EXPANDING HARDWARE-IN-THE-LOOP FORMATION NAVIGATION AND CONTROL WITH RADIO FREQUENCY CROSSLINK RANGING[*]

Jason W. Mitchell[†], Brent W. Barbee[‡], and Philip J. Baldwin[§]

Emergent Space Technologies, Inc., Greenbelt, MD 20770-6334, USA

Richard J. Luquette[¶]

NASA Goddard Space Flight Center, Greenbelt, MD 20771, USA

Abstract

The Formation Flying Testbed (FFTB) at the National Aeronautics and Space Administration (NASA) Goddard Space Flight Center (GSFC) provides a hardware-in-the-loop test environment for formation navigation and control. The facility continues to evolve as a modular, hybrid, dynamic simulation facility for end-to-end guidance, navigation, and control (GN&C) design and analysis of formation flying spacecraft. The core capabilities of the FFTB, as a platform for testing critical hardware and software algorithms in-the-loop, are reviewed with a focus on recent improvements. With the most recent improvement, in support of Technology Readiness Level (TRL) 6 testing of the Inter-spacecraft Ranging and Alarm System (IRAS) for the Magnetospheric Multiscale (MMS) mission, the FFTB has significantly expanded its ability to perform realistic simulations that require Radio Frequency (RF) ranging sensors for relative navigation with the Path Emulator for RF Signals (PERFS). The PERFS, currently under development at NASA GSFC, modulates RF signals exchanged between spacecraft. The RF signals are modified to accurately reflect the dynamic environment through which they travel, including the effects of medium, moving platforms, and radiated power.

Keywords: radio frequency signals, spacecraft crosslinks, relative navigation, delay, signal modulation, real-time, hardware-in-the-loop, formation flying, formation control.

INTRODUCTION

Interest in using distributed spacecraft systems flying in formation as a mission enabling technology continues to grow. Both the National Aeronautics and Space Administration (NASA) and the European Space Agency (ESA) are evaluating formation flying concepts for numerous planned missions. A brief list of currently planned missions include, NASA: Magnetospheric Multiscale (MMS) [25], Black Hole Imager [11, 32], Submillimeter Probe of the Evolution of Cosmic Structure (SPECS) [15], Stellar Imager (SI) [5]; ESA: Darwin [6], Prisma [29], Proba-3 [7]. In addition, *precision formation flying* was evaluated as one of the five candidate technology capability areas for the New Millennium Program's Space Technology 9 (ST9) Project [4, 8].
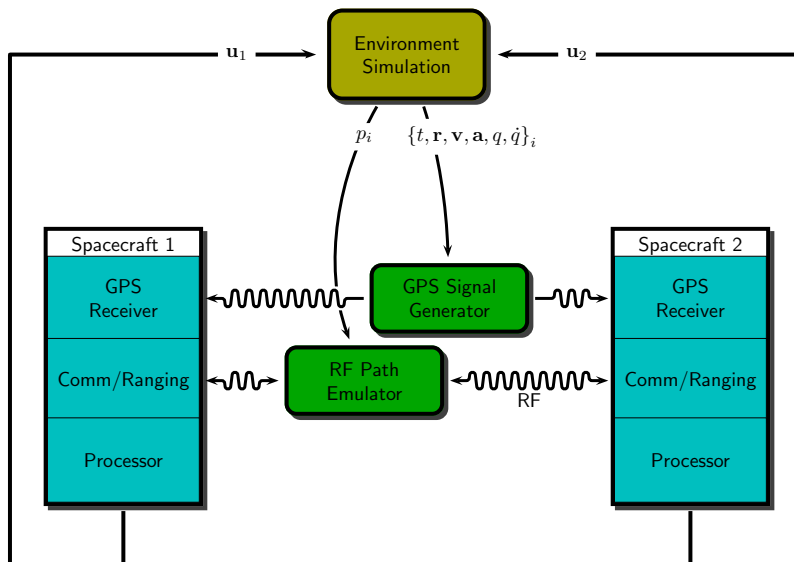
---

[*]This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

[†]Aerospace Scientist, `jason.mitchell@emergentspace.com`.

[‡]Aerospace Engineer, `brent.barbee@emergentspace.com`.

[§]Assoc. Flight Dynamics Engineer, `philip.baldwin@emergentspace.com`.

[¶]Aerospace Engineer, Code 591, `rich.luquette@nasa.gov`.

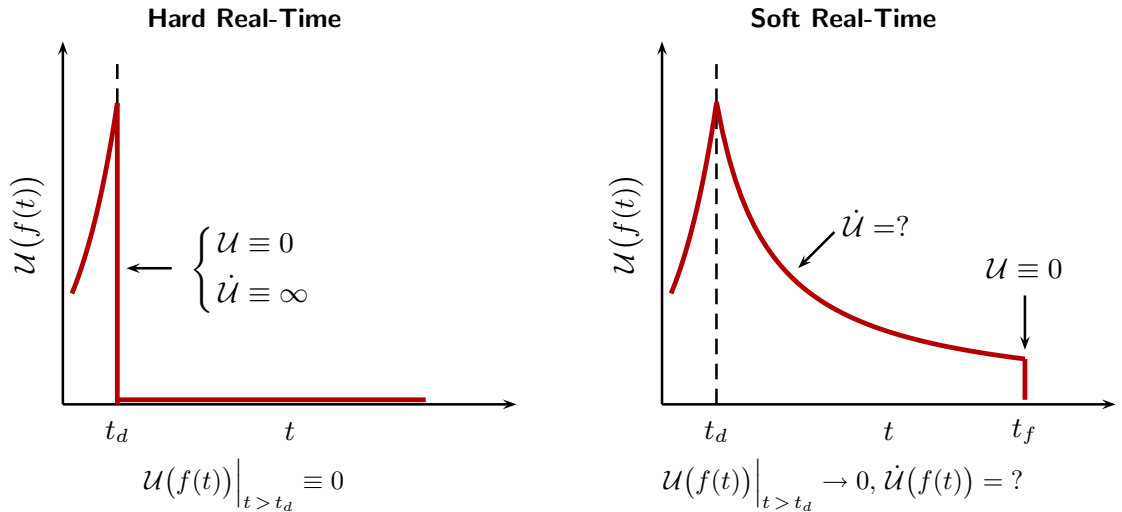**Figure 1.** Example scenario for two satellite Earth orbiting simulation employing GPS and RF ranging.

To reduce mission risk and provide suitable technology validation, algorithms and hardware that enable formation flying must be ground tested to provide sufficient confidence in their on-orbit performance. The Formation Flying Testbed (FFTB) [16, 19, 22] at NASA Goddard Space Flight Center (GSFC) provides a unique environment for designing and testing formation flying Guidance, Navigation, & Control (GN&C) algorithms and hardware. The FFTB enables an end-to-end simulation capability by testing GN&C algorithms in real-time, and in the presence of essential flight hardware, e.g. relative navigation sensors and crosslink transceivers. By including this hardware directly in the closed-loop testing, researchers and engineers gain valuable information about the interaction and performance of their algorithms and essential hardware.

In the following, the current capabilities of the FFTB are presented. Recent developments that enable Radio Frequency (RF) crosslink ranging for use in hardware-in-the-loop simulations are emphasized.

CAPABILITIES

To facilitate the discussion of the FFTB components and capabilities, we first present an example scenario. Figure 1 notionally depicts a simulation of two Earth orbiting spacecraft that includes hardware sensors for GPS and ranging via RF crosslink. The environment simulation provides truth data to each of the hardware environment emulations, viz. spacecraft state information, $\{t, \mathbf{r}, \mathbf{v}, \mathbf{a}, q, \dot{q}\}_i$, for GPS RF signal generation; and channel parameters, p_i, for RF path emulation. Sensor measurements are processed by flight software on each simulated flight processor, which may include formation navigation and control algorithms. Any resulting spacecraft controls, $\mathbf{u}_i$ are fed back to the environment simulation to close the loop.

In this example scenario, we easily identify explicit hardware and software components that provide for environment simulation/emulation, sensor measurements, and spacecraft navigation and control. In addition, hardware-in-the-loop simulations contain two important implicit components: real-time effects and software interface layers to hardware. Finally, a distributed simulation capability requires a method for exchanging information between non-collocated components. Each of these elements are identified and described in the following.

Hard Real-Time

$$\begin{cases} \mathcal{U} \equiv 0 \\ \dot{\mathcal{U}} \equiv \infty \end{cases}$$

$$\mathcal{U}\big(f(t)\big)\Big|_{t>t_d} \equiv 0$$

Soft Real-Time

$$\dot{\mathcal{U}} = ?$$

$$\mathcal{U} \equiv 0$$

$$\mathcal{U}\big(f(t)\big)\Big|_{t>t_d} \to 0, \ \dot{\mathcal{U}}\big(f(t)\big) = ?$$

Figure 2. Characterization of hard and soft real-time in terms of utility function, $\mathcal{U}$.

Real-Time

The term *real-time* is both ubiquitous and multivocal. The common notion of real-time involves only the time-sensitivity of completing a computation. This is only a portion of the overall validity, since the computation also requires logical correctness. Ultimately, the goal is to establish a system that responds predictably with respect to certain deadlines. To capture this dual-nature, we specifically distinguish between *hard* and *soft* real-time in the manner of Robillard et al. [24, pp. 6]:

> Hard real-time applies to activities that must be deterministic; critical activities have deadlines. When this processing fails to meet a deadline, the system has failed. [...] The design emphasis when building systems with hard deadlines is to guarantee that all deadlines will be met.

> Soft real-time is non-deterministic to the extent that an occasional missed deadline can be tolerated as acceptable degraded performance, not a system failure. The value of completing a soft real-time activity decreases after its deadline has passed, but the rate at which the value decreases differs between activities. The operational procedures for dealing with missed deadlines also vary.

These concepts are more easily seen if we introduce the notion of a *time utility function*, $\mathcal{U}$ [14, 17]. This concept is graphically depicted in Figure 2. In each case, there is a computational result, f, whose utility, $\mathcal{U}(f(t))$, is shown as a function of time. For both cases, there is a deadline, t_d, at which the utility of the result is maximum. For *hard* real-time, the utility at any time after the deadline is zero, $\mathcal{U}|_{t>t_d} \equiv 0$. For *soft* real-time, the utility decreases at some rate until reaching time t_f, after which the result has no utility.

Based on these definitions, simulations in the FFTB are generally characterized as *soft* real-time because many of the simulation components use some form of filtering. As a result, they can continue to function in a degraded mode if a deadline is missed. Late measurements may be incorporated to reduce the effect of the missed deadline(s). Thus, if the missed deadline was not the result of a complete system failure, normal performance may be recovered.

Available Hardware

Individual hardware elements are described in the following.

Sensors

GPS Receivers: At present, the FFTB maintains two Orion receivers, four PiVoT receivers, and a single Ashtech G-12 receiver. The Orion receivers from the German Space Operations Center possess a built-in relative navigation feature where GPS measurements are shared directly between the receivers via RS-232. In simulation, both the PiVoT and Orion receivers produce American Standard Code for Information Interchange (ASCII) WinMon format messages over RS-232. The Ashtech receiver is used solely for calibration purposes.

RF Crosslinks: Two pairs of RF crosslinks currently reside in the FFTB. The Low-Power Transceivers (LPTs) provide integrated RF communication and Tracking and Data Relay Satellite System (TDRSS) links. The Crosslink Transceivers (CLTs) support integrated RF communication and relative navigation.

Environment

GPS Simulator: Up to four Spirent Model 4760 GPS Signal Generators are available, producing up to eight L1 RF GPS signal outputs for direct stimulation of GPS receiver antennæ. True spacecraft state information is provided in real-time to the Spirent SimGEN software [26], which drives the Model 4760's GPS RF outputs.

Crosslink RF Path Emulation: The Crosslink Channel Simulator (CCS) [13] emulates the space environment for RF signals used for inter-spacecraft communication and ranging. The CCS provides two uni-directional RF channels emulating a single bi-directional crosslink between two spacecraft. The channel parameters are computed from a channel model and specify the delay, attenuation, and Doppler shift to be applied to RF signals exchanged via crosslink. The channel model receives true spacecraft state information in real-time, and the CCS is commanded in real-time.

Computing Infrastructure

Flight Computers: Standard desktop computers are used as flight processors. This significantly reduces the development burden of formation navigation and control algorithms as compared to the traditional embedded development platforms associated with realistic flight processors. Thus, algorithms may be easily developed and modified directly in the FFTB. This is particularly useful since early algorithm prototyping and development often occurs in programming environments like MATLAB [30] or SciPy [31]. The computers are single processor Pentium 4 (6-series) with a clock speed of 3 GHz and HyperThreading. To approximate the computing power of a more realistic flight processor, these computers can be *down-clocked* using the Enhanced Intel SpeedStep Technology (EIST). The 6-series processors offer eight dynamic frequency steps for processor clocking (kHz): 3000, 2625, 2250, 1875, 1500, 1125, 750, 375. This frequency stepping allows for computational resource constraints that approach flight-like processors during simulation without robbing developers of the computational resources necessary to quickly develop and modify the algorithms.

Network: Distributed simulation components are connected to the FFTB internal Ethernet network, which is a multi-switched, Gigabit, full-duplex, auto-negotiated, Class C network. Critical simulation hardware shares a single switch to minimize latency, while less critical components are connected over multiple switches. Components that require high bandwidth are fitted with multiple Ethernet interfaces that are channel bonded with adaptive send/receive load balancing.

Precision Frequency Reference

Precision reference signals at 1, 5, and 10 MHz are generated by a Symmetricom FTS 4065B Cesium frequency standard.
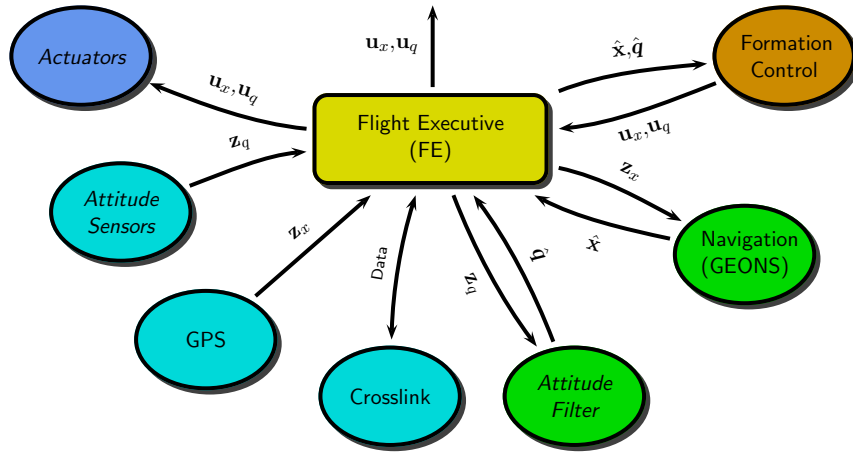
Figure 3. Flight Executive flight software connectivity.

Software

A collection of software components drive the soft real-time hardware-in-the-loop simulations in the FFTB. These components are divided into the following areas.

Environment

The Spacecraft Trajectory and Attitude Real-time Simulation (STARS) software package generates truth spacecraft environment at 10 Hz, and synchronizes with a hardware 1 Pulse per Second (1PPS) signal generated by the Spirent Model 4760 GPS signal generators. STARS accepts maneuver input in the form of a ΔV expressed in the vehicle Radial, In-Track, Cross-track reference frame.

GPS Constellation: Spirent's SimGEN software [26] manages the GPS system simulation via the SimREMOTE remote command interface [27]. STARS provides true spacecraft state information to SimGEN at 10 Hz. SimGEN drives the GPS signal generators to produce appropriate RF output for stimulating a GPS receiver antenna.

RF Path Emulator: Similar to the GPS constellation simulation, the RF path model receives true spacecraft state information at 10 Hz from STARS. Channel parameters are written to the CCS via User Datagram Protocol (UDP) message over Ethernet, also at 10 Hz.

Flight Software

The Flight Executive (FE), shown in Figure 3, represents a federation of components that, together, make up the flight software. The named components that appear in a normal sans serif font are existing components, while named components appearing as *italicized sans serif* text are not yet implemented. The FE collection is a mix of Java, C, and C++ programs and MATLAB scripts. This multi-language support provides a range of development options from rapid prototyping with MATLAB and Java to optimizing algorithms in C or C++ to improve real-time performance.

In the current FE operating scenario, GPS and crosslink measurements are processed with GPS Enhanced On-Board Navigation System (GEONS) [23] for orbit determination and navigation. Any control information generated from the formation controller is provided to the truth environment simulation as previously described. We should emphasize that for each FE instance, beyond initial
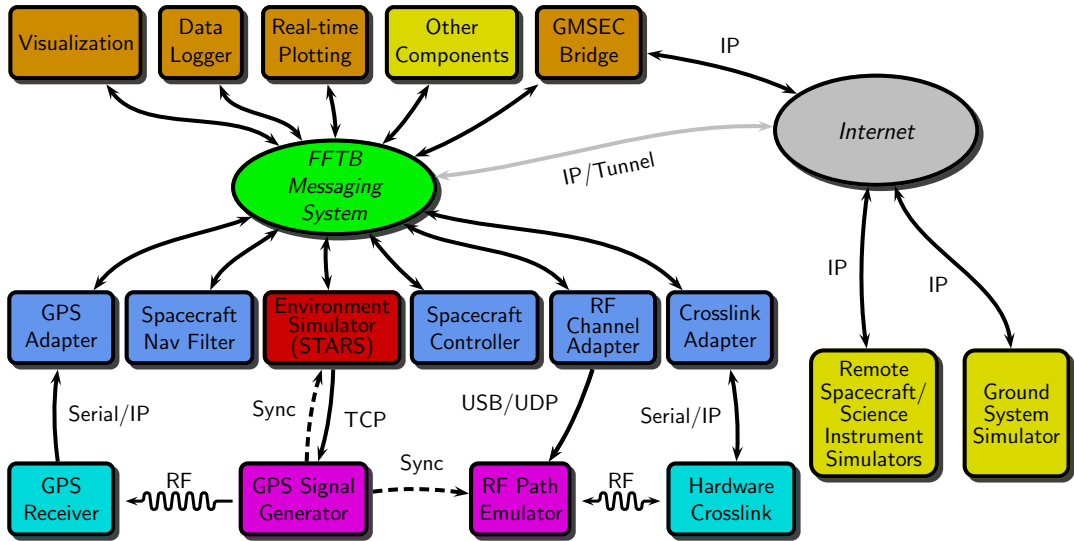
Figure 4. Simulation component connectivity.

configuration file input, no direct true state information is consumed; only sensor measurements of the space environment are provided as input. While GEONS does receive state information via configuration file at start-up, the filter does not start automatically in a converged state.

Component Integration Software

A connectivity for FFTB simulation components is shown in Figure 4. Hardware sensors and hardware environment emulators appear at the bottom of the figure. Essential simulation components appear at the next level above the hardware components. This layer contains the true spacecraft state environment simulator, FE components, etc., as well as *adapter* software. Adapters represent a thin interface layer to hardware components that extracts information from the hardware and provides it to other simulation components, both machine local and distributed, via the FFTB Messaging System (FMS) [19, 20].

FFTB Messaging System: The FMS is a Message-Oriented Middleware (MOM) layer that is designed to significantly reduce the development when integrating new software and hardware components into the existing FFTB software architecture. The FMS uses a customized version of the open-source Spread Toolkit [28] as the core of a *publish/subscribe* messaging system. Message definitions are maintained in an interface control document [9]. Individual messages are formatted in XML [34] with entries composed of empty-element tags containing (*key, type, value*) information; currently supported types are: boolean, integer, long integer, single-precision float, double-precision float, and ASCII strings.

The FMS libraries are available in both Java and C++, and include support for argument marshaling and unmarshaling, input file parsing, type conversion, and other supporting functionality.

Other Software Components

As seen in Figure 4, there are additional software components associated with a simulation. Those components at the top row of the figure are generally terminal components, i.e. components that only subscribe to simulation component outputs. These are typically visualization or logging components. For example, real-time, 3-D, full scene visualization can be performed with the Satellite Tool Kit from Analytical Graphics, Inc. Likewise, real-time engineering visualization for monitoring relative
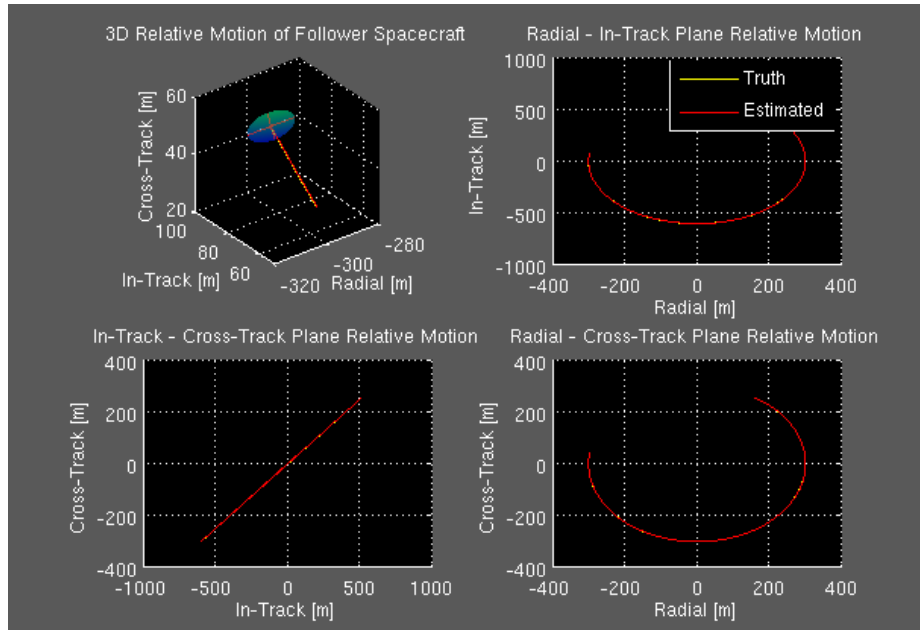
Figure 5. Planar views of the relative motion in the Hill's frame and one 3D view (top left) of the true and relative trajectories with the corresponding covariance ellipsoid.

motion, navigation, and control function can be performed with MATLAB, by the Mathworks, Inc.

One non-terminal component shown in Figure 4 is the GSFC Mission Services Evolution Center (GMSEC) bridge. GMSEC is a message bus middleware produced by the NASA GSFC Mission Services Evolution Center whose goal is to provide integrated mission services through the year 2010. While the FMS subject naming conventions are motivated by GMSEC, the FMS is not directly compatible with GMSEC. The purpose of the GMSEC bridge is to translate messages so that external simulations that use GMSEC as a message bus, e.g. remote spacecraft, science instrument or ground system simulators, can communicate with FFTB simulations.

RECENT ACCOMPLISHMENTS

ST9 Precision Formation Flying

In support of the Precision Formation Flying system technology concept area of the NASA New Millennium Program ST9 program [8], the FFTB successfully developed and demonstrated a real-time Hardware-in-the-Loop (HitL) simulation to satisfy Technology Readiness Level (TRL) 4[1].

Scenario

The ST9 TRL 4 scenario consisted of a *leader* and *follower* spacecraft, flying in a controlled planar formation. The leader flew an uncontrolled, circular, low, Earth orbit at an altitude of 420 km, with an inclination of approximately 97° (dawn-dusk sun-synchronous). The follower relative motion about the leader was controlled to maintain a 300×600 m safety ellipse [10, 22]. The plane of the safety ellipse was inclined up out of the leader's orbit plane about the radial axis by approximately 26.5°. The follower period about the leader on the ellipse was equal to the period of the leader orbit, which was approximately 90 min. Figure 5 shows the relative motion over approximately 70% of an orbit period. Figure 5 was generated by a engineering visualization system, using MATLAB and the

---

[1]TRL 4 represents a successful component and/or breadboard validation in a laboratory environment [18].

Table 1. Steady state Root Mean Square (RMS) error requirements and error performance.[a]

	Req	Case 1	Case 2
Range	2.5 m	0.47 m	1.11 m
Bearing (L→F)	0.83°	0.10°	0.08°
Bearing (F→L)	0.83°	0.05°	0.13°
Relative Position Magnitude	-	0.91 m	1.28 m
Relative Velocity Magnitude	-	3.16 mm/s	4.46 mm/s

[a]Total ΔV consumed for formation control over one orbit period of leader was between 1 and 2 m/s (1-norm) for both cases.

FMS, that monitors simulation performance in real-time.

The STARS truth environment simulation included a 30×30 Earth gravity field model, a Harris-Priester atmospheric drag model, and formation control impulses. The GPS signal generator RF output included realistic signal effects including path delay, ionospheric delay, signal attenuation, and transmit/receive antenna property effects. The GEONS flight software used a 10×10 Earth gravity field model and a drag model slightly less accurate than the Harris-Priester model used in STARS. For ranging, a software crosslink model produced inter-spacecraft pseudorange measurements. The pseudorandom noise model for the software crosslink used a uniform distribution of $\pm 10\%$ true range, which deliberately placed additional burden on the system since a normally distributed noise is expected.

In the scenario each spacecraft flew an Orion GPS receiver for navigation. The receivers were stimulated by the GPS signal generators to produce hardware GPS measurements. The software crosslink model provided pseudorange measurements. GEONS processed these measurements and produced the absolute state estimates for both spacecraft, from which a relative state estimate was formed by differencing. These state estimates were fed to the formation controller, which computed control in the form of ΔV impulses that were fed back to the truth environment simulator (STARS). The formation control algorithms running within the FE accounted for the maximum acceleration dictated by the ST9 propulsion system design.

Results

The ST9 scenario successfully demonstrated sufficiently accurate formation control and relative navigation, validating the formation controller scheme, the Orion GPS receivers, and GEONS for precision formation flying GN&C at TRL 4. The goal was to achieve specific maximum steady-state RMS range and bearing errors, seen in Table 1, where error was defined as the difference between the true and targeted values.

Two distinct simulation initialization cases were treated:

Case 1 nominal initial conditions;
Case 2 large initial position error of approximately 300 m in magnitude.

The system requirements and performance metrics for these cases are summarized in Table 1. From this, it is clear that the requirements were exceeded by a substantial margin. Considering the perturbations included in the scenario, and the overall performance, the demonstration robustly satisfied TRL 4.

Prototype Path Emulator for RF Signals

NASA GSFC is currently developing the next generation of RF path emulator, the Path Emulator for RF Signals (PERFS). This successor to the Crosslink Channel Simulator (CCS), formerly known as the Crosslink Channel Path Simulator (CCPS), will accommodate more bi-directional RF

Table 2. PERFS performance requirements and prototype test results.

	Requirement			Prototype		
	max	min	Δ^b	max	min	Δ^b
Attenuation	90.0 dB	0.5 dB	0.5 dB	63.0 dB[c]	0.5 dB	0.5 dB
Range	3,500.0 km	100.0 m	5.0 cm	10.0 km[c]	922.0 m[c]	16.5 cm
Doppler	5.0 kHz	0.0 Hz	10.0 mHz	5.0 kHz	0.0 Hz	10.0 mHz

[b]Resolution.
[c]Prototype limitation.

channels in support of the Magnetospheric Multiscale (MMS) mission. As part of this development, a prototype unit was constructed to verify feasibility, and was tested for attenuation, range, and Doppler performance based on a combination of requirements taken from both the ST9 and MMS missions [21]. The essential performance requirements are given in Table 2. As noted in Table 2, the prototype had several limitations:

- maximum range limited to 10 km,
- minimum range limited by Surface Acoustic Wave (SAW) filter use,
- maximum attenuation limited to 63 dB,
- tested at the Intermediate Frequency (IF) of 35.42 MHz,
- free running clocks.

As an early, iterative, functional test, a nominally large range was set and then steered by Doppler to produce a range variation of between 1 and 1000 cm. During this sweep, the range was recorded and compared directly to the commanded range. The resulting maximum 40 cm range tracking error indicated proper function.

To verify attenuation performance, various ranges were commanded and the attenuation recorded. The recorded values were compared to free space loss computed from the commanded range. The results indicated desired performance, however it was determined that choosing Digital Step Attenuators (DSAs) with slightly different error properties and increasing their number would better satisfy requirements. An additional component of attenuation testing required the suppression of side-band frequencies. The highest side-band measured was 50 dB below the center frequency peak power and sufficiently outside the 2 MHz bandwidth, and thus, was accepted.

In order to verify preservation of carrier modulation and Doppler tracking, and, as a coarse delay test, GPS Pseudo-Random Noise (PRN) 1 was input to PERFS and the frequency was stepped by 10 Hz every 100 ms for 10 s, producing a total shift of 1 kHz. The recorded data was processed with a software GPS receiver [12]. The carrier modulated data was preserved, and tracked. After accounting for an acquisition bias of approximately 170 m, the range tracking error was less than 10 m maximum magnitude over a range variation between approximately 100 m to 10 km. In addition, it was determined that the prototype exceeded by maximum required Doppler shift. This verified preservation of carrier modulated data and Doppler tracking, and provided an additional coarse delay test.

Determining the range resolution proved to be a challenge. For this test, an input sine wave at IF was split with constant Doppler applied to the signal passing through the prototype. The recorded data was post-processed for the phase residual and estimated measurement noise. The phase residual for several constant applied Doppler values consistently indicated a range resolution of 16.5 cm. For each case, the 3σ estimated measurement noise was well below the computed phase residual, lending confidence to the result.

The minimum range of the prototype was largely bounded by the performance of the SAW filters used. SAW filter specifications indicated a worst case of 1.6 μs of delay, or approximately 960 m of minimum distance for a pair of SAW filters. Measuring the path delay including SAW filters, DSAs, and required amplifiers produced approximately 875 m of minimum distance, while the Field

Programmable Gate Array (FPGA) contributed approximately 50 m to the minimum path distance for a total of 920 m total minimum path distance.

The results of the PERFS prototype testing prove the efficacy of the PERFS concept, and the prototype met or exceeded 7 of 17 production requirements. The testing did indicate that components for the production PERFS need to be chosen carefully to reduce noise, and component clocks must be phase-locked to avoid issues arising from a lack of synchronization. From the insight provided by the prototype testing, we can anticipate that the PERFS production units will meet all requirements.

CURRENT ACTIVITIES

Flight Processors

To provide a more realistic flight processor and development chain for FFTB simulations, four GE Fanuc CompactPCI-7055 single board computers have been procured. Each unit contains a 32-bit PowerPC 750GX processor, a Condor MIL-STD-1553 serial interface, a Picasso-LS RS-644 Low-Voltage Differential Signaling (LVDS) frame grabber, five USB 2.0 ports, three IEEE-1394 Firewire ports, a 10/100 Mbit Ethernet port, a Gigabit Ethernet port, an RS-232 port, and two RS-422 serial ports. The numerous supported interfaces will allow a wide range of hardware to be connected directly to the simulated flight system.

Magnetospheric Multiscale Mission Support

The spacecraft for the MMS mission [25] are in highly elliptical orbits with apogee altitudes up to 25 Earth Radii ($R_\oplus$). The four spacecraft orbits are designed such that the spacecraft are arranged in a pulsating tetrahedral formation with no closed-loop formation control. The Inter-spacecraft Ranging and Alarm System (IRAS) is an essential spacecraft component for the MMS mission. The IRAS enables absolute navigation via a state-of-the-art fast acquisition, weak-signal tracking, GPS receiver [2, 3, 33] and relative navigation via RF crosslink between spacecraft for ranging. It also provides spacecraft communications.

In support of the MMS missions, the FFTB has been selected as the relevant environment for TRL 6[2] testing of the IRAS devices. Current FFTB activities in preparation for testing focus on component test plan, environmental model, and PERFS interface development, as well as limited component testing as hardware becomes available. The six PERFS production units required for IRAS testing must be validated prior to commencing IRAS testing. Test activities for TRL 6 will include: GPS acquisition, tracking, and raw measurement accuracy; GEONS Convergence; and crosslink antenna transition, Time Division Multiple Access (TDMA) function, and raw measurement accuracy. In addition, several full system function tests will include multi-day orbital trajectories that cover selected mission phaseses.

FUTURE DIRECTION

Building on successes such as the ST9 TRL 4 demonstration and the PERFS prototype testing, the Formation Flying Testbed continues to grow and to evolve as a modular, hybrid, dynamic simulation facility for end-to-end guidance, navigation, and control design and analysis of formation flying spacecraft. The core capabilities of the FFTB are expanded significantly by the ability to emulate the RF space environment between spacecraft for communications and navigation. When combined with the integration support provided by the FFTB software architecture for real-time HitL simulations, critical spacecraft crosslink hardware and software algorithms can be included

---

[2]TRL 6 represents a successful system or subsystem model or prototype demonstration in a relevant environment [18].

in-the-loop, providing capability to pursue simulations in new operational regimes that are of great interest:

Automated Rendezvous and Docking: In addition to the precision formation flying technology focus area, the President's Commission on Implementation of United States Space Exploration Policy [1] also identifies *Automated Rendezvous and Docking* (AR&D) as an enabling technology. Developing such simulations in the FFTB will require additional work for attitude sensing, determination and estimation. The addition of LVDS frame grabber hardware presents new possibilities for simulating visual sensors that were not previously available in the FFTB. Likewise, the ability to emulate the RF space environment provides for new possibilities in safe proximity operations and AR&D.

Libration Dynamics: Many of the previously mentioned missions are evaluating precision formation flying about collinear Libration points of the Sun-Earth or Earth-Moon systems. The considerably different orbital regimes between Earth orbit and Libration orbits require commensurately different dynamical models. Previously, incorporating Libration dynamics into the STARS truth environment simulator would have been a difficult task. However, recent improvements made to GEONS may provide a more manageable path to extending simulations to include Libration dynamics in both truth environment and flight software components of the FFTB architecture.

ACRONYMS

ASCII American Standard Code for Information Interchange
CCS Crosslink Channel Simulator
CCPS Crosslink Channel Path Simulator
CLT Crosslink Transceiver
DSA Digital Step Attenuator
$R_\oplus$ Earth Radii
EIST Enhanced Intel SpeedStep Technology
ESA European Space Agency
FE Flight Executive
FFTB Formation Flying Testbed
FMS FFTB Messaging System
FPGA Field Programmable Gate Array
GEONS GPS Enhanced On-Board Navigation System
GMSEC GSFC Mission Services Evolution Center
GN&C Guidance, Navigation, & Control
GPS Global Positioning System
GSFC Goddard Space Flight Center
HitL Hardware-in-the-Loop
IF Intermediate Frequency
IRAS Inter-spacecraft Ranging and Alarm System
LPT Low-Power Transceiver
LVDS Low-Voltage Differential Signaling
MMS Magnetospheric Multiscale
MOM Message-Oriented Middleware
NASA National Aeronautics and Space Administration
PERFS Path Emulator for RF Signals

1PPS	1 Pulse per Second
PRN	Pseudo-Random Noise
RF	Radio Frequency
RMS	Root Mean Square
SAW	Surface Acoustic Wave
SI	Stellar Imager
SPECS	Submillimeter Probe of the Evolution of Cosmic Structure
ST9	Space Technology 9
STARS	Spacecraft Trajectory and Attitude Real-time Simulation
TDMA	Time Division Multiple Access
TDRSS	Tracking and Data Relay Satellite System
TRL	Technology Readiness Level
UDP	User Datagram Protocol

REFERENCES

[1] E. C. Aldridge, Jr. (Chairman). A Journey to Inspire, Innovate, and Discover. Technical report, President's Commission on Implementation of United States Space Exploration Policy, June 2004. `http://tinyurl.com/d28rx`, Accessed Jan 23, 2007.

[2] W. Bamford, L. Winternitz, and M. Moreau. Real-Time Geostationary Orbit Determination Using the Navigator GPS Receiver. In *NASA Goddard Flight Mechanics Symposium*, October 2005.

[3] W. Bamford, L. Winternitz, and C. Hay. Autonomous Navigation at High Earth Orbits. *GPS World*, pages 1–7, April 2006. URL `http://tinyurl.com/3e3d9s`. Accessed August 1, 2007.

[4] D. Beasley and E. Hupp. NASA Selects Advanced Technology Providers. NASA News, July 2005. `http://tinyurl.com/aztru`, Accessed Jan 23, 2007.

[5] K. Carpenter. Stellar Imager. `http://hires.gsfc.nasa.gov./~si/`, Accessed Jan 23, 2007.

[6] European Space Agency. Darwin Mission, . `http://tinyurl.com/86n5h`, Accessed Jan 23, 2007.

[7] European Space Agency. Proba-3 Mission, . `http://tinyurl.com/34apzm`, Accessed Jun 14, 2007.

[8] K. Fisher, Diane and N. J. Leon. New Millennium Program's Space Technology 9 (ST9) Project. `http://nmp.jpl.nasa.gov/st9/`, Accessed Jan 23, 2007.

[9] D. Gaylor and B. Bamford. FFTB Messaging System Interface Specification. Technical report, Emergent Space Technologies, Inc., December 2005.

[10] D. E. Gaylor and B. W. Barbee. Algorithms for Safe Spacecraft Proximity Operations. In *AAS/AIAA Spaceflight Mechanics Meeting*, January 2007.

[11] K. C. Gendreau, W. C. Cash, P. Gorenstein, D. L. Windt, P. Kaaret, and C. Reynolds. MAXIM: The Black Hole Imager. In *Proceedings of the SPIE*, volume 5488, pages 394–402, 2004.

[12] G. W. Heckler and J. L. Garrison. Architecture of a reconfigurable software receiver. In *Institute of Navigation GNSS*, 2004.

[13] C. Hunt, C. Smith, and R. Burns. Development of a crosslink channel simulator. In *Proceedings of the IEEE Aerospace Conference*, volume 2, pages 1322–1328, 2004.

[14] E. D. Jensen. Time/utility functions, 2007. URL `http://tinyurl.com/3yvzb6`, Accessed Aug 2, 2007.

[15] D. Leisawitz. Submillimeter Probe of the Evolution of Cosmic Structure. `http://space.gsfc.nasa.gov/astro/specs/`, Accessed Jan 23, 2007.

[16] J. Leitner. A hardware-in-the-loop testbed for spacecraft formation flying applications. In *Proceedings of the IEEE Aerospace Conference*, volume 2, pages 615–620, 2001.

[17] C. D. Locke. *Best-effort decision-making for real-time scheduling*. PhD thesis, Department of Computer Science, Carnegie Mellon University, Pittsburgh, PA, 1986. URL `http://tinyurl.com/3br2fc`, Accessed Aug 2, 2007.

[18] J. C. Mankins. Technology Readiness Levels, 1995. `http://www.hq.nasa.gov/office/codeq/trl/`, Accessed May 23, 2007.

[19] J. W. Mitchell and R. J. Luquette. Recent developments in hardware-in-the-loop formation navigation and control. In *NASA Goddard Flight Mechanics Symposium*, October 2005.

[20] J. W. Mitchell, D. M. Zakar, R. D. Burns, and R. J. Luquette. A message oriented middleware for a soft real-time hardware-in-the-loop spacecraft formation flying testbed. In *AIAA Modeling and Simulation Technologies Conference*, 2006.

[21] J. W. Mitchell, P. J. Baldwin, R. Kurichh, B. J. Naasz, and R. J. Luquette. Characterization of a Prototype Radio Frequency Space Environment Path Emulator for Evaluating Spacecraft Ranging Hardware. In *AIAA Guidance, Navigation, and Control Conference*, 2007.

[22] B. J. Naasz, R. D. Burns, D. Gaylor, and J. Higinbotham. Hardware-in-the-Loop Testing of Continuous Control Algorithms for a Precision Formation Flying Demonstration Mission. In *18th International Symposium on Space Flight Dynamics*, 2004.

[23] NASA GSFC Mission Engineering and Systems Analysis Division. GEONS Open Architecture Solutions for Onboard Orbit Determination in any Orbit, 2005. `http://geons.gsfc.nasa.gov/`, Accessed Jan 23, 2007.

[24] L. M. Robillard, H. R. Callison, and J. Maurer. Extending the DII COE for Real-Time. *CrossTalk: The Journal of Defense Software Engineering*, 12(6):6–12, September 1999. URL http://tinyurl.com/35az9p (HTML), http://tinyurl.com/3327vo (PDF), Accessed Aug 2, 2007.

[25] D. Smith and G. Colón. Magnetospheric Multiscale Mission. `http://stp.gsfc.nasa.gov/missions/mms/mms.htm`, Accessed Jan 23, 2007.

[26] *SimGEN User Manual*. Spirent Communications, September 2006. Issue 1-20.

[27] *SimREMOTE User Manual and Interface Control Document*. Spirent Communications, September 2006. Issue 2-00.

[28] Spread Concepts, LLC. The Spread Toolkit, 2005. URL `http://spread.org/`. Accessed September 5, 2007.

[29] Swedish Space Corporation. Prisma Mission. `http://tinyurl.com/2p4z3u`, Accessed Jun 14, 2007.

[30] The Mathworks, Inc. MATLAB, 2007. URL `http://www.mathworks.com/`, Accessed Aug 2, 2007.

[31] N. Wagner. Scientific Tools for Python: SciPy, 2007. URL `http://www.scipy.org/`, Accessed Aug 2, 2007.

[32] N. E. White and P. Newman. Micro-Arcsecond X-ray Imaging Mission, 2007. `http://maxim.gsfc.nasa.gov/`, Accessed Jan 23, 2007.

[33] L. Winternitz, M. Moreau, G. Boegner, and S. Sirotzky. Navigator GPS Receiver for Fast Acquisition and Weak Signal Tracking Space Applications. In *ION GNSS, the 17th International Technical Meeting of the Satellite Division of The Institute of Navigation*, pages 1013–1026, September 2004.

[34] World Wide Web Consortium (W3C). Extensible Markup Language (XML) 1.0. Technical report, June 2004. URL `http://tinyurl.com/3kxga`. Accessed August 5, 2005.