



Physics Informed Machine Learning for Path Planning of Space Robots

Rogério R. dos Santos¹, Ijar M. da Fonseca², and Domingos A. Rade¹

¹ Aeronautical Mechanics Division, Aeronautics Institute of Technology (ITA), S. J. dos Campos, SP, Brazil

² Division of Aeronautical and Aerospace Engineering, Aeronautics Institute of Technology (ITA), S. J. dos Campos, SP, Brazil

Abstract: The present study concerns a specific characteristic of the autonomy of space robots, namely the ability to plan the movement of the manipulators in the presence of uncertainties that affect the position of the target, normally due to the dynamic characteristics of the equipment. A physics informed machine learning strategy for the computation of the inverse kinematics of a space robot is proposed. The effects of satellite and manipulator dynamics are taken into account. The performance of the methodology is discussed as the amount of historical data increases. It is found that in some cases the machine learning algorithm is able to calculate the inverse kinematics with great precision. In general, its use results in a significant improvement on the process convergence.

Keywords: support vector machine, inverse kinematics, uncertainty quantification, berthing, space robotics

INTRODUCTION

In contemporary space missions service satellites are employed in a multitude of scenarios. Their ability to manipulate objects and interact with other vehicles are important in the deployment and maintenance of space devices. In this context, highly autonomous space robots are key elements to the success of the mission. The present study concerns a specific feature of the autonomy, namely the ability to plan the motion of the manipulators in the presence of uncertainties affecting the positions of objects and the dynamic characteristics of the equipment involved.

Fonseca et al. (2017) analyze the influence of the manipulator dynamics on the attitude of the satellite. The analysis of the dynamics is performed in the close proximity phase of rendezvous docking/berthing operation. External disturbance of gravity-gradient and the residual magnetic torque are considered in the analysis. Numerical results show that the control laws allow the system stabilization and good performance when the manipulator is in service. The papers by Fonseca et al. (2002, 2004) present an integrated structural and control optimization of a large space structure with a robot arm and tip payloads subjected to the gravity-gradient torque. Constraints from the structure and control disciplines are imposed on the integrated optimization process. The result demonstrates the feasibility of the proposition while taking into account combined constraints. Further, the resulting minimum structural weight represented a saving of 20% of the initial value.

Santos et al. (2022) address the problem of automatic path planning of a manipulator-like spacecraft in orbit. Based on the concept of optimal control and off-line establishment of optimal trajectories, the study proposes a formulation of multi-objective optimization that accounts for multiple aspects of motion. The numerical procedure includes a machine learning strategy that is able to learn from both training data and mission tasks. It improved the convergence rate of the optimization procedure, which led to the precise value of the angle of the joint. The learning strategy was effective for estimating the optimal solution.

In the present paper a Support Vector Machine algorithm is used in conjunction with a mathematical model as an heuristic to improve the rate of convergence of an optimization procedure that establishes the inverse kinematics of a space robot. The main scientific contribution of the present methodology is the proposition of a physics informed machine learning strategy that significantly increases the efficiency in establishing the joint angles for a specific task subject to uncertainty. Effects of satellite and robot dynamics are taken into account along the movement and such influence is incorporated in the learning process as well as the kinematics behavior.

SPACE ROBOTS

The concept of space robot considered in the present paper consists in a space satellite with a serial manipulator attached to it.

The schematic representation of the spaceship is given in Figure 1.

The kinematic model of the industrial serial manipulator describes the influence of joint angles in the placement of the end-effector. Usually, an n -link manipulator is mathematically represented by a set of homogeneous transformations $\mathbf{A}_i, i = 1, \dots, n$ that describe the Cartesian positioning from the base $\mathbf{p}_{base}(x, y, z)$ to the end-effector $\mathbf{p}_{end}(x, y, z)$. Following

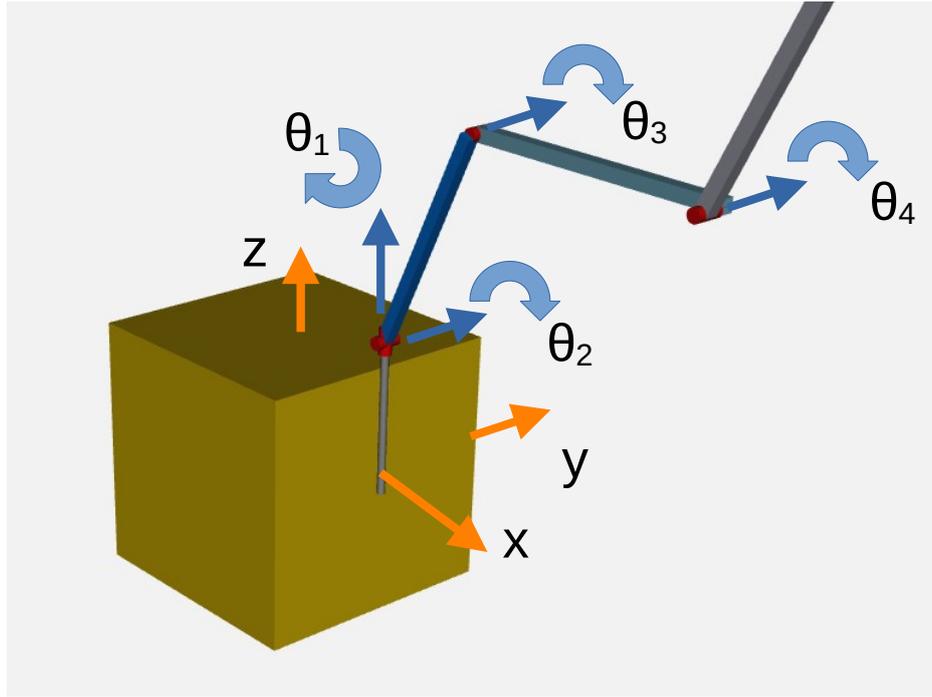


Figure 1 – Space robot composed by a satellite and a serial manipulator.

this convention, the serial manipulator can be represented by the matrix

$$\mathbf{T}_0^n(\mathbf{q}) = \mathbf{A}_1(q_1)\mathbf{A}_2(q_2) \dots \mathbf{A}_n(q_n) \quad (1)$$

where q_i is the i -th joint coordinate, that is, θ_i for rotational joint or d_i for translation joint.

Adopting the Denavit-Hartenberg convention to define \mathbf{A}_i , any serial robot can be represented by Eq.(1). Hence, given the base \mathbf{p}_{base} and joint $q_i(d, \theta)$ values, the Cartesian position of the end-effector \mathbf{p}_{end} is given by

$$\mathbf{p}_{end} = \mathbf{T}_0^n(\mathbf{q})\mathbf{p}_{base}. \quad (2)$$

Solving the inverse kinematics problem means to solve the inverse problem, i.e., given \mathbf{p}_{base} and \mathbf{p}_{end} , find $\mathbf{q}(d, \theta)$ for which Eq. (2) holds. It will require the solution of the inverse problem

$$\min_{\theta_i} |\mathbf{T}_0^n(q_i(\theta_i))\mathbf{p}_{base} - \mathbf{p}_{end}|. \quad (3)$$

The Denavit-Hartenberg parameters of the manipulator, as seen in Figure 1, is shown in Table 1.

Table 1 – Denavit-Hartenberg parameters.

Joint	a (m)	α (rad)	d (m)	θ (rad)
1	0	1.57	0	θ_1
2	1	0	0	θ_2
3	1	0	0	θ_3
4	1	0	0	θ_4

The classical formulation of the serial manipulator, Eq. (2), contains the hypothesis that the base of the manipulator is fixed. Since the satellite is traveling in orbit, in the space scenario the manipulator base is in motion and is subject to the influence of the robot and satellite dynamics. Therefore, to account for the dynamic effects along the movement, a higher fidelity model is considered.

The high fidelity model is build using the OpenModelica software, which provides a multi-body dynamics library, and is used to compute the coupled effects between the robot and the satellite while floating in the space. The parameters are shown in Table 2.

Table 2 – Satellite and robot parameters for dynamics computation.

Device	Dimension _{x,y,z} (m)	Mass (kg)
Satellite	1 × 1 × 1	90
Link 1	1 × 0.05 × 0.05	0.3
Link 2	1 × 0.05 × 0.05	0.3
Link 3	1 × 0.05 × 0.05	0.3

Given the displacement Δ of the satellite, the position \mathbf{p}_{end} of the end-effector and the position \mathbf{p}_{target} of the target, the inverse problem for obtaining the inverse kinematics is given by

$$\min_{\theta_i} \left| \{\mathbf{p}_{end}(\theta_i) + \Delta(\theta_i)\} - \mathbf{p}_{target} \right|, \quad (4)$$

the minimum value of which corresponds to the Euclidean distance between the end-effector and the target. The disturbance due to multiple dynamic effects and coupling between satellite and robot is accounted for by the Δ parameter.

NUMERICAL RESULTS

The satellite and the robotic arm have computational models represented in the OpenModelica software (Fritzon et al., 2006; Elmqvist et al., 1998). The machine learning model and the optimization process were developed through the R software (R Core Team, 2020).

The OpenModelica software is a general purpose modeling system that has been successfully applied in aerospace studies (Lovera, T. Pulecchi, 2006b; Briese et al. 2017; Chen et al. 2017). The paper by Pulecchi and Lovera (2006) discusses the suitability of Modelica language for modeling Attitude and Orbit Control Systems (AOCS). The ability to use multiple coordinate frames through visual connectors, the existence of rich MultiBody Library to describe spacecraft dynamics, and the ability to describe multi-physics domains are some key aspects cited by the authors.

In the conceptual model of the robot and the satellite, as shown in Figure 1, there are four joint angles $\theta_1, \theta_2, \theta_3$ and θ_4 whose values are obtained by eight reference points r_{ij} according to Table 3.

Table 3 – Joint angles and the parametric representation by design variables.

Joint	Design variables	Joint angle
1	r_{11}, r_{21}	$\theta_1(r_{11}, r_{21}, t)$
2	r_{12}, r_{22}	$\theta_2(r_{12}, r_{22}, t)$
3	r_{13}, r_{23}	$\theta_3(r_{13}, r_{23}, t)$
4	r_{14}, r_{24}	$\theta_4(r_{14}, r_{24}, t)$

The optimization procedure encompasses the analysis of eight design variables $\mathbf{r} = [r_{11}, r_{21}, r_{12}, r_{22}, r_{13}, r_{23}, r_{14}, r_{24}]^T$ that will describe the four joint angles $\theta_1, \theta_2, \theta_3$ and θ_4 according to the concept presented in Figure 2, which is equivalent to the correspondence tabulated in Table 3. Furthermore, in the present case $r_{0j} = 0, j = 1, \dots, 4$.

At the beginning of the motion ($t_0 = 0s$), the joint angles are all zero, that is, $\theta_j(0) = 0, j = 1, \dots, 4$, which is implicitly represented by the equations $r_{0j} = 0, j = 1, \dots, 4$.

The target is placed at the Cartesian position (2.5m, 0.0m, 0.7m) relative to the satellite origin and there is an uncertainty box of dimension $\Delta x = \Delta y = \Delta z = \pm 0.21$ m. As a result, the height, width and depth of the berthing box is 0.42 m.

The numerical computations consider the following steps:

- Step 1 – Non-linear programming algorithm solving the inverse problem (Eq. 3);
- Step 2 – Support Vector Machine learning the inverse kinematics solution from historical data;
- Step 3 – Non-linear programming algorithm solving the inverse problem (Eq. 4).

Step 1 is used as an heuristic to determine a good initial guess θ for the optimization in Step 3. This procedure will be referred to as the DH-based algorithm. The second strategy, the SVM-based algorithm, consists of using Step 2 to determine a good initial guess θ for the procedure in Step 3. As a result, the calculation of steps 1–3 characterize the physics informed machine learning strategy addressed in the present study. By evaluating initial cases, the physical model (Step 1) provides data for machine learning (Step 2). As the number of cases increases, the historical data computed in

Table 4 – Number of iterations and final error obtained from DH and SVM strategies.

Case	Target (m)	DH Iterations	DH Error (m)	SVM Iterations	SVM Error (m)
1	(2.500, 0.000, 0.700)	141	0.004	141	0.004
2	(2.710, 0.210, 0.910)	121	0.003	147	0.003
3	(2.710, 0.210, 0.490)	174	0.003	128	0.002
4	(2.710, -0.210, 0.910)	111	0.003	113	0.003
5	(2.710, -0.210, 0.490)	153	0.003	96	0.003
6	(2.290, 0.210, 0.910)	117	0.003	109	0.002
7	(2.290, 0.210, 0.490)	151	0.002	120	0.003
8	(2.290, -0.210, 0.910)	127	0.003	105	0.002
9	(2.290, -0.210, 0.490)	139	0.003	92	0.002
10	(2.471, 0.050, 0.600)	145	0.003	97	0.002
11	(2.600, -0.023, 0.708)	136	0.003	3	0.002
12	(2.576, -0.051, 0.607)	150	0.003	99	0.003
13	(2.486, 0.103, 0.688)	144	0.003	92	0.003
14	(2.428, 0.007, 0.653)	137	0.003	94	0.003
15	(2.478, -0.102, 0.616)	139	0.003	10	0.003
16	(2.559, -0.011, 0.670)	145	0.003	96	0.003
17	(2.568, 0.090, 0.675)	136	0.004	84	0.002
18	(2.583, 0.056, 0.721)	137	0.003	10	0.002
19	(2.405, 0.075, 0.741)	166	0.003	106	0.003
20	(2.535, -0.078, 0.777)	134	0.004	105	0.003
21	(2.501, -0.061, 0.635)	148	0.002	3	0.002
22	(2.596, 0.035, 0.803)	117	0.002	3	0.002
23	(2.518, 0.023, 0.658)	141	0.003	3	0.003
24	(2.417, -0.070, 0.757)	141	0.003	99	0.002
25	(2.461, -0.088, 0.693)	139	0.002	3	0.003
26	(2.398, -0.037, 0.623)	148	0.003	3	0.001
27	(2.456, 0.070, 0.745)	132	0.003	93	0.000
28	(2.493, -0.015, 0.731)	130	0.002	103	0.002
29	(2.525, -0.096, 0.700)	150	0.003	6	0.002
30	(2.507, 0.013, 0.783)	136	0.003	3	0.003
31	(2.545, 0.040, 0.630)	137	0.002	3	0.003
32	(2.442, -0.031, 0.767)	137	0.003	88	0.003
33	(2.423, -0.003, 0.714)	141	0.003	3	0.004
34	(2.552, -0.050, 0.760)	141	0.002	3	0.002
35	(2.443, 0.087, 0.644)	126	0.002	3	0.003
36	(2.531, 0.066, 0.796)	132	0.003	3	0.001

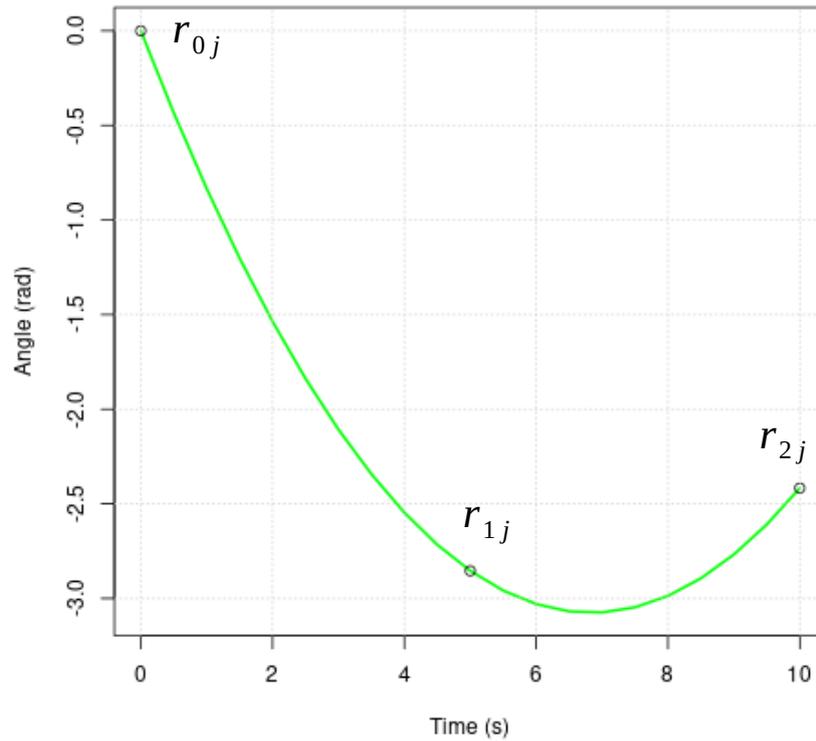


Figure 2 – Joint parameters.

Step 3 will contribute to the machine learning training phase (Step 2). In the following, 36 cases with distinct target points \mathbf{p}_{target} are analyzed. The results provided by the DH and SVM based algorithms are summarized in Table 4.

The error corresponds to the distance between the end-effector and the target, computed through the high fidelity model (Eq. 4). For the analysis of the first case, both algorithms used the initial guess provided by Step 1. For the analysis of the second case, and later on, the SVM learned from historical data provided by the optimization of previous cases. Furthermore, cases that report 3 “SVM Iterations” are situations in which the algorithm provided the final solution of the inverse kinematics as initial guess. The optimization of the high fidelity model (Step 3) required three steps to realize that the initial guess was also a stationary point. The computation of the distance between the end-effector and the target confirmed that the stationary point was an optimal design.

The error dispersion, summarizing the distance between the end-effector and the target for all cases, is shown in Fig. 3. Before the optimization of the high fidelity model (Step 3) the SVM algorithm provided the sharpest estimate for the optimization (Fig. 3, on the left side). After the optimization process (Step 3) the final result (Fig. 3, on the right side) was similar in both cases, and below the threshold $\varepsilon = 0.004$ m. A comparison about the performance of DH-based and SVM-based algorithms is shown in Fig. 4. It is possible to identify a notable trend in decreasing the number of iterations of optimization as the data availability to SVM-based method increases.

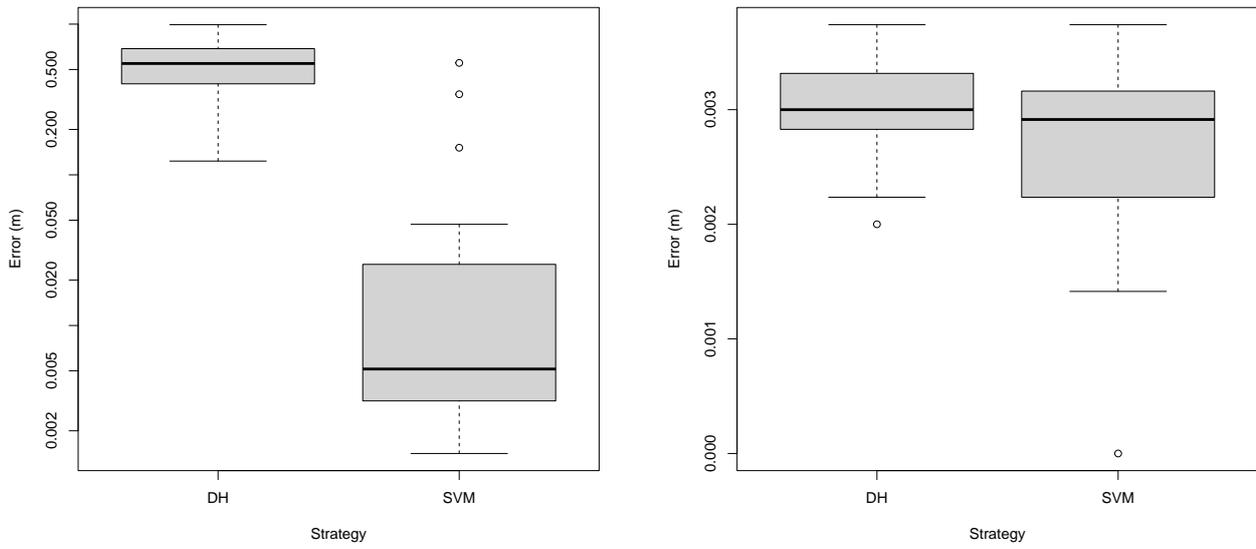


Figure 3 – Error dispersion before (left) and after (right) the optimization process.

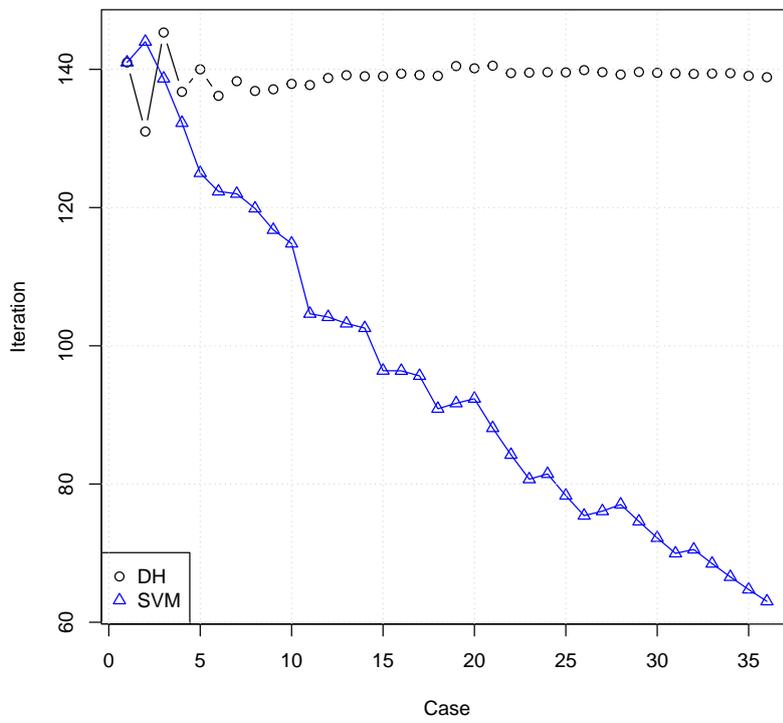


Figure 4 – Average number of iterations required by the optimization process.

CONCLUSION

The present paper proposed a physics informed machine learning strategy for the computation of the inverse kinematics of a space robot, taking into account effects from the dynamics of the satellite and the manipulator. The physics model consisted in a kinematic representation of a rigid robot with fixed base. This information, as a simplified formulation that omits dynamic characteristics, was useful at the training phase of the Support Vector Machine. After using historical data from four cases in the learning phase, the SVM-based algorithm demonstrated superior performance in estimating an initial guess for the optimization procedure.

Using historical data from ten cases, the method was able to infer the solution of the inverse problem before the optimization process. In each case, a new SVM model was trained. The data from the previous cases were kept, but the parameters of an SVM model were calculated from scratch. The convergence and accuracy of the estimate continued to improve as new data were added. It demonstrates the robustness of the present methodology, which kept a record of accurate predictions along with changes in training data. Motivated by the good results obtained from multiple data sets and problem specifications, the authors believe that the present methodology constitutes a contribution to the path planning of space manipulators in the presence of uncertainties that affect the target's position.

ACKNOWLEDGMENTS

D.A. Rade acknowledges the Brazilian research agencies Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) (grant #2018 / 15894-0) and Conselho Nacional de Desenvolvimento Científico e Tecnológico (grant #312068 / 2020-4) for the financial support to his research work. Ijar M. da Fonseca acknowledges the Brazilian Space Agency (AEB-UNIESPACO, grant #104006129).

REFERENCES

- Briese, L.E., Klöckner, A., and Reiner, M., 2017, "The DLR environment library for multidisciplinary aerospace applications", in: Proceedings of the 12th International Modelica Conference, Prague, Czech Republic, May 15-17, 2017, no. 132, Linköping University Electronic Press, pp. 929–938.
- Chen, F., Zhao, G., Yu, Y., Wang, J., Cao, P., 2017, "Modelica-based modeling and simulation of satellite on-orbit deployment and attitude control", in: 4th International Conference on Machinery, Materials and Information Technology Applications, Atlantis Press, pp. 196–204.
- Da Fonseca, I. M., Goes, L. C., Seito, N., da Silva Duarte, M. K., de Oliveira, E. J., 2017, "Attitude dynamics and control of a spacecraft like a robotic manipulator when implementing on-orbit servicing", *Acta Astronautica*, Vol. 137, pp. 490–497.
- Elmqvist, H., Mattsson, S.E., Otter, M., 1998, "Modelica: The new object-oriented modeling language", in: 12th European Simulation Multiconference, Vol. 5, Manchester, UK.
- Fonseca, I. M., Bainum, P. M., 2004, "Integrated structural and control optimization", *Journal of Vibration and Control*, Vol. 10, No. 10, pp. 1377–1391.
- Fonseca, I. M., Bainum, P. M., Lourencao, P. T., 2002, "Structural and control optimization of a space structure subjected to the gravity-gradient torque", *Acta Astronautica*, Vol. 51, No. 10, pp. 673–681.
- Fritzson, P., Aronsson, P., Pop, A., Lundvall, H., Nystrom, K., Saldamli, L., Broman, D., Sandholm, A., 2006, "Openmodelica - a free open-source environment for system modeling, simulation, and teaching", in: 2006 IEEE Conference on Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control, pp. 1588–1595.
- Lovera, M. Pulecchi, T., 2006b, "Object-oriented modelling for spacecraft dynamics: A case study", in: 2006 IEEE Conference on Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control, pp. 1898–1903.
- Pulecchi, T., Lovera, M., 2006, "A modelica library for space flight dynamics", in: Proceedings of the 5th International Modelica Conference, Citeseer.
- R Core Team, 2020, "R: A Language and Environment for Statistical Computing", R Foundation for Statistical Computing, Vienna, Austria, URL <https://www.R-project.org>.
- Santos, R. R., Rade, D. A., da Fonseca, I. M., 2022, "A machine learning strategy for optimal path planning of space robotic manipulator in on-orbit servicing", *Acta Astronautica*, Vol. 191, pp. 41–54.

RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.