ASTRA 2023: 17th Symposium on Advanced Space Technologies in Robotics and Automation
18-20 October 2023, Leiden, The Netherlands

TRAJECTORY OPTIMIZATION AND CONTROL OF MULTIPOD ROBOTS IN ON-ORBIT SERVICING OPERATIONS

Jorge Pomares[1], José L. Ramón[2], Leonard Felicetti[3], M. Olivares-Mendez[4]

[1] University of Alicante, Department of Physics, Systems Engineering and Signal Theory, San Vicente del Raspeig, 03690 Alicante, Spain, jpomares@ua.es
[2] University of Alicante, Department of Physics, Systems Engineering and Signal Theory, San Vicente del Raspeig, 03690 Alicante, Spain, jl.ramon@ua.es
[3] Cranfield University, School of Aerospace, Transport and Manufacturing, Cranfield, MK43 0AL, United Kingdom, leonard.Felicetti@cranfield.ac.uk.
[4] Space Robotics (SpaceR) Research Group, Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg, Luxembourg City, Luxembourg, miguel.olivaresmendez@uni.lu.

ABSTRACT

This paper presents a trajectory optimization method applied to multipod robots performing extravehicular activities. The presented approach automatically determines the leg motion required to achieve a desired location on the exterior of the target spacecraft. A 3D camera is located at the robot body, and a 3D map of the target spacecraft is generated from the point cloud. A trajectory optimization is obtained given the system's initial and desired state, the manoeuvre's total duration, and the number of steps for each leg. From this information, the trajectory optimization approach generates the legs trajectories and contact forces required to guide the multipod robot. Numerical simulations assess the applicability of the proposed strategy in typical operations that can potentially be performed in an extravehicular activity scenario.

1. INTRODUCTION

On-orbit assembly will provide flexibility to many future space mission designs. From building structures for human habitation on Mars or the Lunar surface [1], to building telescopes and large space structures [2][3], the application of robots will play a vital role in accomplishing the upcoming milestones. The assembly of these infrastructures in space will involve intricate tasks with high reliability, efficiency, and safety requirements. These tasks will require the deployment of autonomous robots, particularly for repetitive, structured, and standardized tasks.

Space robots can be categorized into three types based on the nature of the locomotion: stationary, free-flyers, and crawling robots. Stationary robots refer to robots or manipulators firmly attached to a single location on the base spacecraft. The workspace of such systems is directly depending with the length of the manipulator, generally resulting in having long links, and joints able to generate significant forces and torques at the base of the robotic system [4]. On the other hand, free-flying robotic systems are not fixed and thrusters can be employed to manoeuvre the robot within the workspace [5]. In this case, the workspace isn't constrained by manipulator's length but is instead limited by the availability of propellant. The final category pertains to crawling robots, also known as walking robots or multipod robots. These systems can remain physically connected to the primary spacecraft but have the capability to relocate their robot base or body [6]. They can establish connections to the target spacecraft through various means, including designated footholds, grasping structural elements, or employing adhesives such as gecko grippers. Typically, crawling systems have more arms compared to other categories because these manipulators are essential for moving the robotic body along the structure.

This paper presents a trajectory optimization algorithm that can be used to plan the motion and control of a multipod robot to move and achieve a given location from an initial one. The past literature has several studies focusing on path planning methodologies based on optimization for free-flying spacecraft [7] and free-floating and free-flying space manipulators [8][9]. In these previous studies, sequential convex programming

was proposed to obtain kinematically feasible trajectories. Tools such as differential dynamic programming [10] and LQR-based control systems have been employed control and optimize space robot trajectories. Such controllers were applied for path planning an control the Astrobee robot. The approach provides on-orbit free-flyers with the capability to perform on-orbit manipulation while avoiding collisions. In this paper, a new optimization algorithm that considers the constraints imposed by the topology of the target spacecraft is proposed. A trajectory optimization method applied to multipod robots performing an extravehicular task is described.

Various research teams within the space exploration community have created numerous resources for simulating spacecraft operations. Ad-hoc libraries and simulation tools are openly accessible on the internet, like SpaceDyn [11], while others have transitioned into commercial products, such as DCAP [12]. Many research projects have developed specialized simulation tools for analyzing space robot manoeuvres in orbit. However, these tools are often designed for specific simulations, lacking standardization, open-source availability, and rigorous validation. Considering this context, this paper proposes the use of a framework based on the Robotic Operating System (ROS) and Gazebo. These tools have been extended to include the main environmental conditions that space robots might encounter in orbit. The proposed simulation system is available for download [13].

This paper consists of the following parts. Section 2 describes the system architecture and the proposed simulation system. The trajectory optimization problem is presented in Section 3. The constraints of this optimization problem are shown in Section 4 and Section 5. Section 4 describes the robot kinematics and dynamics constraints, and Section 5 the legs constraints. Simulation results are detailed in Section 6, and main conclusions are presented in Section 7.

2. SYSTEM ARCHITECTURE AND ON ORBIT SCENARIO

The system architecture represented in the Gazebo simulation shown in Figure 1 is considered. A multipod robot is supposed to perform extravehicular activities (EVA) outside a target spacecraft. This multipod robot presents ζ legs with n degrees of freedom. The multipod represented in Figure 1 has $\zeta = 4$ legs with $n = 6$ degrees of freedom each and gecko grippers at their end-effectors. These gecko grippers allow the robot to perform a grasping force up to 5 N. The joint coordinates of each leg are represented as $\boldsymbol{q}_i \in \Re^n$ $(i = 1 \dots \zeta)$. This paper presents a path planning strategy for the guidance of the multipod robot doing EVAs. The presented approach automatically determines the leg motion required to achieve a desired location on the exterior of the target spacecraft. The robot's body coordinate frame B, is located at the body centre. The multipod robot is expected to be near the target spacecraft, allowing its leg end-effectors to reach the target spacecraft surface. A 3D camera is located in the robot body. A 3D map of the workspace or target spacecraft is generated form the point cloud obtained by this camera. This map is defined as $m(x, y)$ and provides depth information from a given position x, y with respect to the body reference frame. The multipod centre of mass position and attitude coordinates (Euler angles) with respect to the inertial coordinate frame are denoted as $\boldsymbol{r}(t) \in \Re^3$ and $\boldsymbol{\theta}(t) \in \Re^3$ respectively.

The OnOrbitROS framework is used for the robot simulation. Figure 2 illustrates the core components and structure of OnOrbitROS. This framework operates as a ROS publisher node responsible for generating the trajectories outlined by the entities comprising the simulation. The publisher node supplies the temporal information for each associated entity, along with the values of matrices $\boldsymbol{R}_l$ and $\boldsymbol{t}_l$, denoting the rotational and translational components of the LVLH frame with respect the inertial frame, along with the respective velocity, $\boldsymbol{v}_L(t)$. A plugin called OORplugin is defined to modify the behaviour of the physics engine employed by Gazebo and eliminate factors like gravity, wind, or magnetism while applying the required torques and forces for on-orbit servicing applications (such as gravity gradient). More details about the OnOrbitROS framework can be seen in [14].

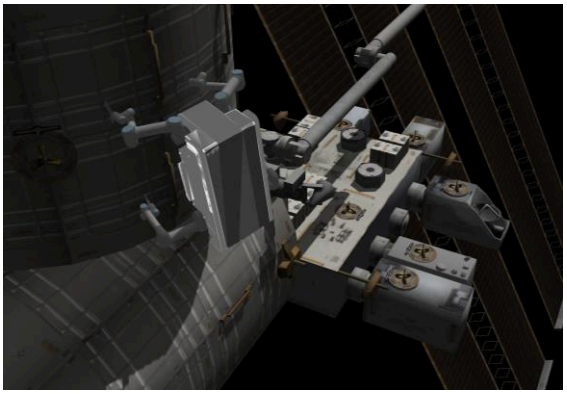

Figure 1. System architecture

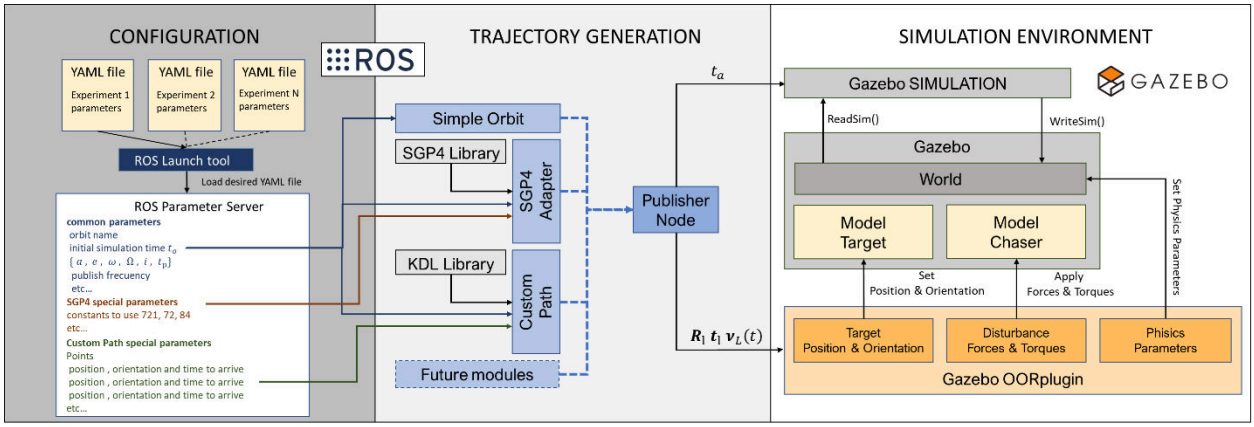

Figure 2. OnOrbitROS architecture

3. TRAJECTORY OPTIMIZATION

Table 1 represents the main formulation of the trajectory optimization method proposed in this paper. This trajectory optimization is obtained given the initial and desired state of the system, the total duration of the maneuver, T, and the amount of steps N for each leg i ($i = 1 \dots \zeta$). From this information, the trajectory optimization approach generates trajectories for the position, $\boldsymbol{r}(t)$, and attitude, $\boldsymbol{\theta}(t)$, of the multipod robot center of mass, and the trajectories, $\boldsymbol{p}_i(t)$, and forces, $\boldsymbol{f}_i(t)$, exerted by each leg, i (see decision variables in Table 1).

Different constraints are included to guarantee the specific kinematic and dynamic properties of the multipod robot. These properties are described in detail in Section 4. In addition, a set of restrictions on leg motions and forces are included to ensure physically feasible behaviours (taking into account the gecko grippers' properties). The nonlinear programming formulation allows the generation of dynamic trajectories for the multipod robot considering different phases based on each leg state. For a given leg i, these phases are denoted as $\Delta T_{ci,j}$ and $\Delta T_{nci,j}$ respectively, where $j = 1 \dots N$. These phases represent the duration of the contact, $\Delta T_{ci,j}$, and non-contact phases, $\Delta T_{nci,j}$. Therefore, for a given leg i, the total duration of the manoeuvre is equal to $\sum_{j=1}^{N} \Delta T_{ci,j} + \Delta T_{nci,j} = T$. The trajectory optimization problem generates phases when only one leg touches the target, phases when all the legs are in contact with the target or even full flight phases.

The generated trajectories allow to achieve the desired robot location considering complex and non-flat target spacecraft. The durations of the contact and non-contact phases for each leg are stablished by the trajectory optimization method and, therefore, the duration of each leg's polynomial $\boldsymbol{p}_i(t)$ and $\boldsymbol{f}_i(t)$. Section 5 analyzes the automatic computation of the legs trajectories and interactions forces required to move the multipod robot body from an initial location $\boldsymbol{r}_0, \boldsymbol{\theta}_0$ to a final desired one $\boldsymbol{r}_d$, $\boldsymbol{\theta}_\mathrm{d}$ in a duration of T seconds.

Table 1. Decision variables and constraints to define the proposed trajectory optimization problem

Decision variables
Multipod centre trajectory: $\boldsymbol{r}(t)$
Mutipod centre attitude: $\boldsymbol{\theta}(t)$
For each leg i ($i = 1 \dots \zeta$)
Phase durations: $\Delta T_{ci,1}, \Delta T_{nci,1}, \dots, \Delta T_{ci,N}, \Delta T_{nci,N}$
Leg trajectory: $\boldsymbol{p}_i(t, \Delta T_{ci,1}, \Delta T_{nci,1} \dots)$
Interaction force: $\boldsymbol{f}_i(t, \Delta T_{ci,1}, \Delta T_{nci,1} \dots)$

Constraints
Initial multipod body state: $\boldsymbol{r}(t=0) = \boldsymbol{r}_0, \boldsymbol{\theta}(t=0) = \boldsymbol{\theta}_0$
Desired multipod body state: $\boldsymbol{r}(t=T) = \boldsymbol{r}_d, \boldsymbol{\theta}(t=T) = \boldsymbol{\theta}_d$
Multipod dynamics constraint: $[\ddot{\boldsymbol{r}}(t), \dot{\boldsymbol{\omega}}(t)]^T = \boldsymbol{F}_d(\boldsymbol{r}(t), \boldsymbol{p}_i(t), \boldsymbol{f}_i(t))$
For each leg i ($i = 1 \dots \zeta$)
Multipod kinematics constraint: $\boldsymbol{p}_i(t) \in \mathcal{K}_i(\boldsymbol{r}(t), \boldsymbol{\theta}(t))$
If leg i is in contact with the workspace ($t \in \mathcal{C}_i$):
The leg end-effector does not slip: $\dot{\boldsymbol{p}}_i(t \in \mathcal{C}_i) = \boldsymbol{0}$
The leg maintains the contact with the workspace: $p_i^z(t \in \mathcal{C}_i) = m(p_i^x, p_i^y)$
If leg i is not in contact with the workspace ($t \notin \mathcal{C}_i$):
The leg does not exert forces: $\boldsymbol{f}_i(t \notin \mathcal{C}_i) = \boldsymbol{0}$
$\sum_{j=1}^{N} \Delta T_{ci,j} + \Delta T_{nci,j} = T$

4. MULTIPOD KINEMATICS AND DYNAMICS

This section outlines the methodology adopted for simulating the kinematics and dynamics of the multipod robotic system.

4.1 Robot kinematics

The kinematics of the multipod robot is defined by the body position and attitude and the legs joint coordinates. This information is defined by the vector $\boldsymbol{\epsilon} = [\boldsymbol{r}^T, \boldsymbol{\theta}^T, \boldsymbol{q}_i^T]^T$, $(i = 1 \dots \zeta)$, where $\boldsymbol{\theta}$ represents the yaw, pitch, roll Euler angles representing the orientation of the robot base with respect to the inertial frame. Additionally, $\boldsymbol{v}_i^b = \left[\dot{\boldsymbol{r}}_i^{b\,T}, \boldsymbol{\omega}_i^{b\,T}\right]^T \in \Re^6$ denotes the twist of each leg *i* end-effector coordinate frame with respect to the multipod robot's body coordinate frame, *B*; $\dot{\boldsymbol{r}}_i^b \in \Re^3$ and $\boldsymbol{\omega}_i^b \in \Re^3$ represents the linear and angular velocity components, respectively. The robot Jacobian $\boldsymbol{J}_i^b \in \Re^{6\times n}$ relates $\boldsymbol{v}_i^b$ and the joint velocities $\dot{\boldsymbol{q}}_i$ of the leg *i*:

$$\boldsymbol{v}_i^b = \boldsymbol{J}_i^b(\boldsymbol{q}_i)\dot{\boldsymbol{q}}_i \tag{1}$$

The following vector is defined $\boldsymbol{v} = [\boldsymbol{v}_b^T, \boldsymbol{v}_i^T]^T$ $(i = 1 \dots \zeta)$, where $\boldsymbol{v}_b$ is the linear and angular velocities of the multipod robot body, and $\boldsymbol{v}_i$ is the vector velocity of the leg *i* end effector frame with respect the inertial coordinate frame. The differential relationship between $\dot{\boldsymbol{\epsilon}}$ and $\boldsymbol{v}$ can be represented as:

$$\boldsymbol{v} = \boldsymbol{J}(\boldsymbol{\epsilon})\dot{\boldsymbol{\epsilon}} \tag{2}$$

where $\boldsymbol{J} = diag\left(\boldsymbol{I}_3, \boldsymbol{T}_b, \bar{\boldsymbol{R}}_b \boldsymbol{J}_i^b\right) \in \Re^{(6+6\zeta)\times(6+\zeta n)}$ is the robot Jacobian which relates the time derivative of $\boldsymbol{\epsilon}$ with the vector $\boldsymbol{v}$. In addition, $\bar{\boldsymbol{R}}_b = diag(\boldsymbol{R}_b, \boldsymbol{R}_b)$, where $\boldsymbol{R}_b$ is the rotation matrix that defines the relationship between the base and inertial frames, and $\boldsymbol{T}_b$ is the transformation matrix between the time derivative of $\boldsymbol{\theta}$, and the correspondent linear and angular velocity, $\boldsymbol{\omega}$.

In Table 1 the system kinematics constraint is defined as a given set of positions that can be achieved by the end of the multipod's legs, $\boldsymbol{p}_i(t) \in \mathcal{K}_i\big(\boldsymbol{r}(t), \boldsymbol{\theta}(t)\big)$. To define the range of possible motions of each leg, $\mathcal{K}_i$, first nominal position for each leg end-effector is defined as $\boldsymbol{p}_{ni}$. The kinematics constraint or range of motion for each leg is defined as a prism centered at $\boldsymbol{p}_{ni}$:

$$|\boldsymbol{R}_b[\boldsymbol{p}_i(t) - \boldsymbol{r}(t)] - \boldsymbol{p}_{ni}| < \xi_i \tag{3}$$

where the edge of the prism that constrains the range of motion, $\mathcal{K}_i$, has a length of $2\xi_i$,

4.2 Robot dynamics

The robot dynamics establishes a connection between acceleration and the forces and torques acting on the robot's body and legs. This relationship is defined in Table 1 and denoted $\boldsymbol{F}_d$. In this case, the system's dynamics relates to the robot body's linear and angular accelerations, denoted as $\dot{\boldsymbol{v}}_b = [\ddot{\boldsymbol{r}}^T, \dot{\boldsymbol{\omega}}^T]^T \in \Re^6$ in the inertial coordinate frame, the joint accelerations of each leg, $\ddot{\boldsymbol{q}}_i$, along with the forces and torques acting on the robot body, also expressed in the inertial frame, and the torques applied to each of the robot's legs. In our scenario, ζ legs are considered, and these system dynamics can be represented in the following equation:

$$\boldsymbol{u} - \boldsymbol{u}_e = \begin{bmatrix} \boldsymbol{M}_{bb} & \boldsymbol{M}_{bi} \\ \boldsymbol{M}_{bi}^T & \boldsymbol{M}_{ii} \end{bmatrix} \ddot{\boldsymbol{\epsilon}} + \begin{bmatrix} \boldsymbol{c}_b \\ \boldsymbol{c}_i \end{bmatrix} \tag{4}$$

where $\boldsymbol{u}$ is the input vector and $\boldsymbol{u}_e = \boldsymbol{J}^T \boldsymbol{h}_e$ shapes the effects of generalized external wrench $\boldsymbol{h}_e = [\boldsymbol{f}_b^T, \boldsymbol{f}_i^T]^T$ at joint level, with $\boldsymbol{f}_i$ the external forces applied on the leg *i*, and $\boldsymbol{f}_b$ the ones applied to the robot base. Additionally, $\boldsymbol{M}_{bb} \in \Re^{6\times6}$ is the inertia matrix of the multipod body, $\boldsymbol{M}_{bi} \in \Re^{6\times n}$ is the coupled inertia matrix of the robot body and the leg *i*, $\boldsymbol{M}_{ii} \in \Re^{n\times n}$ is the inertia matrix of the leg *i*; $\boldsymbol{c}_b$, and $\boldsymbol{c}_i \in \Re^6$ are velocity/displacement-dependent, nonlinear terms for the body and leg *i*, respectively.

Equation (4) represents the free-floating dynamics of the multipod robot. A simplified robot model is employed to analyze how contact forces impact both the linear and angular motion components of the robot's base. We utilize rigid body dynamics to simulate the motion of the robot's base caused by the contact forces exerted on its end-effector legs:

$$\begin{aligned} m\ddot{\boldsymbol{r}} &= \sum_{i=1}^{2} \boldsymbol{f}_i(t) \\ \boldsymbol{M}_I\dot{\boldsymbol{\omega}}(t) + \boldsymbol{\omega}(t) \times \boldsymbol{M}_I\boldsymbol{\omega}(t) &= \sum_{i=1}^{2} \boldsymbol{f}_i(t) \times \big(\boldsymbol{r}(t) - \boldsymbol{p}_i(t)\big) \end{aligned} \tag{5}$$

The movement of the robot's base, characterized by its 6 degrees of freedom, is represented through the linear position of its centre of mass denoted as $\boldsymbol{r}(t) \in \Re^3$. Additionally, $\boldsymbol{\omega}(t)$ is the angular velocity of the robot base. The mass of the multipod robot is represented as m, and $\boldsymbol{M}_I$ is a constant rotational moment of inertia derived from the robot's nominal configuration.

5. LEGS PARAMETERIZATION AND CONSTRAINTS FOR OPTIMIZATION

This section describes the leg constraints and parameterization defined to automatically generate the leg motion by means of the trajectory optimization approach described in Section 3.

5.1 Multipod legs parameterization

Two main states or phases can be achieved by a given leg, *i*, during the robot motion: the leg is in contact with the workspace, $t \in \mathcal{C}_i$, or the leg is not in contact $t \notin \mathcal{C}_i$. This basic consideration is defined to parameterize the legs' motion defined by their corresponding state durations $\Delta T_{ci,j}$ (duration of a contact phase) and $\Delta T_{nci,j}$ (duration of a non-contact phase), where $j = 1 \ldots N$. By changing the duration of these states, different kinds of trajectories can be obtained for the legs. These durations are not established by the user, but the optimizer automatically optimizes these durations for each leg to define the adequate legs motion.

Once the duration of the phases is established, the force and end-effector trajectory profiles should be defined by using polynomials. For each leg end-effector position $\boldsymbol{p}_i(t)$ multiple third-order polynomials are used when the leg is not in contact, $t \notin \mathcal{C}_i$, and a constant value is used when the robot is in a contact phase, $t \in \mathcal{C}_i$.With respect to the contact forces, $\boldsymbol{f}_i(t)$, multiple polynomials represent the contact forces, $t \in \mathcal{C}_i$, and zero force is established when the leg is not in contact with the workspace, $t \notin \mathcal{C}_i$. For a contact phase, when the leg *i* is contact with the workspace, the duration of this contact phase will be established by the optimizer as $\Delta T_{ci,j}$. If we consider the contact forces, the contact forces during a given contact period $\Delta T_{ci,j}$ is codified by three polynomials with equal duration of $\Delta T_{ci,j}/3$. This parametrization allows to codify arbitrary end-effector contact force profiles. In a similar way, the legs' end-effector trajectories are codified by means of three polynomials with equal duration, $\Delta T_{nci,j}/3$, when they are not in contact with the workspace, $\boldsymbol{p}_i(t \notin \mathcal{C}_i)$. The duration of these phases is defined as $\Delta T_{nci,j}$ and the same parametrization as the contact forces is used. We additionally ensure that the end-effector motion and force profiles are continuously differentiable between non-contact and contact phases. Since the optimizer automatically determines the duration of the contact and non-contact phases for each leg, the total duration of each leg motion must ensure that $\sum_{j=1}^{N} \Delta T_{ci,j} + \Delta T_{nci,j} = T$.

5.2 Multipod legs constraints

As shown in Table 1 different constraints are defined depending on the state of each leg. In a contact phase, the end of the leg shouldn't slip during all the time defined for this phase ($\Delta T_{ci,j}$) due the use of the gecko gripper. Therefore, the contact position is maintained, $\boldsymbol{p}_i(t \in \mathcal{C}_i) = \boldsymbol{p}_{ic}$, and a constraint is stablished to guarantee a constant position is kept: $\dot{\boldsymbol{p}}_i(t \in \mathcal{C}_i) = \boldsymbol{0}$. Additionally, as indicated in Section 2, a map function is obtained from the robot 3D camera, $m(p_i^x, p_i^y)$, that provides depth information about the target workspace at a given leg position p_i^x, p_i^y. An additional constraint is included in Table 1 to guarantee that the leg's contact positions are within the surface of the target: $p_i^z(t \in \mathcal{C}_i) = m(p_i^x, p_i^y)$. When the leg is not in a contact phase, an additional constraint is indicated in the optimization problem to guarantee the contact forces are zero: $\boldsymbol{f}_i(t \notin \mathcal{C}_i) = \boldsymbol{0}$.

To assure the contact end effector stability by using the gecko grippers, the obtained tangential forces should be within the respective friction cones. Therefore, the tangential component of the contact force, $\boldsymbol{f}_t$, must be:

$$\boldsymbol{f}_t < \mu \boldsymbol{f}_n \tag{6}$$

where $\boldsymbol{f}_n$is the perpendicular component and μ is the static friction coefficient characterizing the coupling between the gecko gripper and the target surface. Therefore, this equation constrains the largest tangential force component that can be applied at the legs end.

6. RESULTS

This section describes simulation experiments to show the path planning performed by using the optimization method proposed in this paper. Table 2 and Table 3 describe the dynamic parameters of the multipod robot represented in Figure 3.

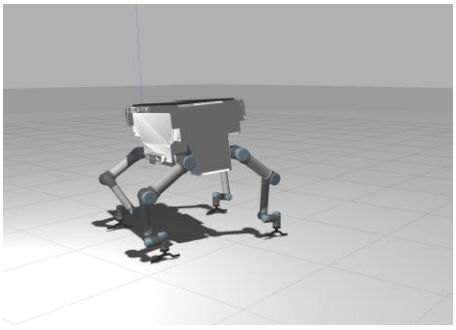

Figure 3. Gazebo simulation of the multipod robot.

Table 2. Main dynamic parameters of the robot body.

	Mass (kg)	Height (m)	Inertia (kg·m2) Ixx	Iyy	Izz	Ixy	Ixz	Iyz
Body Parameters	40	0.843	18.6	15.4	4.1	-0.008	-0.027	0.058

Table 3 Main dynamic parameters of the robot legs.

	Mass (kg)	Length (m)	Inertia (kg·m2) Ixx	Iyy	Izz	Ixy	Ixz	Iyz
Link 1	2.741	0.28	0.0124	0.0042	0.0136	3.6e-05	7.1e-05	-0.0002
Link 2	2.425	0.144	0.013	0.0138	0.0049	1.2e-05	-0.0032	-0.0001
Link 3	0.877	0.274	0.0025	0.0027	0.0012	0.0001	-0.0003	0.0004
Link 4	1.878	0.265	0.0035	0.0044	0.0023	1.3e-05	1.03e-05	-9.7e-05
Link 5	0.409	0	0.0001	0.00014	0.00015	-8.9e-08	-4.4e-08	4.2e-07
Link 6	0.308	0	0.0003	0.0002	0.00017	-1.6e-06	1.7e-06	-1.2e-05

Table 2 details the main parameters of the robot body, and Table 3 the parameters of each of the legs. Given the robot's initial and desired location, the manoeuvre's total duration, and the number of steps for each leg, the trajectory optimization approach generates trajectories for the linear position and attitude of the robot body, and the trajectories and forces exerted by each leg.

In the test case included in this paper, the robot is required to perform a displacement of 2.1 m along the x direction while the motion in the other components should remain constant. Figure 4.a represents the trajectory of the robot body (position and attitude) during the manoeuvre. The orientation of the robot body remains constant, but a displacement of 2.1 m is achieved in x direction while the displacement in y and z directions is the same as the one obtained at the beginning of the trajectory. Figure 4.b shows the evolution of the linear and angular acceleration of the multipod robot body. The optimizer evaluates the compliance with the dynamic constraints at 0.1 second intervals, leading to the segmented pattern observed in the figure. Nonetheless, the behaviour obtained using this sampling rate indicates that a 10 Hz update frequency suffices to produce achievable and dynamically coherent trajectories.

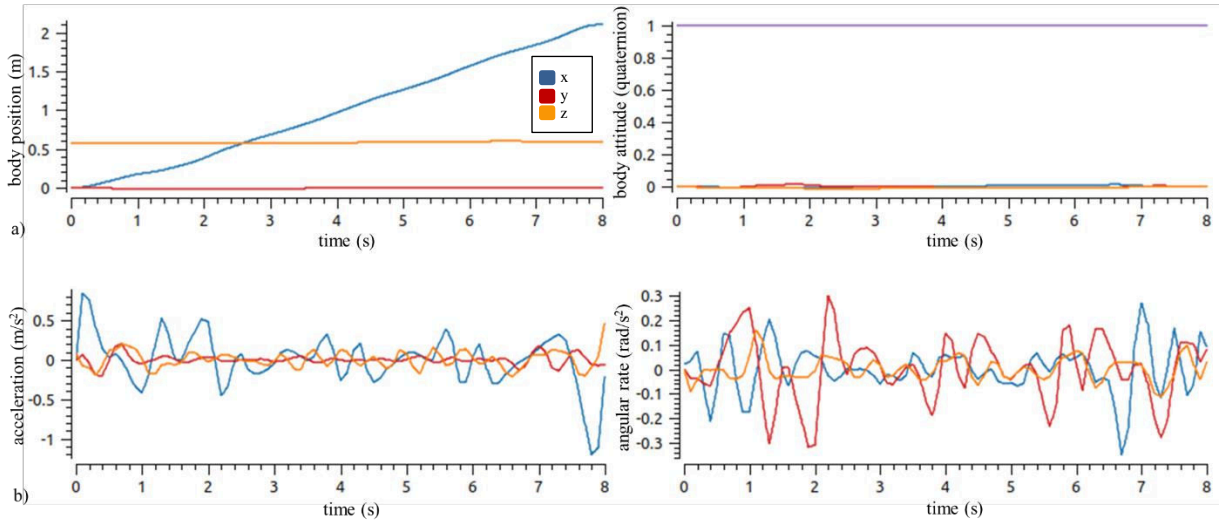

Figure 4. Multipod path planning. a) 3D Position and attitude in quaternions of the robot body. b) Linear and angular acceleration of the robot body.

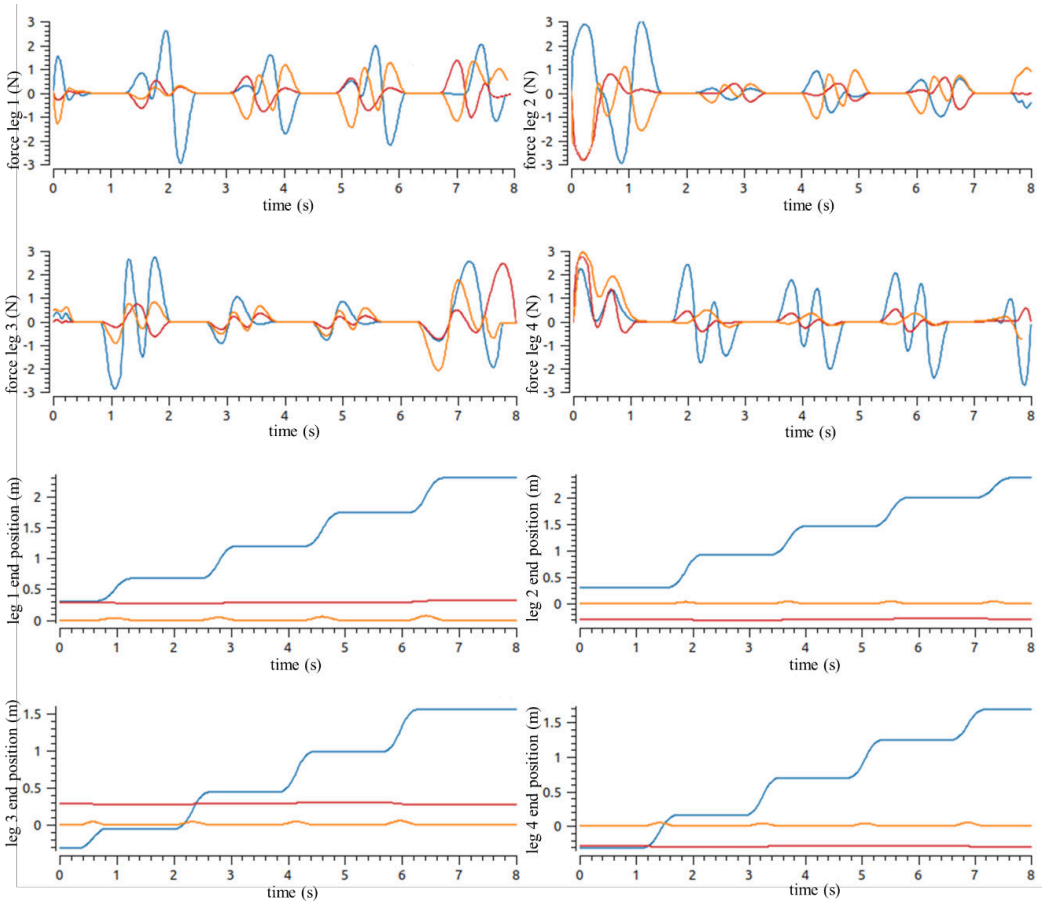

Figure 5. Multipod path planning. Legs end-effector forces and legs end-effector 3D position

Figure 5 shows the leg's end-effector force and position during the operations. The end-effector forces allows to guide the robot body towards the objective. These contact forces are only generated when the corresponding leg is in contact with the workspace thanks to the constraint $\boldsymbol{f}_i(t \notin \mathcal{C}_i) = \mathbf{0}$. As the figure shows, the proposed parameterization for the position and force allows to codify the required end-effector position and contact force profiles.

8. CONCLUSIONS

This paper presented the main characteristics of a multipod robot designed to perform extravehicular activities outside a given target spacecraft. Although the robot comprises four legs with 6 degrees of freedom each, the proposed path planning and control methods are generically formulated. The presented approach automatically determines the leg motion required to achieve a desired location on the exterior of the target spacecraft. A 3D camera is located at the robot body, and a 3D map of the target spacecraft is generated from the point cloud. A trajectory optimization is obtained given the system's initial and desired state, the manoeuvre's total duration, and the number of steps for each leg. From this information, the trajectory optimization approach generates the legs trajectories and contact forces required to guide the multipod robot.

The paper formulates the trajectory optimization problem and defines the adequate robot and trajectories parameterization. Different constraints are included in the proposed approach to guarantee the robot's specific kinematic and dynamic properties. A set of restrictions on leg motions and forces are included to ensure physically feasible behaviours (considering the gecko grippers' properties). The obtained results show that the proposed trajectory optimization approach is adequate to guide multipod robots in EVA.

9. REFERENCES

1. Thangavelautham, J., Chandra, A., and Jensen, E. ,2020, Autonomous Robot Teams for Lunar Mining Base Construction and Operation, in 2020 IEEE Aerospace Conference, 1–16.
2. Lee, N., Backes, P., Burdick, J., Pellegrino, S., Fuller, C., Hogstrom, K., et al., 2016, Architecture for In-Space Robotic Assembly of a Modular Space Telescope. J. Astron. Telesc. Instrum. Syst. **2**, 041207.
3. Belvin, W. K., Doggett, W. R., Watson, J. J., Dorsey, J. T., Warren, J. E., Jones, T. C., et al., 2016, In-space Structural Assembly: Applications and Technology, in 3rd AIAA Spacecraft Structures Conference. San Diego, California, 2163.
4. Komendera, E. E., and Dorsey, J., 2017, Initial Validation of Robotic Operations for In-Space Assembly of a Large Solar Electric Propulsion Transport Vehicle, in AIAA SPACE and Astronautics Forum and Exposition. Orlando, FL, 5248.
5. Jin, M., Yang, G., Liu, Y., Zhao, X., Liu, H., 2018, A Motion Planning Method Based Vision Servo for Free-Flying Space Robot Capturing a Tumbling Satellite, 2018 IEEE 8th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER), Tianjin, China, 883-888.
6. Lee, N., Backes, P., Burdick, J., Pellegrino, S., Fuller, C., Hogstrom, K., et al., 2016, Architecture for In-Space Robotic Assembly of a Modular Space Telescope. J. Astron. Telesc. Instrum. Syst. 2, 041207.
7. Mote, M. L., Hays, C. W., Collins, A., Feron, Hobbs, K. L., 2021, Natural Motion-based Trajectories for Automatic Spacecraft Collision Avoidance During Proximity Operations, 2021 IEEE Aerospace Conference (50100), Big Sky, MT, USA, 1-12
8. Misra, G., Ba, X., 2017, Optimal Path Planning for Free-Flying Space Manipulators via Sequential Convex Programming, Journal of Guidance, Control, and Dynamics, **40**(11), 3019–3026.
9. Misra, G., Ba, X., 2017, Task-Constrained Trajectory Planning of Free-Floating Space-Robotic Systems Using Convex Optimization, Journal of Guidance, Control, and Dynamics, **40**(11), 2857–2870.
10. Albee, K., Ekal, M., Coltin, B., Ventura, R., Linares, R., Miller, D. W., 2022, The RATTLE Motion Planning Algorithm for Robust Online Parametric Model Improvement With On-Orbit Validation. IEEE Robotics Autom. Lett. **7**(4), 10946-10953.
11. Yoshida, K., 2003, The SpaceDyn: a MATLAB toolbox for space and mobile robots, in: Proceedings 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human and Environment Friendly Robots with High Intelligence and Emotional Quotients.
12. Toso, M., Rossi, V., Robinson, M., Battie, F., 2018, DCAP: assessment of a multi-payload insertion problem by means of multibody dynamics", in: Proceedings of the European Conference on Spacecraft Structures, Materials and Environmental Testing, ESTEC, The Netherlands.
13. Ramón, J. L., Pomares, J., and Felicetti, L, 2023, ROS Framework for on orbiting space robots, OnOrbitROS. https://github.com/OnOrbitROS
14. Ramón, J. L., Pomares, J., Felicetti, L. 2023. Task space control for on-orbit space robotics using a new ROS-based framework. Simulation Modelling Practice and Theory. **127**.

2023

Trajectory optimization and control of multipod robots in on-orbit servicing operations

Pomares, Jorge

European Space Agency (ESA)