HARDWARE-IN-THE LOOP TESTS OF AN AUTONOMOUS GN&C SYSTEM FOR ON-ORBIT SERVICING*

Robert Bell, Thomas Morphopoulos, Jon Pollack,
John Collins, James R. Wertz, and Richard E. Van Allen
Microcosm, Inc.

ABSTRACT

Autonomous on-orbit servicing of satellites leads to a significant savings in mission life-cycle costs, especially for expensive satellites and for expensive satellite systems composed of many satellites, such as constellations or close formations. The ability to repair satellites, replace worn components, and resupply critical consumables, such as station-keeping propellant, will lead to increased on-orbit lifetime and lower system costs. Microcosm has recently completed a Phase II SBIR on *Testing of Satellite Servicing and Inspection Elements* for the Air Force Research Laboratory, Space Vehicles Directorate in Albuquerque, NM. In September 2002, Microcosm and Michigan Aerospace Corporation successfully tested the Autonomous Micro-Satellite Docking System (AMDS) in micro-gravity conditions onboard NASA's KC-135 micro-gravity research aircraft. The AMDS was designed for soft-docking capability, tolerance to misalignment, and scalability to the micro-satellite level. These tests successfully demonstrated the functionality and robustness of the AMDS design under a variety of initial conditions. This paper focuses on the results of Phase II tests that were performed at the Naval Research Laboratory (NRL) Spacecraft Robotics Engineering and Controls Lab (SRECL). These tests validated Microcosm's autonomous guidance, navigation and control (GN&C) algorithms with a relative navigation sensor in the loop for satellite servicing and inspection. The NRL simulator is composed of two robotic arms driven by propagated estimates of location and attitude of two independent spacecraft in orbit around the Earth. The simulation models the effect of environmental disturbances on both spacecraft as well as commanded force and torque inputs from the "chaser" spacecraft mock-up flight computer. One arm represents a "chaser" spacecraft and the other arm the "target." The "chaser" spacecraft monitored relative position with an ILRIS-3D LIDAR sensor produced by Optech, Inc., of Toronto, Canada. Microcosm wrote custom software to convert this sensor from a commercial, long-integration time "batch" terrestrial surveying sensor into a 0.25 Hz closed-loop relative navigation sensor. Microcosm also developed and tested relative motion control algorithms. In the SRECL, these algorithms were tested to show spacecraft rendezvous from an initial separation of 15 m down to 4 m, but not actual complete hard docking, due to test facility limitations at the time of the tests. The results of the tests presented in this paper show that the ILRIS-3D sensor provided data with sufficient accuracy to be used for relative navigation. In addition, the guidance laws were tested with an actual navigation sensor in the loop and a modeled thruster-based spacecraft attitude and navigation control system.

LIST OF ACRONYMS

6-DOF	6-Degrees of Freedom
ACS	Attitude Control System
CAM	Collision Avoidance Maneuver
CW	Clohessy-Wiltshire
FOV	Field-of-View
GN&C	Guidance, Navigation, and Control
GTA	Greatest Tangent Arc
HITL	Hardware-in-the-Loop
I/O	Input/Output
ISS	International Space Station
LEO	Low Earth Orbit
LIDAR	Light Detection and Ranging
LRDI	Laser Range Doppler Imager
MAC	Michigan Aerospace Corporation
NRL	Naval Research Laboratory
PD	Proportional-Derivative
PVC	Polyvinyl Chloride
RSW	Radial, In-track, Cross-track coordinate system
SITL	Software-in-the-Loop
SRECL	Spacecraft Robotics Engineering Controls Laboratory
VGS	Video Guidance Sensor
ZCS	Zero Closing Speed

LIST OF SYMBOLS

β	ballistic coefficient
δ_{ij}	Kronecker delta
ϕ	pitch angle (ECI to body)
η	nadir vector

θ	roll angle (ECI to body)
ρ	atmospheric density
ψ	yaw angle (ECI to body)
$\boldsymbol{\omega}$	body angular rate vector
a	semi-major axis
e	eccentricity
f	applied acceleration
$\mathbf{I}$	mass moment of inertia
i	inclination
m	mass
n	mean orbit motion
P	orbital period ($2\pi/n$)
R	distance along radius vector
R-bar	approach along the radius vector
S	distance along velocity vector
V-bar	approach along the velocity vector
W	distance along angular momentum vector
x	distance along body-fixed roll axis
y	distance along body-fixed pitch axis
z	distance along body-fixed yaw axis

1.0 INTRODUCTION

Studies have shown that on-orbit servicing is particularly cost effective for servicing large constellations (Dellacamera et al. [1], Wertz et al. [2]) as well as single satellites that are very costly, such as the International Space Station (ISS) [3]. Wertz et al. [2] performed trade studies on servicing a large low Earth orbit (LEO) communications constellation and found that reuse of the servicing asset was a key factor in reducing the life-cycle cost, even when the space assets being serviced have a comparatively inexpensive replacement cost. Many studies have focused on the potential for on-orbit servicing to decrease life-cycle costs by increasing the useful life of a space asset, especially by refueling. In many cases, however, fuel is not the limiting factor in the life span of a spacecraft. Equipment lifetime also has a significant impact on limiting the useful life of a spacecraft. Though equipment/component replacement is technically possible, it is generally far more difficult than simply refueling. However, a servicing architecture capable only of refueling can still provide a benefit by enabling an asset to launch with very little fuel, thereby increasing the payload mass at launch. Turner [4] makes this argument for geosynchronous satellites, and states that per satellite revenue can be increased through on-orbit servicing.

There are several critical components to a low cost on-orbit servicing architecture, including: a docking mechanism, a relative navigation sensor, and the ability to operate autonomously. A separate portion of this research was devoted to the micro-gravity tests of the Michigan Aerospace Corporation (MAC) autonomous micro-satellite docking system (AMDS). Details of this research are described in Tchoryk, et al. [5]. The AMDS was specifically designed for small satellites, but the design can be scaled up for larger satellites as well [6]. The AMDS is tolerant to misalignment (± 10° in pitch/yaw and ± 180° in roll) and was shown to dock with separations of up to 10 cm, which reduces the requirements on the trajectory control and attitude control systems. Several relative navigation sensors are either available or are being developed for various programs, including the Video Guidance Sensor (VGS) [7], the Laser Range Doppler Imager (LRDI) [8], and the RELAVIS sensor [7], to name a few. Lastly, a key component to cost-effective satellite servicing will likely be autonomy. Autonomous operations have the potential to reduce or eliminate the number of required ground support personnel and the communications equipment and links necessary to support them. Autonomy would be a requirement for inter-planetary missions, where the communications delay is an insurmountable barrier to pilot-in-the-loop operations.

This study has moved the technology needed for on-orbit servicing forward significantly.[†] Previously, a docking mechanism was successfully tested aboard NASA's KC-135 micro-gravity research aircraft [5]. This paper addresses the hardware-in-the-loop (HITL) and software-in-the-loop (SITL) testing of a relative navigation sensor and an autonomous GN&C system. The Naval Research Laboratory (NRL) Spacecraft Robotics Engineering and Controls Lab (SRECL) provided two independent 6-degree-of-freedom (6-DOF) robots for this testing. The "target" spacecraft was modeled by a half-meter by half-meter flat piece of polyvinyl chloride (PVC) attached to one of the robots. The other robot held the "chaser" spacecraft, which consisted of a relative navigation sensor and a "flight" computer. Microcosm wrote custom software to use a commercially available Optech ILRIS-3D light detection and ranging (LIDAR) surveying sensor [10] to work as a relative position sensor in a closed loop. In addition to a command and telemetry interface to the ILRIS-3D sensor, the flight computer software had a Kalman filter to process the sensor data, guidance laws to bring two spacecraft together on-orbit, and GN&C control software. The test results showed that the guidance laws worked as expected, and that the ILRIS-3D sensor provided data accurate enough for use as a relative navigation sensor.

[†] This work was funded through a Phase II SBIR, contract number F29601-02-C-0007, sponsored by the U.S. Air Force Research Laboratory, Space Vehicles Directorate, with matching funds provided by Spectrum Astro, Inc. The authors wish to gratefully acknowledge the generous support of these organizations that made this research possible.

2.0 GN&C SYSTEM DESIGN

2.1 Guidance Laws

One of the primary goals of this study was to develop an autonomous guidance table generator that could be used as part of a feedback control system for rendezvous and docking applications. Two guidance laws were developed in order to accomplish this goal: the greatest tangent arc (GTA) law, and the zero closing speed (ZCS) law. Both were developed from the Clohessy-Wiltshire (CW) equations of motion [11], viz.

$$\begin{aligned} \ddot{R} - 2n\dot{S} - 3n^2 R &= f_R \\ \ddot{S} + 2n\dot{R} &= f_S \\ \ddot{W} + n^2 W &= f_W \end{aligned} \tag{1}$$

where R, S, and W, are the radial, along-track, and cross-track displacements, respectively, of the chaser vehicle relative to the target, n is the mean motion, and f is the forcing term. The homogeneous solution to Eq. 1 (i.e. $\mathbf{f} = \mathbf{0}$) is

$$\begin{aligned} R(t) &= R_o(4 - 3\cos nt) + \dot{R}_o \frac{\sin nt}{n} + \dot{S}_o \frac{2 - 2\cos nt}{n} \\ S(t) &= R_o(6\sin nt - 6nt) + S_o + \dot{R}_o \frac{2\cos nt - 2}{n} + \dot{S}_o \frac{4\sin nt - 3nt}{n} \\ W(t) &= W_o \cos nt + \dot{W}_o \frac{\sin nt}{n} \\ \dot{R}(t) &= R_o 3n \sin nt + \dot{R}_o \cos nt + \dot{S}_o 2\sin nt \\ \dot{S}(t) &= R_o(6n\cos nt - 6n) - \dot{R}_o 2\sin nt + \dot{S}_o(4\cos nt - 3) \\ \dot{W}(t) &= -W_o n \sin nt + \dot{W}_o \cos nt \end{aligned} \tag{2}$$

Eq. 2 is also applicable to impulsive thrust maneuvers. Both GTA and ZCS guidance laws are applicable to R-bar and V-bar transfers.

2.1.1 Zero Closing Speed (ZCS)

The ZCS guidance law seeks to execute a series of rendezvous maneuvers that (a) minimize the possibility of collision by forcing the closing speed at closest approach to be zero, (b) minimize the possibility of thruster plume impingement at closest approach, and (c) keep the target within a specified field-of-view (FOV). Given an initial state, the law is defined by a set of final conditions, viz.

$$\begin{aligned} R(t_f) &= R_f \\ S(t_f) = W(t_f) &= \dot{R}(t_f) = 0 \end{aligned} \tag{3}$$

for the R-bar approach, and

$$\begin{aligned} R(t_f) &= 0 \\ S(t_f) &= S_f \\ W(t_f) &= \dot{S}(t_f) = 0 \end{aligned} \tag{4}$$

for the V-bar approach. The FOV constraint is given by

$$S(t)^2 + W(t)^2 \le R(t)^2 \tan^2 \lambda_{\max} \quad \text{for all } t \in [0, t_f] \tag{5}$$

for the R-bar approach, and

$$R(t)^2 + W(t)^2 \le S(t)^2 \tan^2 \lambda_{\max} \quad \text{for all } t \in [0, t_f] \tag{6}$$

for the V-bar approach. Given the initial relative position ($R_o, S_o, \mathrm{W_o}$) of the chaser with respect to the target, Eq. 2 can be solved for the initial velocity ($\dot{R}_o, \dot{\mathrm{S}}_o, \dot{\mathrm{W}}_o$) and the time of flight (t_f), subject to the final conditions (3) or (4) and FOV constraint (5) or (6), for R-bar or V-bar maneuvers, respectively. The solution method is an iterative one that requires multiple maneuvers or "hops" to complete. The solution for R-bar and V-bar maneuvers will be outlined separately.

2.1.1.1 R-bar Maneuvers

Solving (2), subject to final conditions (3), yields the following solution:

$$\begin{aligned} 3R_o - \left(\frac{2S_o + 12R_o\left[\tan nt_f - nt_f\right]}{4\tan nt_f - 3nt_f} \right) - \frac{(R_o - R_f)\cos nt_f}{1 - \cos nt_f} &= 0 \\ \frac{\dot{S}_o}{n} = -\frac{S_o + 6R_o\left(\tan nt_f - nt_f\right)}{4\tan nt_f - 3nt_f} \\ \frac{\dot{R}_o}{n} = -\left(3R_o + \frac{2\dot{S}_o}{n} \right) \tan nt_f \\ \frac{\dot{W}_o}{n} = -W_o \cot nt_f \end{aligned} \tag{7}$$

The first equation in (7) must be solved iteratively for t_f. The solution determined from (7) may not satisfy the FOV constraint (5). In order to do so, one must adjust the final radial position, R_f, until (5) is satisfied, which will likely lead to a solution that places the chaser further from the target than desired. Thus, another maneuver may be required to reach the desired R_f. The final conditions of the first maneuver are used as the initial conditions for the second maneuver, and a new solution is developed. This process is repeated until the desired radial location is finally reached.

Such a series of maneuvers is illustrated in Figure 1, which required two hops to achieve the desired final radial position. Each hop has the half-inverted teardrop shape that is characteristic of R-bar ZCS maneuvers. The total time of flight will be in direct proportion to the number of hops required to complete the maneuver. As the chaser nears the target, shorter hops are required to remain within the FOV constraint, which further increases the total flight time. Therefore, the fewer the number of hops, the better. For this reason, the equality in the FOV constraint (5) is typically used unless the desired final location can be achieved with only one hop. Notice how the chaser

trajectory goes right up to the outer edge of the FOV in Figure 1. In a real application, care must be taken when selecting the FOV constraint such that there is always margin between the constraint and the actual FOV limit of the relative navigation sensor. The sensor FOV should be sufficiently larger than the constraint to ensure that the target remains visible to the sensor at all times. In addition, some sensors may perform at sub-optimal levels if the target is at the periphery of the sensor FOV.

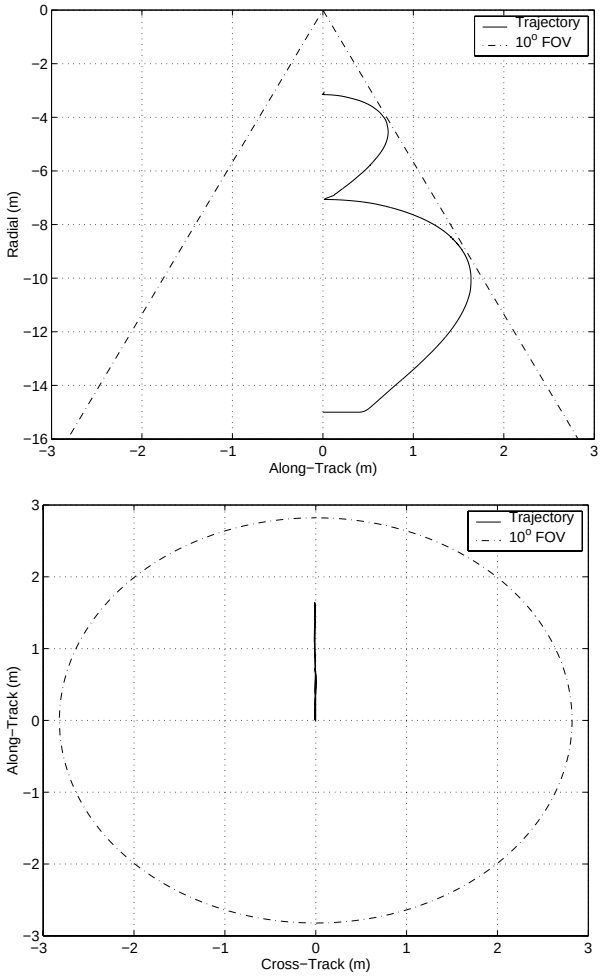


Figure 1. Example R-bar ZCS Maneuver, Chaser Trajectory Relative to the Target in RSW Coordinates

It is possible to execute an R-bar maneuver in which both the final along-track and radial velocities are zero. These thrust-free trajectories, described by Waltz and Meissinger [12], differ from the R-bar ZCS trajectories in that they are only valid for certain initial conditions, viz.

$$\begin{aligned} S_o &= 6R_f(\sin nt_f - nt_f) \\ R_o &= R_f(4-3\cos nt_f) \end{aligned} \tag{8}$$

where $S_f = 0$. In contrast, there are no restrictions on the initial conditions for an R-bar ZCS maneuver. Another advantage of ZCS guidance is that it seeks to keep the target within a given FOV, which is not possible with the zero-velocity approach. However, the price for this convenience is that an R-bar ZCS maneuver will require a terminal thrust in the along-track direction.

2.1.1.2 *V-bar Maneuvers*

The solution for V-bar maneuvers is comparatively simple. Solving (2), subject to final conditions (4), yields the following solutions. For $R_o \neq 0$,

$$\begin{aligned} nt_f &= \mathrm{mod}\left(2\arctan\left[\frac{S_o - S_f}{2R_o}\right] + 2\pi, 2\pi\right) \\ \frac{\dot{R}_o}{n} &= -R_o \cot nt_f \\ \frac{\dot{S}_o}{n} &= -2R_o \\ \frac{\dot{W}_o}{n} &= -W_o \cot nt_f \end{aligned} \tag{9}$$

and for $R_o = W_o = 0$,

$$\begin{aligned} nt_f &= \pi \\ \frac{\dot{R}_o}{n} &= -\frac{S_o - S_f}{4} \\ \dot{S}_o &= \dot{W}_o = 0 \end{aligned} \tag{10}$$

As in the case for R-bar, the solution of (8) or (9) may not satisfy the FOV constraint (6), which requires that the final along-track position, S_f, be adjusted to satisfy (6). Multiple hops may be required to reach the final desired along-track position. An example of a V-bar ZCS maneuver is shown in Figure 2. Each hop has the half 2-to-1 ellipse shape that is characteristic of these maneuvers.

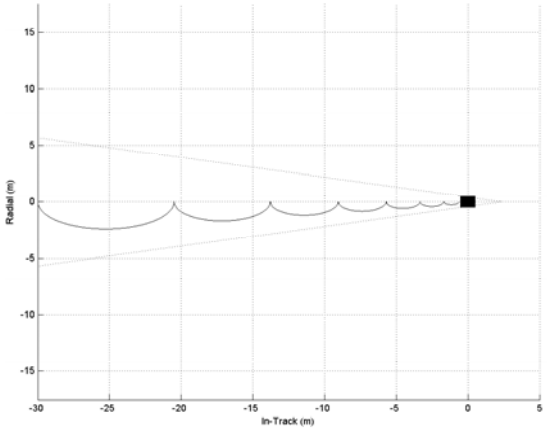


Figure 2. Example of a Positive, V-bar, ZCS Maneuver. The trajectory is indicated by a solid line. The dotted line indicates the FOV constraint. The target is shown at the origin.

Notice the time of flight for V-bar transfers where Ro = 0, which corresponds to a hop with a transfer time of half an orbit period. Thus, for maneuvers requiring several hops, the total transfer time may be multiples of an orbit period. For time-critical applications, V-bar ZCS guidance should be avoided, particularly at high altitudes, where the orbit period is very large. Regardless of the target altitude, the example shown in Figure 2 would have taken 3.5 orbit periods to complete.

2.1.1.3 Characteristics of ZCS Guidance

There are two primary benefits to ZCS guidance over GTA guidance. Because the closing speed at closest approach is 0, there is a reduced risk of a collision. Therefore, safety is a primary benefit, which is particularly the case for an R-bar approach in which the altitude difference between the chaser and target imparts a secular drift velocity in the along-track direction (see Eq. 2 for evidence of this). Another benefit of ZCS maneuvers is that plume-impingement effects are minimized when the chaser is near the target. Since the closing speed is 0 at the point of closest approach, the chaser is moving perpendicular to the target at this point (i.e., in the radial direction for V-bar approaches, and in the along-track direction for R-bar approaches). Thus, braking thrusts are at worst delivered parallel to the target face, but never toward it. These maneuvers also offer some measure of fail-safe protection, should a braking maneuver fail to be applied. Figure 3 and Figure 4 show sample R-bar and V-bar terminal trajectories, using the ZCS guidance law.

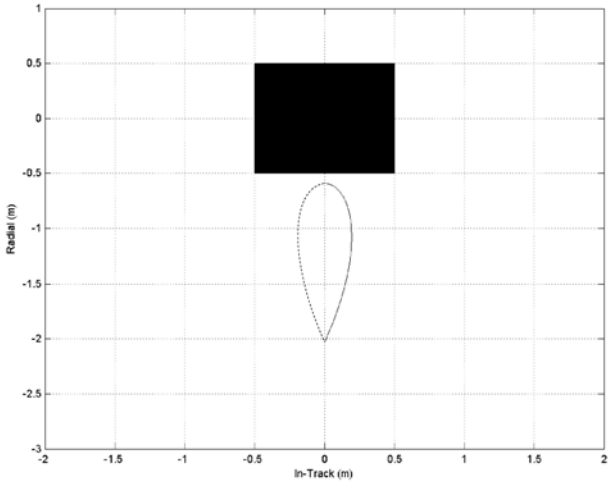


Figure 3. Example of a Terminal R-bar, ZCS Maneuver. The target is shown in black. The nominal chaser trajectory is shown with a solid line. The dotted line indicates the trajectory that would occur if a braking maneuver were not applied at the point of closest approach to the target. Motion is counter-clockwise from the bottom.

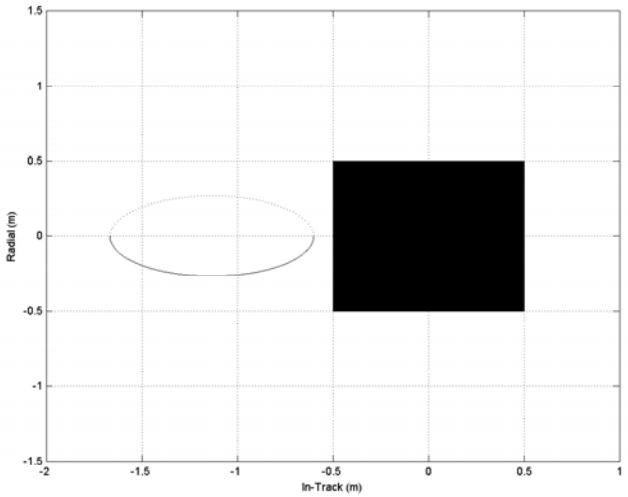


Figure 4. Example of a Terminal V-bar, ZCS Maneuver. The target is shown in black. The nominal chaser trajectory is shown with a solid line. The dotted line indicates the trajectory that would occur if a braking maneuver were not applied at the point of closest approach to the target. Motion is counter-clockwise from the left and is periodic with the same period as the target orbit.

For Figure 3, the motion begins at the bottom of the inverted teardrop and proceeds counter-clockwise towards the target. Nominally, a braking maneuver would be applied at the top of the trajectory that would be directed in the positive along-track direction (i.e., the plume would be directed in the negative along-track direction), and would bring the chaser to a stop 10 cm from the target face. Should the braking maneuver fail, the chaser would continue along the dotted portion of the trajectory back to its starting location. At this point, it would be possible to re-initiate the maneuver. If it is not possible to re-initiate the maneuver, the chaser would continue along an arc that would take it in front of and underneath the target, never to return, which illustrates the passive fail-safe features of this maneuver. At no time is a collision avoidance maneuver (CAM) required, should a thruster failure or sensor outage occur. Similar passive fail-safe features can be observed with the V-bar terminal trajectory, shown in Figure 4. The maneuver is initiated at the far left and proceeds counter-clockwise toward the target. Nominally, a braking maneuver would be applied at the far right of the trajectory (i.e., at the point of closest approach), which would bring the chaser to rest 10 cm from the target face. If the braking maneuver should fail, the chaser would continue along the dotted line and eventually end up at the starting point, one orbit period later. At this point, the maneuver can be re-initiated. If re-initiating the maneuver fails, the chaser will continue along this trajectory, ad infinitum. Unlike the R-bar maneuver, the V-bar maneuver is periodic, under Keplerian motion. However, perturbations (particularly differ-ential drag) could disrupt this periodicity and increase the potential for collision. The redeeming feature is that the motion is much slower than in the R-bar case.

To traverse the entire 2-to-1 ellipse, requires a full orbit period. Thus, a lot of time is potentially available for troubleshooting or executing a CAM.

2.1.2 Greatest Tangent Arc (GTA)

The GTA guidance law seeks to execute a single, two-burn rendezvous maneuver that keeps the target within a specified FOV. Given an initial state, the law is defined by a set of final conditions, viz.

$$\begin{aligned} R(t_f) &= R_f \\ S(t_f) &= W(t_f) = 0 \end{aligned} \tag{11}$$

for R-bar transfers, and

$$\begin{aligned} R(t_f) &= 0 \\ S(t_f) &= S_f \\ W(t_f) &= 0 \end{aligned} \tag{12}$$

for V-bar transfers. The same FOV constraints apply for GTA guidance, as did for ZCS guidance. Namely, (5) and (6) for R-bar and V-bar transfers, respectively. Given the initial position, (2) can be solved subject to final conditions (10) and (11), and FOV constraints (5) and (6), for R-bar and V-bar transfers, respectively. The solution for both R-bar and V-bar is given by

$$\begin{aligned} \frac{\dot{R}_o}{n} &= \frac{R^*(4\sin nt_f - 3nt_f) + 2S^*(\cos nt_f - 1)}{8(1-\cos nt_f) - 3nt_f \sin nt_f} \\ \frac{\dot{S}_o}{n} &= \frac{2R^*(1-\cos nt_f) + S^* \sin nt_f}{8(1-\cos nt_f) - 3nt_f \sin nt_f} \\ \frac{\dot{W}_o}{n} &= -W_o \cot nt_f \end{aligned} \tag{13}$$

where

$$\begin{aligned} R^* &= R_f - R_o(4 - 3\cos nt_f) \\ S^* &= S_f - S_o - 6R_o(\sin nt_f - nt_f) \end{aligned}$$

where the time of flight, t_f, must be chosen to satisfy the FOV constraint. Typically, the closing speed, and therefore the ΔV requirements, increase inversely proportional to the time-of-flight. Thus, t_f is chosen such that the equality in either (5) or (6) is satisfied, in order to minimize the ΔV requirements and the speed with which the chaser approaches the target.

Figure 8 in Section 4.1 show an example of a V-bar GTA maneuver. The execution time of this maneuver was under 300 s total, which illustrates the only advantage of GTA guidance over ZCS guidance. This speed comes at a price. At the point of closest approach, the target is moving toward the chaser, sometimes with considerable speed. For the example shown in Figure 8 (Section 4.0), the closing speed is between 3.5 and 4.5 cm/s. Thus, the potential for collision and plume-impingement is greater than for ZCS guidance. For this reason, GTA guidance is not recommended for terminal approaches. GTA guidance is much more useful to close a large separation gap very quickly. A typical rendezvous engagement may require a mixture of GTA and ZCS guidance. When the target and chaser are separated by large distances (> 100 m), GTA guidance can be used to quickly maneuver the chaser within several meters of the target. Once in proximity of the target, ZCS guidance can be used to perform the final docking maneuvers. However, if the intention is merely to intercept the target, then GTA guidance would be most effective.

2.2 Navigation System

2.2.1 Target Tracking Software for ILRIS-3D LIDAR Sensor

The Optech, ILRIS-3D LIDAR sensor [10] was designed to stare at a target for hours while capturing vast quantities of survey data for batch processing. Microcosm wrote software to use the Optech sensor real-time in a relative navigation control loop. On each cycle, a small set of sensor data was processed to determine the size, range, and centroid of a target. At the end of each cycle, a new command was sent to the sensor to continue tracking the target. The software bypassed Optech's data processing software and interfaced directly to their low level hardware drivers. Although there are a variety of options for scan pattern, the sensor was operated only with a step-stare pattern. The scan area was a square cone defined by the upper left and lower right corner points. For the purposes of the experiment, the target consisted of a square PVC plate, with dimensions 50.8 × 50.8 cm.

Initial acquisition in a lab was complicated by the close proximity of a floor, ceiling, walls, and objects that normally would not be so close to the target on orbit. For our tests, the simplifying assumption was made that there was some initial knowledge of the target's position. A wide bundle of laser shots was commanded in the general direction of the target. The data within a user-defined "expected" range was extracted then averaged to calculate a value for target range. The four edges of the square target were determined by averaging n points, within the expected range, with the largest/smallest values for azimuth and elevation, where n is a user-defined value. A larger n would reduce noise, but increase the risk of miscalculating the location of the edge. For all experiments, we set n = 4. These edge calculations were then used to determine the target centroid and size. Assuming that the target rotates about its centroid, then a rotation along the target normal will increase the apparent target size, but will not change the range or centroid. Likewise, tilting the target surface about its centroid will decrease the apparent size, but will

not change the range nor centroid value. These sensitivities have no effect on the rest of the system, as only the range and centroid values were used in the control laws. After acquisition, the "expected" range was tightened around the prior range reading, plus-or-minus margin for relative motion and target tilt. The above process for determining range, centroid, and size was otherwise unchanged.

On every control cycle the centroid and range information was passed to the Kalman filter while a new scan bundle command was calculated and sent to the Optech sensor to track the target. The user can decide whether to command a bundle with a fixed size or one based on the calculated target size, plus a user-defined margin. In the tests that were conducted, a fixed bundle size was used that was large enough to include the target and expected relative motion. The software automatically increased the commanded bundle angles as the target range decreased. Cycle time was maintained by decreasing shot density as the commanded bundle angles increased. The software took into consideration the blurring caused by target motion. The range and centroid reflected the average location of the target during the scan. A Kalman filter was used to predict forward and mitigate effects of this "motion" delay, as well as delays inherent to the sensor hardware drivers.

We used target centroid as the target tracking point and the data selection criteria was purely based on range. On orbit, the target spacecraft should be the next nearest object so this method should work well for separation distances that are large relative to target spacecraft size. The tracking software will have to be further refined to be useful for ISS type inspections or for target tracking points more specific than simply the spacecraft centroid. In the lab, we found that the walls and background objects could confuse the software. If an object was in the target acquisition window and within the initial assumed target range, it would throw off the centroid calculation and could cause the acquisition to fail. The algorithm was less sensitive when tracking, though an extraneous object within 0.2 m of the target edge and within 0.2 m of the target range would throw off the tracking centroid calculation. Improving the target identification algorithm beyond a basic range check would reduce or eliminate this sensitivity. To work around any lab-induced disturbances, the target was placed on a 0.5 m rod and only scenarios were run that did not move the target within 0.5 m of any wall, the floor, or the ceiling.

A 0.5 by 0.5 m target was tracked successfully from a range of 15 m down to 4 m using a 4-second cycle time. The commanded bundle included about 1,600 shots, which took the sensor only about 0.8 seconds to execute. The majority of the 4-second cycle time was required to accommodate sensor hardware delays and slow data interfaces. It is believed that a major improvement in cycle time could come from optimizing the Optech hardware drivers for use in a real-time mode, rather than in a batch mode. The minimum bundle density was driven by the maximum mirror rate inside the sensor. For a 1,600 shot bundle, this density limit was met when a 0.5 by 0.5 m target came within 2 to 3 m of the sensor. Increasing the cycle time, optimizing sensor drivers, or using a smaller target would allow the sensor to be used at a closer range.

2.2.2 Kalman Filter

An extended Kalman filter (EKF) was created to filter the position measurements provided by the ILRIS-3D and to provide a high quality estimate of the velocities, which were not measured directly. The EKF differs from the linearized Kalman filter in that the state estimates are propagated using the full non-linear equations of motion, while the state correction and error covariance are determined from the linearized dynamics. For the NRL simulations, the non-homogeneous Clohessy-Wiltshire equations (1) were used to propagate the state estimates. Defining the state vector as

$$\vec{\mathbf{X}} = \left[R\ S\ W\ \dot{R}\ \dot{S}\ \dot{Z}\ n\right]^{\mathrm{T}}, \tag{13}$$

(1) can be rewritten as

$$\dot{\vec{\mathbf{X}}} = \mathbf{A}\vec{\mathbf{X}} + \vec{\mathbf{U}}$$

where

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 3n^2 & 0 & 0 & 0 & 2n & 0 & 0 \\ 0 & 0 & 0 & -2n & 0 & 0 & 0 \\ 0 & 0 & -n^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{14}$$

and

$$\vec{\mathbf{U}} = \left[0 \quad 0 \quad 0 \quad f_R \quad f_S \quad f_W \quad 0\right]^{\mathrm{T}} \tag{15}$$

By examining how a small perturbation, $\delta\vec{\mathbf{X}}$, affects the system dynamics, an expression can be derived for $\delta\dot{\vec{\mathbf{X}}}$. Using (13),

$$\begin{aligned} \frac{d}{dt}\left(\vec{\mathbf{X}} + \delta\vec{\mathbf{X}}\right) &= \mathbf{A}(\vec{\mathbf{X}} + \delta\vec{\mathbf{X}}) + \vec{\mathbf{U}} \\ \dot{\vec{\mathbf{X}}} + \delta\dot{\vec{\mathbf{X}}} &= \mathbf{A}\vec{\mathbf{X}} + \vec{\mathbf{U}} + \mathbf{A}\delta\vec{\mathbf{X}} \end{aligned} \tag{16}$$

From the definition of $\dot{\vec{\mathbf{X}}}$,

$$\delta\dot{\vec{\mathbf{X}}} = \mathbf{A}\delta\vec{\mathbf{X}} \tag{17}$$

Equation 15 has the solution

$$\delta\vec{\mathbf{X}} = \mathbf{\Phi}(t, t_o)\delta\vec{\mathbf{X}}_o \tag{18}$$

where,

$$\mathbf{\Phi}(t, t_o) = e^{\mathbf{A}(t-t_o)} \approx \begin{bmatrix} 1+\frac{3}{2}\theta^2 & 0 & 0 & t-t_o & \theta(t-t_o) & 0 & 0 \\ 0 & 1 & 0 & -\theta(t-t_o) & t-t_o & 0 & 0 \\ 0 & 0 & 1-\frac{1}{2}\theta^2 & 0 & 0 & t-t_o & 0 \\ 3n\theta & 0 & 0 & 1-\frac{1}{2}\theta^2 & 2\theta & 0 & 0 \\ -3n\theta^2 & 0 & 0 & -2\theta & 1-2\theta & 0 & 0 \\ 0 & 0 & -n\theta & 0 & 0 & 1-\frac{1}{2}\theta^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{19}$$

with $\theta = n(t - t_o)$. Eq. 16 is valid for small θ (i.e., for small segments of the orbit or for times small compared to the orbit period).

The Kalman filter must be initialized with a state vector estimate, $\hat{\vec{\mathbf{X}}}_n$, and covariance matrix, $\hat{\mathbf{P}}_n$, at the nth time step (The $\wedge$ symbol is used to indicate an estimated quantity). The state vector estimate at the next time step, $\hat{\vec{\mathbf{X}}}_{n+1}$, is determined by integrating (13) with $\hat{\vec{\mathbf{X}}}_n$ as the initial condition. $\hat{\mathbf{P}}_{n+1}$ is determined as follows:

$$\hat{\mathbf{P}}_{n+1} = \mathbf{\Phi}(t_{n+1}, t_n)\mathbf{P}_n\mathbf{\Phi}^{\mathrm{T}}(t_{n+1}, t_n) + \mathbf{Q} \tag{20}$$

where $\mathbf{Q}$ is the 2nd moment of process noise, or the process noise matrix. $\mathbf{Q}$ can be interpreted as the statistical variance of the unmodeled dynamics (the processes ignored in developing the equations of motion). For the current problem, the primary ignored processes are non-linearity and eccentricity. Recall that the CW equations are linearized equations that are valid for two spacecraft in close proximity with small eccentricities. Q is often quite difficult to model analytically and, for the purposes of this study, is left as a model tuning parameter. The corrected state vector and covariance matrix at the next time step are then calculated as follows:

$$\begin{aligned} \hat{\vec{\mathbf{X}}}_{n+1} &= \hat{\vec{\mathbf{X}}}_{n+1} + \mathbf{K}_{n+1}(\vec{\mathbf{Z}}_{n+1} - \mathbf{H}\hat{\vec{\mathbf{X}}}_{n+1}) \\ \mathbf{P}_{n+1} &= (\mathbf{I} - \mathbf{K}_{n+1}\mathbf{H})\hat{\mathbf{P}}_{n+1}(\mathbf{I} - \mathbf{K}_{n+1}\mathbf{H})^{\mathrm{T}} + \mathbf{K}_{n+1}\mathbf{R}\mathbf{K}^{\mathrm{T}}{}_{n+1} \end{aligned} \tag{21}$$

where $\vec{\mathbf{Z}}_{n+1}$ are the observations, or measured quantities, viz.

$$\vec{\mathbf{Z}}_{n+1} = \begin{bmatrix} \alpha_{n+1} & \beta_{n+1} & \chi_{n+1} \end{bmatrix}^{\mathrm{T}} \tag{22}$$

and the Kalman gain matrix, $\mathbf{K}$, is given by,

$$\mathbf{K}_{n+1} = \hat{\mathbf{P}}_{n+1}\mathbf{H}^{\mathrm{T}}{}_{n+1}(\mathbf{H}_{n+1}\hat{\mathbf{P}}_{n+1}\mathbf{H}^{\mathrm{T}}{}_{n+1} + \mathbf{R}_{n+1})^{-1} \tag{23}$$

The observation matrix, $\mathbf{H}$, is equal to the partial derivatives of the observations with respect to the state vector quantities, viz.

$$\mathbf{H} = \frac{\partial\vec{\mathbf{Z}}_{n+1}}{\partial\vec{\mathbf{X}}_{n+1}} \tag{24}$$

For the current study, the measurement set, α, β, and χ, represent the distance of the target from the chaser in the body roll, pitch, and yaw axis, respectively. Therefore, H is simply the transformation matrix from the RSW frame to the spacecraft body frame. Lastly, R is a weighting matrix (that is constant in this case) that consists of the known error variances for the given measurements, i.e.,

$$\mathbf{R} = \begin{bmatrix} \sigma_\alpha^2 & 0 & 0 \\ 0 & \sigma_\beta^2 & 0 \\ 0 & 0 & \sigma_\chi^2 \end{bmatrix} \tag{25}$$

For the ILRIS-3D, the stated measurement accuracy is 1 cm. Experiments conducted at NRL indicate that this number is a 3σ value when the sensor is at rest (i.e., $\sigma \approx 3.3$ mm). This accuracy degrades somewhat when the sensor is in motion, which primarily is a function of how the sensor collects data from the scan window (left-to-right, bottom-to-top) and the overall speed of the scan. The tracking software tries to keep 1,000 points in the scan window, which corresponds to 0.5 seconds of data. If the target motion is large during the scan interval, the range image will be distorted, thereby reducing the ranging accuracy of the sensor. To accommodate these errors, it has been assumed that the range data has a 1σ error of 1 cm in all directions.

2.3 Attitude and Navigation Control Software

The main goal of our testing was to verify the GTA and ZCS guidance laws and determine how well they would work in a HITL test. The thruster model was sized so that the trajectory command would be implemented over a few seconds rather than instantaneously. Even with limited and quantized thrust, the chaser closely followed the projected profile. The resultant error provided an initial condition for the navigation control law to take out. A limit cycle from the minimum thruster impulse provided relative motion due to chaser attitude.

The chaser spacecraft had a 0.25 Hz thruster based Proportional-Derivative (PD) control system for attitude and navigation control. Attitude sensors and attitude performance were not part of this test, so angle positions and rates came directly from the NRL simulation, with no noise or bias added to them. A guidance table was generated semi-autonomously in real-time, based upon the desired FOV constraint, approach path (V-bar or R-bar), and the guidance law to be used (GTA or ZCS). Based upon the current relative position and velocity of the two spacecraft, the guidance table generator would compute the

desired relative position, velocity, and ΔV for each time step of a given rendezvous. Navigation errors were the result of differencing the guidance table estimates of desired relative position and velocity from the Kalman filtered estimates of actual values based on ILRIS-3D data. Commanded torques from the attitude control law and commanded forces from the navigation control law were fed into thruster scheduling logic for a simple model of a 12 thruster system. Each thrust command was quantized to 5 msec, limited to 0.5 N, and subject to a minimum impulse of 25 ms These on-cycle commands were then fed into a thruster model to determine estimated force and torque outputs. The model did not include misalignment or thrust imbalances.

The tests were run to investigate rendezvous, not docking. As such, the relative position, calculated from the centroid of the target vehicle, should be independent of target attitude for a symmetric target. In the laboratory, the target was modeled by a square flat plate, that remained unchanged in the orbit reference frame—as though there was no attitude motion. The target spacecraft was assumed to have no changes in orbital parameters beyond minor variations due to aerodynamic drag.

3.0 EXPERIMENT SETUP

The SRECL, at the NRL, consists of two platform-mounted robot arms that can simulate the full 6-degrees of freedom (6-DOF) relative motion of two spacecraft in neighboring orbits (Figure 5). The chaser platform contained the ILRIS-3D, the "flight" computer (PC-104 form factor running Red Hat Linux), and the power supplies for each device. All of these components were housed in a custom-made PVC enclosure. The target platform contained a sensor target consisting of a simple, flat, PVC plate with dimensions 50.8 × 50.8 × 0.3175 cm. The target shape was chosen to simplify the target detection and tracking algorithms. Figure 6 shows the coordinate frames used during the simulations. The nominal body-fixed x, y, and z coordinates are shown for both vehicles, along with the relationship between the nominal body coordinates and the relative motion frame (or RSW frame) for both V-bar and R-bar maneuvers. Note that the body frame rotates with each body, and can therefore change with time. However, the RSW frame rotates with the orbit, and for the purposes of these experiments, can be thought of as fixed with time.

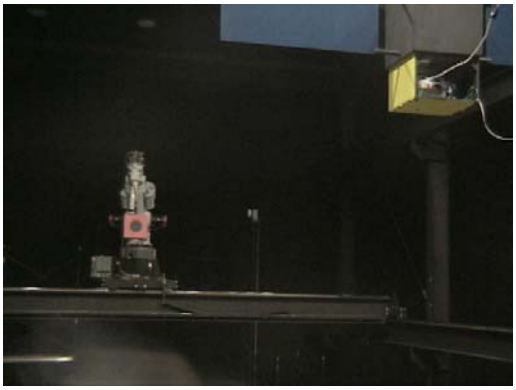


Figure 5. SRECL Lab Setup. Chaser (foreground) and target (background) platforms are clearly visible. The ILRIS-3D was housed in the yellow PVC enclosure (chaser), shown near the top right-hand corner. The target, colored red and black, moves back-and-forth (along-track) and left-and-right (radial) along a system of trolleys. The robot arm itself provides an additional 6-degrees-of-freedom.

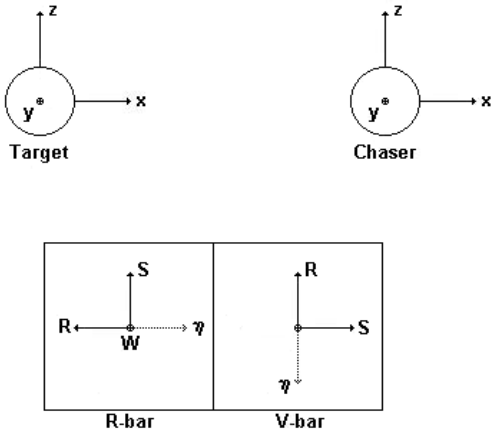


Figure 6. NRL Simulation Coordinate Frames. The body-fixed coordinates for both the target and chaser vehicles are denoted with x, y, and z. The nominal relationship between body-fixed and RSW coordinates for the R-bar and V-bar maneuvers are shown. The vector η denotes the direction to Earth.

There were three independent processors running in the experiments: the robots are driven by a SRECL "truth" model; the ILRIS-3D has an internal processor that controls I/O and commands the mirrors that point the laser; and the chaser platform houses a "flight" computer that runs chaser flight software and a hardware model for the chaser thrusters. Commands and data are passed among the processors using socket-based Ethernet connections. Figure 7 of the simulation software and hardware modules. All communications between the master and slave computers and the slave and ILRIS-3D computers were performed using socket-based Ethernet connections.

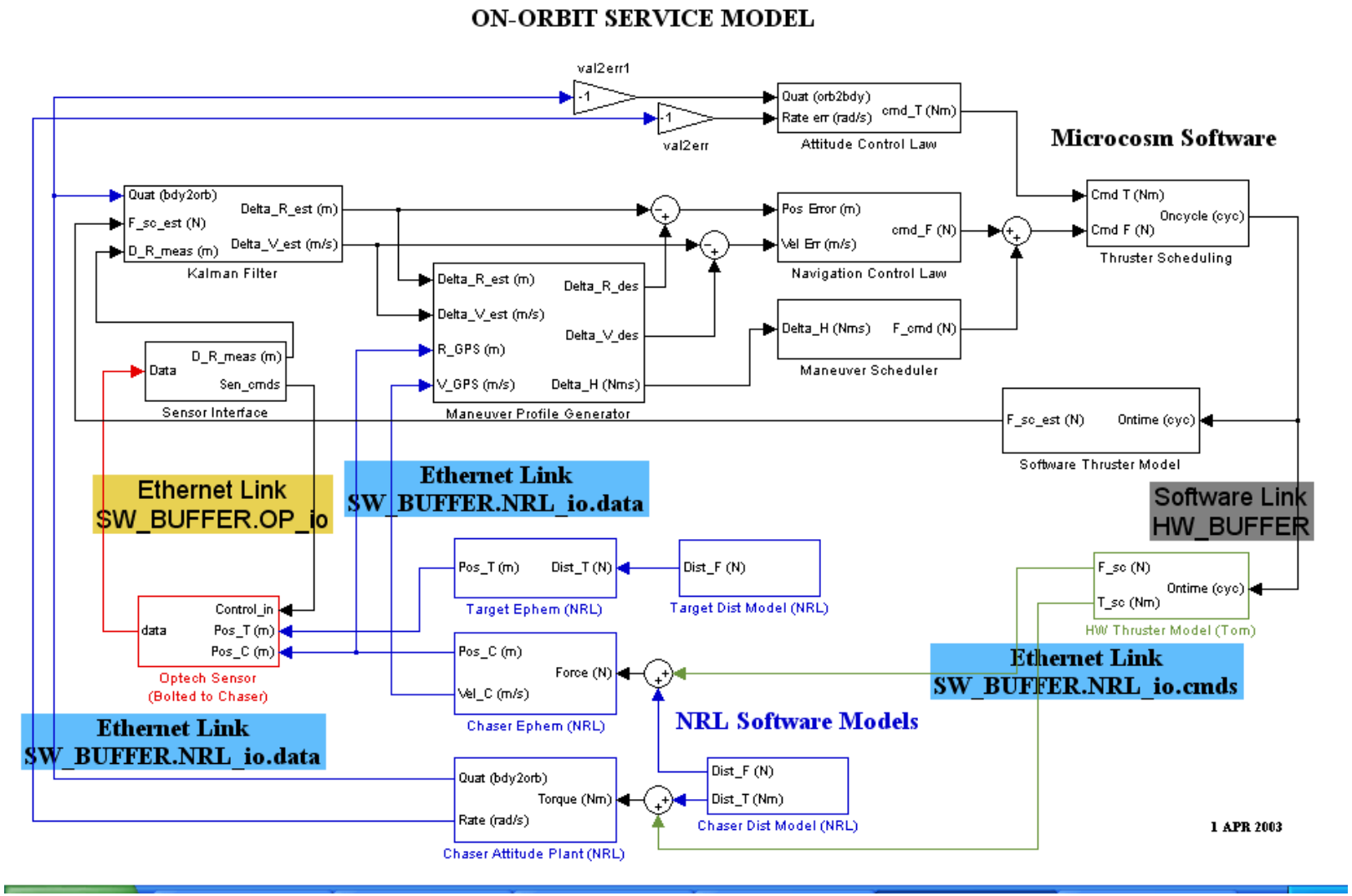


Figure 7. Simulation Software Schematic. The color-coding represents various software and hardware components for the simulation. Black and green represent Microcosm's GN&C software and thruster models. Blue and red represent the NRL software and the Optech sensor, respectively.

The NRL master computer simulation controlled the motion of each robot. The software first integrated the full 6-DOF, non-linear equations of motion for each vehicle and then determined the relative position, velocity, orientation, and angular rate of the target with respect to the chaser vehicle. Only environmental disturbance forces were modeled for the target vehicle; the target vehicle attitude was assumed to be fixed with respect to the orbit reference frame. Calculated environmental disturbance forces and torques were combined with forces and torques commanded by the "flight" computer to drive the 6-DOF propagator for the chaser vehicle. The master simulation provided values for chaser attitude angles and rates to the "flight" computer. All of the relative motion was observed in the target platform, while the chaser platform remained stationary for the entire simulation. Thus, the motion of the target platform was a composite of the simulated trajectory of the chaser spacecraft and the perceived motion of the target due to the attitude motion of the chaser. In other words, the lab coordinate frame was the same as the chaser body-fixed coordinate frame.

The ILRIS-3D has an imbedded processor that controls internal steering mirrors, receives commands from and sends packets of raw data to the "flight" computer. The software drivers for the sensor were not modified or optimized for use in a real-time closed loop application.

An imbedded processor, housed on the chaser platform, ran the "flight" software and was a slave to the master computer simulation. This software included: an Ethernet Sockets based interface for the ILRIS-3D and for the NRL master computer simulation, a Kalman filter, an orbit propagator, a maneuver profile generator, attitude control logic, navigation control logic, thruster scheduling logic, and a model of thruster hardware. Data from the LIDAR sensor was processed to provide a new scan command to be sent back to the sensor as well as a relative position estimate for the Kalman filter. The Kalman filter generated an estimated relative navigation solution (relative position, velocity, and orbit rate). This navigation solution was used by the maneuver profile generator to calculate a guidance table. Trajectory maneuvers were executed in open-loop fashion, based on the simulation time. The navigation control law was driven by the difference between the Kalman filter estimate and the guidance table "desired" values for position and velocity. Conversely, the attitude control law was driven by attitude angle and rate information provided directly by the NRL master simulation. Attitude data did not include sensor noise or biases. The force and torque commands from the control laws were combined to calculate commanded thruster firings. These commands were subject to: 5 msec quantization, 0.5 N limits, and 25 msec minimum impulse duration. Thruster commands were fed through an ideal thruster hardware model to calculate values for commanded force and torque to send to the NRL master simulation.

4.0 RESULTS

The simulation was run for R-bar rendezvous using both GTA and ZCS guidance laws, as well as V-bar rendezvous using a GTA guidance law. Since the NRL simulations were run in real time, a V-bar maneuver using the ZCS guidance law was not performed due to the large amount of time required for such a run (multiple orbit periods). Table 1 lists parameters common to each experiment. The target reference attitude was such that the target body z-axis was always fixed at the zenith (i.e., nadir-pointed). Recall that the LIDAR sensor data was not processed to determine relative attitude information. For this reason, the target attitude was kept fixed. Conversely, the chaser was often given an initial attitude offset that it was forced to correct.

Table 1. Experiment Parameters. Unless otherwise indicated, values for the target and chaser vehicles are the same. The target spacecraft and reference attitude is nadir-pointed.

Reference Orbit Parameters	
A	6878.137 km
E	0.0
I	54.7°
P	5676.977 s
Attitude and Body Rates	
Reference	Nadir-Pointing
Body Rates	$(0, 6.3\times10^{-2}, 0.0)$ deg/s
Mass Properties	
Mass	100 Kg
Inertia	(33, 33, 33) kg·m^2
Perturbations	
J_2	on
Drag	on
ρ	2.82×10^{-12} kg/m^3
β_c (chaser)	63.67 kg/m^2
β_t (target)	127.3 kg/m^2

This section will highlight the results of a typical simulation case, a negative V-bar maneuver using the GTA guidance law. The chaser was constrained to move within a 5° FOV cone, centered on the target. The initial, relative chaser attitude offset was -4° yaw, -5° pitch, and 3° roll (3-2-1 transformation). Figure 8 shows the relative motion of the chaser with respect to the target, in the RSW frame. The target is centered at the plot origin, which is what an observer moving with the target would see. The chaser began the maneuver 15 m ahead of the target in the along-track direction. The trajectory carried the chaser along a CW arc that did not pass beyond the 5° FOV cone, centered

on the target, and came to rest approximately 4 m ahead of the target. Figure 9 shows the same motion but as seen by the chaser and plotted in the chaser body-fixed coordinate frame (i.e., what an observer fixed to the chaser body would see). The ILRIS-3D was placed at the origin of the chaser body-fixed coordinate system. Since the chaser body frame is also the lab coordinate frame, Figure 9 was the motion observed in the lab. Due to the initial attitude offset of the chaser vehicle, the target first appears outside of the 5° FOV cone centered on the sensor (see Figure 9, second frame). The actual FOV of the sensor was ± 20°, so being outside the FOV was not an obstacle. The chaser attitude profile is shown in Figure 10. By 155 s., the chaser corrected its attitude to within 0.5° of nadir pointing. After this point, the control system began a +/- 0.5° limit-cycle, characteristic of thruster-based attitude control. The ACS requirements for the chaser will likely be determined from the pointing accuracy requirements during terminal docking. A 0.5° attitude error is well within the capabilities of a docking mechanism, such as the MAC Autonomous Micro-Satellite Docking Mechanism (AMDS) [5]. Therefore a thruster-based ACS will work for most docking applications.

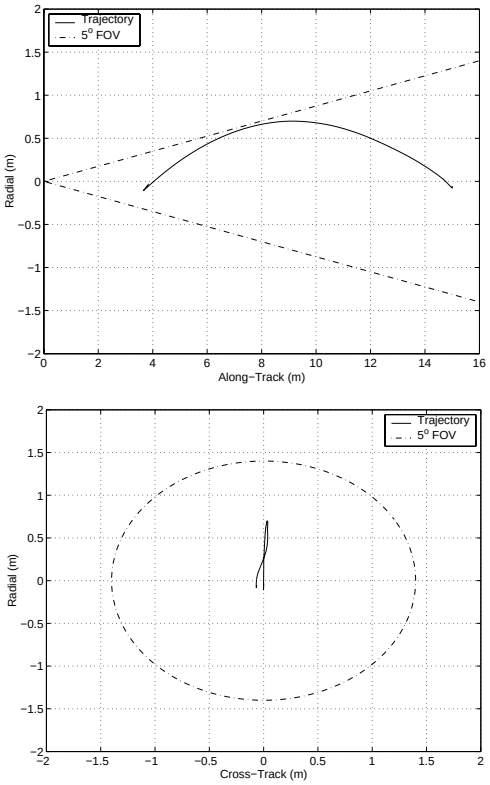


Figure 8. Chaser Trajectory Relative to the Target in RSW Coordinates. Chaser motion is shown as a combination of radial vs. along-track motion (left) and radial vs. cross-track (right). The target is at the origin in each plot. Chaser motion is in the negative V-bar direction.

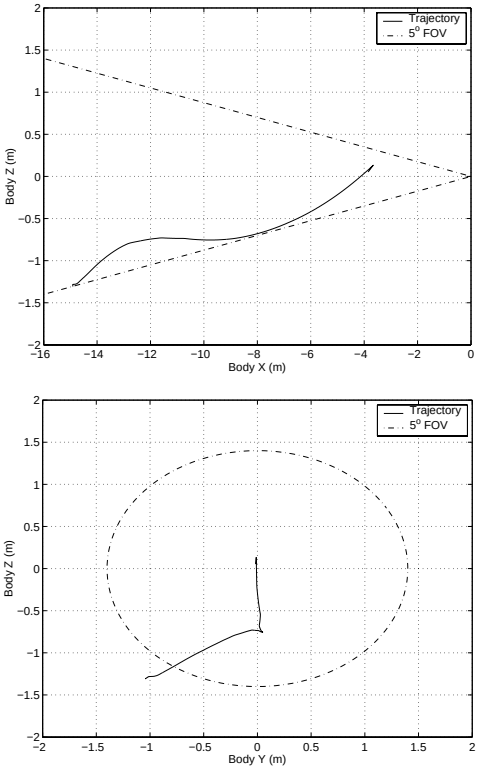
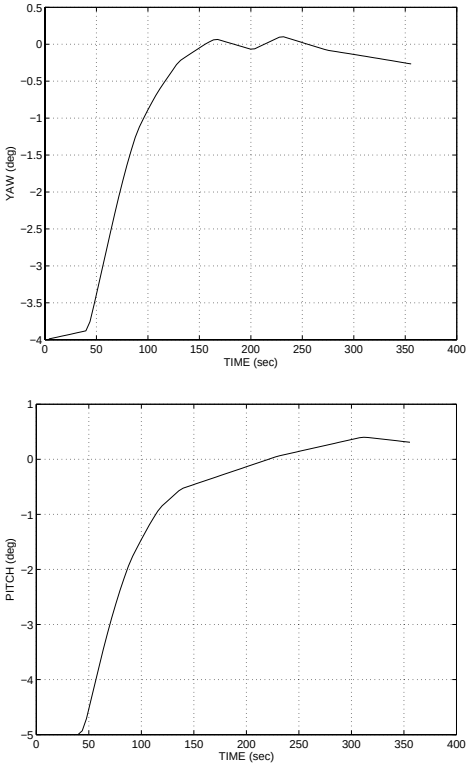


Figure 9. Target Motion Relative to the Chaser in Chaser Body-Fixed Coordinates. Target motion is shown as a combination of body-z vs. body-x (left) and body-z vs. body-y (right). The chaser is at the origin in each plot. Target motion is toward the chaser in the positive body-x direction. The observed motion, as seen by the chaser, is a composite of both trajectory and attitude maneuvering.



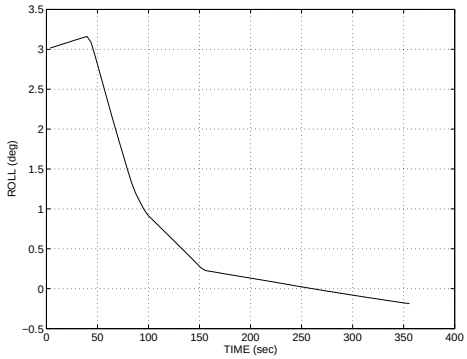


Figure 10. Chaser Relative Attitude. Attitude was determined from a 3-2-1 (yaw-pitch-roll) rotation. The Euler angles represent the deviation from a nadir-pointing spacecraft.

At the end of the maneuver, the chaser overshot the 4-m along-track aim point and then gradually moved back into place, which is more clearly shown in Figure 11, which plots the magnitude of the proportional (position) and derivative (velocity) errors. Two distinct peaks at about 50 and 310 s can be seen in the proportional error. These peaks coincide with the two burns required to start and stop the maneuver. The magnitude of the overshoot at the end of the maneuver (310 s) can be seen to be approximately 27 cm with a maximum velocity error of 4.24 cm/s. This overshoot is typical of the GTA maneuvers and is primarily due to the thruster sizing. The final ΔV (4.24 cm/s) required to stop the maneuver was comparable to the maximum ΔV (4 cm/s) capable of being supplied by the thrusters. For the assumed chaser mass (100 kg) and thrust level (2 thrusters $\times$ 0.5 N = 1 N), the thrusters had to remain on for 4 s. Small thrusters were used to demonstrate the insensitivity of the maneuver profile to thruster limitations. The commanded burns were timed, assuming perfect thrusters with unlimited force capability. The limited thrust of the implementation caused a position error due to a delay in the time to achieve the commanded change in orbital momentum, at both the start and end of the maneuver. Adjusting the timing of all commanded burns based on thruster capability can minimize this error. Even if the overshoot could be reduced, the problem remains that the terminal ΔV maneuver would be directed toward the target, giving rise to plume impingement issues. There is also the potential for collision with the target, if the chaser should fail to stop in time. In addition, the relatively high closing speed of 4.24 cm/s does not leave much time for correcting errors at a separation range of only of 4 m. For these reasons, GTA guidance should not be used for terminal rendezvous operations. However, GTA guidance would be useful for closing a large separation gap, where plume impingement and collision are not an issue. GTA guidance is especially useful when time is of the essence. The transfer shown in Figure 9 took 264 s. A similar transfer using the ZCS guidance law would have taken approximately 1 orbit period (5,677 s).

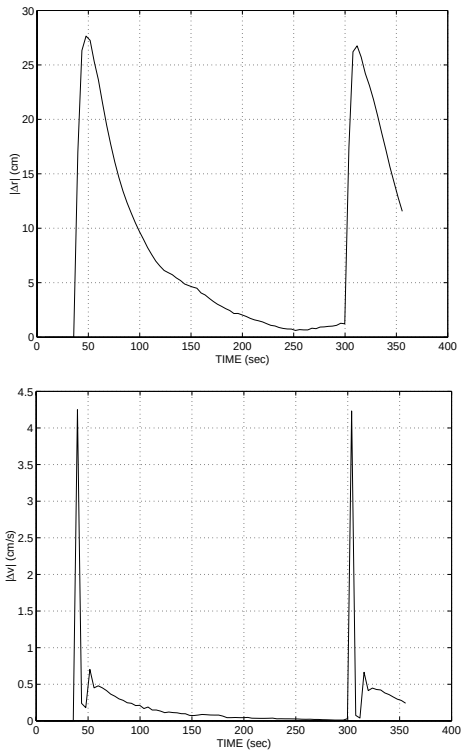


Figure 11. Magnitude of the Proportional and Derivative Errors. These plots represent the magnitude of the difference between the filtered and desired position and velocity (i.e., the control signal).

Figure 12 shows the raw measurement error and the filtered state error as measured against the NRL truth data. These plots illustrate the degree of accuracy to which we were able to estimate the relative position and velocity of the target. The biggest source of error was the measurement delay, or the time interval between when a range measurement was made and when it became available to the control system. For these experiments, the delay was on average about 2.8 s. The effect of delay is illustrated most dramatically in the plot for Δx (top left of Figure 12). While the target is in motion, the measurement error is on average 12.5 cm compared to an average stationary error of 0 cm. The 2.8 s delay was measured after the experiments were complete. During the experiments, an initial guess of 2.0 s was used. Thus, we were able to remove a portion of the delay error using the Kalman filter. This explains the difference between the filtered estimate (solid line) and raw measurements (circles) in the plot for Δx in Figure 12. A more detailed analysis of the errors is left for Section 4.2. Given the rather large, uncorrected biases, the filter does a good job of smoothing the data and providing reasonable velocity estimates, once the filter has converged (approximately 100 s from the maneuver initiation).

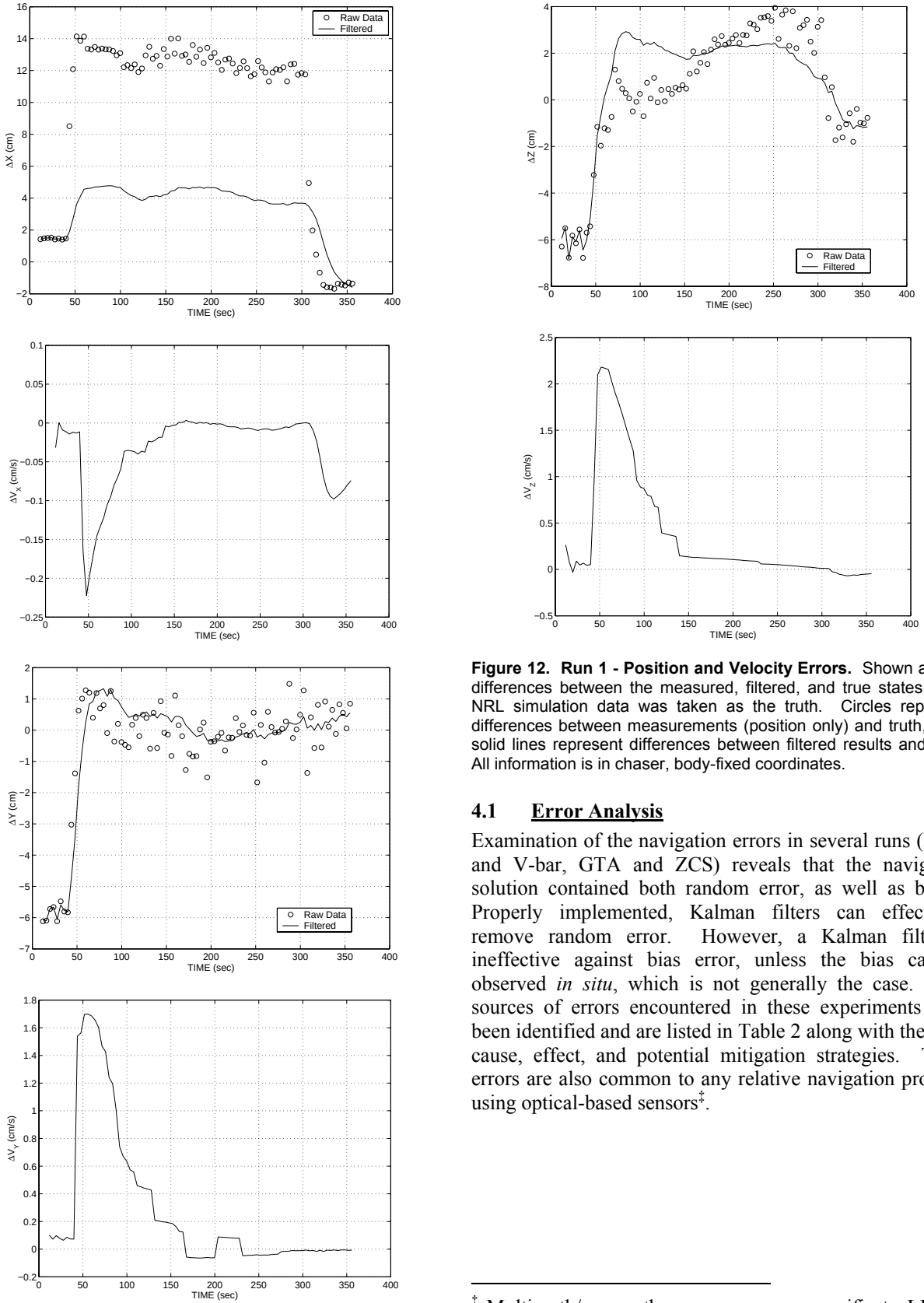


Figure 12. Run 1 - Position and Velocity Errors. Shown are the differences between the measured, filtered, and true states. The NRL simulation data was taken as the truth. Circles represent differences between measurements (position only) and truth, while solid lines represent differences between filtered results and truth. All information is in chaser, body-fixed coordinates.

4.1 Error Analysis

Examination of the navigation errors in several runs (R-bar and V-bar, GTA and ZCS) reveals that the navigation solution contained both random error, as well as biases. Properly implemented, Kalman filters can effectively remove random error. However, a Kalman filter is ineffective against bias error, unless the bias can be observed *in situ*, which is not generally the case. Five sources of errors encountered in these experiments have been identified and are listed in Table 2 along with the error cause, effect, and potential mitigation strategies. These errors are also common to any relative navigation problem using optical-based sensors[‡].

[‡] Multi-path/zero-path errors are more specific to LIDAR sensors than any other type of sensor.

Table 2. Measurement Error Sources.

Error	Cause	Effect	Mitigation Strategy
Misalignment	Sensor frame misaligned with the chaser body frame	Bias that is proportional to the separation distance	• Proper alignment • Measurement of alignment error
Misplacement	Origin of the sensor frame misplaced relative to the origin of the chaser body frame	Constant bias	• Proper placement • Measurement of placement error
Delay	Finite time interval between when measurement was made and when it becomes available	Bias that is proportional to the relative velocity	• High-speed sensor • Measurement of delay
Distortion	1 Apparent size and shape of target distorted by chaser location, orientation, and motion 2 Commanded laser does not cover all edges of target (worse for low density scans)	1 Bias that is proportional to the relative state (all components) 2 Error in azimuth and elevation estimates for target position (range not affected)	• Good image processing • Optimal target selection (if applicable) • High accuracy sensor • Maintain high density scans of target • Change scan pattern to track target edge
Noise	1 Uncertainty of time-of-flight calculation in range measurement 2 Uncertainty in Optech internal mirror positions	1 Scatter of range measurement about a mean value 2 Error in relative position measurement (bias and random)	• Kalman filter • High accuracy sensor • Sensor calibration
Multi-Path/ Zero-Path	Highly reflective materials (e.g., Mylar insulation, mirrors, retro-reflectors)	Multiple range measurements for a single laser shot (multi-path) Reflection directed away from sensor optics (zero-path)	• Avoidance of reflective materials • First-reflection logic • Narrow beam width • Intensity filter

Misalignment, misplacement, delay, and noise errors are probably the most familiar sources of error. These errors can all be measured, if the results can be compared to an error-free data set. In addition, noise can be dealt with effectively by proper application of Kalman filtering techniques. Distortion is an issue when the shape or size of a target is used to measure the relative position and attitude. LIDAR sensors, including the ILRIS-3D produce a range image. In the case of the ILRIS-3D, the image is produced by scanning a single beam over a given angular area using a step and stare method. The image is composed of the range and bearing to each scan point on the target. To develop range and bearing to the target from a range image usually requires some image processing techniques. Target bearing was more sensitive to distortion errors than the range calculation for the experiments presented here because range is measured directly from the time-of-flight of a laser shot. However, bearing is an indirect calculation, based on the detected edges of the target. Geometric distortion can be removed by image processing and minimized by proper selection of targets. Highly symmetric targets are the most desirable. The square plate used in these experiments was a good choice, since the location of the centroid did not vary with orientation. Multi-path/zero-path errors were not encountered in the lab, but are included in Table 3 for completeness. These errors occur when the LIDAR laser shots undergo multiple reflections prior to reflection. These reflections give rise to the appearance of false surfaces in the range image. Conversely, zero-path errors occur when a large fraction of the beam energy is reflected away from the sensor head. The surface need not have a mirror polish for these errors to occur. It has been found that this error is significant even for surfaces coated with semi-gloss paint. Only non-reflective surfaces were used

for these experiments. However, reflective surfaces such as gold foil and Mylar are widely used as thermal control devices on many satellites. Thus, these materials could pose severe difficulties for LIDAR sensors.

Several bias errors are evident from the data. In this section the NRL truth data and the observed measurement error are used to develop scaling laws that minimize the observed bias as possible. These scaling laws account for the effects of misalignment, misplacement, delay, and distortion. Note that these scaling laws were applied during the post-processing of the data. Some data scaling was used during the experiment, but was not sufficient to remove the biases.

Scaling laws that account for misalignment, misplacement, and delay are given in Eq. 17. Here x, y, and z are given in chaser body coordinates. α, β, and γ are misalignment angles, or rotations, about the x, y, and z-axes, respectively. δt is the delay, and v_x, v_y, and v_z are the velocities along the indicated axis. Finally, δx, δy, and δz are the misplacement terms.

$$\begin{aligned}\Delta x &= \underbrace{y\gamma - z\beta}_{} + v_x\delta t + \delta x \\ \Delta y &= -x\gamma + z\alpha + v_y\delta t + \delta y \\ \Delta z &= \underbrace{x\beta - y\alpha}_{} + v_z\delta t + \delta z\end{aligned} \tag{26}$$

Misplacement

Delay

Misalignment

A simplifying assumption is made that all errors contributing to Δy are zero (i.e., α, γ, and δy), with the exception of delay. This assumption is justified because no biases are clearly discernable for Δy in most of the cases run. With this assumption, the equations simplify to

$$\begin{aligned}\Delta x &= -z\beta + v_x\delta t + \delta x \\ \Delta y &= v_y\delta t \\ \Delta z &= x\beta + v_z\delta t + \delta z\end{aligned} \tag{27}$$

In addition to misplacement, misalignment, and delay, another source of error can be seen in the results for Δy and Δz (Figure 12). The peaks in these error plots correspond to the time when the target was near the edge of the 5° FOV cone (i.e., the tangent point). The y and z distortions also appear to be independent of each other. The new scaling laws that include distortion are:

$$\begin{aligned}\Delta x &= -z\beta + v_x\delta t + \delta x \\ \Delta y &= v_y\delta t + y\theta_y \\ \Delta z &= x\beta + v_z\delta t + \delta z + z\theta_z\end{aligned} \tag{28}$$

The error parameters in Eq. 19 determined using a least-squares method, based on several cases. The results are shown in Table 3.

Table 3. Scaling Parameters for Sensor Biases

Parameter	Value
β	0.022°
δt[§]	2.7 sec
δx	9.8 mm
δz	7.9 mm
θ_y	0.06
θ_z	-0.02

Note that for the range of separations considered in the laboratory, β is so small as to be almost negligible. In addition, the misplacement terms are all less than 1 cm. Thus, sensor misalignment/misplacement did not contribute substantially to the errors observed in the laboratory. The only substantial errors were therefore delay and distortion.

When accounted for, these scaling laws substantially reduce the observed biases in all but one case. Figure 13 shows the improved scaled results against the raw measurements for the example run. In the absence of any bias, the errors should all have zero mean value. Table 4 shows all components of the mean error and standard deviation for the scaled data of 3 different cases. With the exception of Δz for the example case (R-bar, ZCS), the results indicate that the mean values and standard deviations are less than 1 cm in all cases, which is well within the stated accuracy of the ILRIS-3D. The Δz results for the example case are somewhat disappointing. Obviously, all sources of error have not been completely accounted for, particularly when there is an initial attitude offset. However, notice that in Figure 13, the scaled result for Δz improves as the time goes to 150 s. After this time the initial attitude error has been largely corrected (see Figure 12), which suggests that the bias is still linked to the relative attitude. It is not clear if the error is due to unaccounted distortion or misa lignment.

[§] The value for δt used in the control law was 2.0 s, which explains the difference between the filtered results and the raw data in the Δx errors for Figure 12.

Table 4. Mean and Standard Deviation of Scaled Measurement Error.

Run	Δx		Δy		Δz	
	Mean (cm)	σ (cm)	Mean (cm)	σ (cm)	Mean (cm)	σ (cm)
Example	0.069250	0.842044	-0.263774	0.759773	-2.448586	3.643453
R-bar GTA	-0.200670	0.959688	0.258248	0.633907	-0.280100	0.775088
R-bar ZCS	-0.504773	0.472918	0.041158	0.607785	0.035388	0.703155

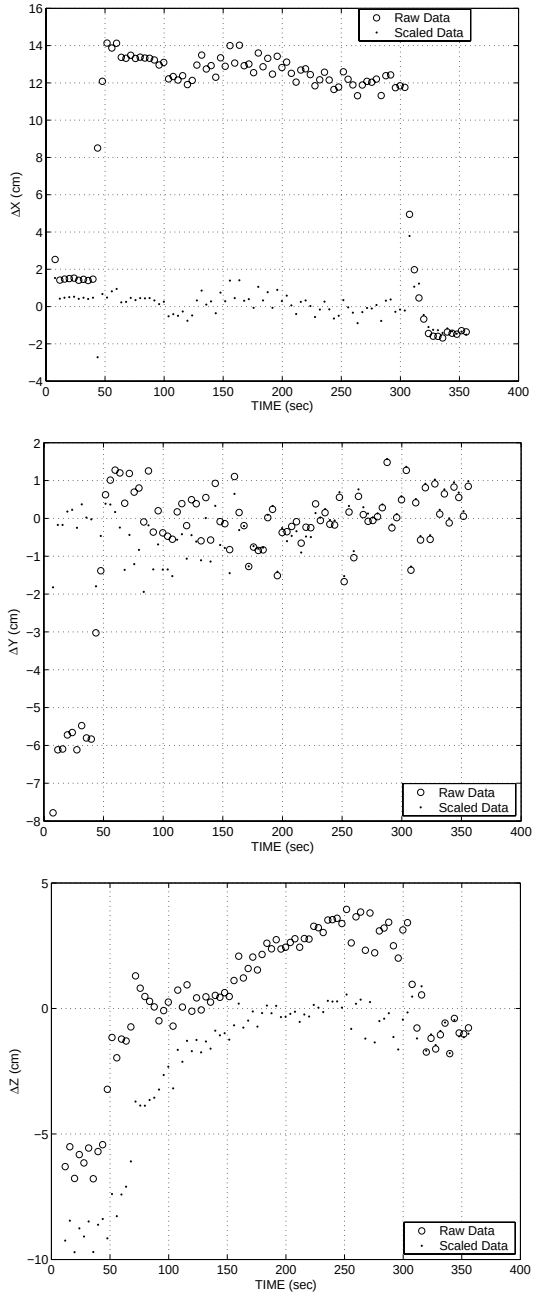


Figure 13. Scaled Measurement Errors

5.0 CONCLUSIONS

In conclusion, the GTA and ZCS guidance laws were incorporated into an autonomous GN&C system for on-orbit servicing that also included a simple Kalman filter and thruster-based PD controller. The GN&C system was validated with HITL and SITL tests at the NRL SRECL facility. The performance of the GN&C system was admirable even in the presence of orbit perturbations, noise, and system bias. In particular, the performance of the ILRIS-3D was extremely good, especially given that its intended function was for static surveying. Without modification, the sensor was placed into a real-time, dynamic application. The 0.25 Hz controller update frequency did not pose any difficulties that could not be mitigated with software. Proper scaling of the data to remove biases reduced the measurement error to within ± 3 cm (unfiltered) when the chaser and target were in close proximity, which is well within the range requirements of 10 cm for the MAC docking mechanism [5]. These experiments demonstrate that autonomous GN&C for on-orbit servicing is achievable with very modest system requirements.

6.0 REFERENCES

[1] Dellacamera, et al., *Logistics Requirements for Space: On-Orbit Servicing (OOS),* PRC Corp., Final Report AFRL-HE-WP-TR-2000-0095, 2000.

[2] Wertz, J. R., et al., In-Space Testing of Satellite Servicing and Inspection Elements, Final Report, AFRL F29601-01-C-0131, Microcosm, Inc., 2002.

[3] Chinnery, A., et al., *Alternate Access to Station, Requirements and Concepts Study*, Microcosm, Inc., Final Report for Contract Number NAS8-00174, 2000.

[4] Turner, A. E., Development of Geosynchronous Satellite Servicing, Space Systems Loral, 2002.

[5] Tchoryk, P., et al., KC-135 Zero G Testing of a Micro Satellite Docking Mechanism, SPIE, 2003.

[6] Tchoryk, P., et al., Autonomous Satellite Docking System (ASDS) Trade Studies, DARPA Phase II SBIR, Contract #DAAH01-C-R015, Michigan

Aerospace and Oceaneering Space Systems, Internal report, March, 2001.

[7] Howard, R. T., et al., *The Video Guidance Sensor – A Flight Proven Technology,* 22nd Annual AAS Guidance and Control Conference, AAS 99-025, Feb 1999.

[8] Smithpeter, C., Nellums, R., A Scannerless Laser Radar for the Primary Guidance Sensor in Autonomous Rendezvous and Docking of Spacecraft, Sandia National Laboratories White Paper.

[9] Optech, Inc., RELAVIS Product Brochure, 2002

[10] Optech , Inc., ILRIS-3D Product Brochure, 2002

[11] Dawson, S., Reynolds, H., Meissinger, H., Spacecraft Orbit and Attitude Systems, Microcosm Press 2001, pp 527-534, section 10.2.3

[12] Waltz, D. M., with contributions by Meissinger, H.F., Supplement to On-Orbit Servicing of Space Systems, Krieger, 1998.

[13] Creamer, G., et al., *NRL Technologies for Autonomous Inter-Spacecraft Rendezvous and Proximity Operations*, J. L. Junkins Astrodynamics Symposium, AAS/AIAA Spaceflight Mechanics Meeting, Paper No. AAS 03-272, College Station, TX, May 23-24, 2003.

[14] Creamer, G., and Hollander, S., *The Spacecraft Robotics Engineering and Controls Laboratory*, 2002 NRL Review, pp. 207-209.