SPHERES OPERATIONS ABOARD THE ISS: MATURATION OF GN&C ALGORITHMS IN MICROGRAVITY

Simon Nolet*, Alvar Saenz-Otero†, David W. Miller‡, Amer Fejzic**

The Synchronized Position Hold Engage Reorient Experimental Satellites (SPHERES) Program, at the MIT Space Systems Laboratory, provides research scientists with a facility to incrementally and iteratively mature estimation, control, autonomy, and artificial intelligence algorithms to advance the field of Distributed Satellite Systems.

Operations aboard the ISS began in May 2006, with a total of five sessions completed by the end of 2006. These sessions achieved a wide range of successes including the demonstration of both formation flight and autonomous docking algorithms.

The docking algorithms were developed incrementally throughout the five tests sessions, making heavy use of the reconfiguration and modularity features of the SPHERES design. The architecture for the docking algorithms is based on the development of smaller simple modules that implement: two estimation algorithms based on extended Kalman filters; mixer functions to convert forces and torques to thruster on/off commands; glideslope, phase-plane, and PID-type controllers; and fault detection and isolation algorithms. The modules were tested piecewise and assembled together to create increasingly complex docking maneuvers. The tests culminated with the successful demonstration of two SPHERES satellites performing cooperative docking to fixed targets, demonstrating the capabilities of safe docking maneuvers, and performing the first microgravity docking to a tumbling target.

INTRODUCTION

The MIT Space Systems Laboratory developed the Synchronized Position Hold Engage and Reorient Experimental Satellites (SPHERES) program to incrementally mature algorithms for Distributed Satellite Systems (DSS) in a microgravity environment. SPHERES was specifically designed to help develop algorithms relevant to guidance, navigation, and control of spacecraft. Throughout the first six months of operations, the following types of algorithms have been implemented and demonstrated:

- Estimation: 6 degrees-of-freedom (DOF) Extended Kalman Filters.

---

* Ph.D. candidate, MIT Space Systems Laboratory, MIT Room 37-356, 70 Vassar St, Cambridge, MA 02139, snolet@mit.edu
† SPHERES Science Lead, MIT Space Systems Laboratory, MIT Room 37-381, 70 Vassar St, Cambridge, MA 02139, alvarso@mit.edu
‡ Director, MIT Space Systems Laboratory, Professor of Aeronautics and Astronautics, MIT Room 37-327, 70 Vassar St, Cambridge, MA 02139
** S.M. candidate, MIT Space Systems Laboratory, MIT Room 37-340, 70 Vassar St, Cambridge, MA 02139, afejzc@mit.edu
• Control: algorithms including PID-type, phase-plane, and glideslope control in 6-DOF.
• Navigation: docking algorithms (with cooperative and uncooperative targets) and satellite formations (linear translations and circular paths).

By operating inside the ISS, SPHERES exploits the microgravity environment to represent the dynamics of complex missions while operating in a risk-tolerant environment. Therefore, SPHERES allows scientists to push the algorithms to their limits in various realistic mission scenarios, learning about both their theoretical and physical limitations.

This paper first presents the motivation behind the SPHERES program and the design philosophy followed to design the facilities of the program. Next, a summary of all the research conducted aboard the ISS to date is presented. The second half of the paper presents an in-depth case study of the iterative maturation of autonomous docking algorithms, as a prime example of the use of the SPHERES program.

MOTIVATION

The motivation for SPHERES arises from the need to develop a wide range of GN&C algorithms for upcoming programs in an environment which allows their incremental maturation. Upcoming missions will require autonomous docking of spacecraft for re-supply (e.g. Orbital Express) and/or assembly (e.g. the Vision for Space Exploration inter-planetary stacks). Other missions will make use of separated spacecraft telescopes (e.g. ESA’s Darwin and NASA’s TPF) to use separated spacecraft interferometry to directly capture the light of distant planets.

While some of the upcoming missions are termed as “demonstration mission”, their cost prohibits scientists from actually testing algorithms on them; they are expected to succeed. Therefore, the algorithms must be tested repeatedly and thoroughly before the mission. The transition from theory to application has been shown to be a challenging process, but one that is necessary. Traditional methods use simulations, ground-based testbeds, neutral buoyancy tanks, and simulated micro-gravity in drop towers and reduced gravity airplanes (see Ref. 3 for a complete review). Yet, none of these facilities provide either the fidelity or time necessary to mature an algorithm as necessary. Therefore, a need exists to provide scientists with a development facility which closely simulates the operational environment without having the risks associated with the planned high-cost missions.

DESIGN PRINCIPLES

Based on substantial previous experience in the development of space technology maturation laboratories, the MIT Space Systems Laboratory created a design philosophy which was followed in the design of the SPHERES program. The design of the project was based on the need to “support the incremental maturation of a wide range of GN&C algorithms that encompass a field of study in a risk-tolerant and representative environment”. Based on these requirements, the following are the major principles which were used in the development of SPHERES:

1. Facilitate the iterative research process
2. Provide focussed modularity
3. Enable a field of study

Facilitating Iterative Research

It is essential for the scientific process that a hypothesis can be tested and modified as experiments are performed. To this purpose the design philosophy considers three main needs: 1) repeat an experiment to validate data 2) modify the experiment design to compare applications of the hypothesis and 3) to modify the hypothesis if necessary. A well developed facility must allow all three options. This must happen while maximizing the amount and flexibility of science time — that devoted to the formulation and modification of hypotheses and the analysis of data — and minimizing overhead time — that devoted to the implementation of the hypothesis in the actual testing environment and collecting the data after tests.
To facilitate the iterative research process, i.e. to minimize overhead time and maximize science time, SPHERES implements the following capabilities:

**Repetition and Reconfiguration** – SPHERES enables scientists to repeat experiments by providing easily replaceable consumables. Further, since all algorithms are programmed in software which can be uploaded to the ISS easily (and independently of any safety controls), scientists can modify individual tests or completely reshape their hypothesis. The availability of the crew to perform simple modifications of the hardware (such as the additions of a proof-mass) creates an extra level of reconfiguration.

**Multi-layered operations plan** – The plan allows scientists to perform research iterations in multiple locations with varying levels of fidelity, science time, and overhead time as presented in Table 1. The iteration period ranges from a few hours to a flexible 4-8 week schedule for operations aboard the ISS. These operations can be further enhanced by the portability of SPHERES, which enables tests at other remote locations such as Reduced Gravity Airplane and Flat Floor facilities.

<table>
<thead>
<tr>
<th>Location</th>
<th>Fidelity</th>
<th>Science Time</th>
<th>Overhead</th>
</tr>
</thead>
<tbody>
<tr>
<td>In-house utilizing the simulation</td>
<td>Low</td>
<td>Flexible</td>
<td>Minimal</td>
</tr>
<tr>
<td>Remotely by sending SW to MIT SSL</td>
<td>Medium</td>
<td>Flexible</td>
<td>Medium</td>
</tr>
<tr>
<td>Locally by visiting the MIT SSL</td>
<td>Medium</td>
<td>Semi-flexible</td>
<td>Minimal</td>
</tr>
<tr>
<td>Remotely by sending SW to the ISS</td>
<td>High</td>
<td>Semi-fixed</td>
<td>High</td>
</tr>
</tbody>
</table>

**Provide Focussed Modularity**

The basis of this principle is that experiments almost always contain basic elements that can support other similar experiments, therefore the design phase of a facility should identify these common elements. The call for focused modularity is to prevent a "do-everything" system which may deviate the facility from meeting its original goals. The generic equipment should be identified after the design of the original experiment; the original design should not be to create generic equipment.

Through its modular approach in software and hardware, the SPHERES facilities can be changed to better reflect the science needs of individual scientists. The primary characteristics of the SPHERES facility which enable reconfiguration and modularity are:

*Satellite bus* - each SPHERES satellite is a physical end-to-end simulation of a spacecraft bus.

*FLASH memory and bootloader* - allows the full software of a SPHERES satellite to be reprogrammed and stored in FLASH.

*Generic Operating System* - the SPHERES Core software\(^5\) implements a generic real-time operating system framework. The software provides scientist with modules that implement algorithms which may be outside of the scientists’ field of expertise (e.g. an autonomy scientist can use available estimation algorithms).

**Enable a Field of Study**

Past experience has demonstrated that to achieve technology maturation a field of study must be researched by several scientists. The combination of their knowledge achieves technology maturation. As such, this principle prescribes that:

- The study of multiple topics requires multiple experiments to be performed.
- Multiple investigators must work on individual topics to cover the whole field of study.
- The laboratory must facilitate bringing together the knowledge from the specific areas to mature understanding of the field of study.
SPHERES was designed to support the field of Distributed Satellite Systems. The MIT SSL identified the theoretical and applied areas of study identified in Table 2 as the ones necessary to cover the field of DSS. To support the field of study, SPHERES implements its operational flexibility through the features below.

<table>
<thead>
<tr>
<th>Area of Study</th>
<th>Current</th>
<th>Future</th>
<th>ISS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Theory</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Metrology</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Control</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Autonomy</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Artificial Intelligence</td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Communications</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Human/Machine Interfaces</td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td><strong>Application</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Docking: Basic</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Docking: Re-supply</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Docking: Assembly</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Docking: Reconfiguration</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Separated Formation Flight</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Tethered Systems</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sample Capture</td>
<td></td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>

*Guest Scientist Program* - An integral part of the SPHERES program, it combines operational and software features to support multiple scientists. It provides a simulation for in-house software development. Scientists have access to tests at the MIT SSL and then aboard the ISS. A flexible yet robust software environment creates the execution framework for the satellites, including the standard algorithm libraries described above.

*Expansion port* – It allows hardware expandability for new science payloads. To date, multiple expansion port items have been developed for ground-based tests. These include:

- Active docking mechanisms which enable research on re-supply between two satellites as well as assembly, and reconfiguration of two and up to five satellite models.
- Tether mechanisms which allow development of control algorithms for two+ satellite systems.
- Video-based docking mechanism to better simulate widely used docking systems.

Through the Guest Scientist Program’s invitation to participate to a wide range of scientists, and the ability to add new capabilities to the hardware, SPHERES successfully achieves its mission to help mature a field of study.

**PROGRAM OVERVIEW**

The SPHERES laboratory for Distributed Satellite Systems consists of a set of tools and hardware developed for use aboard the ISS and in ground-based tests. Three micro-satellites, a custom metrology system (based on ultrasound time-of-flight measurements), communications hardware (2 RF channels), consumables (tanks and batteries), and an astronaut interface are aboard the ISS. Figure 1 shows the SPHERES satellites being operated aboard the ISS and identifies the different elements of the facility. The ground-based setup consist of another set of micro-satellites, a research oriented GUI, and the guest scientist program to allow multiple researchers to use the facility.

The SPHERES satellites were designed to provide the best traceability to proposed formation flight missions by implementing all the features of a standard thruster-based satellite bus. The satellites have fully functional propulsion, guidance, communications, and power sub-systems. These enable the satellites to maneuver in 6-DOF, to communicate with each other and with the laptop control station, and to identify their position with respect to each other and to the experiment reference frame. The laptop control
station is used to collect and store data as well as to upload control algorithms to the satellites. Figure 2 shows a picture of an assembled SPHERES satellite and identifies its main features. Physical properties of the satellites are listed in Table 3.

<table>
<thead>
<tr>
<th>Table 3 SPHERES SATELLITE PROPERTIES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diameter</td>
</tr>
<tr>
<td>Mass (w/tank &amp; batteries)</td>
</tr>
<tr>
<td>Max linear acceleration</td>
</tr>
<tr>
<td>Max angular acceleration</td>
</tr>
<tr>
<td>Power consumption</td>
</tr>
<tr>
<td>Battery lifetime</td>
</tr>
</tbody>
</table>

MATURATION OF GN&C ALGORITHMS ABOARD THE ISS

Three SPHERES micro-satellites launched separately aboard ISS missions 21P (April 2006), ULF1.1 (July 2006), and 12A.1 (December 2006), together with the metrology system (partial on 21P, completed on ULF1.1) and consumables for at least one year of operations. SPHERES became the first autonomous free-flyer vehicle to operate inside the ISS on May 18th, 2006. A total of five test sessions were performed aboard the ISS during 2006: May 18 & 20, August 12 & 19, and November 11.

Test Sessions Objectives

The science objectives for the first five SPHERES test sessions are presented in Table 4. The first two test sessions operated with only one satellite and a limited metrology system (only one beacon). The first session was geared towards hardware checkout and initial demonstrations of basic maneuvers. The second test session began the iterative research process for the development of formation flight, docking, and fault detection algorithms that continued throughout. The third test session was the first to operate with two satellites, with initial demonstrations of formation flight. The fourth test session utilized the complete metrology system to demonstrate incrementally complex docking algorithms. The fifth test session utilized the earlier results of docking maneuvers and fault detection to demonstrate “safe docking” maneuvers and the first docking to a rotating target in microgravity.
Table 4
TEST SESSIONS 1 TO 5 OBJECTIVES

<table>
<thead>
<tr>
<th>Test Session</th>
<th>Objectives</th>
</tr>
</thead>
</table>
| 1 | • Primary Objective: Hardware checkout  
    • Science objectives:  
    • Demonstrate open-loop maneuvers: performance of different thruster mixers  
    • 3-DOF rotations: obtain initial performance of closed-loop (PID) rotations  
    • Autonomous docking: step-wise maneuvers towards autonomous docking |
| 2 | • FLASH fix: correct scaling factors memory corruption  
    • Attitude control demonstrations: complex rotations in 3D  
    • Fault detection and isolation (FDI): detect simulated failures in its thrusters  
    • Metrology: demonstrate ability to hold position  
    • Autonomous docking: perform a docking translation towards a fixed beacon |
| 3 | • Autonomous docking: continue docking demonstrations to fixed beacon  
    • Mass-Identification: on-line calculation of the mass and inertia of the satellite  
    • Two satellite initial tests: demonstrate multiple satellites working together |
| 4 | • Global metrology system checkout and data collection  
    • Complex 3D maneuvers: show “avoidance” trajectories  
    • Two satellite formation flight: demonstrate a leader/follower scenario  
    • Two satellite docking: perform initial tests of two satellites docking |
| 5 | • Mass-Identification: continue tests for on-line mass and inertia calculations  
    • Autonomous docking: perform cooperative and uncooperative docking; integrate failure simulations to demonstrate FDI capabilities and “safe docking”  
    • Formation flight: perform a peer-to-peer formation flight maneuver  
    • System reconfiguration: test algorithms that reconfigure the thruster mixers after satellites dock |

Results Summary

The first five test sessions have had substantial success. Table 5 presents a summary of the result. The first test session showed that the hardware operates correctly aboard the ISS and provided initial data on open-loop control. The second test session demonstrated closed-loop control capabilities, and successfully ran a complete set of fault detection, isolation, and recovery (FDIR) algorithms which provided important results. The third test session showed the ability to perform multi-satellite tests and demonstrated independent steps of docking algorithms. The fourth session fully utilized the global metrology system to demonstrate docking and formation flight algorithms. The fifth test session successfully pushed the limits of autonomous docking to demonstrate in microgravity docking to a tumbling target, safe docking maneuvers (where, in case of thruster failure, the satellite either avoids collision or successfully docks), and initial formation flight tests. Of course, there were several failures of the hardware, procedures, and algorithm implementations along the way. But, due to the risk-tolerant environment developed for SPHERES, these failures provided significant information to improve the facilities and algorithms without any being mission critical.

Iterative Research aboard the ISS

Through these five test sessions the SPHERES team accomplished several iterations on multiple areas under study. The metrology estimator has seen important improvements to its base algorithms, as well as the filtering of noisy data and the use of FDI algorithms to maintain estimator convergence. Formation flight algorithms have been demonstrated with different implementations. Docking algorithms have gone through the most iterations and incremental process so far. The docking algorithms were demonstrated step wise at first, and slowly put together until a complex demonstration was achieved. These iterations are further described in Table 6. The next section presents an in-depth study of the incremental and iterative development of autonomous docking algorithms using SPHERES aboard the ISS.
### Table 5
TEST SESSIONS 1 TO 5 RESULTS SUMMARY

<table>
<thead>
<tr>
<th>TS</th>
<th>Objective</th>
<th>Tests</th>
<th>Result</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Hardware Checkout</td>
<td>2</td>
<td>Success</td>
<td>The checkout included the power-on and bootload capabilities, followed by a prescribed sequence of firings.</td>
</tr>
<tr>
<td></td>
<td>Open-Loop</td>
<td>5</td>
<td>Success</td>
<td>Data provided methods to improve the thruster management module.</td>
</tr>
<tr>
<td></td>
<td>Closed-Loop</td>
<td>8</td>
<td>Failure</td>
<td>Corruption in the scaling factors of the IMU stored in the FLASH of the satellite resulted in failed closed-loop tests.</td>
</tr>
<tr>
<td></td>
<td>FLASH Fix</td>
<td>8</td>
<td>Success</td>
<td>The scaling factors were fixed.</td>
</tr>
<tr>
<td></td>
<td>Attitude Control</td>
<td>4</td>
<td>Success</td>
<td>3D closed-loop rotations completed within the expected performance parameters.</td>
</tr>
<tr>
<td>2</td>
<td>FDI</td>
<td>8</td>
<td>Success</td>
<td>Simulated thruster failures were detected in all the tests.</td>
</tr>
<tr>
<td></td>
<td>Metrology</td>
<td>4</td>
<td>Success</td>
<td>The satellite was able to hold position w.r.t. the ISS frame.</td>
</tr>
<tr>
<td></td>
<td>Docking</td>
<td>6</td>
<td>Partial</td>
<td>Initial tests provided data to improve the estimation and control gains. One test followed the path to dock to the beacon, but contact occurred at too high velocity.</td>
</tr>
<tr>
<td></td>
<td>Docking to a beacon</td>
<td>2</td>
<td>Success</td>
<td>The satellite always had contact in close proximity with the beacon; unexpected motion close to the beacon was observed, leading to improvements in the estimation.</td>
</tr>
<tr>
<td></td>
<td>Mass-ID</td>
<td>11</td>
<td>Partial</td>
<td>All tests were run, but the propulsion system experienced a high-pressure anomaly which prevented some thrusters from opening. Therefore, some tests failed.</td>
</tr>
<tr>
<td>3</td>
<td>Two Satellites:</td>
<td>4</td>
<td>Success</td>
<td>Performed 3-DOF rotations in formation to simulate two satellites always pointing to the same target.</td>
</tr>
<tr>
<td></td>
<td>Formation Flight</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Docking</td>
<td>1</td>
<td>Failure</td>
<td>Because the target satellite was in free drift, the docking satellite estimator was unable to converge.</td>
</tr>
<tr>
<td></td>
<td>Two Satellites:</td>
<td>3</td>
<td>Success</td>
<td>Performed a complex 6-DOF leader/follower formation maneuver.</td>
</tr>
<tr>
<td></td>
<td>Formation Flight</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Docking</td>
<td>4</td>
<td>Success</td>
<td>All attempts traversed and had contact between the satellites (successful pointing). Demonstrated plume impingement when using a glideslope algorithm.</td>
</tr>
<tr>
<td></td>
<td>Docking</td>
<td>7</td>
<td>Failed</td>
<td>These required the estimator to converge. Will be tested in the future.</td>
</tr>
<tr>
<td></td>
<td>Mass-ID</td>
<td>12</td>
<td>Partial</td>
<td>All the necessary data was collected, but due to software errors in the GUI the estimator did not converge.</td>
</tr>
<tr>
<td></td>
<td>Docking</td>
<td>3</td>
<td>Success</td>
<td>Demonstrated docking to a fixed target, safe docking, and docking to a tumbling target.</td>
</tr>
<tr>
<td></td>
<td>Formation Flight</td>
<td>4</td>
<td>Success</td>
<td>The satellites followed a circular path while using communications to synchronize each other. The synchronization did not meet all performance expectation.</td>
</tr>
<tr>
<td></td>
<td>Reconfiguration</td>
<td></td>
<td>Untested</td>
<td>There was not enough time to run these tests.</td>
</tr>
</tbody>
</table>
Table 6
RESEARCH ITERATIONS FROM TEST SESSIONS 1 TO 5

<table>
<thead>
<tr>
<th>Area</th>
<th>TS</th>
<th>Iterations</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metrology</td>
<td>1,2,3,4,5</td>
<td>4</td>
<td>The metrology system began with a limited set of beacons, which resulted in the development of innovative filters. The lessons from that development and the capture of data in TS4 resulted in no need for further development starting with TS5.</td>
</tr>
<tr>
<td>Docking</td>
<td>2,3,4,5</td>
<td>4</td>
<td>The basic maneuvers for docking began development during the second test session. As each session demonstrated individual parts of a full docking mission, these parts were put together to achieve the final demonstrations of TS5. More iterations will follow in future sessions.</td>
</tr>
<tr>
<td>Formation Flight</td>
<td>3,4,5</td>
<td>2</td>
<td>Starting with attitude-only formation flight, the latest iteration of demonstrations aboard the ISS shows the initial maneuvers that are essential for the missions that motivated SPHERES. Further iterations will occur.</td>
</tr>
</tbody>
</table>

CASE STUDY: AUTONOMOUS DOCKING

There exists a clear need to develop a capability to perform routine autonomous docking, i.e. without human intervention or even supervision, to a potentially uncontrolled target. Therefore many hardware and software technologies are currently under development to provide that capability. This section introduces the development and implementation of a GN&C architecture that enabled the first on-orbit autonomous docking to a tumbling target, using the SPHERES facility. This section first defines the problem and approach; it then describes the GN&C architecture and algorithms used. Emphasis is placed on modularity, which significantly facilitated algorithm integration and testing.

Problem Definition and Approach

The objective is to develop and implement a GN&C architecture that will enable safe and fuel-efficient docking of a thruster-based satellite with an uncontrollable target, while maintaining quick response to contingencies. The main challenges are the risk associated with the low recovery time in case of contingencies and the limited amount of fuel onboard. This is especially true when the target is tumbling and has appendages that may interfere with the trajectory of the chaser at some point in time. The problem is scoped for the terminal phase of the docking maneuver: when the two satellites are in close proximity and orbital mechanics effects are minimal, but where obstacle avoidance and issues such as plume impingement must be considered.

The problem is approached at the systems level by integrating existing algorithms spanning multiple areas of research (estimation, control, FDIR, planning), as opposed to developing a specific control algorithm. A series of algorithms is identified, implemented and demonstrated on hardware in a relevant environment.

A GN&C Architecture for Autonomous Docking

In Ref.12, Fehse provides an overview of his concept for a traditional onboard rendezvous (RV) control system based on a typical GN&C architecture. Figure 3 shows an extension of the GN&C architecture (levels of authority) proposed by Fehse. The extended architecture accounts for the idea that communications with human operators may be intermittent or non-existent (illustrated by dashed lines), preventing real-time human intervention. Therefore, systems architects now tend to integrate FDIR capability at all levels. In this extended model, the onboard computer is granted the authority through

---

* A target which lost control authority around at least one of its body axis
the main FDIR module to trigger a collision avoidance maneuver (CAM) if a problem is detected. Further, a solver module which enables on-line planning is monitored by the FDIR to prevent infeasible solutions or other failures. This model requires the chasing satellite to be aware of the health status of the target satellite to enable the MVM module to determine the correct maneuver sequencing. Therefore, remote sensing and/or communications between the satellites are needed.

Figure 3: A GN&C architecture for autonomous docking.

Modularity

One of the key characteristics of SPHERES is its modularity in the development of algorithms. Multiple modules which correspond to each block within the autonomous onboard RV-control shown in Figure 3 were developed. Each software module contains an algorithm to accomplish a simple task. Modularity can provide great benefits including understandability, maintainability, protection and reusability. For example, the architecture developed for autonomous docking was used almost as is for formation flight experiments. Modularity also facilitates cooperation between engineers, even when located at separate sites. It allows the capability to evolve and expand as new technologies become available by replacing an obsolete module rather than redesigning the whole system. Standardization of the interface between the different modules facilitates integration and reusability. The modules are at different levels of implementation. Linear Programming (LP) and Mixed-Integer Linear Programming (MILP) algorithms have been used off-line to determine the paths for safe docking, but on-line solver algorithms which will enable on-line path planning are still under development. The navigation, control, MVM, and FDIR algorithms used to populate the GN&C architecture for autonomous docking are introduced below.

Navigation Software Modules

Two main navigation software modules are used to interpret navigation sensor information: a “global estimator” and a “single-beacon estimator”. Both systems utilize the time-of-flight information from the ultrasonic sensors combined with gyroscope data.
Global Estimator – The global estimator is traceable to a standard GPS-INS sensor suite since it uses time-of-flight information from multiple sources. The system uses an infrared ping (sent out by one satellite) to synchronize the time between all satellites and the beacons. After the ping, the beacons transmit an ultrasound pulse in sequence (20ms apart). All the satellites listen for the ultrasound pulses through their 24 receivers, located four per face, to create a time-of-flight matrix which is used by the global estimator. The time-of-flight data and gyroscope measurements are processed using an extended Kalman filter (EKF) to compute the 6-DOF state. Other spacecraft including ETS-VII, DART, and the former Mini AERCam used a similar approach to compute their position and attitude. The SPHERES global estimator provides position results on the order of a few millimeters when used in a pre-defined testing environment such as the US Laboratory of the ISS.

Single-Beacon Estimator – Because of the availability of only one beacon onboard the ISS for the first three test sessions, a second navigation software module named the single-beacon estimator had to be developed. The SPHERES team developed an ingenious solution to provide the satellite with 6-DOF state estimation using a single ultrasound beacon and three gyroscopes. The attitude of the satellite is solely determined by integrating forward the angular rates provided by the gyroscopes, while the position and the velocity are computed by repetitively determining the location of the external beacon in the satellite’s body frame using the time-of-flight data. Assuming that the attitude of the satellite is maintained fixed during the process, regular sampling of the beacon can provide displacement information by subtracting the updated beacon location to the one initially recorded. However, the accuracy of this method is tied to the drift rate of the gyroscope, since when using the single-beacon estimator the angular deviation caused by gyroscope drift cannot be distinguished from tangential errors. Experimentation showed that integrating the gyroscope data without any external attitude correction led to an attitude drift rate of up to 5 degrees/minute. This drift rate was acceptable for SPHERES tests, which last on the order of a few minutes.

Control Software Modules

A series of control software modules command the onboard thrusters based on closed-loop control techniques.

Mixers - Since the satellites thrusters are single-level on/off type, the force and torque commands requested by the control modules are converted to thruster on/off times using a pulse-width modulator. The mixer matrix uses conservation of thrust impulse [Ns] to determine the on/off time for an equivalent force or torque based on the known fixed thrust. The resulting thruster pulse is centered in the control period. Figure 4 illustrates this process.

Glideslope – A glideslope docking module was developed to control the approach of the chaser to the target. This algorithm has been widely used (Apollo, Shuttle) for trajectory planning in real-time. A glideslope is defined as a straight path from the current location of the chaser to its intended destination. The approach velocity $\dot{\rho}$ is commanded to decrease linearly with the distance-to-go $\rho$, and a safe arrival commanded velocity $\dot{\rho}_f$ is prescribed. The maneuver has to occur in a period $T$ using $N$ thruster firings. The outputs of this controller command a $\Delta\dot{\rho}$ (effectively a force) approximately equally separated in
time. A typical resulting trajectory is shown in Figure 5. The algorithm is not fuel optimal, nor does it account for obstacle avoidance or plume impingement constraints. But it has the property of reducing the nominal $\Delta \dot{\rho}$ conveyed at each burn toward the end of the maneuver, which in some cases can help to reduce the effects of plume impingement.

![Figure 5 Typical approach velocity profile computed by a glideslope algorithm.](image)

**PID Controllers** – Standard PID-type controllers\textsuperscript{1} are primarily used to hold a reference attitude or position. The modules output force and torque commands proportional to state errors. The attitude angular error $\theta_e$ around the Euler axis defined by the unit vector $[n_x, n_y, n_z]$ is expressed in terms of a quaternion as

$$ q = [q_1, q_2, q_3, q_4] = \left[ n_x \sin\left(\frac{\theta}{2}\right), n_y \sin\left(\frac{\theta}{2}\right), n_z \sin\left(\frac{\theta}{2}\right), \cos\left(\frac{\theta}{2}\right) \right] $$

To control the attitude of the satellite, the algorithm uses the quaternion measurements computed by the estimator to approximate angular errors. Each axis is controlled independently. The attitude control law is of the form:

$$ T_i = 2 \cdot K_i \cdot sgn(q_i) \cdot q_i + 2 \cdot K_i \cdot \int \{sgn(q_i) \cdot q_i\} dt + K_i \cdot \omega_i $$

where $sgn(\cdot)$ refers to the signum function.

**Mission & Vehicle Management**

Numerous methodologies have been developed to enable various levels of autonomy onboard a vehicle, allowing for decision making by an onboard computer dictating the pursuit of the mission.\textsuperscript{14} The management software module defines a GN&C mode as a phase of the mission. Each mode is supported by a group of controllers with a pre-determined set of gains or parameters until the fulfillment of an objective. The management module periodically checks the objective until it is met, at which point the mode sequence proceeds. Mode sequencing is currently prescribed a-priori in most cases, except when a CAM is autonomously triggered and the satellite goes to a safe mode. This can occur at any point in the experiment in case a problem is detected and the satellite cannot timely recover.

**Fault Detection, Isolation and Recovery**

A robust GN&C architecture has to include FDIR capability at all levels, from the hardware level to the mission & vehicle management level.\textsuperscript{13,14} Initial FDIR implementation modules exist on SPHERES; more modules are under development. Sensor and actuator FDIR, specifically a thruster FDI technique using INS data\textsuperscript{23}, was independently tested on-orbit using SPHERES.
Following a series of docking experiments performed in the ISS where measurement errors caused the estimator to lose convergence, an ultrasound measurement error detection system was embedded in the EKF used by the global estimator.\textsuperscript{17} Like other Kalman filter techniques, an EKF has built-in fault detection capability through the filter innovation computed during the state update phase. The result of this comparison can be used as a verification if the measurements are coherent with the state estimates. Non-coherence is a sign of a problem. Fault detection occurs when the filter innovation is above a given threshold (determined through simulations that used ISS data). Although the filter innovation provides fault detection capability, it does not necessarily provide fault isolation capability unless the failure has a clear innovation signature. Experimentation on SPHERES has shown that temporary measurement errors produced by the ultrasound system do have a clear signature observed through the filter innovation, therefore validating this technique.\textsuperscript{17}

**ON-ORBIT AUTONOMOUS DOCKING EXPERIMENTAL RESULTS**

This section presents results obtained during the third to fifth sessions, when complete docking maneuvers were tested aboard the ISS. Each incremental experiment builds on knowledge and experience acquired during previous ones. The experiments are presented in chronological order of occurrence, starting from an initial validation experiment using only one satellite and a beacon attached to the wall as a target, and culminating with the first successful microgravity autonomous docking with a tumbling target.

**Docking to a Fixed Beacon**

These experiments used the one beacon available, mounted on the wall, as the target. The single-beacon estimator provided the state estimates necessary to navigate in the test volume. The glideslope docking module controlled the approach along the docking axis, while a PD control module maintained the tangential alignment with the beacon. A non-linear PD-type control module was used to maintain the attitude of the satellite using the control law described in Eq. (1) with an integral gain set to zero.

Figure 6 shows the results for one of these experiments performed on August 12, 2006. The left plot shows the displacement of the chaser satellite as it moves towards the target, with respect to the initial position of the chaser. The target began approximately 1.2 meters away from the –X face of the chaser, resulting in the trajectory shown by the continuous line with a negative slope. The right plot shows the misalignment on the plane perpendicular to the docking axis, which is expected to be around zero. The dotted data indicates state estimates collected during the initialization phase of the experiment, while the solid line data indicates state estimates during the docking approach. In the initialization phase, the satellite induced a random tumble to simulate potential high tip-off rates after deployment. The satellite damped out all rotations using its gyroscopes and initialized its single-beacon estimator to acquire a position and an attitude solution. After successful acquisition, it pointed its –Y face toward the beacon between time 20 and 36 seconds. The position of the beacon in the satellite’s body coordinate frame at time 36 seconds became the reference position for the rest of the experiment. The satellite then initialized its approach toward the beacon. At time 85 seconds, infrared (IR) noise from an unknown external device started to corrupt the ultrasonic data transmitted to the single-beacon estimator and forced the thrusters to remain closed because of an internal safety feature. This caused the satellite to temporarily get off track until time 142 seconds. Nevertheless, it regained control after the IR noise disappeared and eventually made contact with the beacon at time 180 seconds.
A close analysis of the results proved that prior to the appearance of IR noise in the test volume, the actual $\Delta \dot{\phi}$ produced when the glideslope docking module commanded a velocity correction was within 5 millimeters per second of the desired value 70% of the time (larger errors were caused by the initial pulses asking for large $\Delta \dot{\phi}$ and saturating the thrusters). This is known to be acceptable for the purpose of an autonomous docking experiment using SPHERES. The attitude controller displayed good performance prior to the appearance of the IR noise. However, the tangential alignment was not well maintained. Video of the experiment showed that this was caused by nominal attitude errors, which the single-beacon estimator interpreted as tangential errors. Although this experiment validated the glideslope docking module, the attitude controller, and the overall GN&C architecture, it enlightened a problem with the current implementation of the single-beacon estimator leading to tangential misalignments during the approach.

Docking to a Cooperative Target

Two satellites and five external beacons were available for the fourth test session on August 19 2006. This allowed a series of autonomous docking experiments between two floating satellites, with the target actively holding its position and attitude. The same GN&C system used for the “docking to a fixed beacon” experiment was used, but with the global estimator providing absolute state estimates to both satellites. To control itself during the docking approach, the chaser used relative state estimates computed from its own absolute states and the ones broadcast by the target.

Figure 7 shows the results of this experiment. The left plot shows the separation along the docking axis* between the geometric centers of the satellites; the horizontal capture line indicates when the geometric centers are one diameter apart from each other, resulting in docking. The right plot shows the tangential alignment between the satellites. In the initialization phase, both satellites first acquired a position and an attitude fix from the global estimator. Both induced a random tumble, damped it, pointed at each other and rolled into position during the initialization phase of the experiment. For the remaining of the experiment, the target was commanded to hold its position and to keep pointing at the chaser. At time 39 seconds, the chaser initiated its approach. A jump in the state estimates of the target induced an unexpected tumble of the target satellite at time 53 seconds, while another jump in the state estimates of the chaser occurred at time 64 seconds and resulted in a slight collision at time 69 seconds. At time 83 seconds, the chaser initiated its final open-loop capture firing and made contact with the target at time 87 seconds.

The controllers behaved as expected. The root cause of the perturbations in the approach trajectory followed by the chaser was the presence of jumps in the state estimates. A close analysis of the results indicated that no IR noise was present in the environment during this experiment. These jumps are believed

---

* The docking axis is defined as the X body axis of the target satellite for all docking tests involving two satellites.
to be caused by multi-path in the ultrasound signal transmission when the satellites are in close proximity. They were not observed during ground testing using the same GN&C system, probably because of the different beacon locations around the test volume. This experiment clearly illustrates the need for very smooth state estimates when performing close proximity operations. Following the results from this experiment, a decision was made to implement the measurement error detection algorithm introduced above.

Docking to a Free Drifting Target

The objective of this experiment, also run during the fourth test session, was to demonstrate an autonomous docking maneuver with an uncooperative target. The same GN&C system was used as for the fixed target, except that the thrusters on the target were commanded to remain closed after the separation distance dropped below 0.5 meters.

Results are presented in Figure 8. Both satellites acquired a navigation solution, pointed at each other and rolled in position during the initialization phase which lasted until time 27 seconds. The chaser then initialized its approach. Following the interruption in thruster actuation of the target (at time 63 seconds), every time the chaser reduced its closing rate by firing a plume in the direction of the target, the target was pushed away with an increasing force as the chaser closed in. This effect becomes obvious in between time 91 and 112 seconds where the separation distance slowly increases after an important braking pulse by the chaser. Contact occurred at time 117 seconds, as confirmed by the gyroscope data and by video of the experiment. Each plume impingement produced minimal angular rate changes (of less than 0.5 degrees per second) but important velocity changes of up to 6 millimeters per second. Overall, this experiment was very successful and provided valuable information on the potential effect of plume impingement when docking with an uncooperative target.
Docking with a Cooperative Target with Measurement Error Detection Enabled

Using the results obtained from the series of experiments during the fourth test session, FDIR was implemented into the measurement error detection system. This enabled the GN&C architecture to use closed-loop control during the capture stage, as opposed to the open-loop capture firing at 5 centimeter separation used in previous experiments.

During the fifth test session, on November 11 2006, another cooperative docking experiment was conducted. Figure 9 shows the results. The initialization phase completed at time 33 seconds after both satellites pointed at each other and rolled into position. The chaser then initialized its approach toward the target. At a separation of approximately 7 centimeters, the chaser terminated its glideslope approach. It switched GN&C mode, slowly moving down to 4 centimeters separation and precisely aligning itself. It then switched GN&C mode again and demonstrated the capture maneuver with closed-loop control at time 127 seconds; contact occurred but the Velcro (used as capture mechanism) did not latch. For most of the approach, the tangential alignment was maintained within a box of 2cm x 2cm (within docking tolerance), which confirmed the good tuning of the GN&C system. The video of the experiment and communication from the crew confirmed precise alignment at contact.

![Docking Approach Trajectory](image)

Figure 9 Autonomous docking maneuver to a cooperative target with measurement error detection enabled

Close examination of the data showed that, although it did not have an impact on the success of the experiment, messages transmitted between satellites sometime took much longer than expected to reach their destination. This illustrates the importance of handling acknowledgements between satellites when communicating critical information. Also, because of the good attitude and tangential alignment, the contact between both satellites was barely perceived by the gyroscopes, confirming the need for a different docking sensor.

Docking to a Cooperative Target Following a Safe Approach Trajectory

This test, conducted during the fifth test session, follows a trajectory demonstrating safe rendezvous with a simulated detection of a failure toward the end of the trajectory. When detection of a failure occurs, the GN&C system responds by shutting off all thrusters, forcing the satellite to enter a drift mode. The time of the simulated failure detection is determined to occur randomly within the last few seconds of the test, when the optimized trajectory has been guaranteed to be safe in the presence of failures. Since the optimized trajectory is chosen prior to the failure detection time, this test demonstrates the ability of the safety algorithm to handle arbitrary failures that occur during that window. A similar GN&C architecture as in the other docking tests is used, except that a PID control module is used to follow a pre-computed safe trajectory, replacing the glideslope module.

The results of the experiment are shown in Figure 10. During the initialization phase, both satellites acquired a navigation fix, pointed at each other, rolled into position and moved to a separation distance of approximately 90 centimeters (performed by the chaser). The chaser started to follow the pre-
computed approach trajectory at time 50 seconds. The simulated detection of the failure occurred at time 102 seconds, at which point the spacecraft disabled its thrusters and entered a drift mode as expected. The target held its position with little state error (within +/- 1cm along each axis for most of the time) and the chaser spacecraft followed its trajectory accurately. The maximum deviation from the trajectory occurred at the point of maximum curvature in the desired trajectory where the rate of change reached its maximum, indicating that the bandwidth of the desired trajectory should be reduced for future experiments using the same tracking controller.

Video of the experiment showed that the chaser successfully docked 107 seconds into the test, even in presence of a detected failure. The left plot of Figure 10 shows a small approach velocity at the time of contact, consistent with the goal of a safe docking trajectory. Since a safe trajectory is defined specifically in terms of collision avoidance and the terminal docking box was not inside the failure avoidance region (i.e., failures in the terminal docking box result in docking, while failures in the avoidance region result in an avoidance maneuver), the resulting rendezvous is consistent with a safe maneuver.

![Docking Approach Trajectory](image)

Figure 10 Safe Docking: tangential approach at the end of the docking and path following in global frame

**Docking to a Tumbling Target**

The two repetitions of this test during the fifth test session were the first autonomous docking maneuvers to a tumbling target attempted on-orbit. The objective was to validate the GN&C architecture presented in Figure 3 for such a maneuver. To simulate a tumbling target, the target satellite was commanded to maintain an angular rate of -2.25 degrees per second around its z-body axis during the docking approach phase. The use of a constant rate allows comparison with other docking to tumbling target experiments (yet to be performed) that use a pre-computed approach trajectory.

The results of the first run are shown in Figure 11. The initialization phase consisted in both satellites acquiring an absolute navigation fix, pointing at each other, rolling into position and reducing the separation down to 40 centimeters (performed by the chaser). The initialization phase completed at time 56 seconds. The target began its constant rotation about its z-body axis, as illustrated by the rotation angle shown in Figure 11. The chaser then began its approach toward the target while maintaining tangential alignment, resulting in a spiral motion in the plane of rotation as shown in Figure 12 (the origin of the coordinate frame is located at the reference position used by the target satellite). The video of the experiment and the data indicates that contact occurred at approximately 111 seconds with both satellites well aligned. Safe contact occurred in both experiments; the second experiment achieved full capture, resulting in the first successful autonomous docking to a tumbling target on-orbit.

Because of the increased amount of control input required to maintain such a trajectory, the relative trajectory plot shown in Figure 11 appears noisier than the one shown in other docking experiments. The separation is still shown decreasing continuously during the docking approach, and the tangential alignment remained mostly within a 2cm x 2cm box for the last 20 seconds prior to docking.
Video of the experiment shows the astronaut taking pictures of the satellites 109 seconds into the test. Camera flashes and range finders are known sources of IR, which interferes with the metrology system. Figure 13 shows the filter innovation used by the measurement error detection system. A couple of points are observed above the rejection threshold. These occur precisely at times when unexpected IR flashes were also recorded and when the crew was observed taking a picture of the satellites. The faulty measurements were successfully rejected, allowing the state estimates to remain stable in presence of measurement errors. Had the implementation of the fault detection system using EKF innovation been omitted for this experiment, the measurement errors would have caused the state estimates to jump like observed in previous experiments. Therefore, this experiment went beyond its intended objectives and demonstrated the capability of the GN&C architecture implemented on SPHERES to perform an autonomous docking maneuver to a tumbling target in presence of failures like global positioning measurement errors.
CONCLUSIONS

Successful operations of the SPHERES program aboard the ISS have resulted in a number of important lessons learned in the development of space technology maturation facilities, the transfer from theory to application, and the advancement of autonomous docking algorithms. The incremental nature and flexible reconfiguration of the SPHERES facilities enabled the team to begin operation with a limited set of hardware aboard the ISS. The modularity in the system proved essential to enable incremental technology maturation, as the initial tests with limited hardware were fully used towards achieving the goal of docking to an uncontrollable target. Further, the use of modules originally designed for docking purposes greatly simplified the development of formation flight tests.

The presence of multiple unforeseen sources of noise in the metrology system stressed the need to implement FDIR algorithms at all levels of the GN&C architecture. Even after several years of testing the system in ground based facilities, the deployment aboard the space station presented new and unexpected failure modes which could have been caught by FDIR algorithms, but which otherwise prevented full functionality.

Through the development of several series of autonomous docking experiments, the SPHERES team gained important expertise in the design of GN&C architectures for docking. Through this process, multiple lessons were learned about autonomous docking algorithms:

- The effects of plume impingement were demonstrated to be of importance even with a glideslope algorithm whose benefits include reduced closing rate in close proximity to the target.
- The effects of loss of inter-satellite communications were observed and evaluated: docking situations where there are few unmodeled accelerations are reasonably robust to communications failures; when the chaser cannot sense the target and there are accelerations present (e.g., target external disturbances or thruster firings) communications plays a critical role.
- The straight docking maneuvers showed that in the case where two satellites of similar mass are to dock, the reaction of the target satellite to the initial contact can create substantial disturbances; therefore, while traditional architectures shut off all actuation capabilities in the target at the time of capture, these systems will require active elements on both the capture and target satellites.
- The safe docking provided important lessons in the use of models. The lack of aggressive collisions during both safe docking tests, even when the satellites were not following the predetermined paths within the modeled error, showed robustness in the system. However, the results were not exactly as expected since the different models used to compute the safe trajectory were too aggressive; therefore more conservative margins should be used in the future.
- The demonstration of docking to a tumbling target showed the ability to use traditional GN&C docking architectures for this purpose.

The SPHERES team has achieved substantial successes in the maturation of GN&C algorithms over the course of five test sessions aboard the ISS. These advances will help future missions reduce their risk and launch with greater confidence of success. The test sessions aboard the ISS will continue for an extended amount of time, over which the program is expected to grow by providing more scientists with the unique possibility to test their algorithms in a true microgravity environment.
ACKNOWLEDGMENTS

The authors wish to acknowledge the complete SPHERES team at MIT and Payload Systems Inc as well as the DoD and its Space Test Programs office for their funding, launch opportunities, and continued operational support.

REFERENCES