

EXTENDING THE NTUA SPACE ROBOT EMULATOR FOR VALIDATING COMPLEX ON-ORBIT SERVICING TASKS

Konstantinos Machairas⁽¹⁾, Stelios Andreou⁽²⁾, Iosif S. Paraskevas⁽³⁾ and Evangelos G. Papadopoulos⁽⁴⁾

National Technical University of Athens

Department of Mechanical Engineering – Control Systems Laboratory

9 Heroon Polytechniou Str., 15780, Athens, Greece

e-mail: ⁽¹⁾k.machairas@gmail.com, ⁽²⁾stelandreou@hotmail.com, ⁽³⁾isparas@mail.ntua.gr, ⁽⁴⁾egpapado@central.ntua.gr

ABSTRACT

To validate on-orbit servicing strategies, extensive simulations and terrestrial emulation of robotic systems are necessary. The NTUA has developed a Space Robot Emulator to emulate the operation of free-flying robotic servicers, as already presented in previous works. Until recently, the emulator consisted of one robot with two manipulators, capable of hovering over a flat granite table using air bearings, and of planar motion employing CO₂ thrusters and a reaction wheel. To extend the capabilities of the Emulator, a second robot was developed with similar operational and design principles. In this paper, the mechatronic design of the robotic systems is presented. The limitations of the first robot and the developments to overcome them in the second as well as some preliminary validation tests are presented.

1. INTRODUCTION

During the last few years the problem of space debris has become more and more important [1]. Due to this fact the interest of all major space agencies around the world is focused exactly on ways to mitigate the problem. This can be achieved basically with two solutions: removing in a viable way the existing orbiting objects which pose a danger, and minimizing the creation of new ones. Both objectives constitute a large part of On-Orbit Servicing (OOS).

More specifically, the removal of the orbiting space debris can be achieved actively or passively. Various concepts have been proposed towards this aim with various degrees of maturity and different advantages and disadvantages. The favorable method depends on the trade-off between various parameters: orbits, cost, size of debris etc. [2]. Robotic manipulation is of course a preferable method especially when the size of the target is large and difficult to manipulate with other methods (e.g. electrodynamic tethers). On the other hand, the use of robotic systems enables the maintenance of operational satellites that can reduce the possibility of creating new space debris. The maneuverability and the dexterity of robotic systems, enable a large number of possible actions like inspection, ORU replacement, fixing of broken parts, fuel replenishment and others. Towards this, some important missions have been completed,

which proved the advantages of using robots for such tasks. Among others, the Orbital Express [3] and the recent efforts by SSCO [4] represent significant milestones. Other systems can be found in [5]. It is however important to mention the European efforts towards this with DEOS [6].

In all cases however, extensive testing and validation is absolutely necessary prior to launching a robotic system, especially during design and development. Indeed due to the inherent properties of these systems, their complex dynamics make difficult to easily assess their performance in the zero gravity environment. For these reasons, methods which allow even a partial simulation or emulation of the space environment are of great importance. These methods include neutral buoyancy facilities, hardware-in-the-loop systems, parabolic flights, etc. If the planar motion of the robots is adequate, air bearings facilities present a high fidelity emulation system, with a budget which can be adapted according to various requirements, e.g. [7-8]. The Control Systems Laboratory (CSL) in NTUA has developed a planar air bearing emulator, which was presented in [9]. The system consists of a granite table on top of which robots can hover. However, to achieve most complex operations such as robot cooperation, docking, capture of a target and manipulation, and swarm motions, a second robot was necessary. The requirements for the design have been set similar to those of the original robot, but with improved and more compact subsystems. The modularity, compactness, autonomy, the manipulation capabilities and the inclusion of critical satellite components such as reaction wheels were to be preserved.

In Section 2 the concept and the requirements for the NTUA Space Robot Emulator (SRE) are presented. The design and development of the new robot is discussed in Section 3. Particular attention to the control software and the techniques which have been used to overcome communication problems with the xPC Target is given. Some preliminary results which certify the functionality of the new robot are presented in Section 4. In Section 5, the full NTUA Space Emulator setup is presented, and Section 6 concludes this work.

2. CONCEPT AND REQUIREMENTS

Addressing the need of developing the new Space Robot according to the standards set by the existing one, see [9], led to a conceptual design of a cylindrical, fully autonomous structure floating on the granite table and driven by thrusters. However, the expertise gained through the development process of the previous robot in conjunction with problems revealed during its application, determined additional design parameters and requirements that should be taken into account.

While the shape was predefined, restrictions for similarity on general dimensions were also applied. A major requirement for the new design was to increase the available volume in order to avoid overlaps between various components. Smaller and more efficient parts would have to be employed so as to obtain the same or even better performance, within less space. Correspondingly, a demand was set to maintain regional separation among pneumatics, electronics, batteries and all other subsystems. Obviously, easy assembling and modularity were required, in order to effortlessly replace every individual component, if needed.

Another requirement that came up by the use of the first robot was the trunking task. Wires and CO₂ hoses had to be guided so as to prevent their tangling and to provide protection and immobility. These are qualities that have proven indispensable by experience gained from the previous robot and taken into consideration for the new design. A last requirement was the capability of a replaceable front section in order to accommodate different payloads like manipulators, docking subsystem, or any other experimental device is of particular interest.

As far as the electrical/ electronic subsystem is concerned, and in comparison to the first robot's design, the requirements included efforts for cost and volume reduction with respect to the first robot, while retaining or extending (if possible) the functionality. Naturally the new hardware should have been compatible with the attach points of the modified chassis. However, no limitations were imposed on the design principles and methodology, neither on the development process.

Electrical/ electronic and software design were performed from the beginning, irrespective of methods employed on the first robot, and only based on the basic requirements which generally define the emulator, namely: (a) power and computation autonomy of the robot, (b) remote control, (c) localization employing an onboard system of optical sensors, (d) localization through wireless communication with an external camera system, (e) thruster and motor control with the respective drives and custom PCBs.

3. NEW ROBOT

3.1 Mechanical Design

The mechanical design of the new robot started aiming at creating a structure capable to be built inside institution facilities, and mainly at CSL's workshop. Structural parts designed for ease of machining and assembly, maintaining a bottom to top procedure which will result to a targeted and robust configuration. The experience gained and the new requirements defined the procedure to be trailed for design and construction.

To begin with, the robot chassis was built out of standard materials (aluminum alloys and Plexiglas), off-the-shelf structural members and general hardware. Particularly, robot's frame consists of two cylindrical ring type plates which are kept apart using three U channel columns, thus creating a two-floor structure. This design provides approximately 30 lt of available volume to be exploited for subsystems installation, while in its flat top are situated the LEDs for the overhead camera. In more detail, the previously used bottom T base with the air-bearings attached on each vertex of it, was reshaped to a CNC-milled cylindrical plate profile providing increased useful area. Furthermore, the Reaction Wheel (RW) and CO₂ tank pocket were displaced radially and anti-symmetrically, while the latter also received a lowered floor datum under the base bottom surface, allowing for a higher tank and for separating the construction of the several modules, see Fig. 1.

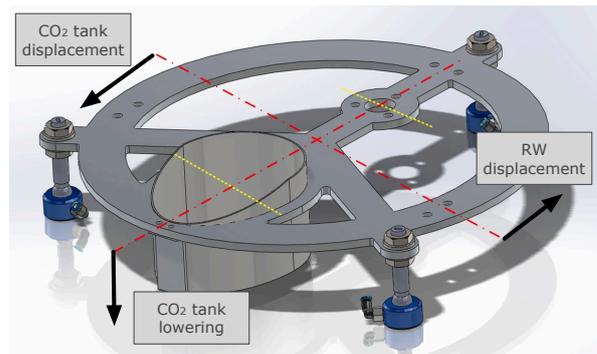


Figure 1. CAD representation of the bottom base.

Proceeding with the strategy of creating an emulator for increased ease of use, a second design optimization took place. The new principle is the employment of a layout where the various subsystems are situated in distinct modules. The new robot is divided in regions where clusters of servoamplifiers, electronic boards, batteries, pneumatic components and computational/ communicational segments are lying together. Every cluster is bordered with Plexiglas planes, which have been designed as shelf, drawer or box and fastened to robot chassis. Special care was paid on achieving quick and simple mounting/ removing. Every component part can be reached by pulling out a drawer or by opening a door.

As a result the process of maintenance is significantly improved and accelerated due the excellent accessibility.

Yet, to comply with the requirements of extending emulator capabilities and to accomplish more complex scenarios, the new mechanical design imposed innovations that increased the precision and interchangeability in the payload. Regarding the precision, a new design allowed for the capability to regulate the height of the optical sensors. Particularly, the three optical sensors which are placed at the bottom of robot and provide its position, are now capable to be set in a delimited distance above the granite table, so as to minimize sensor error.

Also, enhancements have taken place in the field of gripping and transporting the emulator. Hooks and custom-made handles designed and constructed to provide the ability of placing/ removing the robot on the granite table with smooth movements and without vibrations, hence shaking of sensors has been eliminated. Whereas discussing the interchangeability of the payload, the new front panel base has to be mentioned. This sliding-type base can be fastened if needed at the front plane of the bottom base via screws and it can be used to attach any type of payload. Future experiments employing manipulators, docking system or special sensors can be fulfilled by modulating the base in a manner to carry the appropriate equipment.

In addition to structural upgrades, improvements were made in the area of robot pneumatics. The evolution in the relevant components has been exploited to reduce their volume and increase their effectiveness. In more detail, the formerly used industrial-type high pressure regulator at the top of the tank was replaced with a tiny paintball part called “Palmer Stabilizer” capable to regulate and stabilize gas output. In combination with a coiled remote hose, a quick release coupling and a targeted design concerning siting, resulted in a CO₂ supply system that can be assembled in short time and with no need for any wrenches. Even the CO₂ tank itself was improved, as new gas bottles with fill indicators were purchased and employed. Moreover, major changes were introduced in the thrusting system. Brand new thrusting nozzles fabricated by FESTO as standard components, replaced the earlier custom ones. The nozzle pipe diameter was enlarged and consequently thrusting force was raised. Thruster switching valves were substituted by state of the art models providing response time less than 2ms. Maximum switching frequency can now reach 330Hz from 130Hz which was previously; accordingly thrusting PWM frequency can also rise.

Finally, the issue of CO₂ tube trunking was addressed. Previous experience has shown that cables and pipes can cause troubles during use and maintenance of the robot, since they are intermeshing, pulled and ultimately

they are displaced with respect to their correct position/ connection. Thus, the new design foresaw this task and special trunking paths were constructed, so as pipes and wires to remain organized and unshakeable. The new robot can be seen in Fig. 2.

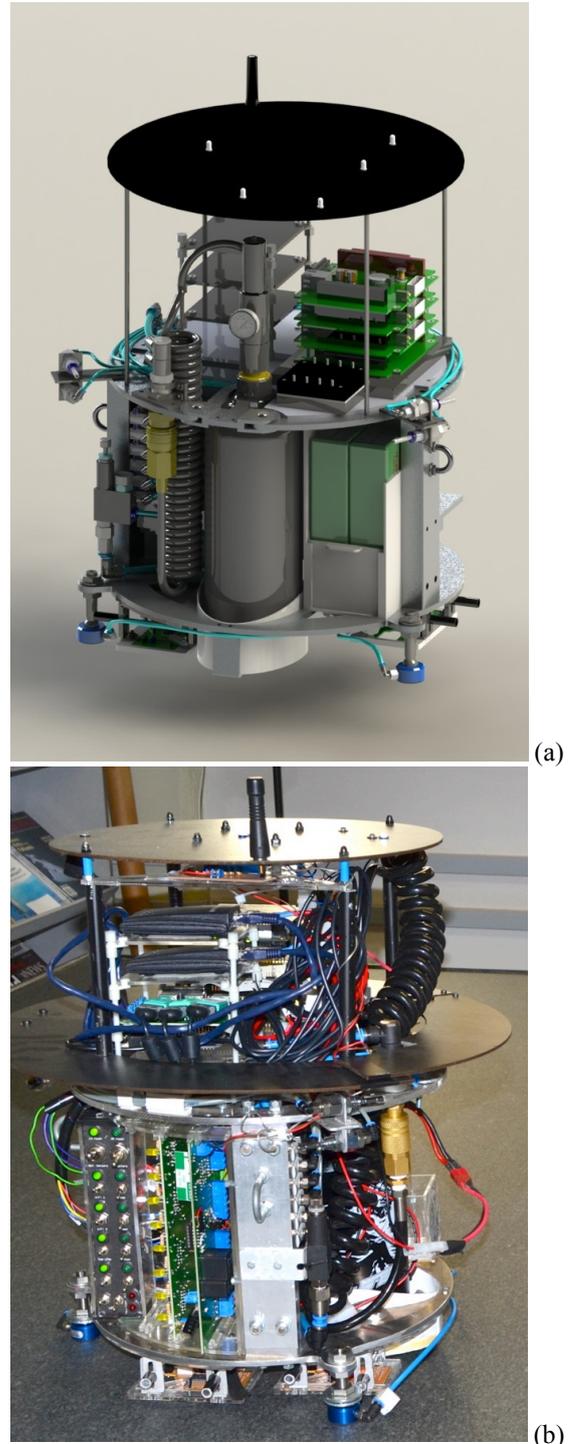


Figure 2. (a) CAD representation of new robot structure and (b) the current configuration.

3.2 Electrical and Electronic Design

The electrical and electronic design was largely based on the previous one, yet it was improved in several aspects, Fig. 3. Power autonomy was achieved using Li-Po batteries. Two DC/DC converters were also employed to supply two optionally isolated circuits (otherwise a common ground connects them): a low-voltage circuit for the computer, the sensors and the wireless modules (nominal voltage 5V), and a high-voltage circuit for the servomotors and the thruster valves (nominal voltage 24V).

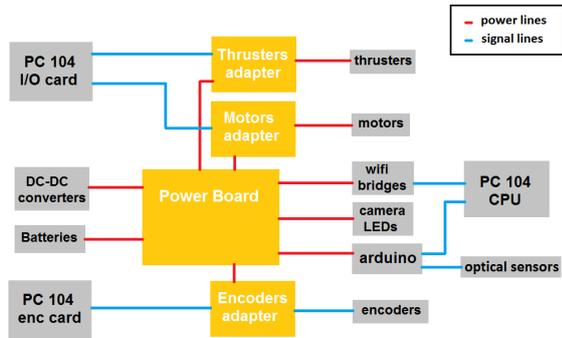


Figure 3. Electrical and electronics subsystem and its interconnections: red lines represent power and blue lines signals.

PCBs have been designed and printed for power routing, thrusters' control, collecting optical sensors' data (this PCB is called "Arduino shield"), and a dedicated PCB has been developed to be used as a control panel, Fig. 4a. In order to attain computation autonomy a PC-104 system is used, including a CPU board (running at 500 MHz), a 48 Digital I/O card and USB flash drive (for booting), Fig. 4b.

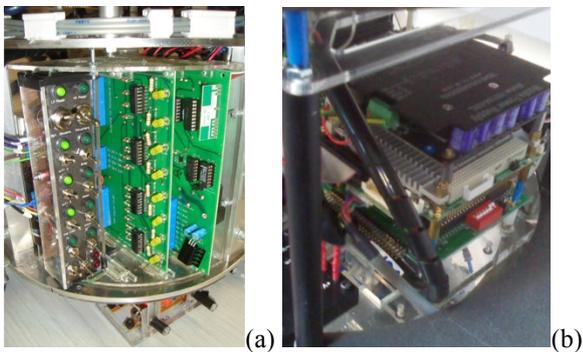


Figure 4. (a) Close up of PCBs and (b) PC104 board.

The localization of the robot is achieved through two different sensors: (a) relative estimation sensors, Fig. 5a, which are comprised by three optical sensors mounted at the bottom of the chassis, for fast estimation and (b) an absolute estimation sensor, that is an off-board overhead camera, which locates LEDs on top of the robot,

Fig. 5b, providing more accurate estimation. An Arduino board with a custom shield is used for collecting optical sensor data. An off-board computer receives data from the camera, does the image processing (in C/C++) and finally transmits the robot pose via UDP/IP (faster than TCP/IP, used in the past) to the on-board CPU. Two wireless Ethernet bridges are used (a) for the remote control of the robot, and (b) to receive real-time data wirelessly from the external camera system, Fig. 6.

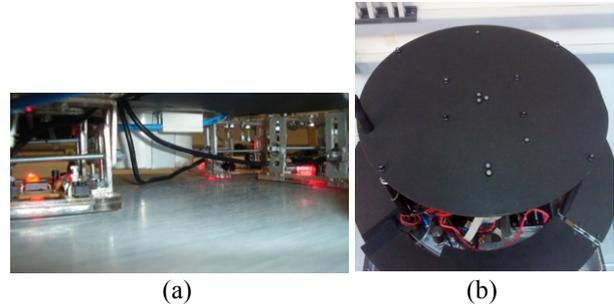


Figure 5. (a) Optical Sensors and (b) LEDs on top of the robot.

3.3 Software Design through Model-Based Design

A modern aspect considers the traditional, text-based approach of embedded system design inefficient to handle advanced control systems development, and indicates that the divide-and-conquer strategy for developing such systems, which means coordinating the resources of people with expertise in a wide range of disciplines, leads to slow development cycles. Focusing also on the experience gained from the first robot's design process, it was clear that developing complex models was difficult, time-consuming, and highly prone to errors. In addition, debugging text-based programs for the first robot was, and still is, a tedious process, requiring much trial and error before a final fault-free model could be created, especially since mathematical models undergo unseen changes during the translation through the various design stages. Trying to eliminate these time-consuming problems underlying the traditional design process, a different approach, called Model-Based Design (MBD), was tried out for the new robot.

Hence, employing this technique, the development of the robot was carried out as a whole, since the design of algorithms, the software development and the final integration on the hardware were studied as interdependent parts. Development was manifested in four steps: (1) modeling the plant, (2) synthesizing a controller for the plant, (3) simulating the plant and the controller, and (4) deploying the controller on real hardware after connecting it to sensors and actuators of the actual plant. An iterative debugging process was carried out by analyzing results on the actual target and updating the controller model. Model based design tools allowed these iterative steps to be performed in a unified visual environment.

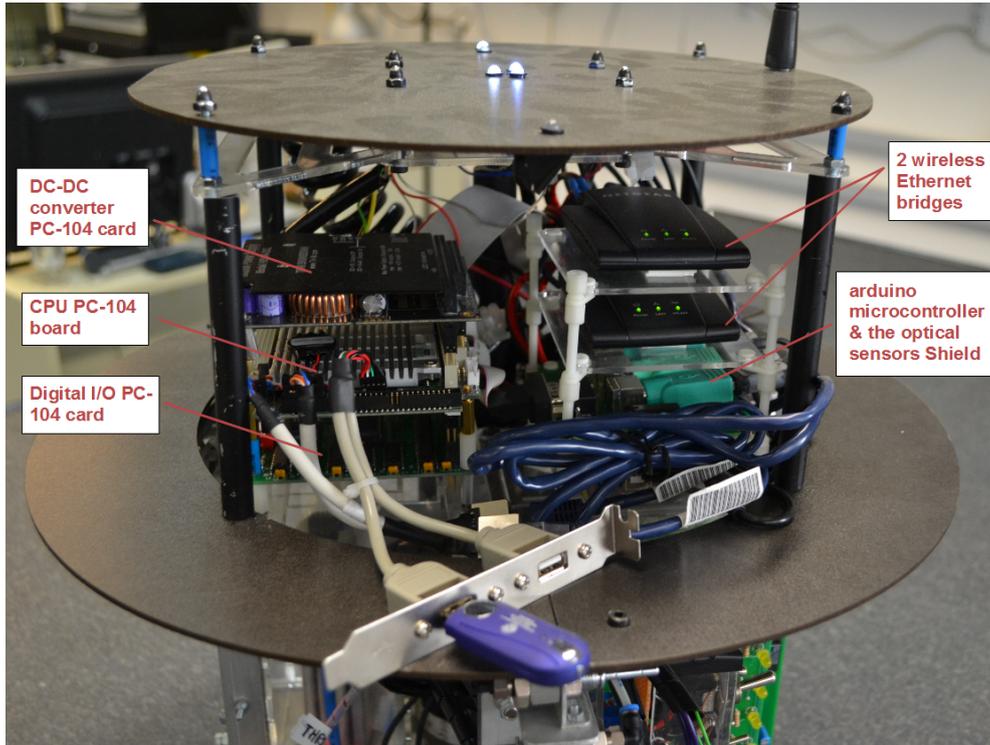


Figure 6. PC-104 Boards and wireless Ethernet bridges used for communications.

The software used was the xPC Target, from Mathworks. In this environment a Matlab/ Simulink model (running on a host PC) is auto-generated into C code, compiled with a third-party compiler, transmitted wirelessly to the robot (target PC) and finally runs in a real-time kernel, Fig. 7. Remote control, monitoring, parameter tuning as well as virtual reality animation (with VRML) were also enabled and extensively used during the development. In this manner, validation and verification of the design were continuously performed throughout the development. As a result, emphasis was put on innovation, while at the same time potential low-level problems in hardware and software were surmounted.

However, during the development phase, several compatibility issues, regarding xPC Target, came to surface, which complicated the design process and finally imposed strict limitations for the hardware selection. Generally, two major problems had to be addressed. The first one concerned the fact that the commonly used USB connection between the optical sensors and the robot CPU could not be implemented within the xPC Target software, since it is a disabled feature by default. To overcome this inconvenience, an Arduino microcontroller was interposed to establish the respective communication, employing the PS/2 protocol to collect optical sensor data, and the RS-232 protocol to transmit them (with or without post-processing) to the CPU. To this

end, an Arduino optical sensors shield (previously mentioned) was designed and manufactured.

A second problem concerned the limited reference for wireless communications within the xPC Target. The solution given was to use two wireless Ethernet bridges connected to the CPU Ethernet ports. This configuration defined the CPU board selection among a very limited catalog of supported PC-104 boards which complied with the requirements. Special focus was finally placed on the real-time wireless communication between the robot's PC (target PC) and the camera PC. This communication required a dedicated link (a second wireless bridge) and a software conversion from TCP (previously used) to the UDP/IP communication protocol.

3.4 Future Developments

The servomotors subsystem is yet incomplete; however there have been preliminary selections for the components of this subsystem. There will be at least five motors with servomotors on board: Two sets of two motors/ servomotors for controlling two 2-dof manipulators and one motor/ servomotor for a reaction wheel. Due to the modularity of the design, the motors can be used also for controlling specialized payloads if necessary, instead of the manipulators. Note that at the current configuration the new robot can easily emulate a free-floating satellite with or without initial motions imposed by the use of the thrusters.

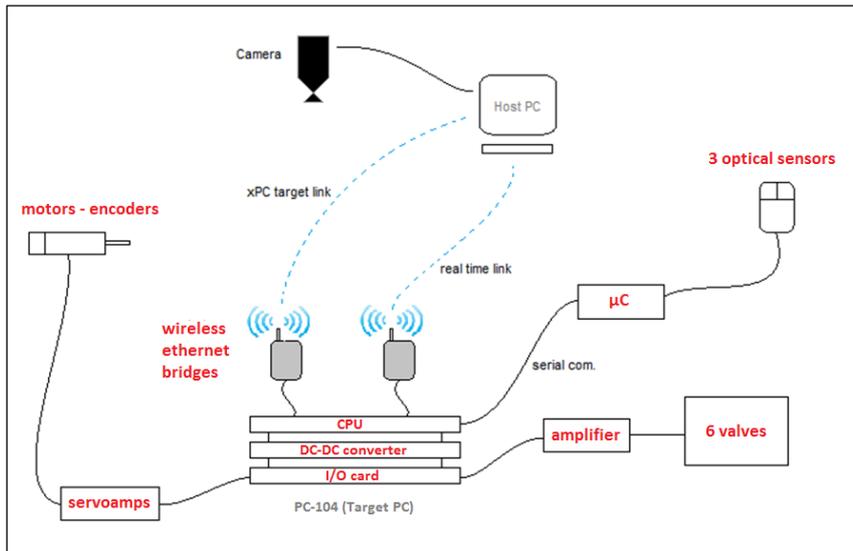


Figure 7. Interconnections between PC104, subsystems and external computers using xPC Target.

4. VALIDATION EXPERIMENTS

To validate the functionality of the new robot and most specifically the combination of the localization subsystems and the xPC Target, a number of experiments have taken place. The robot was forced to move through circular or straight paths. The motion as captured by the overhead camera is presented in Figs. 8 and 9. Similarly, a circular motion as seen by the optical sensors is presented in Fig. 10. However, these experiments only show a rough verification of subsystem functionality, since the sensors have not been calibrated yet. Calibration will be achieved by employing a Phasespace mocap system.

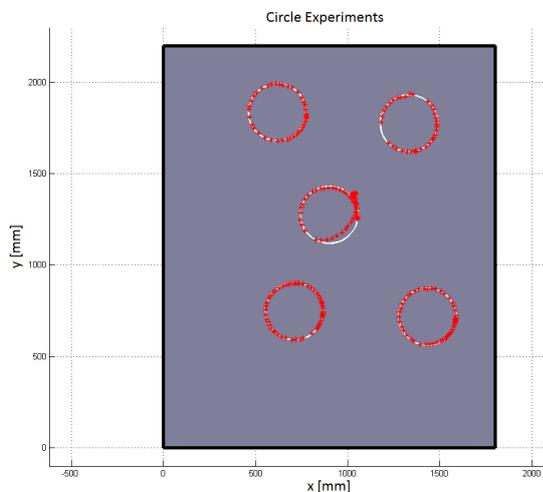


Figure 8. Five experiments on circular trajectories with data received from the camera system. (White: desired trajectory, Red: data after image processing).

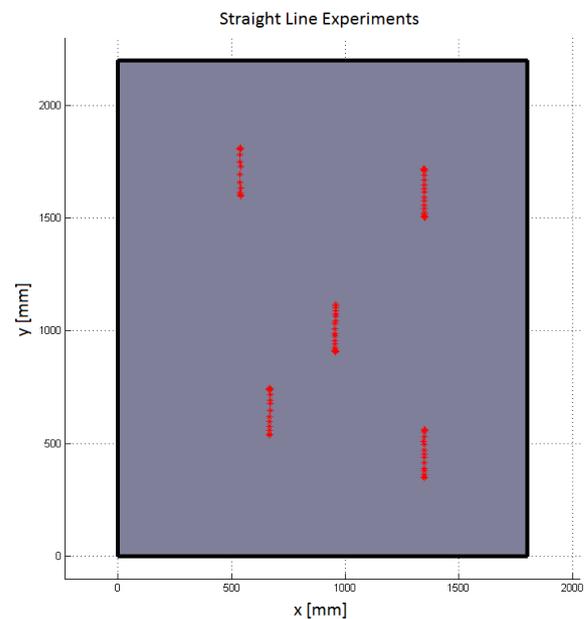


Figure 9. Five experiments on straight line trajectories with data received from the camera system.

5. FULL EMULATOR

Finally both robots have been positioned on the granite table, Fig. 11. Their coexistence on the table was very smooth and allows for interesting experiments in the near future. Although there is some work that should be completed, experiments which require both robots can be performed (as for example the case of a chaser satellite - target satellite).

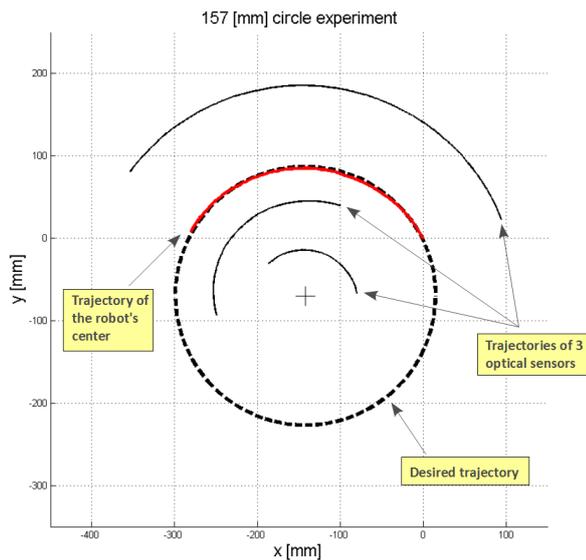


Figure 10. Motion data derived from the optical sensors system during a circular trajectory experiment.

6. SUMMARY

In this paper the extension of the NTUA Space Robot Emulator has been presented. The goal was to develop a system which would augment the existing robot and emulator environment into performing more complex On-Orbit Servicing tasks. In this paper, the mechatronic design, the development of the main subsystems and some preliminary results that certify the functionality of the emulator have been presented, and the system's promise has been shown.

ACKNOWLEDGEMENTS

Mr. Paraskevas' research has been co-financed by the European Union (European Social Fund – ESF) and Greek national funds through the Operational Program

"Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: Heracleitus II. Investing in knowledge society through the European Social Fund.

REFERENCES

1. Liou, J.-C., "Orbital Debris and the Challenges for Orbital Debris Environment Remediation", *39th Committee Space Research (COSPAR) Scientific Assembly*, Mysore, India, 2012.
2. Johnson, L. "Ch. 14 - Reduce the Amount of Debris in Space", *Sky Alert!*, Springer Praxis Books, pp 139-151, 2013.
3. <http://archive.darpa.mil/orbitalexpress/index.html>
4. <http://ssco.gsfc.nasa.gov>
5. NASA Goddard Space Flight Center, "On-Orbit Satellite Servicing Study", Project Report, 2010.
6. Wolf T., Reintsema, D. and Sommer, B., "Mission DEOS – Proofing the Capabilities of German's Space Robotic Technologies", *International Symposium on Artificial Intelligence, Robotics and Automation in Space – i-SAIRAS*, Turin, Italy, 2012.
7. Stoll, E. et al, "SPHERES Interact-Human-Machine Interaction Aboard the International Space Station", *Journal of Field Robotics*, Vol. 29 (4), pp 545-575, 2012.
8. Guglieri, G. et al, "Design and Development of Guidance Navigation and Control Algorithms for Spacecraft Rendezvous and Docking Experimentation", *Acta Astronautica*, 2013 DOI: 10.1016/j.actaastro.2013.02.010
9. Papadopoulos et al, "The NTUA Space Robotic Emulator: Design and Experiments", *International Conference on Intelligent Robots and Systems (IROS) – Workshop on Space Robotics Simulation*, San Francisco, USA, 2011.

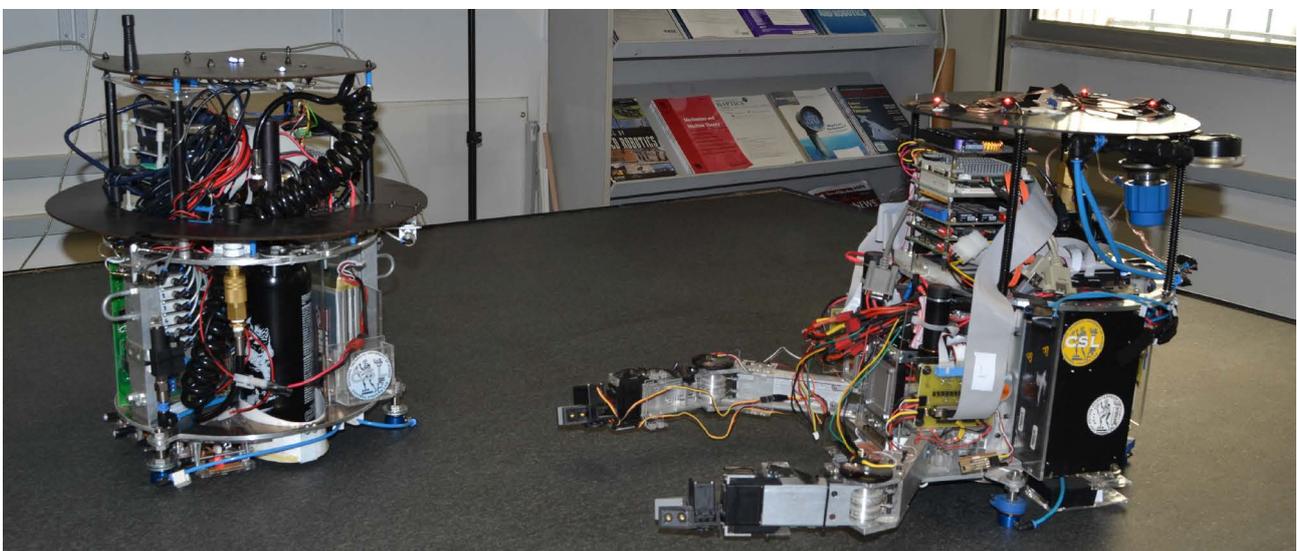


Figure 11. NTUA Space Robot Emulator with both robots.