Vol. 1-333

The Personal Satellite Assistant: An Internal Spacecraft Autonomous Mobile Monitor

Gregory A. Dorais and Yuri Gawdiak
NASA Ames Research Center
MS 269-2
Moffett Field, CA 94035
650-604-4851
gdorais@arc.nasa.gov, ygawdiak@hq.nasa.gov

Abstract—This paper presents an overview of the research and development effort at the NASA Ames Research Center to create an internal spacecraft autonomous mobile monitor capable of performing intra-vehicular sensing activities by autonomously navigating onboard the International Space Station. We describe the capabilities, mission roles, rationale, high-level functional requirements, and design challenges for an autonomous mobile monitor. The rapid prototyping design methodology used, in which five prototypes of increasing fidelity are designed, is described as well as the status of these prototypes, of which two are operational and being tested, and one is actively being designed. The physical test facilities used to perform ground testing are briefly described, including a micro-gravity test facility that permits a prototype to propel itself in 3 dimensions with 6 degrees-of-freedom as if it were in an micro-gravity environment. We also describe an overview of the autonomy framework and its components including the software simulators used in the development process. Sample mission test scenarios are also described. The paper concludes with a discussion of future and related work followed by the summary.

TABLE OF CONTENTS

1. INTRODUCTION

The Personal Satellite Assistant (PSA) project is a NASA research and development activity to design an intelligent, small, free-flying, remote-sensing vehicle capable of autonomously navigating in three dimensions within a pressurized, micro-gravity environment, diagnosing systems in its environment, and interacting with people such that it is useful, easily understood, and easily commanded in a minimally time-consuming manner. The primary operating environment is the International Space Station (ISS), but other environments include the Space Shuttle and future manned spacecraft, such as one designed to carry a crew to Mars. The PSA has various environmental and equipment sensors as well as audio/video human-interface devices. It can be remotely commanded at various levels of autonomy and also can be commanded by simple speech commands and human motions.

Mission Roles

The two primary mission roles the PSA is being designed to address are to improve spacecraft crew productivity and to decrease mission risk by serving as part of an integrated spacecraft systems health management system.

Spacecraft health-management support role—the PSA will provide mobile monitoring, diagnosis, and communication capabilities. The PSA is being designed to supplement the spacecraft's Environmental Control Life and Support System by measuring temperature, pressure, humidity, and various gas levels (e.g., oxygen, CO_2) and recording a visual log as it traverses the spacecraft. The PSA will help diagnose and calibrate spacecraft sensors, temporarily replace faulty environmental sensors, generate acoustic, temperature, and gas concentration maps, locate gas and fluid leaks, filter atmospheric particles, as well as characterize heat sources with its infrared camera.

---

[1] U.S. Government work not protected by U.S. copyright

[2] 2003 IEEEAC paper #1091

Crew productivity role—the PSA will provide several support capabilities including: remote visual monitoring and task recording, video and data display, payload & core system knowledge management terminal, inventory location and tracking, just-in-time training, and standard PDA functions (schedule, notes, activity lists, and calculation functions). These capabilities will directly support on-orbit crews in the daily execution of payload experiment and core system tasks.

To support the flight crews, ground crews, and payload scientists, the PSA can be used for monitoring and communication using its audio and video sensors as well as perform videoconferencing and display a variety data on its LCD screen. The PSA will allow ground crews and scientists to be virtually located inside the spacecraft. Moreover, the PSA's autonomy capabilities will allow remote users to interact with the crew and spacecraft in a human-centered way while providing real-time data collection and communication.

The ISS, for example, is an extremely ambitious operational environment for the crew (3-6 members) with tens of thousands of inventory items to track within it and hundreds of experiments to manage covering a wide spectrum of science disciplines. The overall productivity of the ISS can be increased by automating or otherwise reducing the crew time required to perform tasks as well as enhancing or enabling science activities that would otherwise not be performed due to insufficient crew time by means of a PSA.

This paper describes the PSA project rationale, high-level functional requirements, project implementation approach including a description of five prototypes and their test facilities, current project status, autonomy technology components, test mission scenarios, and future mission goals.

2. PSA RATIONALE

The impetus for the PSA project is to develop an intelligent system that increases crew productivity and reduces risk for manned missions. It was originally an element of the NASA Cross-Enterprise Technology Development program and has subsequently been adopted by the Intelligent Systems project in the NASA Computing, Information, and Technology program, which primarily focuses on enhancing its autonomy, and the NASA Engineering Complex Systems (ECS) program, which is responsible for the overall development effort to raise its technological readiness level.

The ECS program was formulated by NASA to address growing concerns about the agency's ability to develop, operate, and maintain the complex systems required to meet our current and future mission objectives.

During the program formulation phase the ECS program identified four common problem classes associated in most NASA systems:

1. Limited system and trade space analysis capabilities
2. Poor understanding of system, human and organizational risk
3. Incomplete knowledge acquisition and communication
4. Inadequate state assessment and brittle control strategies

Based on these problem classes, the ECS program developed a three-pronged WBS solution approach:

1. System Reasoning & Risk Management
2. Knowledge Engineering for Safe Systems
3. Resilient Systems and Operations

The PSA project is part of the Resilient Systems and Operations thrust. PSA's main focus is addressing problem class number four: "*Inadequate state assessment and brittle control strategies*", but it also addresses problem class number three "*Incomplete knowledge acquisition and communication.*"

The PSA near term objectives are organized around addressing these two problem areas during the operations and maintenance phases of a mission lifecycle. Within these phases, the PSA provides support in two distinct parts of in-flight spacecraft system operations: spacecraft health management and crew productivity. The spacecraft health management goal relies primarily on developing advanced Fault Detection, Isolation, & Recovery (FDIR) technologies for Environmental Systems. The crew productivity goal relies primarily on advanced Knowledge Management (KM) technologies. In addition, underlying many of the advanced FDIR and KM technologies are advanced autonomy technologies to reduce the user time and complexity required to benefit from the FDIR and KM technologies.

Fault Detection, Isolation, & Recovery

In this area, system designers have to make difficult trade-offs on the appropriate number, type, location, and redundancy of environmental sensors to meet critical life support requirements. In spacecraft such as the ISS and Space Shuttle, the Environment Control & Life Support System (ECLSS) Crit. 1 system is required to meet two fault tolerant requirements. Part of meeting these requirements may be achieved through redundancy. Redundancy in general has a significant penalty in terms of weight, volume, and cost. Additionally, it is very susceptible to common cause failures. The PSA's mobility capability can have a significant impact on the redundancy strategy by providing sensing in an autonomous & dynamic fashion. Instead of having to hardwire a large quantity of sensors that might never be used and absorbing the associated weight, volume, development time, and cost hits in a spacecraft design, the PSA "on-demand redundancy" significantly lowers the number of overall sensors that are required to be installed. Additionally, the mobile nature of the PSA platform allows for a more thorough and systematic monitoring of a given spacecraft module thereby detecting anomalies and failures with a higher probability than a given fixed sensor.

The specific PSA FDIR functional goals include:

Fault Detection—

- Monitoring of environmental gases: (e.g., oxygen, CO_2), temperature, pressure, and humidity (and keeping crew members informed as to when changes may cause health risks)
- Detecting and monitoring internal structural flaws or failures: cracks and leaks
- Off-gassing of harmful chemicals: hydrazine, others
- Sensor failures/calibration errors

Isolation—

- Pinpointing the location of gas leaks or heat sources
- Pinpointing the location of structural flaws or failures
- Identifying specific failed sensors

Recovery—

- Taking over the role of a failed fixed sensor
- Calibrating a fixed sensor after it has been replaced
- Augmenting the monitoring of the environment in a given location by operating multiple PSA's in the region (such as on the Mir Space Station after the fires were put out—additional sensors at locations affected by the fire would have been helpful to make sure hot spots were closely monitored or if new flare ups occurred, they were quickly identified).

The FDIR environmental capabilities of the PSA are applicable to both core and payload systems. Beside the obvious implications for crew and payload safety, there is also the key aspect of ancillary data quality for science payload research. During certain payload experiment phases additionally monitoring can be delivered on demand to improve the quality/resolution of the ancillary environmental data being collected.

Another FDIR design goal for the PSA is for it to be able to intelligently interact with the crew, spacecraft systems (e.g., ECLSS), and other multiple PSA's. The mobility aspect of the PSA enables flexible verification and optimization of tasks for dealing with environmental system failures. Spacecraft systems can use the PSA to verify common false positive sensor feedback before having to activate expensive (power-wise, schedule) and (sometimes more dangerous backup)[3] systems. Ground and flight crews can use the PSA's as virtual extensions of their senses enabling them to cover more options and collect information faster than by using traditional resources. Moreover, multiple PSA's can be coordinated to cover more space quickly in hunting down, isolating, and recovering from failure scenarios.

Knowledge Management

The PSA KM functional goals cover: crew personal assistant functions (schedules, notes, data access, etc.), core and payload procedure support, inventory tracking, training, data recording, and maintenance. These features allow the crew to leverage cutting edge information technologies to improve their own productivity. The PSA will tap into both the core and payload networks and will be able to interact with crew via keyboard and voice recognition interfaces. These inputs and the PSA's intelligence will allow it to function in complementary, parallel, and independent modes with the crew. Examples include "look ahead" capabilities involving checking crew members schedules, current location, the next activity requirements, spacecraft mode & status, and doing inventory checks to make sure appropriate resources are available for the next task at hand. The training, maintenance, procedure support functions will heavily leverage the PSA multi-media features including: audio and video I/O. These components will enable video conferencing capabilities allowing remote personnel to dynamic interact with the onboard crew. Remote manipulation of the PSA will allow collaborative inspections, procedure support and experiment consultations to take place while freeing both of the crewmember's hands to focus on execution of a given task in the position and at the location desired.

A Day in the Life of the Personal Satellite Assistant

Consider the following hypothetical scenario where a PSA interacts with a crewmember and ISS equipment and systems throughout the day. This scenario showcases the range of PSA functionality and flexibility to support on orbit operations.

The PSA assigned to astronaut Mary onboard the International Space Station wakes her up to her favorite song. As Mary wakes up, the PSA uses its wireless network access to status networks and servers to provide a user-prioritized list of information Mary likes to have in the morning: action items, ISS location and mode, any overnight anomalies, e-mail from home, and world news. As she exercises and eats breakfast, the PSA briefs her on her last in-space physical as well as preps her for the first experiment of the morning. The PSA reads the procedures,

---

[3] Going to backup systems often entail new procedures, new systems being turned on for the first time for extensive use and other embedded risks

her ground notes, and the biography on the Principle Investigator (PI) of the experiment.

As she and the PSA move to the experiment station, the PSA notices an inconsistency between its environmental sensors and the station's in the Node 2. An auxiliary PSA is requested by the ground crew to verify the readings.

Mary's morning experiment involves the use of an ISS glovebox: a sealed, windowed environment with two gloves attached to a wall that permits a crew member to use both hands to manipulate objects in the environment without breaking the seal. Mary's PSA's control is transferred to the ground to allow the PI to be virtually present as Mary conducts the experiment on orbit. The PI is able to fly around the operation and give real-time consultation and evaluations while Mary is free to use both hands to conduct the tasks. The PI also takes advantage of the PSA being around and requests for additional ancillary environmental data to be collected by the PSA and integrated into the experiment logs.

As Mary finishes up the experiment ahead of schedule, the PSA notifies Mission Control who updates her schedule. The PSA checks the latest updated schedule for the next task to make sure all the resources are available for Mary to perform it. The PSA discovers that a wrench is missing for the core system filter replacement task next on Mary's list and asks Mary for permission to initiate a search. Mary grants permission and the PSA initiates a search pattern for the wrench using RFID to identify and verify inventory within the station. The PSA fleet commander, an intelligent server monitoring and supporting all the PSA's, is notified of Mary's PSA inventory activity and dedicates two more unassigned PSA's to help conduct the inventory sweep. The three PSA's working together institute an efficient search pattern. The wrench is quickly found in one of the new modules and the crew is notified of the location and the inventory database is updated.

As Mary and her PSA approach their next task, the PSA, constantly monitoring real-time status data and Mary's preferences, detects an opportunity event: a hurricane has developed over Cuba and Mary likes to monitor them and take pictures when she can. Since the next task is a 2-person activity and the second crewmember hasn't arrived yet, Mary gets permission from Mission Control to take time for her hobby.

During the core system filter replacement, the ground crew requests permission to use the PSA for a nearby structural inspection. The astronauts don't need the PSA at the moment and the ground team uses the PSA's thermal imaging capabilities to inspect a trouble spot in a nearby docking node. Nothing is found, but for safety reasons the ground has the PSA provide additional monitoring capability during an automated docking procedure. This enables the ground to have extra sensor data without requiring crew time or increasing risk to the crew.

After the docking, the PSA returns to Mary and supports her through the rest of the day. After Mary and the rest of the human crew goes to sleep, the PSA recharges in preparation for its assigned patrol duty. It's given several priority areas to monitor where recent mishaps have occurred; a fire and a pressure leak are high on the ground crew's list to monitor. As the PSA moves out on its evening patrol, it passes by another PSA system at the sensor location that it had identified earlier as having a calibration error. The sensor in question has since failed. The other PSA is now providing substitute sensing until a new sensor is installed.

3. FUNCTIONAL REQUIREMENTS

In order to develop a system that can meet the goals previously described, we defined a number of requirements. In this paper, we will discuss a few of the higher-level functional requirements that may be of particular interest.

Requirement 1—create a self-contained portable device with environmental sensors, computational capabilities to analyze the data and perform diagnoses, and a display for viewing sensor data. The sensors are to function inside the ISS and similar operating environments. The high priority sensors include those that measure local temperature, atmospheric pressure, humidity, and gas concentrations including O_2 and CO_2. Lower priority sensors include visible-light still and motion cameras, thermal imager, Geiger counter, NDIR spectrometer, electromagnetic detector, RFID tag detector for inventory management, microphone, and a directional acoustic detector array for localizing emissions. Our first attempt at designing a system that met this requirement resulted in a "breadboard system" that could conceivably have been packaged into a system similar to the Star Trek® "tricorder." However, additional requirements caused us to abandon the tricorder paradigm.

Requirement 2—stamp the sensor data with the time and a 6-DOF position of the sensors relative to the environment. The 6 degrees-of-freedom (DOF) correspond to X, Y, Z translations and yaw, pitch, roll orientations relative to a global origin. Although meeting the time element of this requirement can readily be achieved with a clock, satisfying the position element is challenging. A number of approaches were examined resulting in two major solution classes. The first requires engineering the environment with active devices, such a beacons for a local GPS system, or passive fiducial marks that can be detected by sensors on the device. The second class of solutions includes those that require motion, proximity, and feature detectors on the device that enabling registering to an internal map. The benefits of engineering the environment are that it is much easier to create such a system and the portable sensor device can be smaller. The disadvantages include the costs to retrofit, safety-qualify, and maintain active electromagnetic emission devices on ISS; the mass of the equipment required for coverage throughout ISS; and the difficulty in testing. The benefits of enabling the device to localize itself include avoiding the disadvantages of engineering the environment, not being dependent on external devices for operation, and the same sensors used for self-localization can be used to detect dynamic objects not on the PSA's internal map. The challenge is that the general ability for a mobile system to accurately determine where it is and how it is moving relative to other objects in its environment is one area where sophisticated computer systems still lag behind the computational capabilities found in many "simple" biological creatures. Our current approach is to develop a system that can do self-localization using a combination of stereo-cameras to build depth maps and sense motion by means of optic flow algorithms and fuse this with data from a 6-DOF inertial measurement unit (accelerometers), and proximity sensors. As necessary, we can mitigate risk by engineering the environment as needed.

Requirement 3—"station-keep" on command by maintaining a fixed position and orientation relative to its environment. Note that the environment, i.e., the ISS, is continually in motion as it orbits the Earth and performs minor attitude adjustments. Although the device is useful if it is held by a crewmember or fixed to a surface, the more that can be done with the device that doesn't require crew time, the better. For many tasks, crewmembers require both hands so the sensor device cannot be held. Moreover, some monitoring tasks require taking a measurement at the same location for a period of time. By satisfying this requirement, a crewmember's valuable time can be applied to other tasks.

Requirement 4—navigate to various positions on command, avoiding static and dynamic obstacles. This requirement can be viewed as a corollary of requirements 2 and 3. If the system already has the sensors, controllers, and actuators to determine its absolute position and maintain it, enabling it to navigate requires no additional hardware. Allowing the system to navigate to various positions again increases the flexibility of the system while decreasing the crew time required to perform a task. For example, searching for a leak or a measuring gas concentrations throughout a module can be quite time-consuming. The task is more efficient if it doesn't require a crewmember to be present even if the task takes longer.

Requirement 5—minimize the time required by the crew while enabling crewmembers to command the system at the level of autonomy they desire. This requirement is in keeping with the general principle that crew time is extremely valuable. In some cases, this means that totally autonomous systems are preferable to manual systems. However, there are cases where autonomous systems require more crew time because the overhead in figuring out how to command the system to do what is desired autonomously is greater than doing it manually. A simple example of this phenomenon involves VCRs where people do not bother to program them because it is too complicated and too easy to make a mistake as opposed to performing the task manually. For more complex systems, in particular the system we are discussing, this problem is magnified significantly. Consequentially, the requirement is essentially for the system to be adjustably autonomous. If necessary, another system, such as the environmental life support system can command it to localize a heat source without requiring any crew intervention. In another task, a crewmember can command it to go to a certain location and notify him upon arrival, at which time the crewmember teleoperates the system as desired. In order to achieve this requirement, the system must have mixed-initiative planning, scheduling, and execution capabilities, and the ability to effectively communicate with the human operator so the operator understands what the system is doing and why it is doing it, and the system can interpret what the operator wants and can translate it into commands it can execute.

Requirement 6—perform continuous active hybrid temporal-variable diagnostics on its environment and equipment in it. We define a diagnostic system here to be one that determines the sets of likely states of the system being diagnosed consistent with the observations and the model of the system. A temporal-variable diagnostic system can use observations that change over time, e.g., recognize trends. A hybrid diagnostic system is one that can reason given both continuous-valued and discrete-valued observations. Typically, different approaches are used for continuous and discrete-valued observations, but many systems require that both be reasoned about simultaneously. An active diagnostic system is one that determines what additional observations are needed to disambiguate the state of the system being diagnosed. For example, consider a system with a HIGH-TEMPERATURE warning light that is on. Two possible diagnoses are that the system is indeed overheating or the temperature sensor is faulty. By actually checking the temperature of the system, we can then determine more accurately which of these two diagnoses is more likely correct. One of the uses of this portable sensor device is as part of a larger Integrated Vehicle Health Management (IVHM) system so having this diagnostic capability can increase the likelihood of early detection and accurate diagnosis of problems without requiring crew time.

Although there are several other requirements, these six functional requirements effectively constrain the space of possible solutions. Other notable requirements involve safety, reliability, and ease-of-use. In particular, a smaller overall size and longer operation between recharges is better.

4. PSA PROTOTYPES AND TEST FACILITIES

The PSA project is using an iterative, rapid prototyping development approach. We started by envisioning the end product in our minds. In 1998, the project's first year, a concept model, shown in Figure 1, was developed.

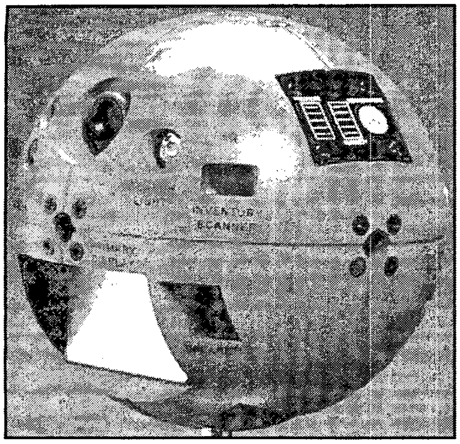

Figure 1 - PSA Original Concept Model

With this model, we sought to answer the question, "if we could build it, would we want to?" The effort spent in developing a handheld mockup was well worthwhile. Not only did team members find having a physical concept model useful to convey ideas, we found people outside the project grasped what we were doing much more readily when we used the model in our presentations.

Technology Challenges

Having fleshed out the concept, we began a critical analysis of the problems and risks as well as the needed technologies currently or imminently available, and those that needed to be developed. The tall-pole issues we identified were:

1. Continuous 6-DOF position and velocity estimation of the PSA relative to its objects within its environment and the environment, i.e., spacecraft, itself.
2. Adjustably autonomous control, from high-level mission commands to low-level controller commands, and the corresponding user-interfaces
3. Active hybrid diagnosis capabilities with multiple agents
4. Safety qualification. Standards for autonomous IVAs inside ISS have not been developed. Other safety concerns include autonomous battery recharging and acoustic limits.
5. Onboard wireless bandwidth available.
6. Miniaturization of component technologies while protecting against radiation-induced failures. With the exception of the LCD, the smaller, the better for a safety and usability.

The project's primary foci are mitigating risk on issues 1 – 3. Issues 4 – 6 are also being worked but at a lower level. In addition, significant effort was invested in engineering prototypes, simulators, and test facilities in order to validate that we have captured and addressed the salient issues.

For example, spacecraft technology had basically solved the propulsion and attitude control problem by means of cold-gas thrusters and reaction wheels. Although reaction wheels that met our requirements were not commercially available, we were convinced they could readily be developed. Safety concerns of having and refueling a high-pressure tank that would enable extended operation soon caused us to rule out cold-gas as a primary means of propulsion. However, it does remain as a option for a special-purpose PSA, such as one for exploring depressurized regions of a spacecraft. Instead, we began examining fans and blowers for propulsion.

Unlike most spacecraft, the PSA is required to navigate in close proximity to both static and dynamic objects. Moreover, unlike ground vehicles, the PSA has no "brakes" to stop and safe itself with while it determines where it is relative to what's around it and what it should do in an environment that is continually moving. Moreover, simply moving one meter forward is problematic since the PSA has no direct way of measuring odometry whereas a ground vehicle can move fixed distances by simply counting wheel rotations. Unfortunately, the error that accumulates by attempting to determine distance traveled by double integrating the measured accelerations is useful only for very short distances.

PSA Model 1

As previously discussed in the functional requirements section, we began looking at using stereo vision and engineering the environment as possible methods for performing continuous localization. In addition, we also began looking at fusing stereo depth maps with proximity sensors for obstacle avoidance. In order to perform tests, in 1999 we developed at 3-DOF testbed called the Model 1, which is shown juxtaposed to the concept model in Figure 2.

In the Model 1 design, no attempt was made to conform to the packaging eventually required. By equipping the Model 1 with a stereo-vision systems, we are able to perform vision-based localization and visual servoing. My means of a wireless Ethernet, a high-level model-based autonomous control system located on a remote server commands the Model 1 to execute various missions.

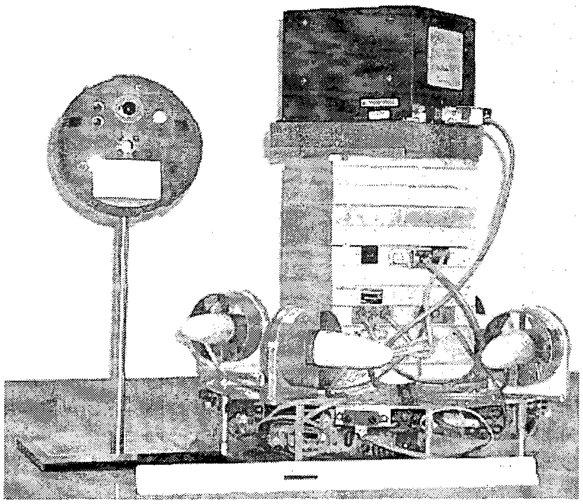

Figure 2 - PSA Concept Model (left) and PSA Model 1 Prototype (right)

Low-friction Planer Test Facility

To test the Model 1 prototype in 3-DOF (X, Y, yaw) and subsequent prototypes in up to 5-DOF (vertical translation is not supported), we have developed a low-friction planer test facility. It consists of a 12' x 12' granite table polished and leveled so that it functions as a smooth, horizontal plane. A 12" circular plate was attached to the bottom of the Model 1 on which a small compressor was mounted that pressurizes the space between the plate and the table. This enables the Model 1 to float on a thin cushion of air as it translates and rotates on the table propelled by its fans. Above the table, a digital camera was mounted so that most of the table is within its field of view. Three LED's were mounted on the top of the Model 1 so that machine vision software can track the actual location of the Model 1, i.e., provides ground truth. We use this external localization capability to measure the accuracy of the onboard position estimation system. In addition, for unit test purposes we can configure the onboard control system to use the externally-calculated position data stream as feedback to achieve its commanded position and velocity trajectories.

PSA Model 2

While the Model 1 was being tested, development of a 6-DOF Model 2 and an accompanying 6-DOF micro-gravity test facility was underway. We were well aware that the position and velocity estimation problem in a 3-DOF system was much simpler than in a 6-DOF system. The existing algorithms we examined that work well in a 3-DOF system did not work at all in a 6-DOF system, primary due to the inability to disambiguate motion between the various degrees of freedom. To address this, we began a joint-research effort with SRI International to extend elements of 3-DOF vision systems it developed to 6-DOF. We developed a vision testbed that supports testing vision algorithms that use one to four stereo-pair cameras. The Model 2 was developed to support up to four stereo pair cameras. In addition, a 12" sphere was created directly from CAD drawings using a stereo-lithography process.

Propulsion and attitude control 6-DOF (X, Y, Z, yaw, pitch, roll) were achieved using 6 fan pairs located in 6 ducts. It would have been preferable to use reaction wheels for several reasons including they would provide tighter control, quieter operation, and greater energy efficiency. However, reaction wheels that met our specifications would have to be custom built so they were scheduled to be implemented as part of the Model 3 prototype.

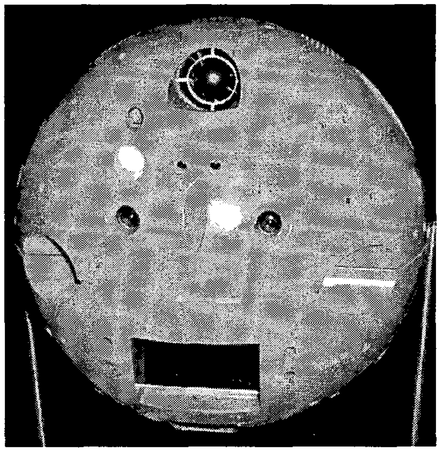

Figure 3 - PSA Model 2 supported on pitch & yaw gimbal

The Model 2 became operational in 2001 and currently is being tested on a pitch & yaw gimbal shown in Figure 3. It is 12" in diameter to enable the use of commercially available components as much a possible and to give space to make changes after it was constructed. For example, the primary core is comprised of a PC104 stack of boards. The gimbal is mounted on a pressurized plate, which permits it to float on the table in the low-friction planer test facility similar to the Model 1, permitting motion in 4-DOF.

The Model 2 has a 3.8" diag. LCD located at the center of its front lower hemisphere. The LCD can be used to display data generated locally as well as data received via its wireless network, e.g., text terminals, images, schematics, videos. The LCD was purposefully designed to be small since we expect the flight version of the PSA to be significantly smaller where fitting a large LCD is problematic. The smaller the display, the closer the user must be to it. However, users generally prefer larger displays and the PSA will attempt to maintain a safe distance from a user that gets too close.

Micro-gravity Test Facility

The Model 2 is also being tested in a micro-gravity test facility currently under development. Currently, the micro-gravity test facility supports 5-DOF motion (X, Y, Z, yaw, pitch). A 3-DOF gimbal, which permits yaw, pitch, and roll motion, is expected to be operational soon.

The micro-gravity test facility is roughly 36' long, 13' wide, and 8' high. This is large enough so that the interior volume of any one ISS module can fit underneath. It consists of a 3-DOF (X,Y,Z) bridge-crane-like mechanism that supports a passive 3-DOF gimble that permits free spinning in yaw, pitch, and roll. The bridge moves up and down the length of the facility. The trolley moves along the bridge permitting the trolley to move to any (X,Y) coordinate in the test facility. A crane on the trolley raises and lowers a gimbal attached to it. The object to be tested is mounted in the gimbal and balanced so that it freely spins and doesn't "wobble." The micro-gravity test facility can be operated in the following four modes.

1. Velocity mode. This is the most common mode used to control bridge cranes. When a crane axis motor is activated, it runs at a specified velocity until stopped.
2. Position mode. The crane servos to a specified (X,Y,Z) location then stops.
3. Force mode. The commanded force increases or decreases the crane motor velocities. When no force is commanded the crane motors continue at a constant velocity.
4. Force-neutralization mode. Instead of commanding motor forces, sensors located on the trolley and gimbal sense translation forces (X,Y,Z) acting on the gimbal payload and these signals are interpreted by the crane motors as force commands. The Z-axis signal is modified so that the constant force of gravity is zeroed out.

Both force modes (3 & 4) are currently being tested and expected to be operational in early 2003. These modes can be used to simulate micro-gravity as well as various fractions of Earth gravity. However, the force neutralization mode is the primary micro-gravity simulation mode. Either a human operator or the PSA Model 2 will be able to operate the micro-gravity test facility in any of these modes.

PSA Model 3

While testing continues on the Model 2, the preliminary design of the Model 3, shown in Figure 4, is nearing completion. The Model 3 is scheduled to be operational by the end of 2003.

If it is deemed important for risk mitigation, the Model 3 will be tested onboard a KC-135 aircraft performing a series of dives to simulate short periods of micro-gravity as used by astronauts for trainings. Otherwise, these tests will be first performed using the Model 4.

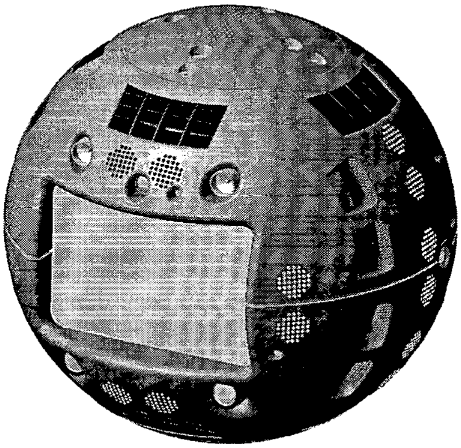

Figure 4 - PSA Model 3 Preliminary Concept Drawing

The most notable difference in the Model 3 compared to the Model 2 is the use of two blowers and four reactions wheels for propulsion and attitude control. Each blower, one is located at the top and the other at the bottom, exhaust through for actuated vents to propel the PSA. Although it is possible to control yaw, pitch, and roll with only three reaction wheels, a fourth reaction wheel enables momentum to be shifted among the reaction wheels to help avoid saturation of any one wheel. Saturation occurs when a reaction wheel can no longer provide torque because the rotational velocity of the wheel cannot be increased further. If necessary, the blowers can be used to provide a torque to desaturate the wheels.

Another notable difference in the Model 3 is that it will include additional environmental sensors, including a thermal imager. When completed, the Model 3 will be oversized and not space qualified, but otherwise will have all the capabilities planned for the flight model.

PSA Model 4

The primary focus of the Model 4 development is to miniaturize the prototype and increase its energy efficiency as opposed to adding capabilities. A 12" sphere is uncomfortably large for people to interact with and its associated mass decreases its operational time between recharges. Moreover, increased mass generally increases the risk of unsecured objects causing injury or damage. The development cost of the sphere increases significantly as the sphere diameter decreases. Moreover, the user benefit gained diminishes as the size decreases. Currently, we are targeting the Model 4 to be 8-9" in diameter.

The blowers and reaction wheels used in the Model 3 were specifically designed so that they can be scaled down without undesirable side effects. The most expensive task of the Model 4 development is expected to be moving from the COTS electronics, in particular the PC104 boards, used in the Models 1-3, to a custom chip set.

The largest non-structural component of the Model 4 by volume and mass will be its battery. We have begun researching various approaches to address this including using batteries as structural components, e.g., the external shell itself. However, concentrating the mass density in the sphere center reduces the energy required to turn the PSA. Moreover, there are operational advantages to quickly being able to replace a battery rather than waiting for it to recharge. Another design consideration is that batteries that can be charged and discharged quickly tend to not hold as much energy as ones that don't. Note that a critical propulsion measure is how quickly can the PSA stop, which is a function of its mass, velocity, and the maximum battery discharge rate. Initially, a hybrid battery approach seems to hold the most promise.

PSA Model 5

The Model 5 is targeted as the first PSA prototype to be flight-tested onboard a spacecraft. Its primary design focus will be to address outstanding issues related to flight qualification, which include component off-gassing, radiation shielding, battery safety, meeting acoustic and electronic noise standards, and other safety issues. In addition, the Model 5 is expected to incorporate design changes resulting from tests performed on the Model 4. Given funding and schedule constraints as well as user feedback on the usability of the Model 4, we may also attempt to reduce the diameter of the Model 5 to as small as 6".

5. AUTONOMY FRAMEWORK AND SIMULATORS

An autonomy framework designed to address the previously discussed operational requirements has been developed and is depicted in Figure 5. The same software is used to command the PSA Model 1 and Model 2 as well as the PSA in simulation. Care was taken to design and implement this framework so that it is applicable to a wide range of free-flying vehicles and are exploring applying it to other domains, e.g., UAV.

The user can issue commands to the PSA through the Crew GUI. Also, the user can issue verbal commands to and receive spoken notifications generated by the PSA via a headset. Other external systems, including other PSA's, can directly and simultaneously issue commands to the PSA, which will attempt to resolve any conflicts. Finally, the PSA itself can generate commands in keeping with its high-level goals and periodic task schedule.

The PSA autonomy framework is comprised of a number of control elements, which are represented as boxes in Figure 5. The current implementation is distributed over three processors, as indicated by the dashed boxes, which are connected by wireless Ethernet. Each of these three subsystems and the control elements it contains is briefly discussed below. Note that the framework design and many of its elements draw their heritage from the model-based, goal-achieving, temporally-flexible NASA "Remote Agent" autonomy software flight-validated on the Deep Space One spacecraft in 1999 [1].

Onboard Control System Elements

The onboard control system is responsible for sensing, sensor analysis (e.g. object fault recognition), state estimation (e.g., position estimation), hardware actuation (e.g., motor currents), and real-time reactive control (e.g., obstacle avoidance), generally with sub-second latency. This system is designed to enable local operation of the PSA even when communication with the off-board system is lost, which may occur during a flight emergency.

Local Path Planner—generates a trajectory between two waypoints that takes into account locally sensed obstacles When given a third waypoint, the trajectory passes through the second waypoint. The local path planner performs limited trajectory repair in case of a path plan failure, e.g., blocked path.

High-level controllers—primary responsibility is to translate the trajectory into a sequence of 6-DOF [position, velocity, and acceleration] setpoints for the low-level controllers.

Low-level controllers—primary responsibility is to translate the setpoints into motor force commands to achieve the specified PSA motion.

PSA Hardware—the sensors and actuators with their associated drivers. These include fan motor controllers, stereo cameras, environment sensors, proximity sensors, and an LCD.

Monitors—signal processing loops that abstract the data generated by the sensors. They run from being as simple as indicating that a proximity sensor has fired to continually calculating 6-DOF positions and velocities by fusing the stereo camera, 6-DOF accelerometers, and proximity sensors.

Communication Manager—responsible for managing message traffic and executing certain message handlers. Serves same role in both off-board systems.

Off-board Autonomy System

The off-board autonomy system is responsible for high-level autonomous control including inter-agent communication and coordination (including humans), goal management, decomposing high-level tasks (planning) into commands that can be executed by the onboard control system, e.g., waypoint commands, constraining task times (scheduling), command sequencing (plan execution), and reasoning about sensor data provided by the onboard control system, e.g., for diagnosis, and for plan repair, e.g.,

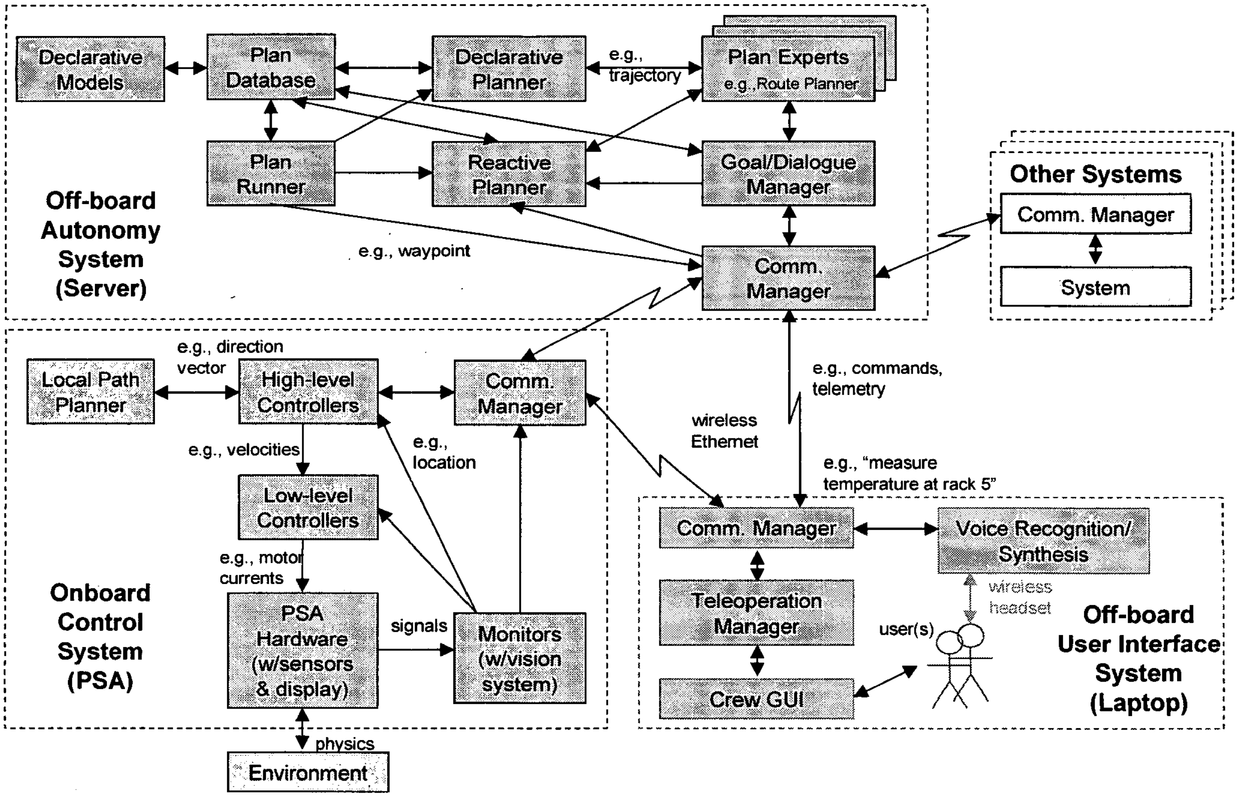

Figure 5 - PSA Autonomy Framework

onboard control system is unable to achieve a waypoint. Architecturally, this system could be integrated onboard the PSA. It is off-board to permit increased computational power that is not constrained by onboard size, power, and communication constraints.

Declarative Models—contains the library of constraints used by the Plan Database that define a set of coordinated state machines. A constraint may simply specify that Task A must precede Task B by at least 10 seconds but not more than 20 seconds. The constraint may also functionally relate the parameters of tasks A and B as well as specify preconditions as to when it applies.

Plan Database—contains the plan being executed and is responsible for automated sub-goaling of tasks, i.e., determining the set of sub-tasks required to achieve a task, and for maintaining flexible plans, i.e., the propagation of valid task variable domains that are minimally restricted without violating a constraint. It has been implemented using the EUROPA plan database developed at the NASA Ames Research Center. EUROPA is a derivative of the model-based, temporally-flexible Remote Agent Plan Database described in [2], an earlier version of which was demonstrated on Deep Space One [1]. The plan database represents a temporal, constraint-based network of tokens that defines the past, the present, and flexibly-defined future states and actions of the system. Each token represents the "state" of a state variable for a period of time and represent tasks that achieve or determine the state. The token data structure is a tuple that specifies the state variable, the procedure and its arguments that is invocated when the token is "executed," and the token start and end time bounds. The plan database supports multiple timelines with constraints on and between tokens. If none of the constraints are violated for a given instantiation of the plan database, the database is defined to be consistent.

Declarative Planner—responsible for scheduling outstanding tasks, and the related sub-tasks generated by the plan database, as well as making decisions regarding constraining the domains of task variables. This element is implemented by a variation of the Remote Agent Model-based Planner/Scheduler described in [2] and as specified by the Intelligent Distributed Execution Agent (IDEA) architecture [3]. More specifically, the declarative planner is responsible for generating a consistent, flexible plan in the plan database given a start and end horizon time bound, an initial state of the timelines at the start time, and a set of goals. A flexible plan is loosely defined as a set of timelines, each consisting of tokens on each timeline, token order constraints that prevent overlapping tokens on the same timeline, and token procedure variable constraints. Plan flexibility is characterized by the set of decisions yet to be made in a plan database that is consistent. A plan identification function is used to determine which of the outstanding decisions must be made in order to have a valid plan. The search process and decision selection priorities are determined in part by user-defined heuristics. Complex plans can require considerable computation time. The proper set of heuristics can dramatically reduce the time required. The declarative planner is called to initialize the plan database and also is called during plan execution as specified by the plan being executed. It is typically called to plan for a period of significant duration sufficiently in the future such that the deliberative planner will complete prior to the start time of this period, but not so far in the future that the initial state at the future start horizon is not known with high confidence.

Reactive Planner—responsible for insuring that the Plan Database is in a state such that the tasks to be executed at a specified time are unambiguous. It has been implemented as described in [3]. In many respects, as implemented the Reactive Planner is very similar to the Deliberative Planner described above, although that not need be the case. The salient differences between the two planners are that the Reactive Planner reasons over a shorter, more immediate time horizon, typically ending just after the current execution time; its plan identification function is more restrictive so decisions that were postponed by the Deliberative Planner must now be made; the time allocated for planning is relatively very short, typically less than a few seconds, and cannot be exceeded without a fault; and in the event of a plan deliberation or execution failure, the Reactive Planner is responsible for local plan repair or if necessary generating a standby plan to safe the PSA and calling the Deliberative Planner. Plan repair may be necessary for several reasons including tasks completing too late or too early, task return state variables posted to the Plan Database make it inconsistent, and new tasks have been added to the Plan Database for immediate execution that cause a conflict.

Plan Experts—computational procedures, called by a planner, that return information used by the planner to make planning decisions, typically regarding token variable values. For example, a route planner expert is called by either the declarative or reactive planner to determine the time, route, and energy required to move between two points in the environment or to cover a certain space. The route planner expert has access to a global map that can be updated with sensed obstacles. A route plan request is typically made by the deliberative planner as part of developing the initial plan, but may also be called by the reactive planner to develop an alternate route if necessary, e.g., the route is blocked or there is insufficient energy to complete the current plan. In addition, a user may initiate a request to answer a hypothetical question about a particular goal.

Plan Runner (command sequencer)—responsible for "executing" tokens in the plan database at the appropriate time. Executing a token involves calling the procedure with its arguments defined by the token, updating the plan database with the token return values when the procedure terminates, constraining the plan database so that planners only have limited ability to change the past, and calling the Reactive Planner, as described above, as needed to update the plan database. The plan runner implemented is described in more depth in [3].

Goal/Dialogue Manager—acts as an arbiter between the autonomous control system and other agents, including people. It retains state regarding its interaction with the other agents, e.g., recalls the subject of a previous sentence spoken by the user. As an arbiter, this element serves two roles: a goal manager and a dialogue manager. The goal manager essentially acts as a meta-planner for the declarative planner. As stated above, the declarative planner requires a start and end horizon time bounds, an initial state of the timelines at the start time, and a set of goals. The goal manager interacts with the user to determine this information. This may include negotiation of goals when all goals are not achievable or supporting mixed-initiative planning for hypothetical situations. The dialogue manager is responsible for acting as an intelligent interface with other agents. When interacting with people, it can converse with a person speaking a restricted natural language responding as appropriate to spoken commands and queries, i.e., it inserts, changes or removes tokens in the Plan Database or responds to user queries by querying the planner experts and Plan Database. Currently, the integrated Dialogue Manager is simplistic. A more sophisticated dialogue manager tested on a stand-alone simulator is presented in [4]. The integration of such a dialogue manager remains as future work.

Off-board User Interface System

The user-interface system is responsible for enabling the user to interact with the PSA by commanding and displaying information. It provides situational awareness, sensor-data views, plan views, and commanding capabilities. This includes interfaces for interactively creating and modifying the plan and teleoperation. Our intent is for this interface to support operation at various autonomy levels that can be dynamically changed and range from teleoperation to high-level autonomous control.

Voice Recognition and Synthesis—responsible for speech-to-text and text-to-speech. The voice recognition subsystem essentially converts an audio signal into a parsed text stream. In the past, we have used commercial products to accomplish this. We anticipate that we can continue to use such products, upgrading them as improvements are made. However, it may be necessary to filter the audio signal for noise. Conversely, the voice synthesis subsystem essentially converts text commanded by the Dialogue Manager or the Plan Runner into speech via the user headset or remote speakers. Similarly, we use a commercial product for this purpose.

Teleoperation Manager—responsible for executing user commands that can be handled within the User Interface system and providing support for converting GUI-generated commands into commands executable by the Autonomy system, e.g., path plan editing. Also, it supports two force-feedback 3-DOF joysticks or one 6-DOF joystick for 6-DOF teleoperation (x, y, z, yaw, pitch, roll) in position, velocity, or acceleration modes.

Crew GUI—responsible for displaying the sensor data, 3D rendering the PSA position in its environment, displaying and editing PSA plans, and directly commanding the PSA. Included in the displayed sensor data is the real-time video stream generated by the PSA. In addition, by using a camera mounted on the Crew GUI display, the Crew GUI supports teleconferencing.

Simulators

A variety of software simulators have served a crucial role in the software development process. They permit unit testing of components being developed as well as system integration tests when software changes are made. Our primary simulator is configurable so that it can replace as requested various hardware and software components as needed for testing.

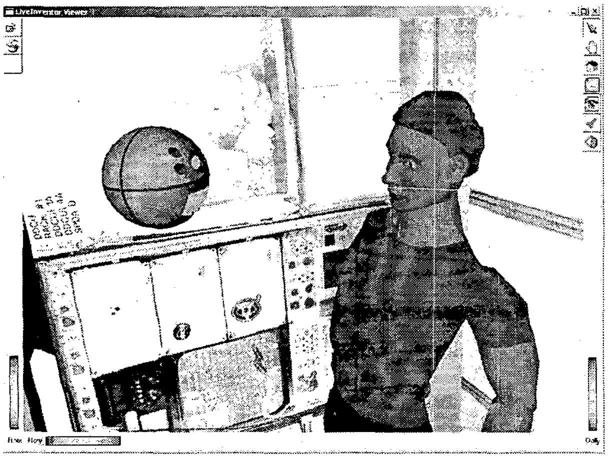

Figure 6 - 3D simulator screenshot of PSA in ISS with crewmember

We have recently integrated our PSA-specific simulator with a general-purpose 3D simulator, which provides 3D-rendered graphics and object dynamics. The current version is a synthesis of the graphics provided by the SGI Open Inventor™ object-oriented 3D toolkit built on top of Open GL® and the CMLabs Vortex rigid-body physics simulator. By providing VRML and collision models of the ISS, we can navigate multiple PSA's throughout the ISS and interact with simulated crewmembers, payloads, and objects. In Figure 6, a PSA is shown with a crewmember in the ISS U.S. Lab "Destiny" module.

In addition, this simulator has a scripting capability for controlling the environment. We have added an environment simulator to it to simulate fires, pressure leaks, and other faults to test the diagnostic capabilities of the PSA and its autonomous control system. Although software simulators can accelerate the software development process, we also use the same autonomy software, often executing similar scenarios, to control the physicals PSA prototypes in the physical simulators. These scenarios, some of which are described in the next section are helpful in testing the fidelity of the software simulators as well as the PSA hardware and software.

6. SAMPLE MISSION TEST SCENARIOS

In order to measure the system capabilities with reference to the operation requirements and to identify the challenging problems, several scenarios have been developed. These scenarios are designed to be executed both in simulation as well as with the prototype hardware in the test facilities. These scenarios perform a valuable role in measuring our current capability levels. They are also useful for regression testing. As software and hardware changes, we can run these scenarios to demonstrate we have not lost any capability. As the functionality of the autonomous control system, the prototypes, and the environment simulators increase, we raise the bar by increasing the complexity of the scenarios. The current scenarios that the system is being designed to address are:

Scenario A: Robust generation of an ISS module environment map

Description—
PSA will create an environment map of the ISS module by traversing the space in a serpentine path recording the environment sensor readings along the way. During this activity, its path will be blocked by static obstacles (some of which are known of ahead of time) and moving obstacles. At one point the PSA will be interrupted to be teleoperated and then perform a station-keeping task at a location specified by an ISS Rack Locker name, after which it will complete its original environment-mapping task.

Purpose—

- Demonstrate navigation to several waypoints in an environment that has static and dynamic obstacles.
- Demonstrate mixed-initiative execution including autonomous task interruption and resumption, guarded teleoperation, and visual servoing by command.
- Demonstrate generation of a near-optimal 6-DOF route plans
- Demonstrate obstacle detection and avoidance
- Demonstrate stereo vision-based 6-DOF localization and map registration

Scenario B: Participate in the diagnosis and recovery of an ISS module fault

Description—
A fixed sensor in the ISS module signals a high temperature to the Environmental Control Life Support System

(ECLSS). However, it is not known whether the sensor is defective or the source or the heat. PSA is given a command by ECLSS to go the fixed sensor location and verify the temperature at that location. If PSA confirms the fixed sensor is correct, PSA is to locate the heat source and signal the source to ECLSS, will then power down the locker at that location. Once PSA verifies that the temperature has returned to normal, it returns to its docking bay. If the fixed sensor is not correct, PSA is to stay at that location until the fixed sensor is made operational. Once PSA verifies the sensor, PSA returns to its docking bay.

Variation Summary—

1. Perform with faulty fixed sensor
2. Perform with overheating locker

Purpose—

- Demonstrate Integrated Vehicle Health Management
- Demonstrate cooperative multi-agent planning and execution
- Demonstrate generation of a near-optimal 6-DOF route plans
- Demonstrate stereo vision-based 6-DOF localization and map registration

Scenario C: Fault Detection and Cooperative diagnosis of an ISS module atmosphere leak

Description—
PSA is commanded to perform a routine task to monitor an ISS locker. While en route, PSA detects a drop in pressure in the module. It interrupts its current task and performs a set of directional microphone sensor readings to determine the cause is a leak to space and then PSA isolates the general location of the leak. PSA reports this information to ECLSS, which then dispatches and external spacecraft mobile monitor to the general location outside station where it images the region of the leak to get visual confirmation.

Purpose—

- Demonstrate Integrated Vehicle Health Management
- Demonstrate dynamically changing plan to respond to fault detected in the environment
- Demonstrate multi-agent cooperative diagnosis

Scenario D: Cooperative Data Collection and Crew Instruction for Performing Interactive Mission Science Experiments

Description—
Crewmember commands PSA to follow him to an ISS rack where he will perform an experiment. When he arrives, he commands PSA to point at the locker where he will work. After positioning PSA as desired, he commands PSA to maintain that position and start recording the video and audio. He commands PSA to brief him on experiment X and to instruct him on the first step of the experiment. Once the crewmember completes that step, he requests the next step and so on until all steps of the experiment are completed. He then commands the PSA to visually servo to face his face to record a summary of the experiment while he is moving. He then instructs PSA to stop recording and return to its docking bay, which it does.

Purpose—

- Demonstrate automated data collection
- Demonstrate human – autonomous system collaboration
- Demonstrate autonomous teleconferencing with face-tracking
- Demonstrate person following
- Demonstrate automated task instruction
- Demonstrate spoken language commanding and reporting

Scenario E: Long-term mixed-initiative planning and optimization including inventory tracking

Description—
PSA is given a list of visual servoing goals with time constraints and is requested to generate a near-optimal plan to achieve the goals. The goals will be such that it will be necessary to schedule multiple battery recharges in order to achieve them. The operator will dynamically change the plan prior to its execution. During the execution, PSA will monitor the location of inventory items it senses as it passes by. PSA will encounter static and dynamic obstacles in the environment. Due to an inaccurate battery model, PSA will have to replan to prevent running out of power prior to recharging at the docking bay. Once PSA has completed the goal list, it given a list of inventory items to locate, some of which it passed by. PSA responds with the locations of the items it senses and then generates a plan to explore the areas of the ISS module it did not previously explore in order to locate the other items.

Purpose—

- Demonstrate near-optimal path plan generation
- Demonstrate resource planning
- Demonstrate static and dynamic obstacle avoidance
- Demonstrate mixed-initiative plan generation
- Demonstrate spoken language commanding and reporting
- Demonstrate inventory item sensing and location tracking

7. FUTURE WORK

Future research and development efforts will focus on system-level active hybrid diagnosis, fleet operations (several PSAs working together to handle environmental problems) as well as autonomous operations with spacecraft command and control systems (instead of human commanding/teleoperating). Long-term functional upgrades might include adding effectors, e.g., arms, capable of control panel operation, payload maintenance, re-supply, and repair.

One of the future goals of the PSA R&D is to develop key functional capabilities and prove the feasibilities for micro robotic checkout and repair of autonomous spacecraft and probes. This long-term goal is driven by the recent failures in various low earth satellites, deep space and planetary probes. These failures are particularly troublesome because of our inability to properly identify the contributory failure factors and therefore leave NASA and industry at risk of repeating the same failures in the future.

A generic mission scenario would involve an expendable, golf-ball sized PSA to launch itself from the probe or satellite at key points in the mission: orbit insertions, systems checkout, major maneuvers, etc. This ability to do a fly-by type of checkout would help insure the spacecraft is ready for its next major event and help isolate contributory failure factors in the event of a mishap occurring later in the mission. The other mode of the PSA to operate in would be post-failure. In this role the PSA would support the fault detection, isolation and recovery processes. By being able to inspect the vehicle externally many possible failure scenarios could be evaluated and properly closed out. For some failures special effectors on the PSA might be able to assist in the recovery process (especially latch failures, or icing, or leaks where special efforts could be designed to affect these scenarios (torches, robotics arms, sprays, etc.).

To address this future vision objective several technologies would have to be demonstrated:

Miniaturization of the PSA down to a golf-ball sized spacecraft—this design goal would enable the PSA to be carried on small satellites and probes without incurring a significant volume and weight penalty. The size, and thereby hopefully cost, would allow several PSA's to be carried as expendable devices to maximize coverage and minimize collision potential

Highly developed vision and remote sensing capabilities—these technologies would enable the PSA to navigate around delicate and complex subsystems such as antenna's, power arrays, external sensors, latch and other activation mechanisms, and other structural components. The ability to do close inspections would be key to the fault detection and isolation requirements as well as supporting some of the recovery efforts. Non-destructive inspections such as thermal or others would also significantly support the fault detection and isolation goals.

Precise robotic effectors—these technologies, some of which already exist, would enable the PSA not only to conduct inspections, but also, in limited form support recovery operations. These include anything from spraying sealants to freeing latches, or re-pointing instruments to re-heating equipment.

Advanced intelligence systems—this technology would be required to enable the PSA to efficiently help diagnosis a system failure. The drivers for this capability are as follows: remote operations – which cause time delays that would make manual operations infeasible; complex environments – the structural architectures of the satellites or probes would require very precise and efficient operations that would make human operations, if feasible, very taxing, and high risk; efficient fault detection and isolation – PSA's that could do mishap investigation and analysis autonomously would enable optimized inspections potentially saving precious resources instead of having to wait on remote human collaborations.

Another useful spacecraft mobile robot is a multi-armed "monkey-sized" PSA is to perform various tasks from the mundane, e.g., changing filters, to the critical, e.g., repairing a leak. Consider a future mission where a spacecraft is sent to Mars and remains in orbit unoccupied while the crew is stationed on the Martian surface. This large PSA could be used to monitor and maintain the flight worthiness of the spacecraft and reduce mission risk.

One of the challenges of this project is balancing on the edge of the possible. If we were to incorporate all the fascinating ideas from the PSA team and others interested in the project, it would most likely be so technically impractical we would not end up deploying anything. However, if we were to descope them all, we would probably end up with something so unhelpful and difficult to use it would not be worth doing at all. By having these long term visions in mind now, the PSA team and supporting research organizations can optimize both investments and technology requirements to help meet both short term and long term requirements.

8. RELATED WORK

At this time, no free-flying vehicles have been deployed for performing operations inside spacecraft in flight. However, this work stands on the shoulders of a large body of work on satellite development and control. A similar vehicle, the Sprint AERCam, designed to be teleoperated outside of spacecraft, was successfully flight-tested outside the STS87 space shuttle flight in 1997 [5]. Sprint was a free-flying spherical robot that weighed about 35lbs and was 14" in diameter. It had no localization and proximity sensing capabilities. It had 12 nitrogen-gas thrusters for propulsion and attitude control. Its primary mission sensors were two color video cameras for supporting teleoperation, for providing video support for crew extra-vehicular activities (EVAs), or performing reconnaissance in lieu of an EVA. An effort to create a mini AERCam is ongoing at JSC [6]. Work on cooperative astronaut/mobile-robot operations is being done with the SCAMP SSV, an underwater vehicle designed to simulate a spacecraft [7]. The Synchronized Position Hold, Engage Reorient, Experimental Satellites (SPHERES) project at the MIT Space Systems Laboratory, consists of three 8" spherical spacecraft for testing formation flying of spacecraft and multi-agent control algorithms [8], [9]. The vehicles use CO_2 thrusters for propulsion and attitude control. SPHERES is scheduled for an ISS flight experiment in 2003. Four external-fixed IR/Ultrasound beacons are used by the SPHERES for localization in their workspace, ~6'x6'x6'. Onboard sensors are limited to IR and ultrasound receivers for beacon detection.

9. SUMMARY

We presented the ongoing research and development effort to design an internal spacecraft autonomous mobile monitor and the accompanying autonomous control software that is applicable to a wide range of free-flying vehicles. The rationale for the effort was presented as well as an illustrative scenario that shows how a PSA might be used once deployed. We discussed the high-level functional requirements of the project followed by a description of five PSA prototypes of increasing complexity and fidelity, of which two have been deployed and the third is being designed. We also briefly discussed the micro-gravity test facility, which allows us to fly the PSA prototypes on the ground as if they were onboard the ISS. The autonomy framework for intelligent flight vehicle control being developed and tested as part of this project was also presented. Several sample missions being used to test the prototypes and the autonomous control system were also outlined. We concluded with a discussion of both the short-term and long-term future work in the area of autonomous mobile vehicles for in-flight spacecraft support.

ACKNOWLEDGMENTS

The work presented is the paper is the result of the extraordinary efforts of many talented people far too numerous to individually identify. Rather than erring by not specifically acknowledging anyone, we ask to be pardoned by those we have omitted and gratefully acknowledge the contributions of Rick Alena, Daniel Andrews, Kevin Bass, Michael Belgarde, Jeff Bradshaw, Jeff Brown, Phil Canlas, Ling-Jen Chiang, Howie Choset, Salvatore Desiano, John Dowding, Vineet Gupta, Peter Hamlett, Jim Hanratty, James Hieronymous, Denver Hinds, Beth Ann Hockey, Lynn Hofland, David Husmann, Frankie James, Paul Keas, Linda Kobayashi, Kurt Konolige, David Kortenkamp, Brian Koss, John Loch, Nghia Mai, John Marmie, Mike McIntyre, Sergio Mendoza, Nicola Muscettola, Charles Neveu, Keith Nicewarner, Ana Papasin, Richard Papasin, Eric Poblenz, Manny Rayner, Mark Shirley, Mark Sibenac, Doug Smith, Adam Sweet, Hans Thomas, Vinh To, Mark Turner, Kristen Vollrath, Brian Williams, and Serge Yentus. In addition, we acknowledge the financial and management support provided by the NASA Cross-Enterprise Technology Development Program, the Computing, Information, and Technology Program, and the Engineering Complex Systems Program.

REFERENCES

[1] Douglas Bernard, et al., "Final report on the Remote Agent experiment." *Proceedings of the New Millennium Program DS-1 Technology Validation Symposium*, Pasadena, CA, February 8-9, 2000.

[2] Ari K. Jonsson, et al., "Planning in interplanetary space: theory and practice." *Proceedings of the 5th Artificial Intelligence Planning and Scheduling Conference*, Brekenridge, CO, 2000.

[3] Nicola Muscettola et el., "A unified approach to model-based planning and execution." *Proceedings of the Sixth International Conference on Intelligent Autonomous Systems*, Venice, Italy, 2000.

[4] Manny Rayner, Beth Ann Hockey, and Frankie James, "A compact architecture for dialogue management based on scripts and meta-outputs." *Proceedings of Applied Natural Language Processing (ANLP)*, 2000.

[5] "NASA AerCam Sprint" *Project website.* http://tommy.jsc.nasa.gov/projects/Sprint, April 28, 1998.

[6] Steven Fredrickson, "Mini AerCam." *Video Presentation at NASA's Johnson Space Center*, October 11, 2001. http://explorer.arc.nasa.gov:8080/ramgen/Archive/Robotics/vits.rm.

[7] Ella M. Atkins, Jamie A. Lennon, and Rhiannon S. Peasco, "Vision-based following for cooperative astronaut-robot operations." *Proceedings of the IEEE Aerospace Conference*, Big Sky, MT, March, 2002.

[8] David Miller et al, "SPHERES: a testbed for long duration satellite formation flying in micro-gravity conditions." *Proceedings of the 2000 AAS/AIAA Space Flight Mechanics Meeting, AAS 00-110*, American Astronautical Society, Clearwater, Florida, 1/23-26/2000. http://ssl.mit.edu/spheres/Docs/AAS-AIAA.pdf.

[9] Steve Sell, "SPHERES: Synchronized Position Hold, Engage, Reorient, Experimental Satellites." *Slide Presentation at the Jet Propulsion Laboratory Center for Integrated Space Microsystems*, Pasadena, CA, April 2, 2002. http://cism.jpl.nasa.gov/events/workshop/SPHERES.pdf.

Gregory A. Dorais is a Co-Principle Investigator of the PSA and is its project manager as well as its autonomy lead in the Autonomy and Robotics Area at the NASA Ames Research Center. He was a member of the Remote Agent Executive development team and was the Remote Agent System Integrator for the Deep Space One spacecraft. He also was a co-developer of the IDEA autonomous control system based on the Remote Agent. He was a member of the Design for Safety Program Planning team where he led the Resilient Systems and Operation project. He was also a co-Principle Investigator on a Human-Centered Autonomy project. He has performed autonomous rover research at JPL and remote sensing research at General Motors Research. At the University of Michigan, he performed autonomy research, was a member of the rover development team, and developed an automated system for inferring classification and forecasting models from data. He received his Ph.D. from the University of Michigan in Computer Science with a concentration in Intelligent Systems. His research interests include autonomous systems, robotics, machine vision, and machine learning.

Yuri Gawdiak is a Co-Principle Investigator of the PSA and originated its conceptual design. He is currently the program manager for the Engineering Complex Systems program. He has had many roles in the development of the ISS, including Network Engineer and Technical Management Information System manager at NASA headquarters. He was also on the Space Station Redesign Team B at NASA Langley Research Center as a Payload Avionics Design Architect and integrated portable computers into the ISS baseline design. At NASA Johnson Space Center, he worked on the development of payload avionics and developed the Wireless Network experiment for the ISS. At NASA Ames Research Center, he was the PI for the Wireless Network Experiment on STS-76/Mir 21 and performed research that led to a wireless LAN baseline augmentation on the ISS. He also held management roles in the Aviation Safety Program and led the development of the Surface Movement Advisor system for airport ground traffic management in the Airspace Systems Program. Prior to his work at NASA, he was a Systems Analyst for the Defense Communication Agency working on the World-wide Military Command and Control system and the Human-Computer Interface lab manager at University of Maryland. He has a B.S from Carnegie Mellon University in Information Systems with a concentration in Ergonomics.