Smoothing-Based Estimation of an Inspector Satellite Trajectory Relative to a Passive Object

Timothy P. Setterfield
Space Systems Laboratory
Massachusetts Institute of Technology
77 Massachusetts Ave. 37-360
Cambridge, Massachusetts, 02139
tsetterf@mit.edu

David Miller
Space Systems Laboratory
Massachusetts Institute of Technology
77 Massachusetts Ave. 37-327
Cambridge, Massachusetts, 02139
millerd@mit.edu

John J. Leonard
Marine Robotics Group
Massachusetts Institute of Technology
77 Massachusetts Ave. 32-232
Cambridge, Massachusetts, 02139
jleonard@mit.edu

Alvar Saenz-Otero
Space Systems Laboratory
Massachusetts Institute of Technology
77 Massachusetts Ave. 37-381
Cambridge, Massachusetts, 02139
alvarso@mit.edu

Abstract—**This paper presents a method of obtaining the *maximum a posteriori* estimate of an inspector satellite's trajectory about an unknown tumbling target while on-orbit. An inspector equipped with radar or a 3D visual sensor (such as LiDAR or stereo cameras), an inertial measurement unit, and a star tracker is used to obtain measurements of range and bearing to the target's centroid, angular velocity, acceleration, and orientation in the inertial frame. A smoothing-based trajectory estimation scheme is presented that makes use of all the input sensor data to estimate the inspector's trajectory. Open-source incremental smoothing and mapping (iSAM2) software is used to implement the smoothing-based trajectory estimation algorithm; this facilitates computationally efficient evaluation of the entire trajectory, which can be performed incrementally, and in real time on a computer capable of processing 3D visual sensor data in real time. The presented algorithm was tested on data obtained in 6 degree-of-freedom microgravity using the SPHERES-VERTIGO robotic test platform on the International Space Station (ISS). In these tests, a SPHERES inspector satellite with attached stereo cameras circumnavigated a passive SPHERES target satellite, making visual observations of it. The results of these tests demonstrate accurate estimation of the inspector satellite's trajectory.**

TABLE OF CONTENTS

1. INTRODUCTION 1
2. BACKGROUND AND RELATED WORK 2
3. APPROACH..4
4. RESULTS AND DISCUSSION..........................8
5. CONCLUSION 10
REFERENCES 10
BIOGRAPHY 11

1. INTRODUCTION

Many resident space objects, such as orbital debris, defunct satellites, and natural objects like asteroids and comets, are of unknown appearance, uncooperative, and tumbling. There are several reasons to want to visit these resident space objects with an inspector satellite: to deorbit or deflect dangerous debris; to inspect and/or repair defunct satellites; or to make scientific observations of natural objects. Unmanned autonomous inspector satellites have been proposed by numerous organizations to perform these tasks [1], [2], [3], [4], [5], [6]. After performing orbital rendezvous with the resident space object (referred to as "the target" herein), a mission designed to accomplish these objectives would involve one or more of the proximity operation phases below [2]:

1. Observing and planning, during which the inspector satellite maintains a safe distance from the target, acquiring information about it and planning its next steps.
2. Final approach of the inspector toward the desired docking or grasping location on the target.
3. Docking to or capture of the target satellite.
4. Stabilization of the potentially tumbling target through application of external torques.

For missions whose sole objective is to inspect the target from a distance, only Phase 1 would be executed. For missions seeking samples from the target, Phases 1-3 would be executed before detaching and returning to Earth. For missions intending to deorbit, deflect or repair the target, Phases 1-4 would be required. The separate phases of proximity operations are illustrated in Figure 1.

The research described herein focuses on algorithms for Phase 1, where the inspector satellite maintains a safe distance from the target, acquiring information about it and planning its next steps. The broader objective of this research is to obtain a sufficiently detailed and accurate estimate of the target's structure and dynamics to serve as a platform to plan and execute the objectives of Phases 2-4. For successful execution of these phases, several pieces of information are required. Firstly, the time history of the inspector's pose is needed in order to disambiguate the motion of the inspector from that of the target. Secondly, a 3D visual map of the target sufficient to determine the relative pose between the inspector and the target is needed. Thirdly, estimation of the target's rotational motion and dynamic properties is required to predict the motion of the target through Phases 2-4. This paper focuses on the first of these three objectives, although development of algorithms for the second and third objectives is ongoing at the MIT Space Systems Laboratory.

Two critical insights drive the approach taken to recover the trajectory estimation approach taken in this paper. Firstly, the target object has a rotational motion that is unknown *a priori*, meaning that the estimation scheme must never depend on an assumption of known target orientation. This excludes

978-1-5090-1613-6/17/$31.00 ©2017 IEEE

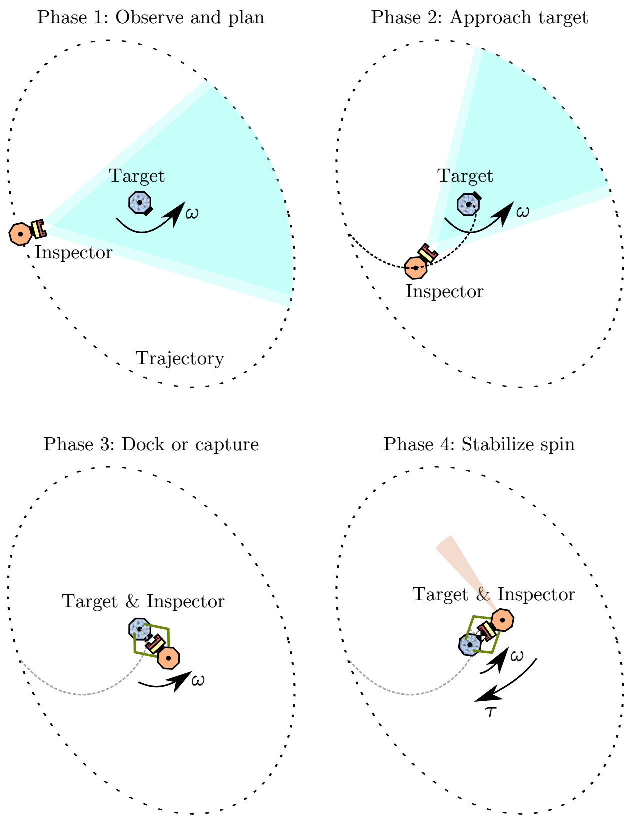


Figure 1. Overview of concept of operations for proximity operations. This paper focuses on Phase 1: Observe and plan.

approaches such as visual odometry that purely use visual observations of the target. Working within these constraints, the approach presented herein treats the target as a single point, obtaining only a range and bearing measurement to its approximate geometric centroid. The remaining degrees of freedom in the problem are resolved using a star tracker to measure absolute orientation. Secondly, in future work, the estimate of inspector trajectory will subsequently (or simultaneously) be used in mapping and motion estimation of the target. This means that the approach used herein must facilitate the mapping of the target, where associating temporally separated measurements of the same location improves mapping accuracy (so called "loop closure"). In terrestrial simultaneous localization and mapping (SLAM), similar challenges have led to the widespread adoption of batch smoothing-based approaches over traditional recursive filtering. The approach presented herein draws on efficient incremental smoothing and mapping (iSAM2) methods [7] developed in the terrestrial SLAM community to implement a smoothing-based method of estimating the inspector trajectory.

This paper will overview background and related work in Section 2, detail the approach taken in Section 3, discuss the results in Section 4, and conclude in Section 5.

2. Background and Related Work

Resident Space Objects

The evolution of tracked man-made objects in space from the beginning of the space age is shown in Figure 2. Only 6% (slightly above 1000) of the tracked objects are operational spacecraft, 30% are non-operational but intact objects, and the remaining 64% are fragments [8]. Thus 94% of man-made objects in space are not actively controlled, and undergo entirely passive motion, as is assumed of the target herein.

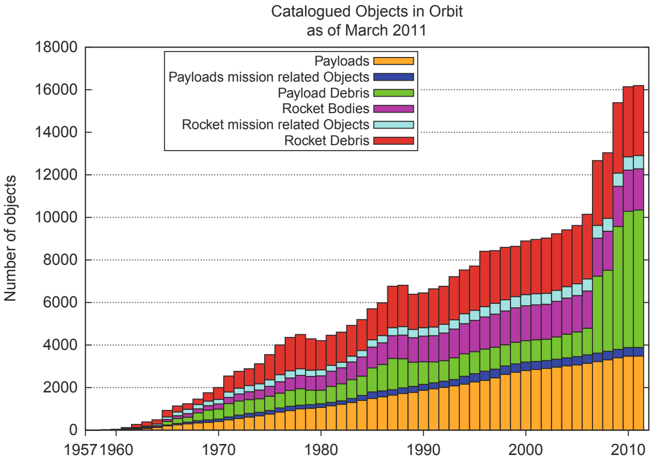


Figure 2. Space objects > 10cm in size tracked by the US Space Surveillance Network (SSN) [8].

Satellite Inspection and Servicing—Missions to defunct, non-operational, but intact satellites, typically aim to restore their previous functionality by repairing, refueling, or reboosting them. Of 2431 orbital payloads launched from 1981-2000, 190 experienced on-orbit failures [1]; many of these failures could have indeed been remedied by servicing if the required systems were available [9]. Because of the high cost of these failures, robotic on-orbit servicing has been thoroughly pursued [2]. Although autonomous satellite servicing has not yet been performed on an uncooperative satellite [10], it has been investigated with demonstration missions of increasing complexity, and useful missions to defunct satellites may soon be possible.

JAXA's ETS-VII successfully demonstrated the automated undocking, station keeping, and redocking of two cooperative satellites in 1998 [11]. The XSS-10 maneuverable satellite was launched in 2003, performing a circumnavigation of its expended Delta II second stage [12]. The follow-on mission XSS-11 was launched in 2005 and performed at least 75 natural motion circumnavigations of its expended Minotaur I rocket's upper stage [13]. The Orbital Express mission in 2007 demonstrated "short range and long range autonomous rendezvous, capture and berthing, on-orbit electronics upgrades, on-orbit refueling, and autonomous fly-around visual inspection" of a cooperative target satellite [14]. The NASA Satellite Servicing Capabilities Office (SSCO) is enabling satellite servicing in LEO or GEO for satellites not originally designed for servicing. Raven, an SSCO project scheduled for a 2016 launch, will use a visible camera, an infrared camera, and a flash LiDAR (a Light Detection And Ranging instrument) to determine the relative poses of incoming and outgoing vehicles visiting the ISS [15]. SSCO's planned Restore-L mission aims to refuel Landsat 7, an on-orbit satellite, near the end of its planned operations in 2020 [16]. Orbital ATK also plans to offer mission extension services for satellites nearing their end of life using their Gemini bus as a servicing platform [4].

Space Debris— For orbital debris, the primary driver for proximity operations is to either deflect it or capture and deorbit it. Since orbital velocities are extremely high (e.g. ~7.8 km/s in LEO), collisions with even small pieces of orbital debris can endanger robotic and manned missions. Large orbital debris, such as the 11% of space debris comprised by intact upper rocket stages [8], can collide with other debris, exacerbating the problem, and thus has been an early target for removal [17].

The European Space Agency (ESA) is pursuing the capture and deorbit of debris using nets, tethers, harpoons, and ion-beam shepherding [18], and has recently approved an active debris removal mission called e.deorbit, projected to be launched in 2023 [6]. A Singapore-based satellite services company called Astroscale is also planning on providing commercial space debris removal services using its 90 kg ADRAS 1 satellite [5].

Natural Objects—For natural objects, such as asteroids or comets, proximity operations are typically undertaken to make scientific observations or to take scientific samples of the surface. Missions such as Hayabusa, Dawn, and Rosetta all travelled into orbit around an asteroid or comet. There has also been recent interest in near-Earth asteroids by companies such as Planetary Resources, since many are rich in precious metals, and harbor water that could be used to make rocket propellant outside of the Earth's gravity well [19].

Relative Navigation

Spacecraft capable of proximity operations typically possess a range and bearing sensor for use with a cooperative or uncooperative target. For this purpose, the Gemini and Apollo missions used radar, the space shuttle missions used radar and a laser rangefinder, and the Soyuz missions used arrays of antennas [11]. The target was cooperative in all of these manned missions. JAXA's ETS-VII and Orbital Express used laser ranging with retroreflectors, and XSS-11 used LiDAR [11]. Of these robotic missions, the target was cooperative in all but XSS-11; however, XSS-11 served only as an inspector and did not dock with its targets.

Recent work using similar sensors to those required in this paper has focused on filtering approaches. Woffinden and Gellar outline a technique for performing angles-only navigation using an extended Kalman filtering approach [20]. In their method, a monocular camera is used to measure bearing (from position in the image) and range (from the size of the target in the image) to the target. Barbee et al. [21] simulated estimation of an inspector's trajectory using a navigation filter for the various phases of rendezvous with a noncooperative target using a range and bearing sensor on the inspector satellite.

Although the authors are not familiar with any orbital systems that have used smoothing in place of filtering, Steiner et al. [22] demonstrated the use of incremental smoothing approaches for terrain relative navigation of landing spacecraft.

Smoothing-Based Estimation

Smoothing-based estimation is a batch method of estimation which attempts to find the *maximum a posteriori* estimate of a set of random variables $\mathbf{\Theta}$. This is accomplished by finding optimal values $\mathbf{\Theta}^*$ that maximize the joint probability of $\mathbf{\Theta}$ and the noisy measurements $\mathbf{Z}$. Sometimes a set of commanded control inputs $\mathbf{U}$ is also included to parametrize the problem; herein these control inputs are treated as random variables with low covariance. Thus the smoothing-based estimation problem can be defined as finding the optimal vector $\mathbf{\Theta}^*$.

$$\begin{aligned}\mathbf{\Theta}^* &= \underset{\mathbf{\Theta}}{\operatorname{argmax}} \, P\left(\mathbf{\Theta}, \mathbf{Z}, \mathbf{U}\right) \\ &= \underset{\mathbf{\Theta}}{\operatorname{argmax}} \, P\left(\mathbf{Z}, \mathbf{U} \mid \mathbf{\Theta}\right) P\left(\mathbf{\Theta}\right) \end{aligned} \tag{1}$$

Assuming that each measurement is Gaussian, and using the monotonic natural logarithm function, this problem can be reduced to the problem of minimizing the sum of several probabilistic factors [23], [7]:

$$\begin{aligned}\mathbf{\Theta}^* &= \underset{\mathbf{\Theta}}{\operatorname{argmin}} \, -\log\left(P\left(\mathbf{Z}, \mathbf{U} \mid \mathbf{\Theta}\right) P\left(\mathbf{\Theta}\right)\right) \\ &= \underset{\mathbf{\Theta}}{\operatorname{argmin}} \, \frac{1}{2} \sum_i \, \|h(\boldsymbol{\theta}_i) - \mathbf{z}_i\|^2_{\mathbf{\Sigma}_i} \end{aligned} \tag{2}$$

where $h(\boldsymbol{\theta}_i)$ denotes the measurement model as a function of a subset of $\mathbf{\Theta}$, $\mathbf{z}_i$ is the measurement value, $\mathbf{\Sigma}_i$ is the measurement's covariance, and $||\boldsymbol{\epsilon}||^2_{\mathbf{\Sigma}} \equiv \boldsymbol{\epsilon}^T \mathbf{\Sigma}^{-1} \boldsymbol{\epsilon}$ is the Mahalanobis distance from the exponent of a multivariate Gaussian distribution.

Using this framework, several Gaussian measurements can be incorporated so long as there exists a measurement model h in terms of the variables $\mathbf{\Theta}$ and an estimate of the measurement covariance $\mathbf{\Sigma}_i$. Gaussian prior belief can also be incorporated using the simple measurement model $h(\boldsymbol{\theta}_i) = \boldsymbol{\theta}_i$. For real-world problems, the resultant optimization problem is typically a large nonlinear, non-convex problem. The state-of-the-art incremental smoothing and mapping technique (iSAM2) exploits the structure and sparsity of the problem to obtain efficient solutions which can often be computed in real time [7]. The open source GTSAM software, which implements iSAM2, is used for smoothing-based estimation in this paper.

Rotation Parameterization

Herein, rotation matrices are defined by how they change the frame in which a vector is expressed: i.e. ${}^B\mathbf{t} = {}^B_A\mathbf{R}\,{}^A\mathbf{t}$. Rotation in three dimensions lives on a manifold called the rotation group SO(3) and has only three degrees of freedom. Although three parameter representations will always contain a singularity, they are commonly used for representing uncertainty and performing optimization near the identity rotation. For these applications, the exponential map can be used to map from rotation matrices to angle-axis vectors ($\mathrm{SO}(3) \Rightarrow$ so(3)) and logarithmic maps can be used to map angle-axis vectors to rotation matrices ($\mathrm{so}(3) \Rightarrow \mathrm{SO}(3)$), where so(3) is a subspace of $\mathbb{R}^3$ inside the ball of radius π. Defining $\boldsymbol{\theta} \in \mathrm{so}(3)$ to be a vector in the direction of the rotation axis, with a magnitude θ equal to the rotation angle in radians, the mappings are defined as follows:

$$\boldsymbol{\theta}^{\wedge} = \begin{bmatrix} 0 & -\theta_3 & \theta_2 \\ \theta_3 & 0 & -\theta_1 \\ -\theta_2 & \theta_1 & 0 \end{bmatrix} = -\left[\boldsymbol{\theta}^{\wedge}\right]^T \tag{3}$$

$$\mathbf{R} = \exp\left(\boldsymbol{\theta}^{\wedge}\right) = \mathbf{I}_3 + \frac{\sin(\theta)}{\theta}\boldsymbol{\theta}^{\wedge} + \frac{1-\cos(\theta)}{\theta^2}\left[\boldsymbol{\theta}^{\wedge}\right]^2 \tag{4}$$

$$\boldsymbol{\theta} = \left[\boldsymbol{\theta}^{\wedge}\right]^{\vee} = \log\left(\mathbf{R}\right)^{\vee} = \frac{\theta}{2\sin(\theta)}\left(\mathbf{R}-\mathbf{R}^T\right)^{\vee} \tag{5}$$

$$\theta = \arccos\left(\frac{\mathrm{tr}\left(\mathbf{R}\right)-1}{2}\right) \tag{6}$$

SPHERES-VERTIGO Test Platform

The algorithm described in this paper has been tested using data from the SPHERES (Synchronized Position Hold Engage Reorient Experimental Satellites) test platform. The SPHERES facility for guidance, navigation and control (GNC) testing consists of a set of hardware and software tools developed for the maturation of metrology, control, and autonomy algorithms. Three SPHERES satellites are currently aboard the ISS. VERTIGO (Visual Estimation for Relative Tracking and Inspection of Generic Objects) is a hardware upgrade to the SPHERES satellites that enables vision-based navigation research (see Figure 3). Attached to the SPHERES expansion port, VERTIGO includes stereoscopic cameras, an embedded Linux computer, and associated software libraries and drivers. The VERTIGO computer shares telemetry and control data with the SPHERES satellite and is fully integrated in the GNC and autonomous decision making. Two VERTIGO units are currently aboard the ISS. More details of the SPHERES-VERTIGO test platform can be found in [24].

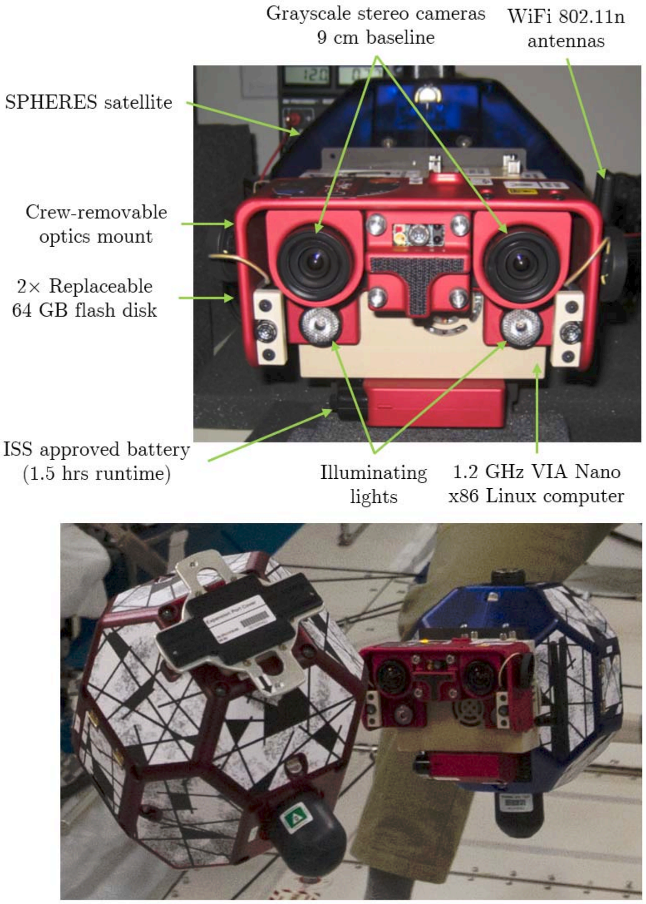


Figure 3. (Top) The major components of the SPHERES-VERTIGO test platform. (Bottom) SPHERES-VERTIGO in operation on the ISS.

3. Approach

The objective of this paper is to recover the *maximum a posteriori* estimate for the full 6-DOF trajectory of the inspector satellite. Specifically, this involves solving for the most probable time sequence of the state vector $\mathbf{x}$ over a sequence of time intervals $\mathcal{I} = \{0, 1, ..., I\}$:

$$\begin{aligned} \mathbf{x}_{\mathcal{I}} &= \left\{{}^{W}\mathbf{r}, {}^{W}_{B}\mathbf{R}, {}^{W}\mathbf{v}, \mathbf{b}^g, \mathbf{b}^a\right\}_{\mathcal{I}} \\ \mathbf{x} &\in \left\{\mathbb{R}^3, \mathrm{SO}(3), \mathbb{R}^3, \mathbb{R}^3, \mathbb{R}^3\right\} \end{aligned} \tag{7}$$

where ${}^{W}\mathbf{r}$ is the position of the inspector center of mass in the world frame, ${}^{W}_{B}\mathbf{R}$ is the rotation matrix that resolves vectors from the inspector body frame B to the world frame W (i.e. ${}^{W}\mathbf{t} = {}^{W}_{B}\mathbf{R}\,{}^{B}\mathbf{t}$), ${}^{W}\mathbf{v}$ is the velocity of the inspector center of mass in the world frame W, $\mathbf{b}^g$ is the gyroscope bias, and $\mathbf{b}^a$ is the acceleration bias.

To determine the full 6-DOF trajectory of the inspector satellite, a smoothing-based estimation approach is adopted. This section will discuss the required measurements from the inspector satellite, the choice of coordinate frames, the smoothing-based formulation of the problem, its observability, implementation details, and the experimental design.

Required Measurements

In order to use the methodology herein to determine the relative trajectory of an inspector satellite about a target object, a minimum set of measurements are required. The first is a measurement of range and bearing from the inspector to the target's approximate centroid. This can be fulfilled by using an averaged depth from radar, LiDAR, or stereo cameras. The second is a measurement of orientation of the inspector in the inertial frame. This can be fulfilled by using star tracker measurements.

Although not critical for making the trajectory observable, adding an independent internal measure of motion increases the accuracy, reliability, and real-time availability of the state information. For rotation, this role can be filled by onboard gyroscopes. For translation, this role can be filled by a combination of commanded forces and relative orbital dynamics models.

Thus the following time series of measurements and control inputs are required:

$$\mathbf{Z} = \left\{\mathbf{Z}^r, \mathbf{Z}^b, \mathbf{Z}^s, \mathbf{Z}^g, \mathbf{Z}^a\right\} \quad \mathbf{U} = \left\{\mathbf{u}^f\right\} \tag{8}$$

where $\mathbf{Z}^r$ are range measurements, $\mathbf{Z}^b$ are bearing measurements, $\mathbf{Z}^s$ are star tracker measurements, $\mathbf{Z}^g$ are gyroscope measurements, $\mathbf{Z}^a$ are acceleration measurements, and $\mathbf{u}^f$ are the commanded forces on the inspector.

Autonomous inspector satellites capable of rendezvous and docking with an uncooperative target will typically have the sensors required to fulfil these measurement requirements. The measurements are visualized in Figure 4.

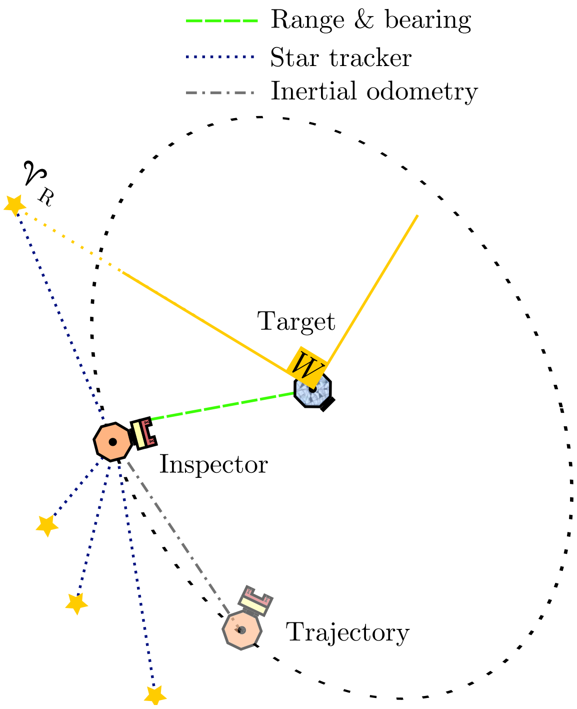


Figure 4. Measurements required for the trajectory estimation algorithm described in this paper.

Coordinate Frames

A local world coordinate frame W is defined as having its origin at fixed the target's center of mass, but axes aligned rotationally with inertial space. This means that range and bearing measurements of the target's geometric centroid are approximate range and bearing measurements of the origin of W. It also means that the star tracker of the inspector satellite can directly measure its orientation in the world frame W. The inspector has two attached coordinate frames. The body frame B is attached at its center of mass and aligned with its principal axes of inertia; calibration of the gyroscopes allow them to sense the rotation of the inspector in this frame. The camera frame C is located at a known position and orientation on the inspector. This is the frame from which the range and bearing to the target's centroid are measured. Figure 5 illustrates the locations of each coordinate frame.

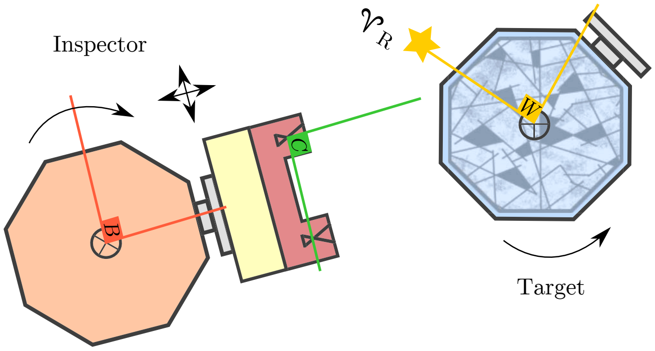


Figure 5. The world W, body B, and camera C coordinate frames.

Smoothing-Based Formulation

To estimate the time series of inspector states from Equation 7, the trajectory estimation problem was formulated as a batch optimization problem of the type described in Equation 2. Each summand in Equation 2 represents a probabilistic factor to be minimized. By adding each measurement as a separate factor, all of the measurements can be incorporated into the optimization. The four types of factors used in the estimation scheme herein are described below.

Range and Bearing Factor—Since the world frame W has its origin at the target's center of mass, range and bearing measurements of the the target's geometric centroid from the inspector are considered approximate range and bearing measurements of the origin of W. The range measurement $z^r = r_{C\text{to}W}$ is the magnitude of the distance between the visual sensor and the approximate target centroid; it has one degree of freedom. The range measurement can be assumed to have a Gaussian noise with covariance Σ_r, and can be applied to the optimization problem as a factor ρ^r as follows:

$$\begin{aligned} P\left(z^r \mid {}^W\mathbf{r}, {}^W_B\mathbf{R}\right) &\sim \mathcal{N}\left(z^r, \Sigma_r\right), \; z^r \in \mathbb{R} \\ \rho^r &= \left\| \left\| {}^W\mathbf{r} + {}^W_B\mathbf{R}\, {}^B\mathbf{r}_{B\text{to}C} \right\| - z^r \right\|^2_{\Sigma_r} \end{aligned} \tag{9}$$

where ${}^B\mathbf{r}_{B\text{to}C}$ is the known body to visual sensor distance in the body frame B.

Similarly, the bearing is the direction to the world origin from the inspector's visual sensor. It has two degrees of freedom and is represented by a unit vector. The evaluation of error is performed in the $\mathbb{R}^2$ tangent space of the unit sphere, as shown in Figure 6. Here, the tangent space (tangent plane) is defined locally about the predicted bearing value $\hat{\mathbf{p}}_1$ by heuristically chosen basis vectors $\hat{\mathbf{b}}_1$ and $\hat{\mathbf{b}}_2$. The bearing error $\boldsymbol{\epsilon}_b \in \mathbb{R}^2$ between the current estimate $\hat{\mathbf{p}}_1$ and measured value $\hat{\mathbf{p}}_2$ is defined as the arc length s in the plane of the angular error. It can be found as follows:

$$\begin{aligned} \boldsymbol{\epsilon}_b\left(\hat{\mathbf{p}}_1, \hat{\mathbf{p}}_2\right) &= \begin{bmatrix} \hat{\mathbf{b}}_1^T \\ \hat{\mathbf{b}}_2^T \end{bmatrix} s\,\hat{\mathbf{a}} = \begin{bmatrix} \hat{\mathbf{b}}_1^T \\ \hat{\mathbf{b}}_2^T \end{bmatrix} \theta\,\hat{\mathbf{a}} = \begin{bmatrix} \hat{\mathbf{b}}_1^T \\ \hat{\mathbf{b}}_2^T \end{bmatrix} \theta \frac{\mathbf{a}}{\|\mathbf{a}\|} \\ &= \begin{bmatrix} \hat{\mathbf{b}}_1^T \\ \hat{\mathbf{b}}_2^T \end{bmatrix} \arccos\left(\hat{\mathbf{p}}_1 \cdot \hat{\mathbf{p}}_2\right) \frac{\hat{\mathbf{p}}_2 - \left(\hat{\mathbf{p}}_1 \cdot \hat{\mathbf{p}}_2\right)\hat{\mathbf{p}}_1}{\sqrt{1 - \left(\hat{\mathbf{p}}_1 \cdot \hat{\mathbf{p}}_2\right)^2}} \end{aligned} \tag{10}$$

where $s = \theta$ since $\hat{\mathbf{p}}_1$ and $\hat{\mathbf{p}}_2$ are unit vectors. The bearing measurement $\mathbf{z}^b = {}^C\mathbf{r}_{C\text{to}W} / \|{}^C\mathbf{r}_{C\text{to}W}\|$ can be assumed to have a Gaussian noise with covariance $\boldsymbol{\Sigma}_b$, and can be applied to the optimization problem as a factor ρ^b as follows:

$$\begin{aligned} P\left(\mathbf{z}^b \mid {}^W\mathbf{r}, {}^W_B\mathbf{R}\right) &\sim \mathcal{N}\left(\mathbf{z}^b, \boldsymbol{\Sigma}_b\right), \; \mathbf{z}^b \in S^2 \\ \rho^b &= \left\| \boldsymbol{\epsilon}_b \left(\frac{-{}^C_B\mathbf{R}\left({}^B_W\mathbf{R}\, {}^W\mathbf{r} + {}^B\mathbf{r}_{B\text{to}C}\right)}{\left\|{}^B_W\mathbf{R}\, {}^W\mathbf{r} + {}^B\mathbf{r}_{B\text{to}C}\right\|}, \mathbf{z}^b \right) \right\|^2_{\boldsymbol{\Sigma}_b} \end{aligned} \tag{11}$$

where ${}^C_B\mathbf{R}$ is the known rotation from body frame B to camera frame C.

A variable ${}^W\mathbf{r}_t$ may be introduced to represent the position of the target's center of mass. A prior belief ϕ_{r_t} with a very low covariance is placed on the position of this "landmark". Nominally, the mean is set to ${}^W\mathbf{r}_t = [0\ 0\ 0]^T$ (as described above), but when convenient this location can be shifted.

In this case, the expected range and bearing measurements described above must be modified to appropriately consider this shift.

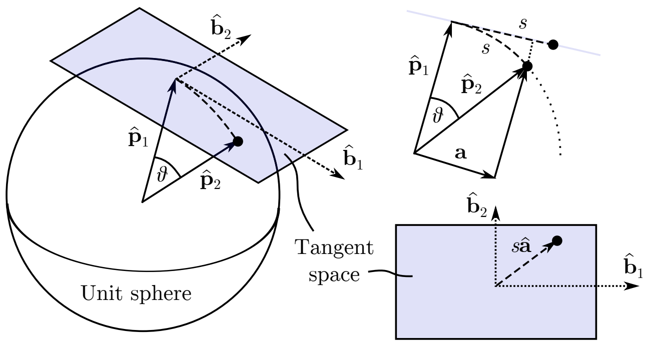


Figure 6. Evaluation of bearing error between the current estimate $\hat{\mathbf{p}}_1$ and the measured value $\hat{\mathbf{p}}_2$ in the tangent space.

Star Tracker Factor—Star tracker measurements give a direct measurement of the inspector's orientation in the world frame. The measurement $\mathbf{z}^s = {}^W_B\mathbf{R}$ is assumed to have Gaussian noise and can be applied directly as a partial factor σ to the orientation component ${}^W_B\mathbf{R}$ of the inspector's state $\mathbf{x}$. The covariance of this measurement $\mathbf{\Sigma}_s$ is determined by the accuracy of the star-tracker. Note that the logarithmic map described in Section 2 is used to reduce the difference in orientation to its three parameter representation. The partial factor σ can then be defined as follows.

$$P\left(\mathbf{z}^s \mid {}^W_B\mathbf{R}\right) \sim \mathcal{N}\left(\mathbf{z}^s, \mathbf{\Sigma}_s\right),\ \mathbf{z}^s \in \mathrm{SO(3)}$$
$$\sigma = \left\| \log\left((\mathbf{z}^s)^T\, {}^W_B\mathbf{R} \right)^\vee \right\|^2_{\mathbf{\Sigma}_s} \qquad (12)$$

Initial State Prior Factor—A prior belief is applied to the initial state. In practice, this will be provided at the hand-off between a long range estimation scheme and that described herein. Here, the "measurement" $\mathbf{z}^{x_0} = \mathbf{x}_0$ is the state estimate at hand-off, which can be assumed to have Gaussian uncertainty with covariance $\mathbf{\Sigma}_{x_0}$, and can be applied as a factor ϕ_{x_0} as follows:

$$P\left(\mathbf{z}^{x_0} = \mathbf{x}_0\right) \sim \mathcal{N}\left(\mathbf{z}^{x_0}, \mathbf{\Sigma}_{x_0}\right)$$
$$\mathbf{x}_0 \in \left\{\mathbb{R}^3, \mathrm{SO(3)}, \mathbb{R}^3, \mathbb{R}^3, \mathbb{R}^3\right\}$$
$$\phi_{x_0} = \left\| \mathbf{x}_0 \ominus \mathbf{z}^{x_0} \right\|^2_{\mathbf{\Sigma}_{x_0}} \qquad (13)$$

where $\mathbf{x}_a \ominus \mathbf{x}_b$ indicates vector subtraction for all vector space components of state and $\log\left(\mathbf{R}_b^T\, \mathbf{R}_a\right)^\vee$ for the SO(3) component (i.e. body orientation).

Inertial Odometry Factor—The preintegrated inertial odometry factor, described in Forster et al. [25], was designed to summarize a batch of IMU readings. It provides an estimate of body motion as well as Jacobians expressing the sensitivity of this estimate to changes in navigation state estimates at the beginning and end of the measurement batch. Although the factor was intended to accept gyroscope and accelerometer inputs and operate in the influence of Earth's gravity field, the mathematics and open source implementation accept any valid angular rate and acceleration measurements in any desired gravity field. Herein, gyroscope measurements are used for the angular rate measurements, the gravitational field strength is set to zero, and thruster forces $\mathbf{u}^f$ divided by vehicle mass m are used in lieu of accelerometer measurements. Although not performed herein, the incorporation of accelerations from relative orbital dynamics could be performed using this framework. The noise on these acceleration"measurements" is small and based on expected thruster force variation. The navigation state in [25] is identical to that in Equation 7. However, the acceleration bias $\mathbf{b}^a$ has a different interpretation herein than in [25]. In [25], it represents biases in the accelerometer. Herein, when a thruster-only model is used, it represents systematic directional biases in thruster force. The mathematics behind this factor are complex, and the interested reader is referred to Forster et al. [25] for the details. In summary, a batch of angular rate and acceleration "measurements" $\left\{\mathbf{z}^g = {}^B\omega, \mathbf{z}^a = \mathbf{u}^f/m\right\}_{i,j}$ in the interval of interest from timestep i to timestep j has a probability of occurring that can be assumed to be Gaussian with covariance $\mathbf{\Sigma}_{ga}$, and can be applied to the optimization problem as a factor ψ as follows:

$$P\left(\textstyle\int_{t_i}^{t_j} \left\{\mathbf{z}^g, \mathbf{z}^a\right\} dt \mid \mathbf{x}_i, \mathbf{x}_j\right) \sim \mathcal{N}\left(\textstyle\int_{t_i}^{t_j} \left\{\mathbf{z}^g, \mathbf{z}^a\right\} dt, \mathbf{\Sigma}_{ga}\right)$$
$$\textstyle\int_{t_i}^{t_j} \left\{\mathbf{z}^g, \mathbf{z}^a\right\} dt \in \left\{\mathbb{R}^3, \mathrm{SO(3)}, \mathbb{R}^3, \mathbb{R}^3, \mathbb{R}^3\right\}$$
$$\psi = \left\| \Delta\mathbf{x}_{i,j} \ominus \textstyle\int_{t_i}^{t_j} \left\{\mathbf{z}^g, \mathbf{z}^a\right\} dt \right\|^2_{\mathbf{\Sigma}_{ga}} \qquad (14)$$

where $\int_{t_i}^{t_j} \left\{\mathbf{z}^g, \mathbf{z}^a\right\} dt$ and $\Delta\mathbf{x}_{i,j} = \mathbf{x}_j \ominus \mathbf{x}_i$ are the measured (from preintegrated measurements) and the predicted (from current state estimates) changes in navigation state respectively.

Complete Factor Graph— With each factor defined, it is possible to create a graphical representation of the joint probability distribution called a factor graph (Figure 7). Here, the circular nodes represent the random variables to be solved for: a sequence of states $\mathbf{x}_{\mathcal{I}}$. The square nodes represent the probabilistic factors ρ, σ, ϕ_{x_0}, and ψ described above. Edges are drawn between factors and the variables on which they conditionally depend, providing an illustration the probability distribution's structure. Herein it is assumed that range and bearing measurements and star tracker measurements arrive with a much lower frequency than the measurement frequency of the IMU. Preintegrated IMU measurements can then connect adjacent states. In the illustrated factor graph, star tracker and range and bearing measurements are applied concurrently, but it is valid to apply one at a time if their arrival times are mismatched. It is of no utility to create a state variable node without a connected star tracker or range and bearing factor unless a state estimate is required between measurements. Ultimately, the creation of variable nodes and application of factors is at the user's discretion; the factor graph shown is representative of the approach taken in this paper.

Observability

The observability of this problem can be seen intuitively, and so an explanation is provided here in lieu of a formal proof (see also Figure 8). A measurement of range between the inspector and the target restricts the location of the inspector to a sphere around the target. The star tracker measurement of orientation then restricts the three rotational degrees of freedom of the inspector; however, the inspector can still reside at any point on the sphere. The remaining two degrees of freedom are restricted by the bearing measurement to the target, which places the inspector satellite at a specific point on the sphere. The observability of velocity follows from

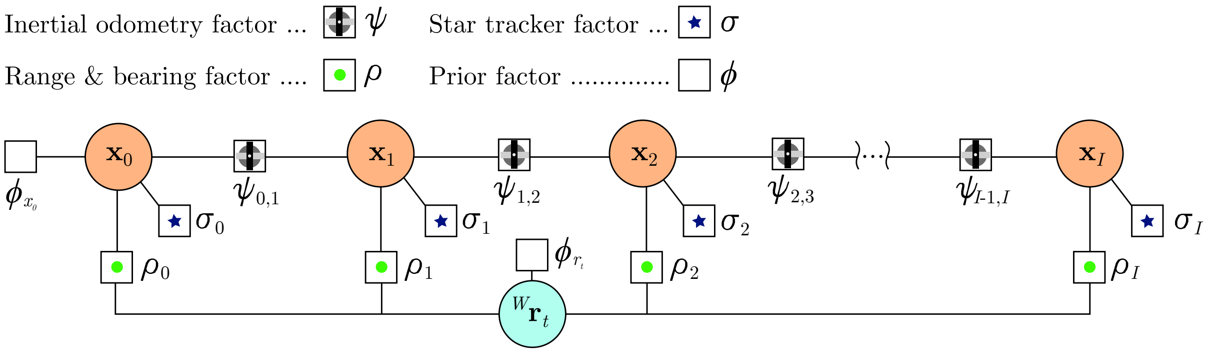


Figure 7. A factor graph representing the joint probability distribution for the inspector trajectory estimation problem. Circular nodes, representing random variables to be solved for are connected via edges to square nodes, representing probabilistic factors.

the time rate of change of position, and the observability of gyroscope and acceleration biases follows from the accurate knowledge of position and observability of velocity respectively.

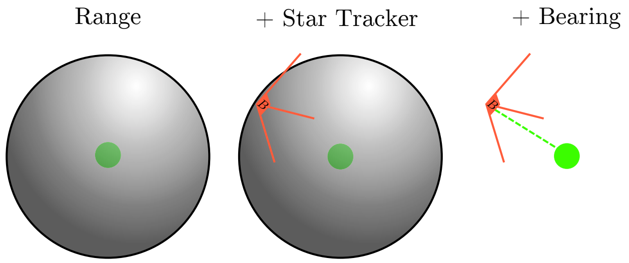
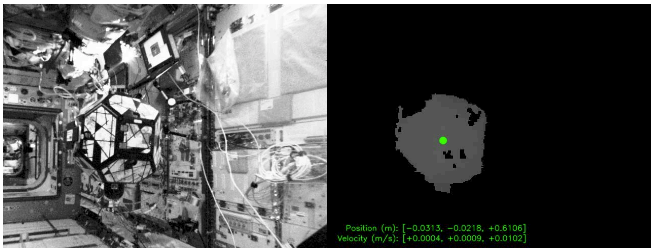


Figure 8. The six degrees of freedom of pose are successively restricted by knowledge of the range, orientation (via star tracker), and bearing.

Implementation

Range and Bearing Measurements— To obtain range and bearing measurements, the depth map provided by the 3D sensor is thresholded such that anything beyond a prescribed distance is considered to be part of free space and not the target object. It is assumed that the entire target object lies within the frame, and the remaining measurements are averaged to obtain a range and bearing measurement to a single point. The rate of image acquisition and processing on VERTIGO was approximately 2.34 Hz and an example of the output data from this procedure is shown in Figure 9.

Depth measurements will all be from the inspector-facing side of the target, and therefore are not an exact measurement of the target's geometric centroid. Typically, these measurements will indicate that the target's centroid is closer to the inspector than it actually is. In practice, if the distance between the inspector and the target is significantly large compared with the dimensions of the target, the error induced will be small as a percentage of range. However, to mitigate this problem at close range, a range bias term can be added using an approximate expected value based the *a priori* knowledge of the target's size. On the SPHERES system, this is implemented by approximating the near side of the target SPHERES satellite as a hemispherical shell; in this case the range bias is $\frac{1}{2}r_s$, where r_s is the (known) radius of the SPHERES satellite. The range measurement is thus corrected



```
Position (m): [-0.0313, -0.0218, +0.6106]
Velocity (m/s): [+0.0004, +0.0009, +0.0102]
```

Figure 9. (Left) Grayscale image from the left camera of the VERTIGO stereo vision system. (Right) A depth map – formed from an equalized, subsampled stereo image – thresholded to include only the foreground, with the target centroid shown in green. The depth map was obtained using the `libelas` open source library [26].

as follows:

$$z^r_{corr} = r_{C\text{to}W} = z^r + \frac{1}{2}r_s \tag{15}$$

Note that the mass distribution of the target object may not be known. Thus even a perfect measurement of the geometric centroid may be an inaccurate measurement of the target center of mass for objects with non-uniform density. This effect is ignored herein.

Star Tracker Measurements— A star tracker provides measurement of the inspector's orientation in the rotationally inertial world frame W. For the SPHERES test platform, this world frame is attached to the ISS; five ultrasonic beacons at known locations are typically used within an extended Kalman filter for localization within the test volume [27]. The orientation accuracy of this estimator is $\sim 3°$ when the satellites remain inside the designated test volume (surrounded by all five beacons). For the test herein, however, the inspector satellite briefly leaves the test volume, and the accuracy of the beacon-provided orientation degrades substantially. To mitigate this problem, a feature-based visual-inertial odometry algorithm was run separately using the same data to obtain a better estimate. For this, the ISS walls provided static background features for use as a reference to the world coordinate frame W. Estimated orientation from this algorithm reflects the motion of the inspector satellite more accurately than the beacon-based method, and is used herein to provide star tracker measurements; the full pose from this method is also used herein as the most accurate available "ground truth".

Hand-Off State Measurement— In the SPHERES experiments, the hand-off state measurement required in the initial state prior factor is provided by the state estimate from the onboard SPHERES extended Kalman filter [27]. A generic covariance matrix with the expected covariance is provided, although the current covariance from the extended Kalman filter could also be used.

Angular Rate and Acceleration Measurements—An independent measurement of motion was provided by gyroscopes attached to the inspector satellite and forces commanded to the thrusters. Body-aligned gyroscopes on the inspector SPHERES satellite were read at 1000 Hz by a digital signal processor. Because of communications restrictions between the SPHERES satellite and the VERTIGO Linux computer, 20 temporally adjacent measurements were averaged to provide 50 Hz gyroscope data to the smoothing-based estimator described herein.

The SPHERES satellites use 12 cold gas fired thrusters for maneuvering; all 12 are connected to the same pressure system. When n_{th} thrusters are open, each thruster provides a thrust force of $F_{th} = 0.94^{n_{th}-1} F^0_{th}$, where F^0_{th} is the thrust force with a single thruster open. If $\mathbf{u}^{th}$ is a 12×1 vector with values of 1 for open thrusters and values of 0 for closed thrusters, then the commanded body force $\mathbf{u}^f$ is given by:

$$\mathbf{u}^f = \mathbf{M}_f \left(0.94^{n_{th}-1}\,\mathbf{u}^{th}\right) \tag{16}$$

$$\mathbf{M}_f = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & -1 & -1 \end{bmatrix}$$

where $\mathbf{M}_f$ is the force mixer matrix. The SPHERES satellite sends the thruster on and off times in milliseconds to the VERTIGO Linux computer. These times are used to create one acceleration "measurement" for every gyroscope measurement (i.e. at a rate of 50 Hz). Batches of angular rate and body force data were collected in between images and input together into the inertial odometry factor.

iSAM2 Implementation— The computational machinery required to implement the smoothing-based estimation algorithm herein was mostly available in the Georgia Tech. Smoothing and Mapping C++ package (GTSAM 4.0) [28]. The centroid of the target was input as a `Point3` object located at its commanded coordinates (generically, this should be set as the origin). The built-in `BearingRangeFactor` was modified slightly to accept a transform between the body and the sensor pose, resulting in the upgraded factor `BearingRangeFactorWithTransform`. The `PoseRotationPrior` factor was used to apply the star tracker measurement to the orientation component of the inspector's pose. The pose $\{{}^W_B\mathbf{R}, {}^W\mathbf{t}_{W\text{to}B.}\}$ in the state vector $\mathbf{x}$ was implemented using `Pose3`, the velocity was implemented using `Vector3`, and the biases were implemented using `imuBias::ConstantBias`. The inertial odometry factor was implemented using the `CombinedImuFactor`, with gravity set to zero and the angular velocity of the world coordinate frame set to accurately reflect the rotation of the ISS every 91.4 minutes from orbital motion (note that in free-space the world coordinate frame would be rotationally fixed to the stars and therefore have zero angular velocity).

Experimental Design

The SPHERES-VERTIGO test platform was used to obtain data for testing the algorithm described herein. One SPHERES satellite without any peripherals was commanded to hold position near the center of the test volume to act as the target object. In certain tests the target was commanded to tumble, and in others it was commanded to maintain a fixed orientation. Another SPHERES satellite, with the attached Linux computer and stereo cameras, was made to circumnavigate the target satellite.

To simulate a representative trajectory for the inspector, a 110×50 m, 91.4 minute free orbit ellipse was scaled down to a 1.1×0.55 m, 5 minute free orbit ellipse to fit within the space constraints of the Japanese Experiment Module (JEM), and time constraints of a test session. The position of the center of mass was commanded to follow this trajectory using a PD controller together with localization from the SPHERES ultrasonic beacon system and onboard extended Kalman filter. In making the best use of the available space in the JEM, the inspector satellite was commanded to go slightly outside the test volume where the performance of ultrasonic beacon localization degrades. All firing times of the thrusters were recorded as part of the experiment. The effects of actual orbital dynamics onboard the ISS were ignored in this experiment (except for the angular velocity of the world frame) because of its short duration and the negligible impact of these effects compared to other significant disturbances such as airflow.

The orientation of the inspector was controlled using the visual measurements of bearing. Here, proportional orientation control was used to keep the centroid of the target in the center of the field of view of the cameras, while a differential term damped the applied torque based on the inspector's body angular rates.

The algorithm presented in this paper was not complete at the time of testing, so the test session was used to collect data for the offline processing presented herein.

4. Results and Discussion

The data presented in this section was collected October 11th, 2015 during SPHERES Test Session 74B. Two tests were run executing the program described in Section 3; in one of the tests the target satellite was tumbling, and in one of the tests the target satellite was static. In both tests, the position of the inspector satellite correctly followed the free orbit elliptical trajectory. However, when briefly leaving the test volume, the performance degraded as expected. Orientation control of the inspector was most challenged at the point of closest approach to the target, where the largest slew rate was required. In the case of the tumbling target test, this caused the inspector to lose visual contact with the target at these locations; this is not believed to be a consequence of the fact that the target was tumbling, as the centroid measurements are nearly unaffected by the rotational motion of a SPHERES satellite. As a result, however, the best uninterrupted data was obtained for the static target test; thus analysis of this data is included herein.

Although images were acquired, processed online, and used for control at approximately 2.34 Hz, only every 5th image was used in the data analysis herein, meaning that the temporal spacing between states is approximately 2.14 s. The depth map was thresholded to ignore anything closer than 0.1 m and anything beyond 0.9 m. The target satellite was placed at ${}^W\mathbf{r}_t = [0\ 0.2\ 0]^T$ m.

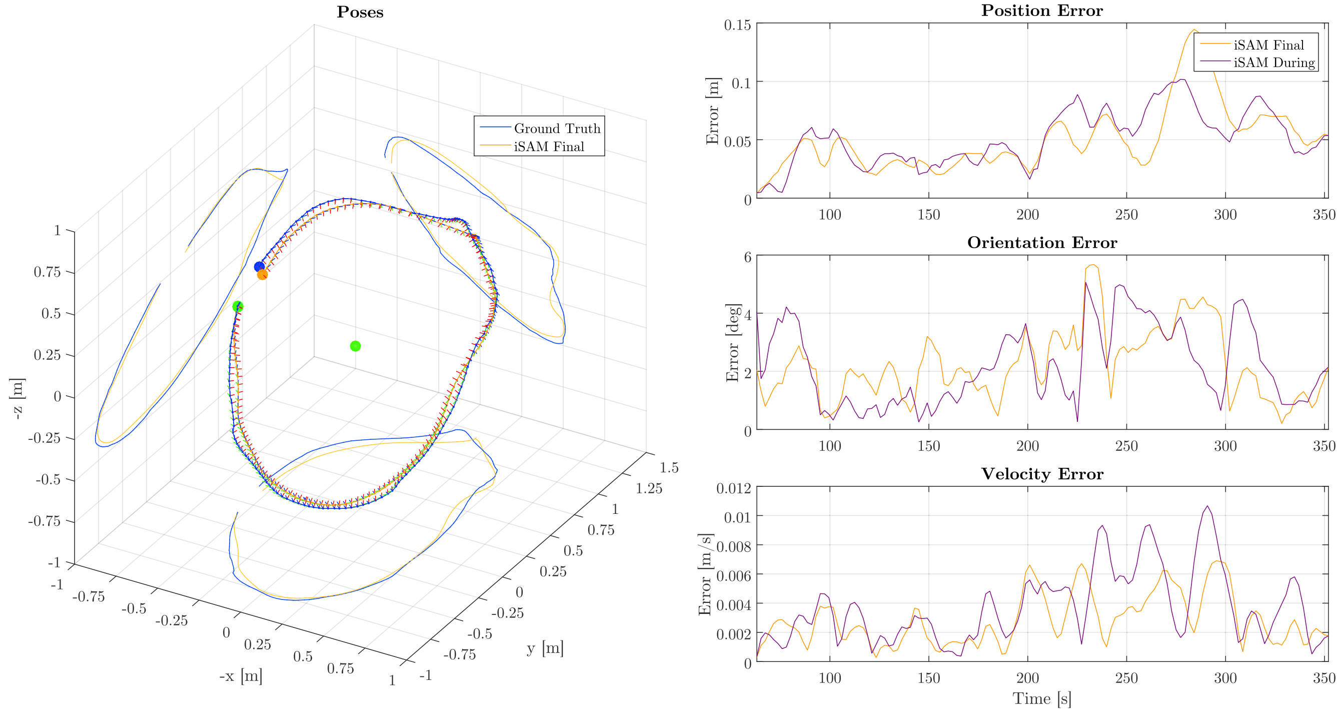


Figure 10. (Left) The ground truth (blue) and final estimated (orange) trajectories of the inspector while it performs an elliptical circumnavigation of the target (green central circle). The trajectories both begin at the green circle on the upper-left. (Right) Error in position, orientation, and velocity for both the final estimate (orange) and the real-time estimate available during the experiment (purple).

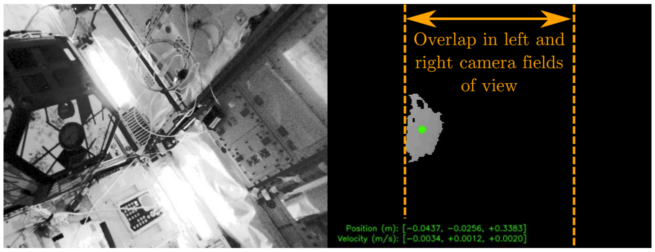
Accuracy

When using iSAM2, the batch problem is solved incrementally, meaning that an estimate of the entire trajectory is available in real time. The state estimates were therefore recorded in two ways. Firstly, the best estimate of current state available during the run was stored. This is the result that would be available to the inspector in real time. Secondly, the best estimate of the entire trajectory at the last timestep is recorded. This is the *maximum a posteriori* estimate of the trajectory, which is useful if mapping is to be performed over the inspection time period. The two series of time estimates are very similar. Figure 10 shows the final estimated poses compared with the visual-inertial odometry "ground truth" (left). The state errors for both the final and real-time results are also shown (right). There is generally reasonable agreement between the results and the "ground truth", with < 0.15 m position error, $< 6°$ orientation error, and $< 0.012\frac{\text{m}}{\text{s}}$ velocity error. The final and real-time errors are similar throughout the run, and converge at the final timestep when they both estimate the final state with the same set of measurements. The average errors in the trajectories are shown in Table 1.

Table 1. Average errors in position, rotation, and velocity.

Avg. Error	iSAM2 Final	iSAM2 During
Position [m]	0.0496	0.0507
Rotation [°]	2.22	2.20
Velocity [m/s]	0.00284	0.00370

There is one section of the trajectory where a notable error develops. From 185 s to 240 s, the assumption that the target object is completely contained in the field of view of the 3D visual sensor is violated. During this time period, a portion of the target satellite is outside the overlapping field of view area in which depth can be sensed (see Figure 11). When this occurs, the range and bearing measurement is based only on the available depth data, which deviates from the target's actual centroid. This emphasizes the fact that accurate pointing and adequate inspection distance are required for this estimation method.



Position (m): [−0.0437, −0.0256, +0.3383]
Velocity (m/s): [−0.0034, +0.0012, +0.0020]

Figure 11. Range and bearing result at 218.9 s. The target satellite is slightly outside the field of view of the left camera (left), and well outside the overlap in the left and right camera field of view (right).

Computational Performance

Computation was performed on an Intel Core i7-3630QM 2.4 GHz laptop on a Linux virtual machine with 8 GB of allocated RAM. The speed of the algorithm is shown in Table 2. The required processing time for this 288.9 s (4.8 min) test was under 16 s, including the image processing time, meaning that this algorithm will able to run in real time on a computer capable of performing image processing in real time. Processing the stereo images into depth maps accounted for the majority of the time. The incremental computational time of the algorithm is not expected to grow over time, since cycles are not introduced in the factor graph and the GTSAM implementation of iSAM2 [7] ensures that at each iteration, only nodes that undergo major changes are updated.

Table 2. Computational performance of the algorithm for the results presented (136 state nodes).

	Total	Per node	vs. Real time	% of total
Image proc. time	15.7 s	0.115 s	18×	98.9%
iSAM2 time	0.18 s	0.0013 s	1625×	1.1%

5. Conclusion

This paper has presented a smoothing-based method of estimating an inspector satellite's trajectory relative to a passive object. The problem was formulated as a factor graph, with measurement information being provided by a range and bearing sensor, a star tracker, a gyroscope, and knowledge of the commanded forces. Performing smoothing instead of filtering adds the convenience of being able to include asynchronous measurements and accommodate sensor outages; the graph formulation also allows mapping of the target to be incorporated and solved in the same architecture. Herein, the GTSAM implementation of incremental smoothing and mapping (iSAM2) was used to accurately and efficiently recover the trajectory of a SPHERES-VERTIGO inspector satellite circumnavigating another SPHERES satellite acting as a passive target object. The computational performance of the algorithm indicates that it can be run on computers capable of onboard real-time image processing.

Acknowledgment

The authors would like to thank Max Yates and Kevin van Hecke for their help in preparing the SPHERES-VERTIGO code, astronauts Kimiya Yui and Scott Kelly for executing the SPHERES Test Session 74B during which the presented data was collected, and the SPHERES teams at NASA AMES and MIT for support during the test session.

References

[1] A. Ellery, J. Kreisel, and B. Sommer, "The case for robotic on-orbit servicing of spacecraft: Spacecraft reliability is a myth," *Acta Astronautica*, vol. 63, no. 5-6, pp. 632–648, 2008.

[2] A. Flores-Abad, O. Ma, K. Pham, and S. Ulrich, "A review of space robotics technologies for on-orbit servicing," *Progress in Aerospace Sciences*, vol. 68, no. 1, pp. 1–26, 2014.

[3] NASA, "Satellite Servicing Capabilities Office." [Online]. Available: http://ssco.gsfc.nasa.gov

[4] O. ATK, "Mission Extension Services," 2016. [Online]. Available: https://www.orbitalatk.com/space-systems/human-space-advanced-systems/mission-extension-services/default.aspx

[5] Astroscale, "ADRAS 1," 2016. [Online]. Available: http://astroscale.com/services/adras-1

[6] ESA, "e.Deorbit." [Online]. Available: http://www.esa.int/Our_Activities/Space_Engineering_Technology/Clean_Space/e.Deorbit

[7] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard, and F. Dellaert, "iSAM2: Incremental Smoothing and Mapping Using the Bayes Tree," *The International Journal of Robotics Research*, vol. 31, no. 2, pp. 216–235, 2012.

[8] H. Klinkrad, "Orbital Debris and Sustainability of Space Operations," in *Handbook of Satellite Applications*. New York, NY: Springer New York, 2013, pp. 1145–1174.

[9] B. R. Sullivan and D. L. Akin, "A survey of serviceable spacecraft failures," *AIAA Space 2001 Conference and Exposition*, 2001.

[10] B. Naasz, D. Zimpfer, R. Barrington, and T. Mulder, "Flight Dynamics and GN&C for Spacecraft Servicing Missions," in *AIAA Guidance, Navigation, and Control Conference*, 2010, pp. 1–16.

[11] D. C. Woffinden and D. K. Geller, "Navigating the Road to Autonomous Orbital Rendezvous," *Journal of Spacecraft and Rockets*, vol. 44, no. 4, pp. 898–909, 2007.

[12] T. M. Davis, T. L. Baker, T. A. Belchak, and W. R. Larsen, "XSS-10 microsatellite flight demonstration program results," in *AIAA/USU Conference on Small Satellites*, 2004.

[13] AFRL, "XSS-11 micro satellite," 2005. [Online]. Available: http://www.kirtland.af.mil/shared/media/document/AFD-070404-108.pdf

[14] R. B. Friend, "Orbital Express Program Summary and Mission Overview," in *SPIE Defense and Security Symposium*, vol. 695803, 2008, pp. 1–11.

[15] M. Strube, R. Henry, E. Skelton, J. V. Eepoel, N. Gill, and R. Mckenna, "Raven: An on-orbit relative navigation demonstration using International Space Station visiting vehicles," in *AAS GN&C Conference*, 2015.

[16] A. Alessandro, "NASA's Restore-L Mission to Refuel Landsat 7, Demonstrate Crosscutting Technologies," 2016. [Online]. Available: https://www.nasa.gov/feature/nasa-s-restore-l-mission-to-refuel-landsat-7-demonstrate-crosscutting-technologies

[17] E. Levin, J. Pearson, and J. Carroll, "Wholesale debris removal from LEO," *Acta Astronautica*, vol. 73, pp. 100–108, 2012.

[18] K. Wormnes, R. Le Letty, L. Summerer, H. Krag, R. Schonenborg, O. Dubois-Matra, E. Luraschi, J. Delaval, and A. Cropp, "ESA technologies for space debris remediation," *6th European Conference on Space Debris*, 2013.

[19] P. Resouces, "Planetary Resources — The Asteroid Mining Company." [Online]. Available: http://www.planetaryresources.com

[20] D. C. Woffinden and D. K. Geller, "Relative Angles-Only Navigation and Pose Estimation For Autonomous Orbital Rendezvous," *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 5, pp. 1455–1469, 2007.

[21] B. W. Barbee, J. R. Carpenter, S. Heatwole, F. L. Markley, M. Moreau, B. J. Naasz, and J. Van Eepoel, "A Guidance and Navigation Strategy for Rendezvous and Proximity Operations with a Noncooperative Spacecraft

in Geosynchronous Orbit," *The Journal of the Astronautical Sciences*, vol. 58, no. 3, pp. 389–408, 2011.

[22] T. J. Steiner, T. M. Brady, and J. A. Hoffman, "Graph-based terrain relative navigation with optimal landmark database selection," *IEEE Aerospace Conference Proceedings*, 2015.

[23] F. Dellaert and M. Kaess, "Square Root SAM: Simultaneous Localization and Mapping via Square Root Information Smoothing," *The International Journal of Robotics Research*, vol. 25, no. 12, pp. 1181–1203, 2006.

[24] B. E. Tweddle and D. W. Miller, "Computer Vision-Based Localization and Mapping of an Unknown, Uncooperative and Spinning Target for Spacecraft Proximity Operations," PhD, Massachusetts Institute of Technology, 2013.

[25] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "On-Manifold Preintegration Theory for Fast and Accurate Visual-Inertial Navigation," *IEEE Transactions on Robotics*, pp. 1–18, 2015. [Online]. Available: http://arxiv.org/abs/1512.02363

[26] A. Geiger, M. Roser, and R. Urtasun, "Efficient large-scale stereo matching," in *Asian Conference on Computer Vision (ACCV)*, 2010.

[27] S. Nolet and D. W. Miller, "Development of a guidance, navigation and control architecture and validation process enabling autonomous docking to a tumbling satellite," Ph.D. dissertation, Massachussetts Institute of Technology, 2007.

[28] F. Dellaert, "Factor Graphs and GTSAM: A Hands-on Introduction," pp. 1–27, 2012.

Biography

Timothy P. Setterfield *received the Bachelors of Science degree in Mechanical Engineering from Queens University, Kingston, ON, Canada, and the Masters of Applied Science degree in Mechanical and Aerospace Engineering from Carleton University, Ottawa, ON, Canada. At Carleton University, he was an integral member of the Kapvik rover team. His prior work experience includes the mechanical design of a miniature broadband seismometer and coordinating graduate student research in micro and hypergravity at the European Space Research and Technology Centre, Noordwijk, the Netherlands. He is currently a PhD Candidate in Aeronautics and Astronautics at the Massachusetts Institude of Technology.*

David Miller *David Miller is the Jerome Hunsaker Professor of Aeronautics and Astronautics at the MIT Department of Aeronautics and Astronautics, having earned his S.B., S.M., and Sc.D. degrees there. He is the director of the Space Systems Laboratory, former NASA Chief Technologist and vice-chair of the Air Force Scientific Advisory Board, and recipient of numerous awards for his multitude of flight systems*

John J. Leonard *John J. Leonard is Samuel C. Collins Professor of Mechanical and Ocean Engineering in the MIT Department of Mechanical Engineering. He is also a member of the MIT Computer Science and Artificial Intelligence Laboratory (CSAIL). His research addresses the problems of navigation and mapping for autonomous mobile robots, developing techniques to handle uncertainty in sensing and actuation. He holds the degrees of B.S.E.E. in Electrical Engineering and Science from the University of Pennsylvania (1987) and D.Phil. in Engineering Science from the University of Oxford (1994). Prof. Leonard joined the MIT faculty in 1996, after five years as a Post-Doctoral Fellow and Research Scientist in the MIT Sea Grant Autonomous Underwater Vehicle (AUV) Laboratory. He is the recipient of an NSF Career Award (1998), an E.T.S. Walton Visitor Award from Science Foundation Ireland (2004), the King-Sun Fu Memorial Best Transactions on Robotics Paper Award (2006), and he is an IEEE Fellow (2014).*

Alvar Saenz-Otero *Alvar Saenz-Otero is the Director of the MIT Aeronautics and Astronautics Department Space Systems Laboratory. He was in the original team that created the SPHERES satellite and led their ISS operations from 2006 to 2014. His research concentrates on development of facilities for technology maturation, space systems engineering, and avionics research. He has PhD in AeroSpace Systems, and MEng and BS degrees in Electrical Engineering and a BS degree in Aero/Astro, all from MIT.*