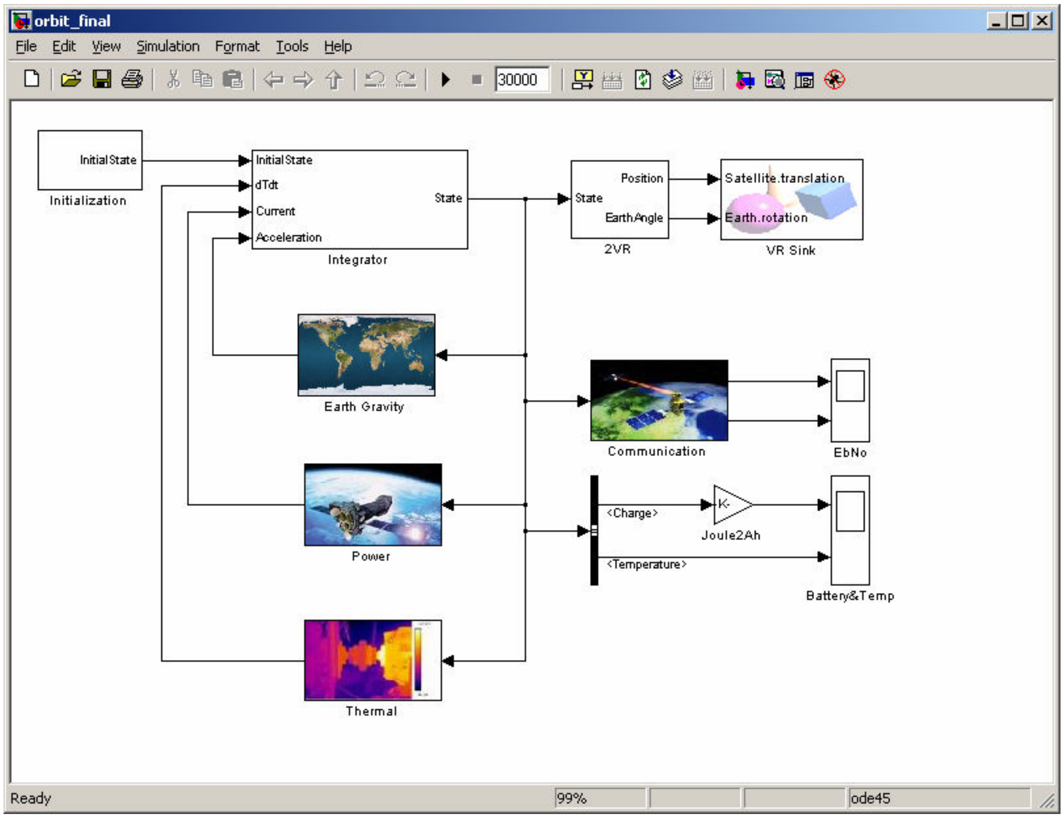
-Figure 5-
Acceleration caused by Earth's gravity

Simulator Upgrade

Once the basics of simulation and simulation architecture are understood, subsystems can be added in two different ways. The first is for analysis purposes, which depend on and have no influence on the current state variables. A good example is the analysis of a communications subsystem. In very simple terms the signal to noise ratio will only depend on the distance to a ground station and visibility conditions. Since this is only a function of the current state and time (radius vector and position of the Earth in the inertial frame), no new state variable needs to be defined. /Figure 6/ depicts the communication subsystem as a one way street with no feed back.

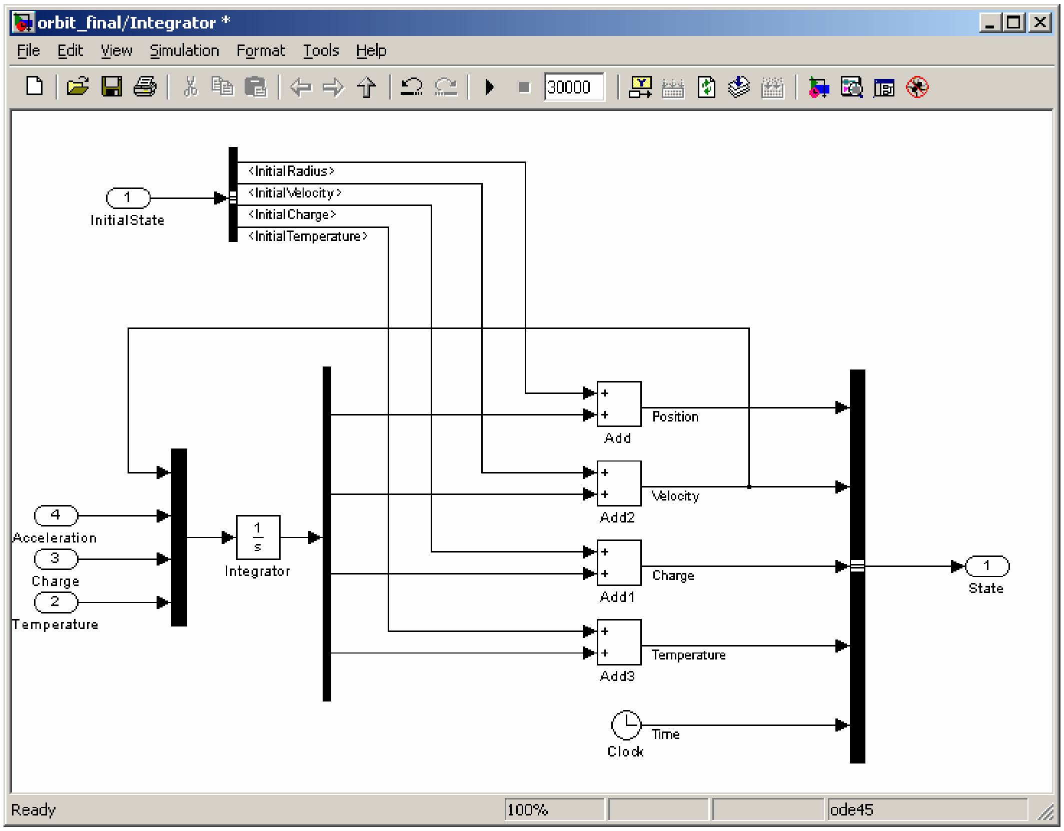
The second possibility is the expansion of the state vector by a new state variable, whose derivative may only depend on it and the other state variables. One example may be the temperature of a solid spherical satellite. The new state variable will be the temperature T and its derivative: $\frac{dT}{dt} = \frac{Q_{in} - Q_{out}}{mc_p}$.

Since $Q_{out} = \sigma \varepsilon A_{surf} T^4$ depends on T and Q_{in} depends on shadow conditions, a derivative function can be developed. /Figure 6/ and /Figure 7/shows this addition in the feedback loop and in the integrator.



-Figure 6-
Addition of Subsystems to the Spacecraft Simulator

This simulator is now ready to perform analysis for the spacecraft systems configuration as mentioned earlier. This simulator can be used to define requirements single subsystems pose on the overall spacecraft and vice versa. It is clear that by adding new or more complex subsystems, the state variables need to be redefined and the integration and initialization needs to be updated. Since the parameter definition is part of the initialization, parametric studies can be performed for sizing and subsystem specifications.



-Figure 7-
Addition of State Variables to the Integrator of Spacecraft Simulator

Simulation in the Space Systems Laboratory

For experimentation and verification purposes the university supplies the aerospace engineering students with an experimental space systems laboratory course combination, AE325/AE326 which supports several different ABET outcomes and objectives primarily through student's hands on experience. In order to optimize the learning experience, all students are required to take a single lecture based course AE325 which enables the students to analyze the experiments both analytically and numerically. Coupling this class with the hands on experimental class AE326 allows for a merger of theory, numerical simulation and experimentation.

AE 326 is broken up into the following major subjects: propulsion, angular momentum/attitude, power supplies & electronics, energy transfer and space environment. Each subject consists of a lecture, followed by an in class assignment where the students perform all necessary calculations by hand. These assignments include basic back of the envelope type of equations to analyze the system they are assembling and testing in AE326. The results are presented in a laboratory report written individually by students in the same basic format as AIAA journals. This lab report is the only permitted crib sheet for the quiz which concludes each subject.

During the first meeting of the AE 326 class students are introduced to the safety protocol in the machine shop by the machinist. During five of the fourteen other three hour labs, the students are asked to perform laboratory experiments on various ‘satellites.’ Each ‘satellite’ is completely student built during previous semesters and is an autonomous entity with solar panels, circuit boards, power systems, communication systems and attitude control systems. The experiments are designed to give results which must be compared against the numerical model designed in AE325. Examples of such experiments are measuring the pressure decay in the storage chamber and thrust of the air gas thrusters as a function of time and measuring the angular location of a satellite on the air bearing as a function of time during and following the thruster burns. During the other nine AE326 meetings the students also prepare a semester project in which each section builds their own ‘satellite’ or improves on a previously existing ‘satellite’ in such a way as to progress the hardware on which they and future classes of AE326 have to experiment. During the nine other meetings students also have time to work on their numerical models on the computers in the lab room.

The propulsion section introduces the Mach number - Area relationship as well as the isentropic equations and the thrust equation. Students are first requested to create a Matlab file which solves for the subsonic/supersonic Mach number in a converging/diverging cold gas thruster chamber-nozzle. They then create a numerical model of the cold gas thrusters used as attitude control thrusters on the student satellite floating on the air bearing. The model must track the decaying storage pressure in both cases of having a regulated and unregulated burn. The thrust is then plotted as a function of time. The numerical results are compared against the actual thruster by means of a small thrust stand using four strain gauges. After the model is proven adequate for one experimental implementation, the input parameters are varied and compared to other scenarios.

The angular momentum/attitude lab requires student’s to re-use the numerical integration routines from the previous thruster lab. The model must use the force calculations from the attitude control thrusters to predict the torque and acceleration of the satellite (taking into account a loss due to aerodynamic drag in the room) as it accelerates rotationally on an air bearing. The numerical model must be verified by graphing against the experimentally measured data points.

Simulation in the Aircraft Capstone Sequence

The AE421 Aircraft Detail Design course is the second of a two-part capstone sequence during which design teams perform detail design on an aircraft conceptualized in the Aircraft Preliminary Design course. The course requirements include the fabrication and testing of both wind tunnel and structural models representative of a chosen aircraft component. An emphasis is placed on using test as a means of verifying analytical predictions. Aerodynamic coefficient data is analytically defined and substantiated through wind tunnel testing, and computer generated finite element models are created to predict structural failure of physical aircraft component models.

The latter method allows for simulation of strain and deflection of the structural models that are constructed, mounted, and loaded to best simulate the configuration and flight environment of the conceptualized aircraft. In the past, this simulation has only occurred prior to test to assist in defining the predicted failure load and locations where strain and deflection measurements should be obtained. However, with the addition of improved facilities at Embry Riddle Aeronautical University in Prescott, it was recently discovered that a substantial improvement in course instruction occurs when students are able to perform real time simulation of their structural models as they are being tested. The computer facilities resident in the new Structures Lab allow design teams to monitor strain and deflection measurements relative to their computer simulation as the test is occurring. This process allows students to perform real time model verification and make adjustments to their testing sequence by simulating anomalies that occur during the test. By combining theoretical analysis, computational simulation, and verification through experimentation, the Aircraft Detail Design course offers students an opportunity to implement tools learned in previous courses and apply them to real aircraft design problems.

Conclusions

Since the complete development cycle of spacecraft and aircraft are based on simulations, students need to be prepared to understand, create, and verify their own simulations. This is being done at ERAU during classes leading to the capstone design sequence and during the design classes itself. Requirements documents, test plans, and system specifications and validations all have a simulation component. Verification and visualization with hands on approach supplement the understanding of the design process in the laboratories. This complies with the student outcomes and objectives, required by ABET.

/1/ ***Anonymous***, NASA Systems Engineering Handbook, SP610S, June 1995

/2/ ***Press, William*** et al., Numerical Recipes, The Art of Scientific Computing, Cambridge University Press, 1989

Appendix

Partial Reproduction of NASA’s Project Life Cycle Chart

Formulation			
Pre-Phase A Advanced Studies	Phase A Preliminary Analysis	Phase B Definition	
Mission Feasibility	Mission Definition	System Definition	Preliminary Design
MCR ▼ -Study Plan -Follow-on Plan -Mission Goal and Objectives -Concept/Design Evaluation Criteria - Mission Concepts -Operation Concepts -Life Cycle Cost Estimates -Feasibility Assessment	**MDR** ▼ -System Engineering Management Plan -Information Management Plan -Eng. Master Plan/Master Schedule -Risk Management Plan -Mission Need Statement -Functional Mission Concept -Preliminary System Specification -Science Requirements -Trade& Analysis Results -Technology Development Plan	**SDR** ▼ -Program/Project Management Plan -Development Plan -Statement of work -System Spec. Plan -Configuration Management Plan -Document Structure Tree - System Concept & Architecture -System Specification -Interface Requirements -Environmental Specification -Human System Standards -Concept/System Evaluation Criteria -Development Test Pans -Engineering Tests -Hardware/Software List -Risk Analysis -Development Test Results -Technology Development Requirements	**PDR** ▼ -Work Breakdown Structure -Technical Performance Measurement Plan -Contamination Control Plan -EEE Part Management Plan -Parts Control Plan -Environments Control Plan -Integ. Log. Support Program Plan -EMI/EMC Control Plan -Produc./Manufacturability Prog. Plan -Reliability Program Plan -Quality Assurance Plan -Applicable Standards -Design to Specifications -Vendor H/W & S/W Specification -Disposal Requirements -Specification Tree -Drawing Tree/Eng. Drawing List -Interface Control Document -Payload to Carrier Integration Plan -Verification Plans -Verification Requirements Matrix -Environmental Impact Statement

Implementation				
Phase C Design	Phase D Development			Phase E Operations
Final Design	Fabrication & Integration	Preparation for Deployment	Deployment & Operational Verification	Mission Operations
CDR ▼	**SAR ▼**	**FRR ▼**	**ORR ▼**	**DR ▼**
-Manufacturing Plan -Tech. Performance Measures Report -Materials and Processes Control Plan -Integrated Logistics Support Plan	-Operations Plan	-Certification of Flight/Launch Readiness		
-Build to Specifications -Manuf. Processes Requirements -Design Disclosure -Operational Limits and Constraints -Integrated Schematics -Spares Provisioning List -Qualification Items -Launch Operations Plan -Transition to Operations Plan -Disposal Plans -Acceptance Plans -Acceptance Criteria -Verif. Requirements and Specifications -Verification Procedures -Integration and Assembly Plan - Instrum. Program and Command List	-Operations Procedures -Training Plan -User Manuals -In-Flight Checkout Plans -Computer Resource Integrated Support Document -Verification Data -Waivers	-H/W % S/W End Items -Operations Data -Launch Facility C/O Results -Go/No-Go Criteria	-Operations Evaluations Results -Problem/Failure Reports -Technical Manuals/Data -Trained Personnel	-Mission Products -Sequential Production -Replacements and Upgrades