65th International Astronautical Congress, Toronto, Canada, October 2014.

IAC-14-C1.6.4x21108

DEVELOPMENT OF AN INTEGRATED SPACECRAFT GUIDANCE, NAVIGATION, & CONTROL SUBSYSTEM FOR AUTOMATED PROXIMITY OPERATIONS

Peter Z. Schulte
Graduate Research Assistant, Space Systems Design Laboratory
Georgia Institute of Technology, Atlanta, Georgia, United States
pzschulte@gatech.edu

David A. Spencer
Professor of the Practice, Space Systems Design Laboratory
Georgia Institute of Technology, Atlanta, Georgia, United States
david.spencer@aerospace.gatech.edu

This paper describes the development and validation process of a highly automated Guidance, Navigation, & Control (GN&C) subsystem for a small satellite on-orbit inspection application. The resulting GN&C subsystem performs proximity operations (ProxOps) without human-in-the-loop interaction. The paper focuses on the integration and testing of GN&C software and the development of decision logic to address the question of how such a system can be effectively implemented for full automation. This process is unique because a multitude of operational scenarios must be considered and a set of complex interactions between various GN&C components must be defined to achieve the automation goal. The GN&C subsystem for the Prox-1 satellite is currently under development within the Space Systems Design Laboratory at the Georgia Institute of Technology. The Prox-1 mission involves deploying the LightSail 3U CubeSat, entering into a leading or trailing orbit of LightSail using ground-in-the-loop commands, and then performing automated ProxOps through formation flight and natural motion circumnavigation maneuvers. Operations such as these may be utilized for many scenarios including on-orbit inspection, refueling, repair, construction, reconnaissance, docking, and debris mitigation activities. Prox-1 uses onboard sensors and imaging instruments to perform its GN&C operations during on-orbit inspection of LightSail. Navigation filters perform relative orbit determination based on images of the target spacecraft, and guidance algorithms conduct automated maneuver planning. A slew and tracking controller sends attitude actuation commands to a set of control moment gyroscopes, and other controllers manage desaturation, detumble, thruster firing, and target acquisition/recovery. All Prox-1 GN&C components are developed in a MATLAB/Simulink six degree-of-freedom simulation environment and are integrated using decision logic to autonomously determine when certain actions should be performed. The complexity of this decision logic is the main challenge of this process, and the Stateflow tool in Simulink is used to establish logical relationships and manage data flow between each of the individual GN&C hardware and software components. Once the integrated GN&C simulation is fully developed in MATLAB/Simulink, the algorithms are autocoded to C/C++ and integrated into flight software. The subsystem is tested using hardware-in-the-loop on the flight computers and other hardware.

I. PROX-1 MISSION DESCRIPTION

Prox-1 is a small satellite mission designed, built, and operated by students at the Georgia Institute of Technology (Georgia Tech) under the University Nanosatellite Program at the Air Force Research Laboratory (AFRL). The primary mission of Prox-1 is to demonstrate automated relative trajectory control in Low-Earth Orbit with an uncooperative target for an on-orbit inspection application[1]. The target spacecraft for this mission is The Planetary Society's LightSail, a 3U CubeSat that demonstrates solar sail technology[2]. LightSail is stowed inside of Prox-1 during launch and then deployed using a Poly Picosat Orbital Deployer (P-POD) device.

The mission begins with deployment of Prox-1 as a secondary payload from the SpaceX Falcon Heavy launch vehicle. Prox-1 then uses magnetic torque rods to detumble the spacecraft. Once angular rates have been nulled, a spacecraft checkout phase ensues where on-orbit functionality is established. Following Prox-1 checkout, LightSail is deployed. A period of time is allowed for the two spacecraft to drift apart, and orbit determination is performed on the ground to determine the trajectories of both vehicles.

After orbit determination has been completed, ground commands maneuver Prox-1 to within visual sensor range of LightSail using a cold gas thruster developed at The University of Texas at Austin[3]. At this point, automated operations begin. The Prox-1 GN&C subsystem acquires the target spacecraft in its thermal imager field of view for relative navigation and maneuvers the spacecraft into a formation flight in a leading or trailing orbit with respect to LightSail.

Entry into automated ProxOps is then commanded from the ground. During ProxOps Phase I, Prox-1 performs a rest-to-rest maneuver to move from the initial formation flight location to a point closer to LightSail and station-keeping capability is demonstrated for multiple orbits. Then, during ProxOps Phase II Prox-1 enters into a Natural Motion Circumnavigation (NMC) of LightSail using a relative elliptical orbit. During the ProxOps mission phases, Prox-1 performs all maneuvers without communication from the ground or cooperation from LightSail.

After Prox-1's primary mission is complete, it performs on-orbit inspection to image the deployment of LightSail's 32 m^2 solar sail. Finally, once all primary and secondary mission requirements are completed, Prox-1 is deorbited using a deployable drag device.

II. GUIDANCE, NAVIGATION, & CONTROL (GN&C) SUBSYSTEM OVERVIEW

The Guidance, Navigation, & Control (GN&C) subsystem for Prox-1 is made up of navigation components used to determine the satellite attitude and trajectory state, and guidance and control components to control the attitude and perform propulsive maneuvers.

II.I Navigation Components

Each of the navigation components is used to determine the state of the spacecraft based on inputs from various sensors. An inertial attitude determination filter (implemented as an extended Kalman filter) computes Prox-1's attitude quaternion and angular velocity vector using rate gyroscope, magnetometer, and sun sensor measurements. A microbolometer is used to capture infrared images of LightSail, which are fed into a set of image processing algorithms and a relative orbit determination filter to obtain relative position and velocity[1,4]. Accelerometer measurements are also taken onboard Prox-1, and a visual camera captures images for use on the ground. Global Positioning System (GPS) measurements are used for onboard inertial orbit determination.

II.II Guidance Components

Prox-1's guidance algorithms evaluate state information and determine a set of maneuvers to reach a desired state with respect to LightSail. Translational guidance algorithms utilize relative orbital elements to calculate the desired state and artificial potential functions for collision avoidance to stay away from a defined "keep-out-zone"[5]. A target acquisition algorithm determines appropriate slew maneuvers to center the target spacecraft in Prox-1's imaging instrument field of view (FOV). This capability is used for initial target acquisition and for recovery of the target if visual contact is lost. Also, a detumble algorithm is used to damp high angular velocities after launch vehicle separation, and a desaturation algorithm performs angular momentum management to prevent the control moment gyroscopes (CMGs) from saturating.

II.III Control Components

Finally, Prox-1 has control components to implement maneuvers commanded by the guidance algorithms using various actuators. A slew and tracking controller (STC) uses a CMG unit developed by Honeybee Robotics Spacecraft Mechanisms Corporation for primary attitude control and performs many functions, including slewing to the proper attitude before commanding a thrust maneuver. The STC can also use the CMGs to track LightSail so that it remains in Prox-1's imager FOV. A thruster controller commands the cold gas thruster to fire for a specified amount of time based on STC outputs. Finally, a torque rod controller is used for secondary attitude control and to implement commands from the desaturation and detumble algorithms using three single axis magnetic torque rods designed and manufactured in-house at Georgia Tech.

III. SIX DEGREE OF FREEDOM (6DOF) SIMULATION ENVIRONMENT

Most components of the GN&C subsystem are developed and tested within a six degree-of-freedom (6DOF) simulation environment using MATLAB and Simulink version 2012a. This "block diagram" environment allows real-time or accelerated simulation of GN&C component interactions with sensors and actuators and with other GN&C components.

III.I Simulink Features

Simulink provides many different block libraries to develop complicated simulation and control functions. One key Simulink block allows the developer to include embedded MATLAB code within the Simulink environment by connecting data ports that correspond to input and output variables of a MATLAB function. Simulink also provides Scope blocks, which allow the developer to view data plots changing in real time as a simulation runs. Another key Simulink tool called Stateflow allows graphical development and testing of decision logic using state machines. The Simulink environment also has the ability to connect and run simulations on external hardware using xPC Target. This capability allows for accelerated simulation speeds for complicated models such as the electrical dynamics model of the CMGs. xPC Target also provides the capability to connect to embedded processors for hardware-in-the-loop testing.

III.II 6DOF Environmental Framework

The framework for the 6DOF simulation environment includes space environment models for tracking time, Earth rotation, Sun, Earth, and Moon location, and Earth magnetic field dipole. The simulation uses perturbed two-body orbital mechanics and rigid body attitude dynamics for both Prox-1 and LigthSail. Environmental disturbances modeled include J2-J6 Earth oblateness effects, gravity gradient torque, aerodynamic drag, magnetic torque, solar radiation pressure, and 3rd body gravitational effects from the Sun and Moon. The states tracked by the simulation include attitude quaternions, inertial positions and velocities of both spacecraft, and relative position and velocity, all of which can be transformed between various reference frames. In addition, a MATLAB initialization script is run prior to starting the simulation to allow the user to set various constants and initial conditions for different simulation cases.

III.III Hardware Plant Models

The 6DOF simulation environment also includes plant models for various sensors and actuators on Prox-1. These models provide a realistic environment for development and testing of GN&C components by estimating the physical and electrical responses of sensor and actuator hardware based on test data and manufacturer specifications. In addition, a power production model has been developed to track the estimated amount of power produced by the solar panels during various ProxOps phases based on the position and attitude of Prox-1 relative to the Sun. Finally, an image generation tool produces simulated images of LightSail as seen by Prox-1 based on their relative positions and attitudes.

IV. GN&C COMPONENT DEVELOPMENT

Two main activities are completed within the 6DOF simulation environment: individual GN&C component development and testing of integrated GN&C components. Each GN&C component for Prox-1 is developed individually in a basic version of the simulation environment. Initially, simple equations are used in place of complex plant models, and often constant values are used to represent inputs from other GN&C components. Once basic functionality of the component is established, it is connected to the 6DOF simulation framework for further development and testing. Several examples of component development will be explained.

IV.I Slew and Tracking Controller (STC) Development

The Slew and Tracking Controller (STC) produces a torque command for the CMGs based on the current angular state (attitude, angular velocity, and angular acceleration) given a desired angular state. Early development of the STC involved using a basic standalone Simulink block diagram, shown in Figure I, with a constant input for desired attitude quaternion and zero vector inputs for desired angular velocity and

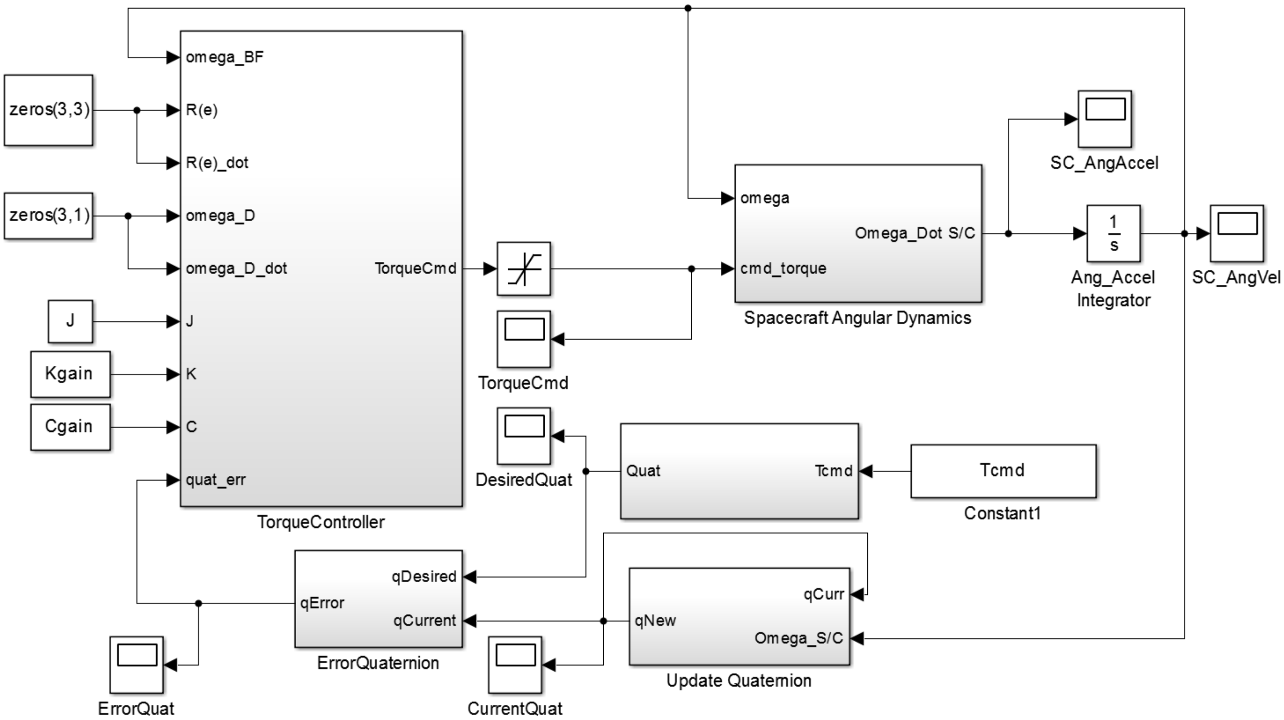

Figure I: Standalone Simulink block diagram for initial slew and tracking controller component development acceleration. This initial simulation included basic unperturbed angular kinematic equations and assumed perfect execution of the desired torque commands.

Once a basic torque control law was developed and tested, additional logic was written to transform other expected inputs, such as a desired direction to fire the thruster (Tcmd), into quaternion representation.

In subsequent stages of development, additional logic was added to allow tracking of the relative position vector to maintain LightSail within Prox-1's imager field of view (the "tracking" element of the slew and tracking controller). Also, upon integration of the environmental components and hardware plant models of the 6DOF simulation, a cost function was added to determine the optimal direction for solar panel pointing. This cost function allows Prox-1 to track LightSail as a primary pointing objective while simultaneously maintaining maximum power production as a secondary objective.

As additional complexities were added to the STC, it became apparent that simply developing additional logic using more Simulink blocks would lead to an unwieldy design solution. Stateflow provided an intuitive and elegant way to overcome this issue. A Stateflow chart was developed, as shown in Figure II which allowed the creation of various modes within the STC: Standby, Slew_to_Thrust, Wait_for_Thruster, and Track. Each mode is represented by a tan box in the chart, and blue arrows between the boxes represent transitions between the modes.

The Stateflow chart is a block within the Simulink diagram for the STC, as shown in Figure III. Inputs are fed into the Stateflow block based on the angular state and desired conditions, and outputs from the block are fed into the torque control law. Boolean variables are set within and outside the Stateflow chart to determine which mode is active. Each transition arrow contains a logical condition. When the condition connected to an outgoing arrow from the active mode is met, the chart will change its active mode.

Code within each mode determines what the STC should be doing, calls embedded MATLAB functions within the chart (shown in the top left corner as a group of silver blocks), and sets the output data that is sent to the torque control algorithm. Stateflow even animates the chart while the simulation is running, adding a bold blue outline around the active mode, which moves along transition arrows when a new mode is activated. In this way the developer can create, test, and debug the mode logic by running the simulation and watching the live animation along with Scope data.

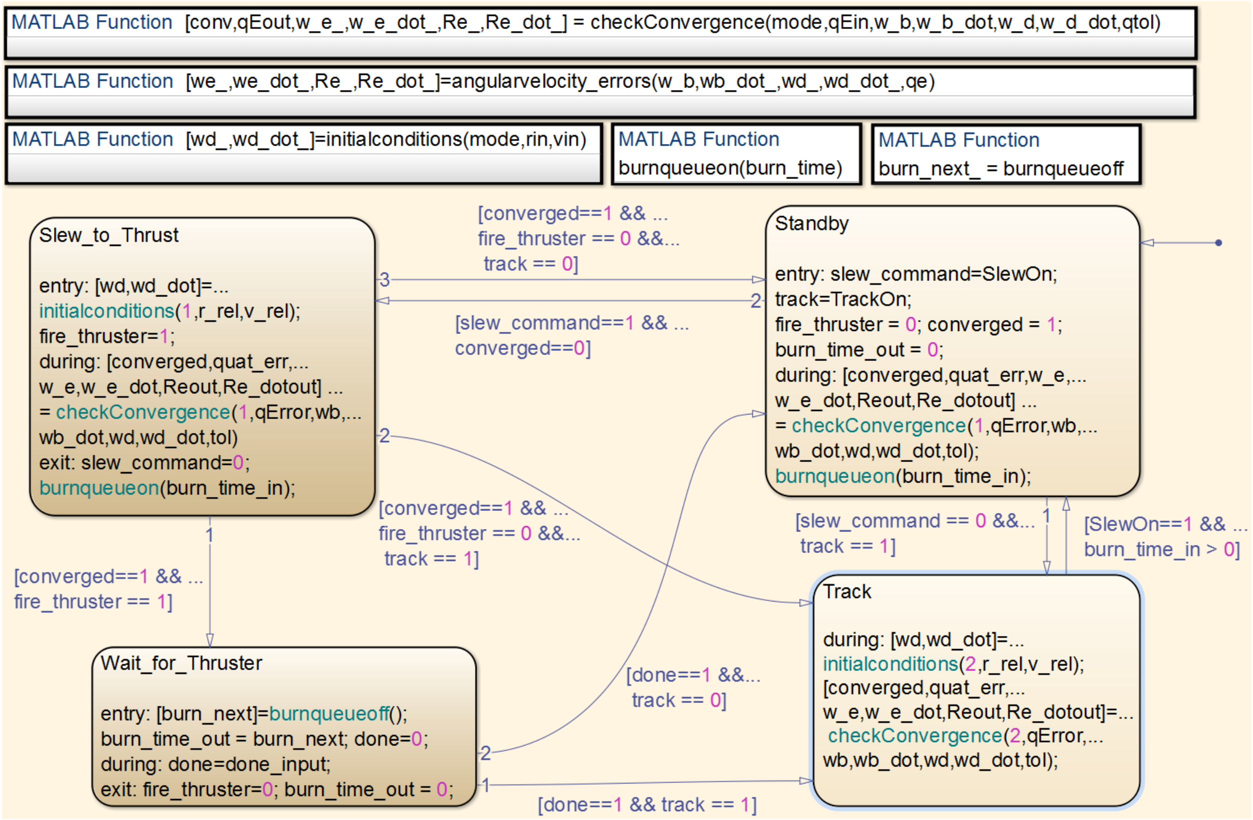

Figure II: Stateflow diagram for slew and tracking controller mode logic, including transition conditions

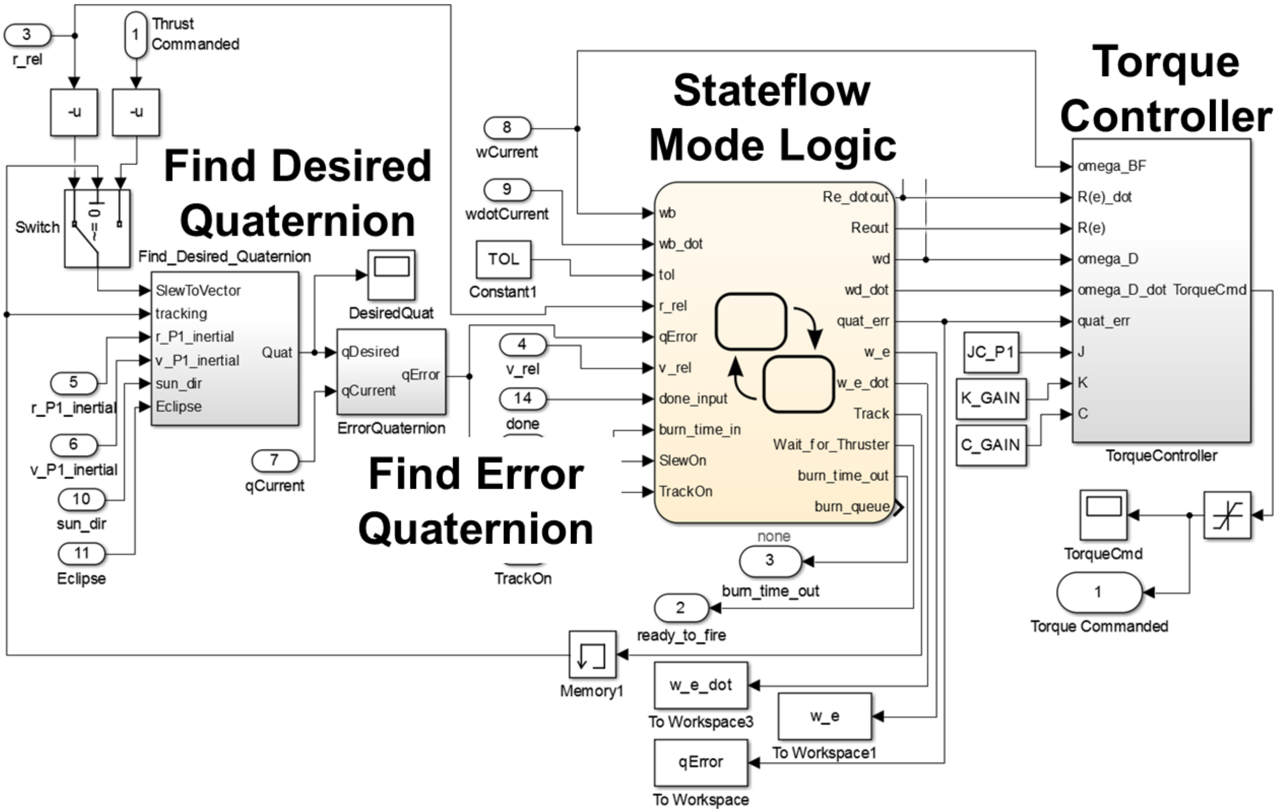

Figure III: Simulink diagram for the slew and tracking controller model. The Stateflow chart is a tan block in the middle, inputs are connected to blocks by arrows from the left, and outputs are connected from blocks to the right.

IV.II Thruster Controller (TC) Development

The thruster controller (TC) is a simple algorithm that controls the Prox-1 thruster state. The TC outputs a simple on/off command to the thruster based on a burn duration and execution time received from the STC. The inputs from STC are total burn time and a "ready-to-fire" Boolean variable.

Early versions of the simulation before the STC was added omitted the thruster controller and assumed that the thruster fired in the desired direction as soon as the guidance algorithms commanded a burn. With the STC added in the loop, it is necessary to delay the thruster firing to allow the STC to slew Prox-1 to the proper orientation before firing the thruster. The thruster controller performs this function by waiting to execute a burn command until the "ready-to-fire" flag is set to true. Future versions of TC will account for the flight thruster's physical response time.

Initially, the TC logic was developed in a standalone Stateflow diagram, shown in Figure IV. This simple

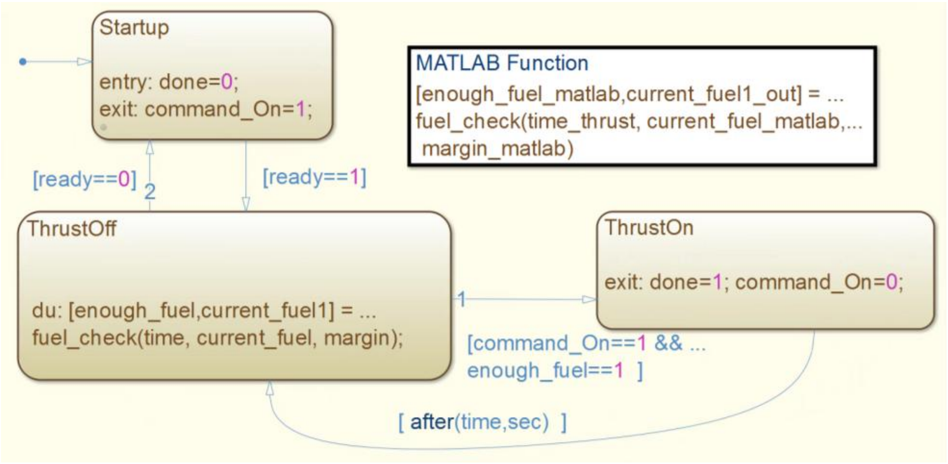

Figure IV: Standalone Stateflow chart for TC logic

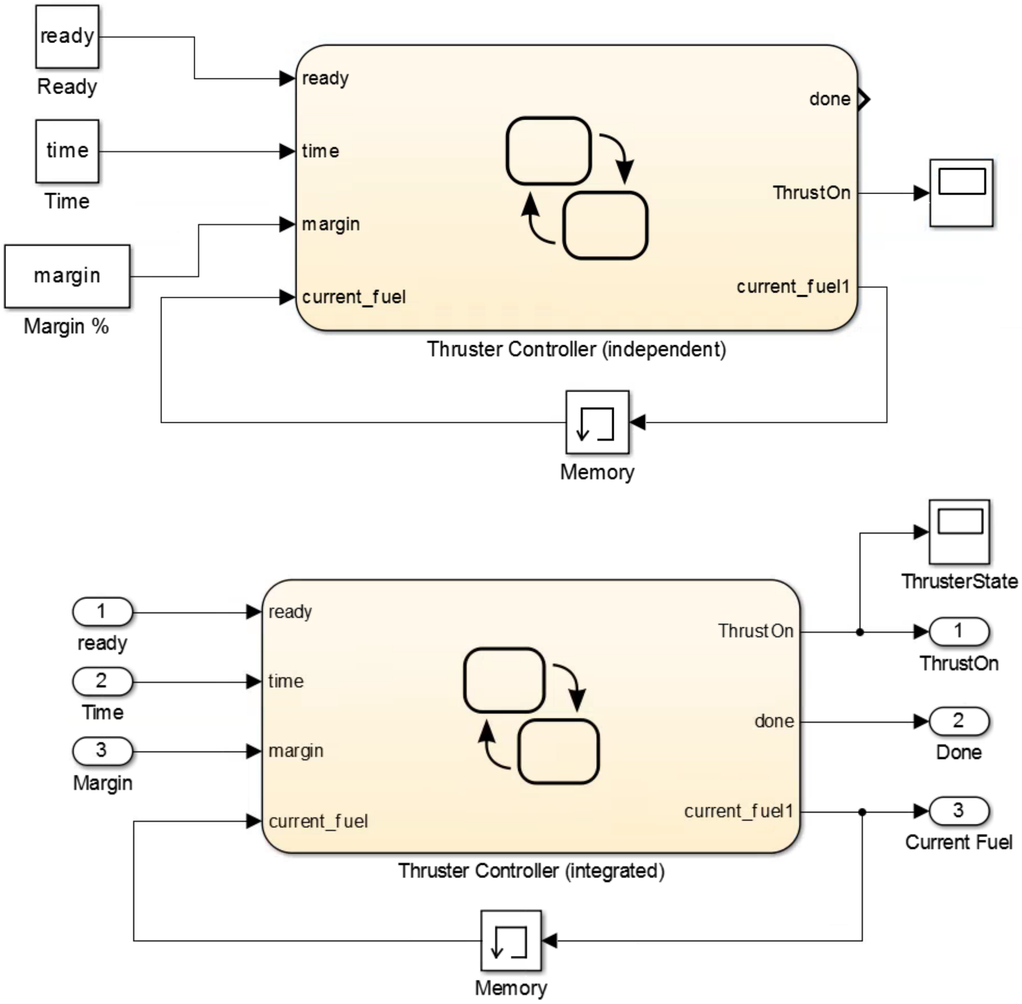

Figure V: Simulink diagrams for the Thruster Controller. Top is the standalone version, bottom is the integrated version.

diagram has three states: Standby, ThrustOff, and ThrustOn. The chart will remain in standby until the "ready" flag becomes true. At that time, a check for available fuel is run, and if sufficient fuel is available, the thruster is fired for the specified amount of time. After the firing is complete the "done" flag is set to true, and the state changes to ThrustOff until "ready" becomes false again. Initially, the standalone TC chart was fed by constants for the ready flag and burn time that were defined in the initialization file. It was soon modified to receive inputs from and send outputs to external models, as shown in Figure V. This allows integration with the STC and thruster plant models.

IV.III Detumble Controller (DTC) Development

The detumble controller (DTC) determines when Prox-1 has an angular velocity that is too high and calculates a magnetic moment to slow the rotation rate of the spacecraft using its torque rods. Originally, the DTC was developed in a simplified version of the 6DOF sim with no other controllers present. A high initial angular velocity condition was input to the sim and the controller was developed to damp that initial condition. DTC was then integrated into the 6DOF sim.

The DTC Stateflow logic diagram is shown in Figure VI. When the DTC is initialized, it enters the Activation_Of_controller state and several logical flags are initialized. Then the DTC transitions to Begin_Detumble, where a MATLAB function calculates the required magnetic dipole moment to damp the rotation, and a command to produce the moment is sent to the torque rods via the torque rod controller. DTC then transitions to Check_Angular_Velocity and a second MATLAB function determines if the angular velocity limits have been satisfied. If angular velocity is not within limits, then DTC returns to Begin_Detumble and calculates a new magnetic moment. If the angular velocity is within limits, a wait timer is started. As the timer advances, the angular velocity check continues. Once the timer expires, DTC moves into Standby_Detumble until the angular velocity limits are violated again.

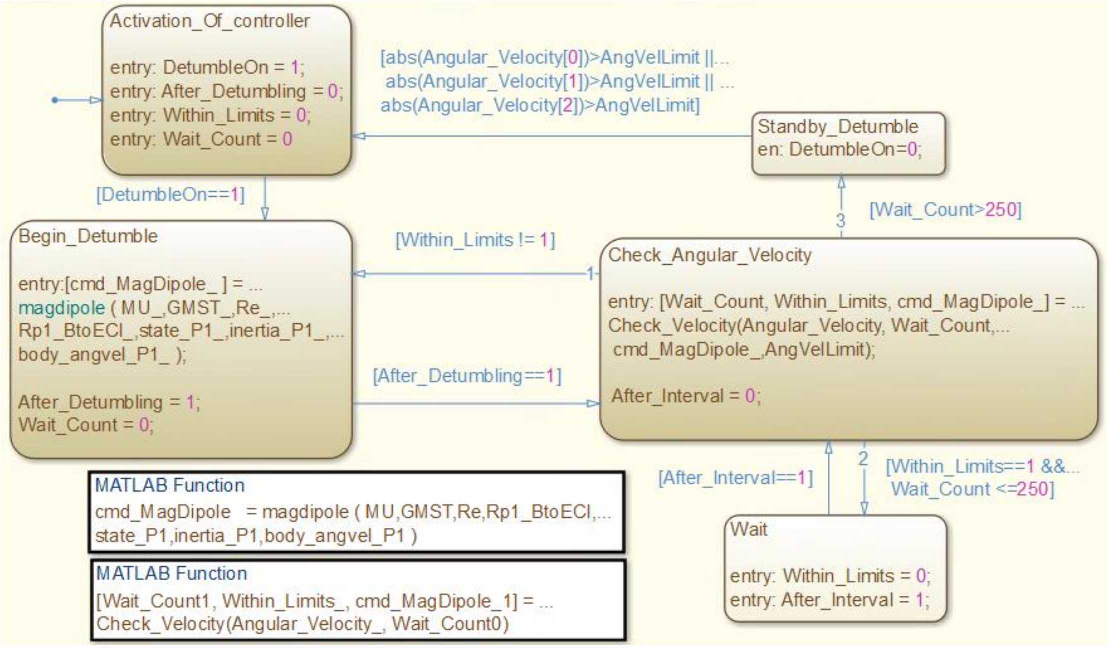

Figure VI: Stateflow logic diagram for the Detumble Controller to generate magnetic dipole command

V. GN&C SIMULATION INTEGRATION

Once an individual GN&C component is developed, it is ready to be integrated into the 6DOF Simulation environment along with other components. During this integration process further developments and modifications are made to each of the components to ensure that they work well together. Also, as components are linked together, test simulation cases are run to ensure that each component is performing as expected. An example follows to illustrate the process of component integration.

To achieve the current version of the "master simulation" in the 6DOF environment, the GN&C team begins with the standalone simulation used to test the guidance algorithms. This initial sim, shown in Figure VII, contains the foundation for the 6DOF sim: the Prox-1 and LightSail plant and environment models. The only GN&C component in this version of the sim is the Guidance block, which receives relative position and velocity inputs directly from the spacecraft plant (rather than from navigation) and outputs thrust commands directly to the thruster plant (assuming instantaneous slewing to the proper attitude). Although these simplifications do not completely represent the reality of operating Prox-1, they are sufficient to develop a functional guidance block. To account for effects of other components as they are added, the guidance block is continually tuned throughout the integration process.

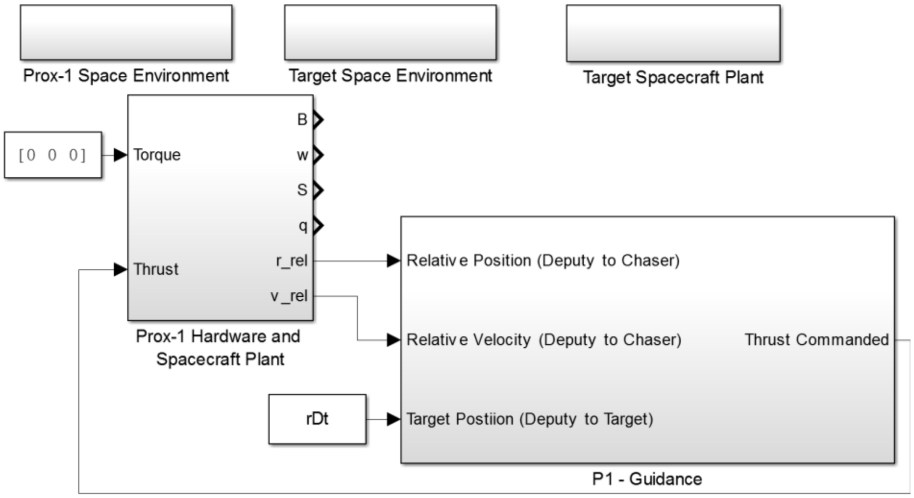

Figure VII: Standalone guidance testing simulation

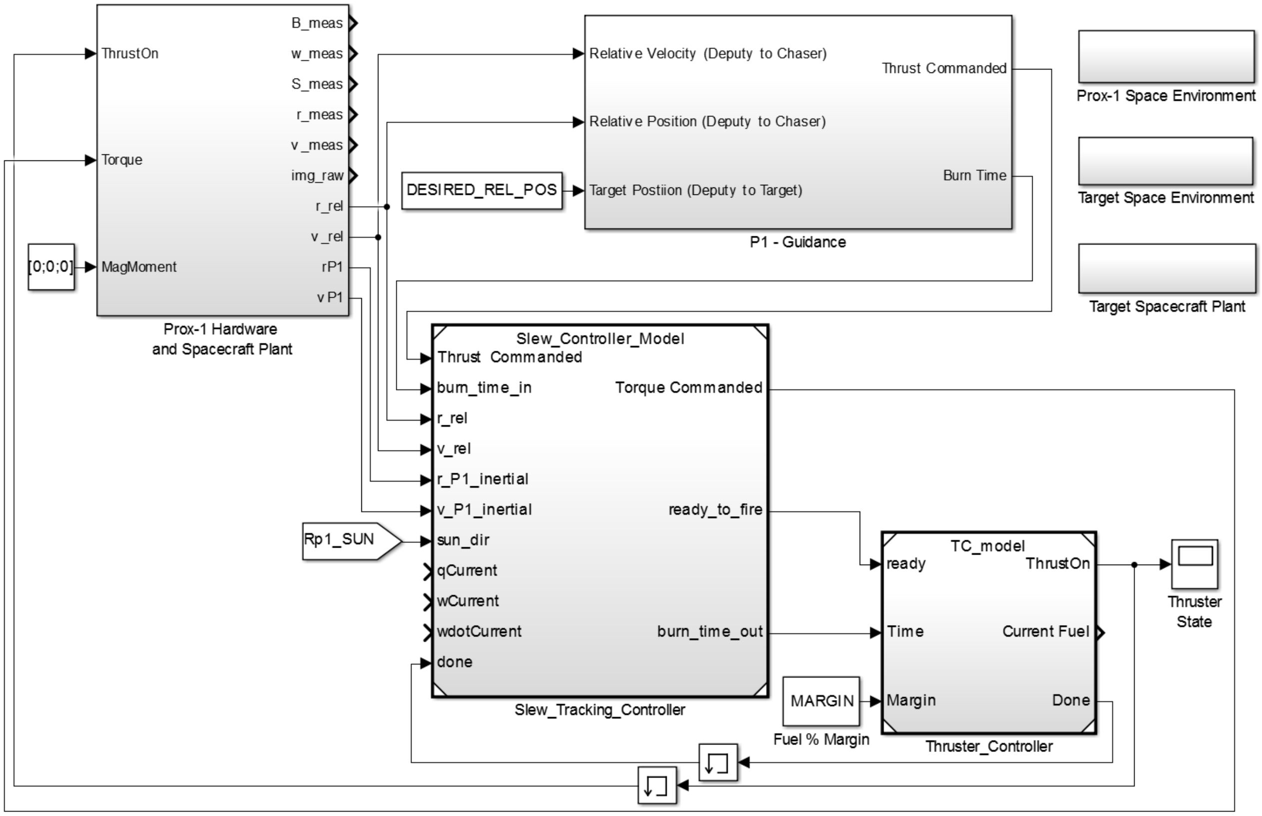

Figure VIII: Integrated sim including guidance, slew & tracking, and thruster controller components

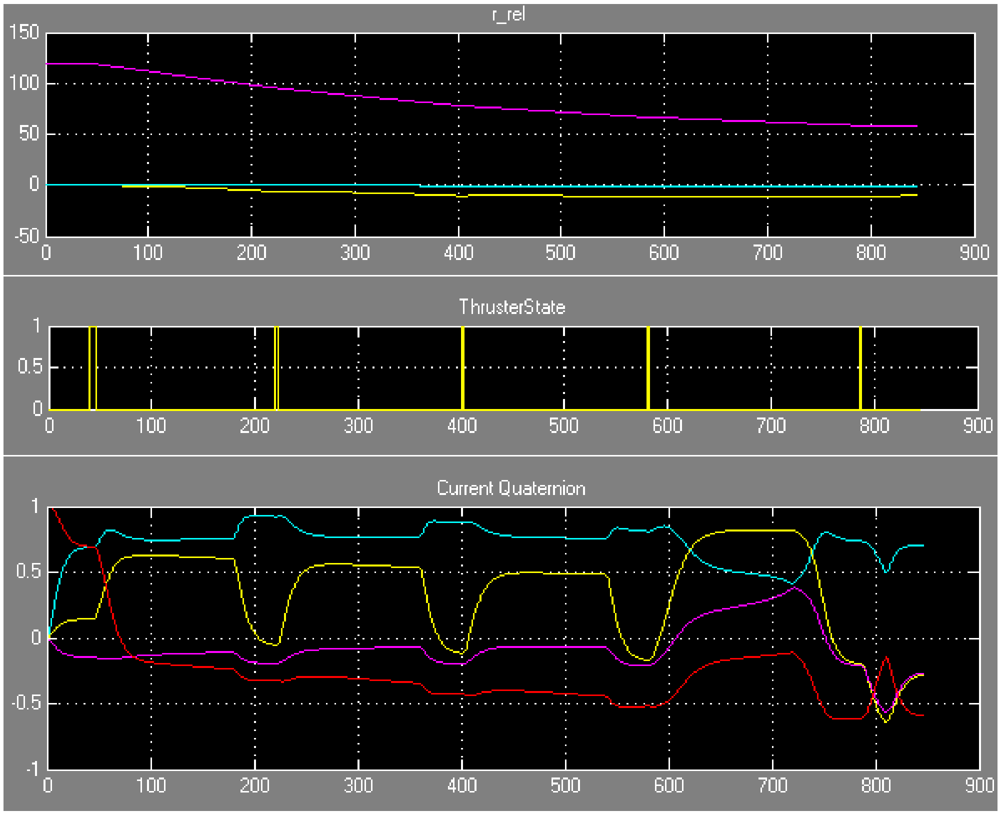

Figure IX: Integrated sim results for Guidance/STC/TC

The next step in the integration process involves bringing in the STC and TC, described in Sections IV.I and IV.II respectively. The resulting integrated sim is shown in Figure VIII. The outputs of the Guidance block are altered to include both a thrust command (unit vector direction) and a burn time associated with each burn command; both of these are sent to the STC. The STC also takes inputs of inertial and relative position and velocity directly from the spacecraft plant, as well as the sun vector directly from the space environment models, again bypassing the navigation filters.

Torque commands from the STC are output directly to the spacecraft plant model (bypassing the complex and computationally intensive CMG model). The “ready-to-fire” flag and burn time are sent from STC into TC, which sends thruster firing commands to the plant models and feeds back a “done” flag to STC that indicates when the firing is complete.

Once all of these connections are made, an integrated functional test of the simulation is run and scope outputs such as those in Figure IX are used to determine if the components are functioning properly. From top to bottom, the plots in Figure IX show the relative position between Prox-1 and LightSail in meters (magenta = along-track, cyan = cross-track, yellow = radial), the thruster state (1 = on, 0 = off), and Prox-1’s attitude quaternion (a unitless 4 element vector with values ranging between -1 and 1). The timescale for all of the scope outputs is in seconds.

The relative position starts with a 120 meter offset in the along-track direction, which closes to 50 meters as the guidance algorithms command thruster burns. The dips in the attitude quaternion plot represent slew maneuvers that point the thruster in the desired direction to execute burns commanded by the guidance algorithms. These burns are then shown as step functions on the thruster state plot. After many tweaks, the desired result is achieved, and the next component can be integrated. At each stage of integration, all of the components are verified to perform in the expected manner. If they do not, design changes are made and the integration cycle is iterated.

Finally, the DTC is added to the simulation and mode logic is developed in a Stateflow diagram, resulting in the master simulation shown in Figure X. The DTC block takes inputs from the spacecraft plant and environment models and outputs a magnetic moment directly to the torque rod plant model (bypassing the torque rod controller for now). The plant and environment models, boxed in red in Figures X and XII, are only used for simulation and will not be coded into flight software.

The tan block in the middle of the master simulation is the Stateflow GN&C mode logic, shown in Figure XI. As of this writing there are two states in the GN&C mode logic, Detumble and ProxOps; additional GN&C modes will need to be added as development of the integrated autonomous system continues. Detumble is the initially active state (to damp any high initial angular rates), and once angular rates have been damped

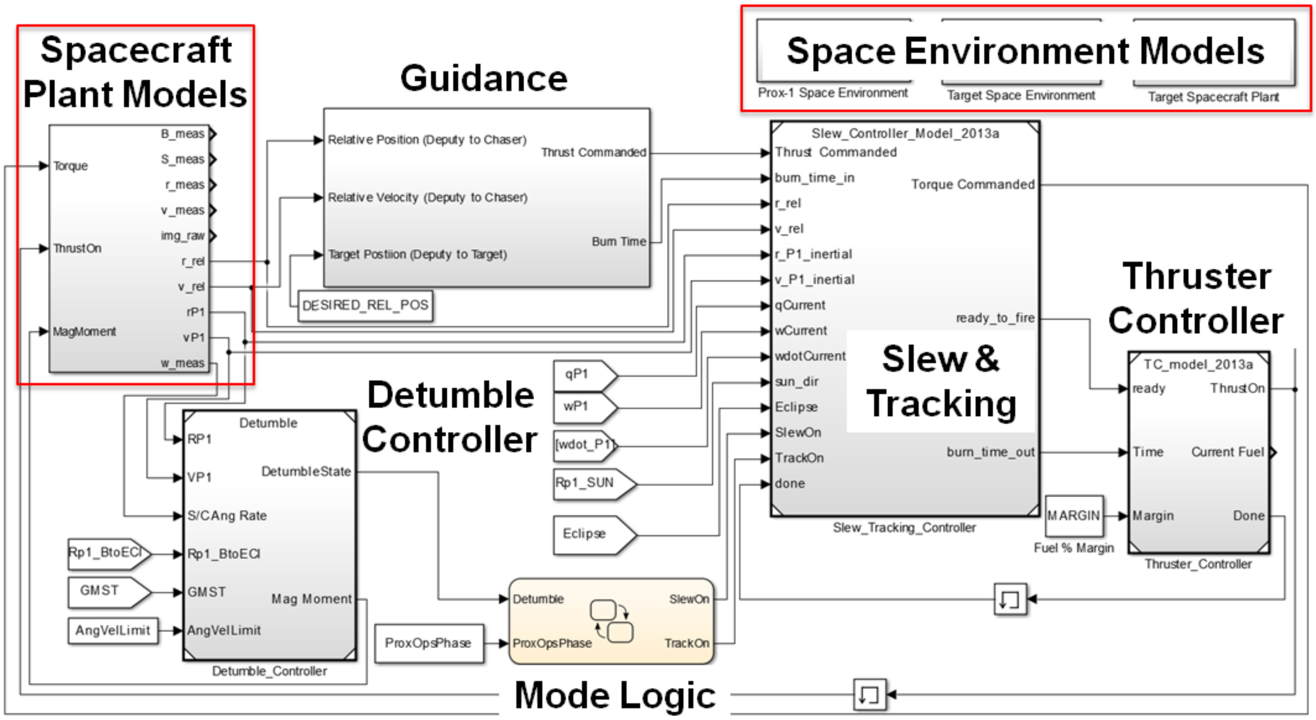

Figure X: Master simulation including guidance, slew & tracking, thrust control, detumble control, and mode logic

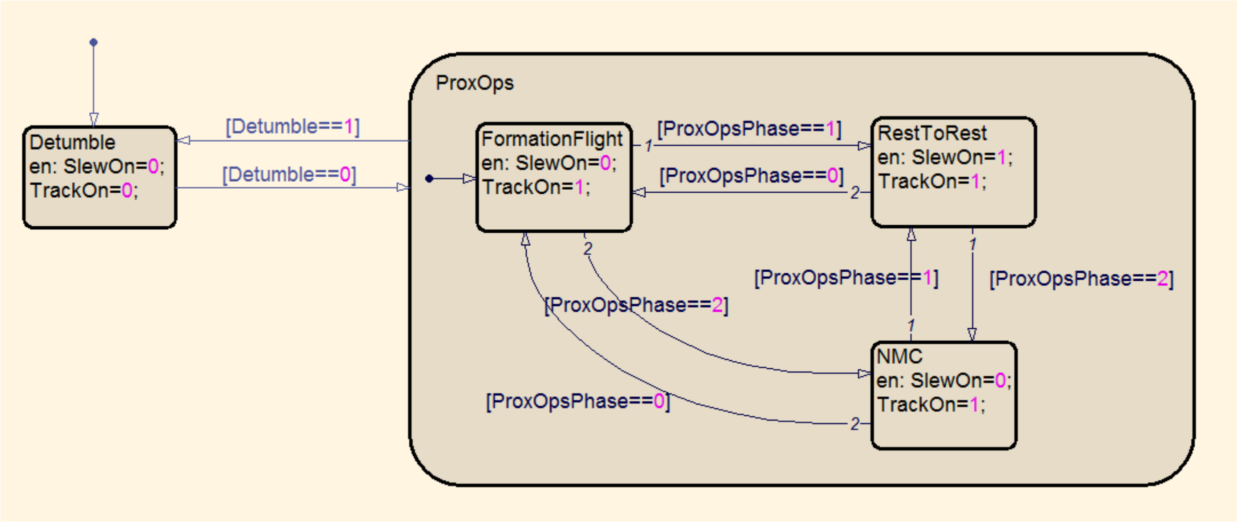

Figure XI: Stateflow mode logic for integrated master simulation

to within acceptable limits, the Detumble flag is set to zero and ProxOps mode is activated. ProxOps mode enables or disables both slew and tracking based on which ProxOps phase is active (phase 0 = FormationFlight, phase 1 = RestToRest, phase 2 = NMC). Both the Detumble and the ProxOpsPhase flags are inputs to the mode logic block, and SlewOn/TrackOn are outputs. In order for the STC to enter its SlewToThrust mode, the SlewOn flag must be set to 1. To enter the Track mode, the TrackOn flag must be set to 1. If both flags are set to 1, STC will perform commanded slew maneuvers then return to tracking LightSail. If neither flag is set to 1, STC will remain in Standby mode.

VI. CURRENT AND FUTURE WORK

As of this writing, the Prox-1 GN&C team is completing the development and integration of various components in Simulink, including the target acquisition and recovery controller, inertial attitude

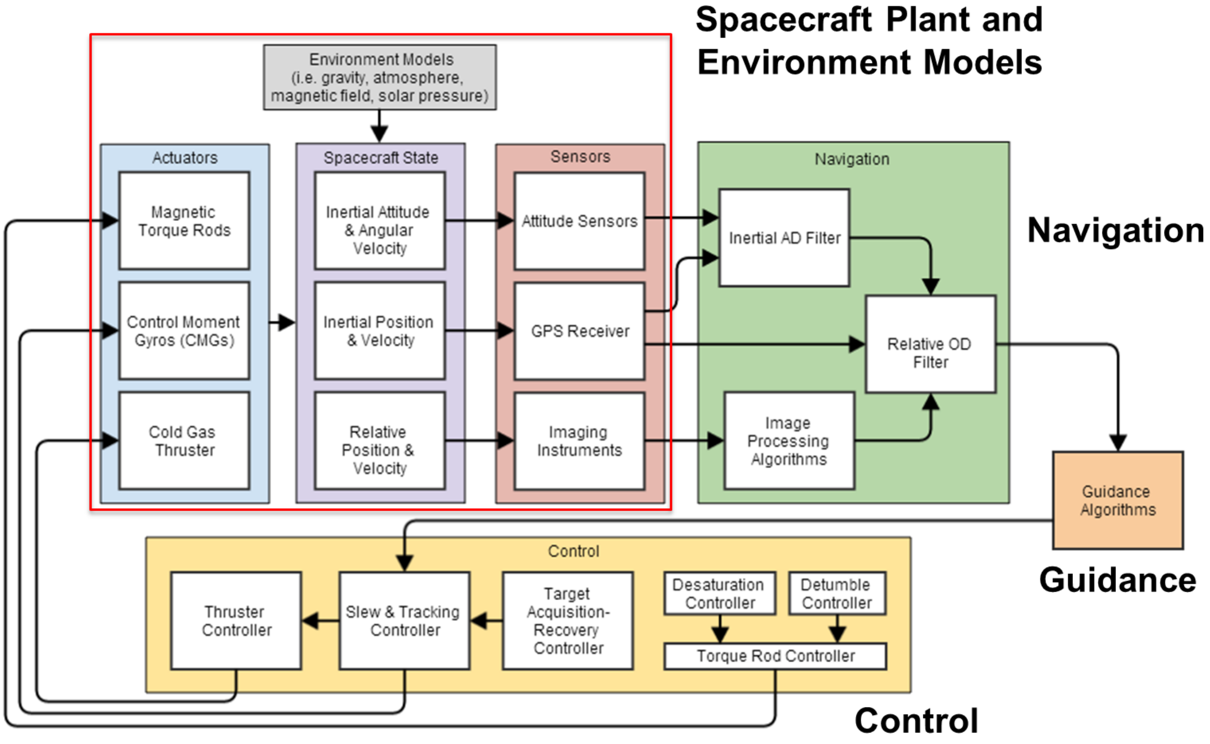

Figure XII: Flowchart showing final GN&C component interactions determination filter, desaturation controller, and torque rod controller. Updates and upgrades are also being made to other components. When all components are in their final form, the entire GN&C subsystem will be tuned and tested rigorously within Simulink. The flowchart in Figure XII shows how all of the components will eventually fit together in the final GN&C subsystem design.

The Simulink design of the GN&C algorithms will be autocoded into C/C++ using Simulink Coder for integration with flight software. Some modifications will be made to ensure the models are codable, including avoiding the use of incompatible MATLAB functions and converting most or all computations to single precision. This process of GN&C algorithm development and integration in Simulink and later autocoding into flight software has been pioneered by the Orion spacecraft team at NASA's Johnson Space Center in Houston, Texas[6].

Once all of the GN&C algorithms have been autocoded, the C/C++ versions will be tested on a PC using Simulink, xPC Target, and/or a flight software environment. These tests will be followed up with two or three levels of hardware-in-the-loop testing. First, a processor-in-the-loop test will determine that the algorithms are compatible with Prox-1's flight computer. Then, an integrated "Day in the Life" test will bring together all of Prox-1's hardware and software systems. Finally, an optional third test will place mock-ups of Prox-1 and LightSail on an air-bearing floor at NASA's Marshall Space Flight Center in Hunstville, Alabama and evaluate spacecraft system performance in a full operational scenario.

VII. CONCLUSION

A complete autonomous Guidance, Navigation, and Control (GN&C) subsystem is being developed for the Prox-1 student satellite project. This subsystem enables the spacecraft to perform inertial and relative navigation using various sensors, plan ProxOps maneuvers to approach and circumnavigate an uncooperative target, and execute those maneuvers using various actuators. Each GN&C component is developed within the MATLAB/Simulink 6DOF simulation environment and integrated together to create a complete design solution. By autocoding this design into C/C++ with Simulink Coder, integration and testing of GN&C algorithms are greatly simplified. Finally, hardware-in-the-loop testing will validate that the algorithms can be executed as intended on flight-like hardware.

VIII. ACKNOWLEDGEMENTS

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE-1148903. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

The authors wish to acknowledge the United States Air Force Office of Scientific Research and Air Force Research Laboratory (AFRL) University Nanosatellite Program for their funding support and technical guidance throughout the Prox-1 flight project lifecycle. Also the Prox-1 team thanks the United States Department of Defense Space Test Program (STP) for providing a ride-along slot for Prox-1 on the STP-2 launch. Josué Muñoz of AFRL is acknowledged for the original development of the 6DOF MATLAB/Simulink simulation environment.

The authors would also like to thank the following Prox-1 team members who also contributed to the work described herein: Sean Chait, Richard Zappulla, Manan Gandhi, Meet Patel, Aayush Sharma, Midhun Mathew, and Tony Nguyen, as well as all of those who have worked on the Prox-1 and R^3 projects at Georgia Tech.

---

[1] Chait, S. and Spencer, D., "Prox-1: Automated Trajectory Control for On-Orbit Inspection," AAS 14-066, *37th Annual AAS Guidance & Control Conference*, Breckenridge, Colorado, Jan-Feb 2014.

[2] C. Biddy and T. Svitek, "LightSail-1 Solar sail Design and Qualification," *41st Aerospace Mechanisms Symposium*, Pasadena, California, May 2012.

[3] Arestie, S, Lightsey, E.G, and Hudson, B, "Development of a Modular, Cold Gas Propulsion System for Small Satellite Applications," *Journal of Small Satellites*, Vol. 1, No. 2, October 2012, pp. 63-74.

[4] Walker, L. "Automated Proximity Operations Using Image-Based Relative Navigation," SSC12-VII-3, *26th Annual USU/AIAA Conference on Small Satellites*, Logan, Utah, August 2012.

[5] Spencer, D. "Automated Relative Proximity Operations Trajectory Control Using Relative Orbital Elements," Ph.D. Dissertation, Guggenheim School of Aerospace Engineering, Georgia Institute of Technology (pending).

[6] Jackson, M.C. and Henry, J.R., "Orion GN&C Model Based Development: Experience and Lessons Learned," AIAA-2012-5036, *AIAA Guidance, Navigation, and Control Conference*, Minneapolis, Minnesota, August 2012.