Design of free-flyer robots for in-space cooperative additive manufacturing

João Vale[1], Rodrigo Ventura[2] and Marco Leite[3]

***Abstract*—The growing need for larger assets in Space, ranging from commercial space stations to exploration beyond Earth orbit, will require the capability of manufacturing and assembling large structures in micro-gravity. Considering the use of autonomous space robots to perform these tasks, this thesis studies the optimal design of a free-flyer robot for mobile manipulation with an emphasis on assembly and additive manufacturing. We propose a robot architecture comprising a dexterous six Degrees of Freedom (DoF) parallel manipulator attached to a free-flyer robot body endowed with six DoF propulsion. The design methodology decouples the parallel manipulator from the robot body. For the parallel manipulator, we define performance metrics for workspace volume and accuracy. We employ multi-criteria optimization to determine the geometric parameters which best tradeoff defined metrics. For the robot body propulsion, we find which geometries result in maximum thrust and torque along all directions, in both force and torque space, thus maximizing maneuverability. The construction of a Ground based prototype is studied and efforts towards its realization were conducted.**

***Index Terms*—Space robotics; In-orbit manufacturing and assembly; Mobile manipulation; Multi-objective optimization.**

I. Introduction

Space exploration is limited by current launching vehicles and systems: Constraints in mission design are imposed by the harsh launch conditions and limitations in payload volume and weight. The integration of in-orbit robotized manufacturing and assembly of parts, vehicles, and structures could be advantageous to mission design, reducing logistical requirements, launch system constraints, and Earth manufacturing constraints. With cooperative in-orbit manufacturing and assembly tasks in mind, this thesis proposes the design of ACROBAT: an aerial free-flying robot for pressurized micro-gravity environments (e.g ISS). The nature of its propulsion system disallows the use of said system in vacuum. Instead, the intent is for this to be a steppingstone towards the design and development of spacecraft.

A. Free-Flying Robots

A Free-Flying robot can be defined as a vehicle capable of navigating and maneuvering the space with six DoF. Multiple aerial Free-Flying Robots have been tested in micro-gravity aboard the ISS: SPHERES [1] is a free-flying robot operated by NASA, designed as a testbed for formation flight. SPHERES uses twelve carbon-dioxide cold gas thrusters for its propulsion. Astrobee [2] is in some ways the successor of SPHERES, building on SPHERES's legacy and lessons learned. Astrobee's propulsion system is relatively simpler, comprised of two large fans inside it with twelve duct valves for control. The Astrobee is also equipped with a 2 DoF manipulator. Int-Ball [3] is a free-flying robot operated by JAXA, designed to be an autonomous mobile camera using twelve small, encapsulated fans for its propulsion.

Space CoBot [4] is a aerial robot designed for indoor micro-gravity environments. Its propulsion system is comprised of six propellers, similar to the ones employed in traditional Earth multi-rotors. Unlike Earth multi-rotors, the propellers are placed so that the kinematics are holonomic.

B. System Description

Figure 1 shows a CAD model of the first iteration of ACROBAT. The first iteration was designed to be a Ground Prototype It can be decoupled into two main components:

1) A robot body equipped with six propellers, displaced such that the kinematics are holonomic, meaning the robot can freely move with six DoF. Considering the target environment, propellers fans were chosen instead of other alternatives, such as cold gas thrusters. This is because they are simpler, faster to prototype, and easier to integrate into the design. The robot body will also house the main computer, batteries and required electronics and sensors (e.g IMU, camera). The CAD model of the robot body is depicted in fig. 2, and it was conceptualized to have a distance between two adjacent propellers of 19cm.
2) In isolation, the robot body has unbounded motion, but a low actuation bandwidth. In other words, ACROBAT can have any position within the environment but the propeller's spin-up and spin-down time constrain the robot's fine movement capability which is detrimental for the targeted tasks. To enhance ACROBAT's manipulation capabilities, we attached a six DoF six Revolute-Spherical-Spherical (RSS) parallel manipulator, similar to HEXA [5] to the robot body, with a task dependent tool as the end-effector. Its CAD model can be seen in fig. 3.

C. Approach

This thesis aims to study how we should determine the geometric parameters of the proposed robotic system in order to maximize performance. To do so, we will decouple the

[1] Instituto Superior Técnico, Universidade de Lisboa
joao.luis.bruno.vale@tecnico.ulisboa.pt
[2] ISR-Lisboa, Instituto Superior Técnico, Univ. Lisboa
rodrigo.ventura@isr.tecnico.ulisboa.pt
[3] IDMEC-Lisboa, Instituto Superior Técnico, Univ. Lisboa
marco.leite@tecnico.ulisboa.pt

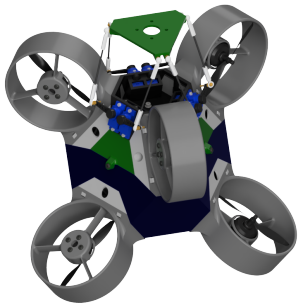

Fig. 1. ACROBAT CAD model.

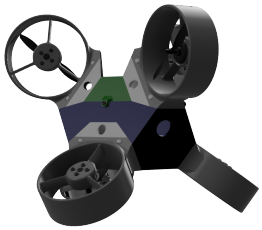

Fig. 2. Robot Body.

manipulator from the robot body and treat each as an individual system. To design the manipulator, we wish to maximize both work-space volume and accuracy. In the case of the robot body, we will look to maximize the maximum force and torque possible in any direction. We will also compare the optimized designs with ACROBAT's first iteration presented in section I-B.

D. Background

The formulated approach is a multiple objective optimization problem. This type of problem can be generically formulated as

$$\begin{aligned} \text{Minimize} \quad & \boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_n(\boldsymbol{x})) \\ \text{w.r.t.} \quad & \boldsymbol{x} \in \mathcal{X} \end{aligned} \tag{1}$$

(2)

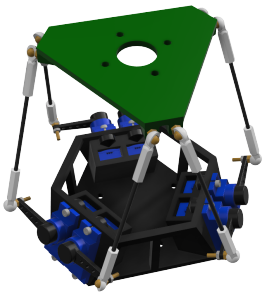

Fig. 3. Six RSS Manipulator.

where n is the number of objective functions f_i, $\boldsymbol{x}$ the vector of design parameters and $\mathcal{X}$ the feasible design set. Generally, it is impossible to find a global solution which optimizes all objective functions. A solution $\boldsymbol{x}^* \in \mathcal{X}$ is said Pareto Optimal if and only if there does not exist another point $\boldsymbol{x} \in \mathcal{X}$, such that $\boldsymbol{f}(\boldsymbol{x}) \leq \boldsymbol{f}(\boldsymbol{x}^*)$, and $f_i(\boldsymbol{x}) < f_i(\boldsymbol{x}^*)$ for at least one function [6]. The set of Pareto Optimal points form the Pareto frontier. A solution is chosen from the Pareto frontier doing trade-off analysis between the objective functions.

The literature of parallel manipulator design optimization is vast. However, strategies often consist in defining performance metrics relevant to the application, and using an optimization methodology to find the design parameters which improve these metrics [7]–[10]. The need for a multi-objective methodology, arrives from the fact that the performance metrics can play opposing roles. For example, an increase in workspace volume can lead to a decrease in dexterity [11], [12].

E. Contributions

Besides this thesis, a few contributions were made during the development of this thesis. First, a methodology was presented for the design of parallel manipulators and thrust actuated rigid bodies. An implementation of ACROBAT in a realistic physics simulator was done, to be used as a test-bench for future algorithms. Lastly, efforts were conducted in the construction and development of ACROBAT's prototype.

The developed design methodologies were presented [13] in the AeroBest 2021 conference, which took place in July 21-23, 2021 in Lisbon, Portugal. Aerobest is an ECCOMAS Thematic Conference on Multidisciplinary Design Optimization of Aerospace Systems.

II. Manipulator Design

A. Design Parameters

The manipulator is composed of a mobile platform where the end-effector is located, and a fixed base. The k-th Revolute-Spherical-Spherical arm, with $k \in \{1, \ldots, 6\}$, closes the loop between the platforms, connecting base anchor $\boldsymbol{B_k}$ to the platform's anchor $\boldsymbol{M_k}$. As depicted in fig. 4, each anchor $\boldsymbol{B_k}$ is connected to each arm by an actuated revolute joint. A rigid link of length h connects $\boldsymbol{B_k}$ to $\boldsymbol{H_k}$, and a rigid link of length d connects $\boldsymbol{H_k}$ to $\boldsymbol{M_k}$. The joints located at $\boldsymbol{H_k}$ and $\boldsymbol{M_k}$ are passive spherical joints. From fig. 4, we can obtain the following loop closure equation

$$\boldsymbol{i_k} = \boldsymbol{h_k} + \boldsymbol{d_k} = \boldsymbol{T} + \boldsymbol{R}\boldsymbol{m_k} - \boldsymbol{b_k}, \tag{3}$$

where $\boldsymbol{T} = \begin{bmatrix} x & y & z \end{bmatrix}^T$ is the translation and $\boldsymbol{R} \in \mathbb{R}^{3\times3}$ is the rotation matrix which define the mobile platform's pose $\boldsymbol{p} = \begin{bmatrix} x & y & z & \gamma & \theta & \psi \end{bmatrix}^T$. $\boldsymbol{R}$ can be described by three Euler angles (γ, θ, ψ), following the ZYZ convention

$$\boldsymbol{R}(\gamma, \theta, \psi) = \boldsymbol{R}(Z, \theta)\boldsymbol{R}(Y, \gamma)\boldsymbol{R}(Z, \psi). \tag{4}$$

According to fig. 5, the base anchor arrangement is defined by vector $\boldsymbol{b_k}$

$$\boldsymbol{b_k} = \begin{bmatrix} r_b\cos(\theta_b) & r_b\sin(\theta_b) & 0 \end{bmatrix}^T \tag{5}$$

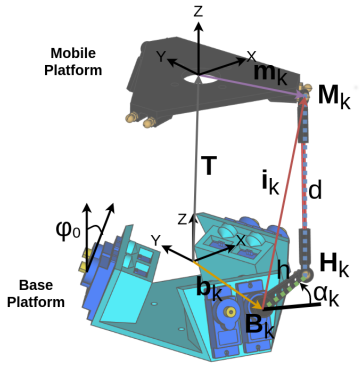

Fig. 4. Platform's dimensions.

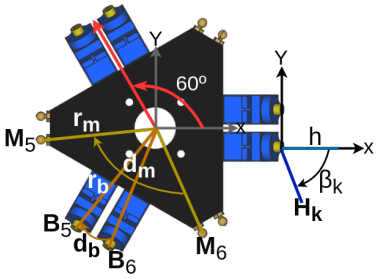

Fig. 5. Platform's dimensions.

where d_b is the angle between the anchor pairing, r_b the base platform radius and θ_b given by

$$\theta_b = \frac{2\pi\lfloor\frac{k+1}{2}\rfloor}{3} + (-1)^k d_b. \tag{6}$$

The mobile platform anchor arrangement is defined by vector $\boldsymbol{m_k}$

$$\boldsymbol{m_k} = \begin{bmatrix} r_m\cos(\theta_m) & r_m\sin(\theta_m) & 0 \end{bmatrix}^T \tag{7}$$

where d_m is the angle between the anchor pairing, r_m the mobile platform radius and θ_m given by

$$\theta_m = \frac{2\pi\lfloor\frac{k+1}{2}\rfloor}{3} + (-1)^k d_m. \tag{8}$$

The k-th revolute joint arm orientation is defined by angles ϕ_k and β_k

$$\phi_k = (-1)^{k+1}\phi_0 \qquad \beta_k = \frac{2\pi\lfloor\frac{k+1}{2}\rfloor}{3} + (-1)^k\beta_0 \tag{9}$$

where ϕ_0 and β_0 describe the revolute joint arm orientation in space.

In conclusion, to dimension the described manipulator, design parameters r_b, r_m, d_b, d_m, ϕ_0, β_0, d and h must be specified. The resulting vector of design parameters $\boldsymbol{s}$ is given by $\boldsymbol{s} = (r_b, r_m, d_b, d_m, \phi_0, \beta_0, d, h)$.

B. Evaluation

An important step for manipulator design is the choice of performance metrics to evaluate and compare different designs. Consider a vector of design parameters $\boldsymbol{s} \in \mathbb{R}^m$, a performance metric function f must evaluate $\boldsymbol{s}$ according to certain performance metric so that

$$L = f(\boldsymbol{s}) \tag{10}$$

where $L \in \mathbb{R}$ is the performance of $\boldsymbol{s}$ in a defined criterion. f can lack a closed-form solution, making methods to compute an approximate solution necessary.

The choice of metrics is heavily dependent on the tasks. Given the tasks at hand of assembly and additive manufacturing, we consider that both workspace volume and accuracy are desirable.

1) Workspace Volume: Workspace volume is a metric used to measure the size of a workspace. Workspace volume dimension depends on the type of workspace that is being measured. For instance, if the constant orientation workspace is considered, the workspace volume is a 3D volume. Formally, $V_{\mathcal{W}}$ is given by

$$V_{\mathcal{W}} = \int_{\mathcal{W}} d\mathcal{W} \tag{11}$$

where $\mathcal{W}$ is the targeted workspace.

2) Accuracy: A measure of accuracy must evaluate how a small displacement of the active joints $\delta\boldsymbol{\alpha}$ translates into a displacement of the end-effector pose $\delta\boldsymbol{p}$. These small active joint displacements may originate from the sensor noise, calibration errors or even thermal expansion and compression. Actuator displacement can be modeled by the kinematic Jacobian [14] such that

$$\delta\boldsymbol{p} = \boldsymbol{J}(\boldsymbol{p})\delta\boldsymbol{\alpha}. \tag{12}$$

Considering a actuator displacement bounded by a hyper-cube of side two

$$\|\delta\boldsymbol{\alpha}\|_\infty \leq 1, \tag{13}$$

if all actuated joints share a type (e.g. all actuated joints are revolute), the maximum displacement in translation $\sigma_{p,\infty}$ and the maximum displacement in rotation $\sigma_{r,\infty}$ is given by [15]

$$\sigma_{p,\infty} = \|\boldsymbol{J}_t\|_\infty \qquad \sigma_{r,\infty} = \|\boldsymbol{J}_r\|_\infty \tag{14}$$

where $\boldsymbol{J}_t$ and $\boldsymbol{J}_r$ are blocks of $\boldsymbol{J}$ such that $\boldsymbol{J} = [\boldsymbol{J}_t^T \boldsymbol{J}_r^T]^T$, obeying

$$\boldsymbol{v} = \boldsymbol{J}_t\dot{\boldsymbol{\alpha}} \qquad \boldsymbol{\omega} = \boldsymbol{J}_r\dot{\boldsymbol{\alpha}}. \tag{15}$$

Designwise, $\sigma_{p,\infty}$ and $\sigma_{r,\infty}$ measure translational and rotational sensitivity to actuator displacement and are representative of manipulator accuracy.

The aforementioned metrics only evaluate the performance in a given pose. To evaluate the sensitivity of the manipulator over the entire workspace, we propose the GTSI and the GRSI, given by

$$GTSI = \frac{\int_{\mathcal{W}} \sigma_{t,\infty}\, d\mathcal{W}}{\int_{\mathcal{W}} d\mathcal{W}} \qquad GRSI = \frac{\int_{\mathcal{W}} \sigma_{r,\infty}\, d\mathcal{W}}{\int_{\mathcal{W}} d\mathcal{W}}, \tag{16}$$

which must be minimized.

C. Design Space

This section will address the bounds and constraints of the manipulator's design space. Let $\mathcal{S}$ be a bounded box, representing the feasible design parameter space. Given the bounds imposed by the parameterization context and application, we assume that $\mathcal{S}$ is a bounded box so that $\mathcal{S} = \{0.01 \leq r_b \leq 0.06, 0.01 \leq r_m \leq 0.06, 0.1 \leq d_b \leq \frac{\pi}{3}, 0.1 \leq d_m \leq \frac{\pi}{3}, -\frac{\pi}{3} \leq \phi_0 \leq \frac{\pi}{3}, 0 \leq \beta_0 \leq \pi, 0.01 \leq d \leq 0.2, 0.01 \leq h \leq 0.2\}$.

However, not all $\boldsymbol{s}$ inside the bounded box $\mathcal{S}$ produce feasible manipulators. Two types of constraints still apply:

1) Design parameter vector $\boldsymbol{s}$ which lead to kinematically impossible manipulators; and
2) Design parameter vector $\boldsymbol{s}$ which lead to kinematically legal manipulators, but physically impossible given inter-link interference.

With this in mind, we apply a set of constraints of the type $C(\boldsymbol{s}) \leq 0$ to the design space.

D. Optimization

Given the antagonic nature of workspace volume and accuracy, we wish to find the set of $\boldsymbol{s}$ which best compromise those criteria. Considering this, we consider necessary to employ a multi-objective optimization methodology. Specifically, we wish to maximize the workspace volume ($V_{\mathcal{W}}$) while minimizing the global translational sensitivity GTSI and global rotational sensitivity GRSI. We can easily translate the problem into an all-minimization problem by considering the workspace volume symmetric ($-V_{\mathcal{W}}$).

With this considerations in mind, we can formulate the following multi-objective optimization problem:

$$\begin{aligned} \text{Minimize} \quad & (-V_{\mathcal{W}}, GTSI, GRSI) \\ \text{w.r.t.} \quad & \boldsymbol{s} \in \mathcal{S} \\ \text{subject to} \quad & C(\boldsymbol{s}) \leq 0\,. \end{aligned} \tag{17}$$

As far as we know, the performance metric functions presented in eqs. (11) and (16) lack a closed-form solution, so we will approximate their value by discretizing the work-space. In our implementation, we will use a constant orientation work-space (meaning that $\boldsymbol{R} = \boldsymbol{I}$), but the methodology can be easily expanded to consider the reachable work-space, at the cost of a heavier computational burden.

To compute a point-wise approximation of the Pareto front of the optimization problem formulated in section II-D, we will use the Direct Multisearch for Multiobjective Optimization [16] algorithm. Given that eq. 11 and eq. 16 lack a closed-form solution, their derivative also has no close-form solution, so a derivative-free method was chosen.

E. Results

The point-wise approximation of the Pareto front can be found in fig. 6 with a total of 3424 non-dominated points found after the algorithm completed a total of 2×10^5 objective function evaluations.

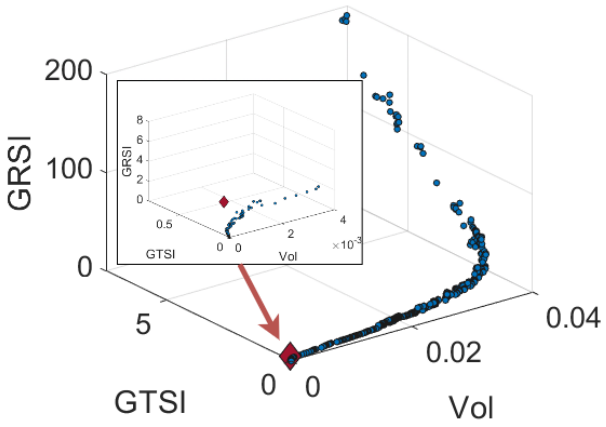

Fig. 6. Point-wise Pareto Front Approximation. The red diamond represents ACROBAT's first iteration manipulator.

From these results, we found that as the volume increases, both the rotational and translational sensitivity to actuator displacement also increase, resulting in a decrease in accuracy. On the other hand, fig. 6 suggests that the translational accuracy grows linearly with the rotational accuracy.

But one questions that arises is how should we choose a solution from the Pareto set. One approach to choose a final design is to search the Pareto set to find the solution with GTSI closer to ACROBAT's sub-optimal manipulator. This approach makes sense given ACROBAT's propulsion system lower actuation bandwidth, where the manipulator might need a larger workspace, while maintaining the same tanslational accuracy, to compensate the propulsion's system shortcomings. Table I compares the design parameters and performance between both designs and figs. 7 and 8 presents a visual representation of the joint placement. The color scheme goes as follows: The red patch is the mobile platform, the green patch the fixed platform, the magenta and blue line-segment are rigid links of size h and d, respectively. The optimized design has a substantial increase in workspace volume while maintaining the same accuracy.

TABLE I
DESIGN PARAMETERS AND PERFORMANCE OF ACROBAT'S FIRST ITERATION MANIPULATOR AND AROBAT'S FIRST ITERATION MANIPULATOR.

Design	Vol (10^{-3})	GTSI	GRTS
ACROBAT	0.408	0.163	2.539
Optimized	6.110	0.162	2.252

III. Robot Body Design

A. Propulsion Model

A single propeller i, rigidly attached to the body frame $\mathcal{B}$ coincident with the Center of Mass (CoM), produces a thrust $\boldsymbol{F_i}$ and torque $\boldsymbol{M_i}$ on the vehicle body while rotating at speed

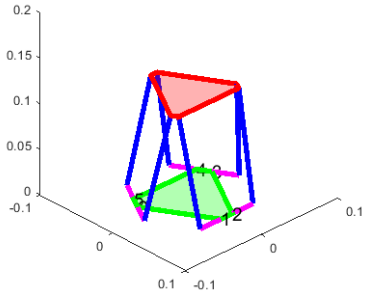

Fig. 7. ACROBAT's manipulator.

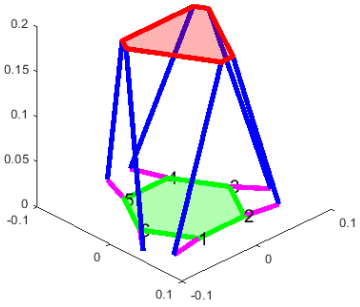

Fig. 8. Optimized design.

n_i (in revolutions per second) [4]. $\boldsymbol{F_i}$ results directly from the propeller thrust

$$\boldsymbol{F_i} = f_i \boldsymbol{u_i} \qquad f_i = K_1 \left|n_i\right| n_i \tag{18}$$

where $\boldsymbol{u_i}$ is a unit vector aligned with the propeller's axis of rotation and K_1 is a propeller constant. $\boldsymbol{M_i}$ is caused by the propeller non-central thrust $\boldsymbol{F_i}$ and reaction torque τ_i

$$\boldsymbol{M_i} = \boldsymbol{r_i} \times \boldsymbol{F_i} - \tau_i \boldsymbol{u_i} \qquad \tau_i = w_i \, K_2 \left|n_i\right| n_i \tag{19}$$

where $\boldsymbol{r_i}$ is the propeller position relative to $\mathcal{B}$, w_i is, for a positive or forward thrust, -1 if the propeller rotates clockwise or 1 if it rotates anti-clockwise and K_2 is another propeller constant.

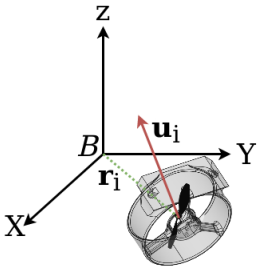

Fig. 9. Notation of a single propeller, relative to the body's CoM.

Propeller constants K_1 and K_2 are given by

$$K_1 = \rho D^4 C_t \qquad K_2 = \frac{\rho D^5}{2\pi} C_P \tag{20}$$

where ρ is air density, D is propeller diameter, C_T is the thrust coefficient and C_P the power coefficient. C_T and C_P are blade dependent coefficients [17]. We assume that the propeller has similar K_1 and K_2 for both directions of rotation.

Consider an actuation signal vector $\boldsymbol{q}$, with i-th actuation signal following $q_i = |n_i| n_i$, resulting in

$$\begin{bmatrix} \boldsymbol{F_i} \\ \boldsymbol{M_i} \end{bmatrix} = \boldsymbol{a_i} q_i \tag{21}$$

where

$$\boldsymbol{a_i} = \begin{bmatrix} K_1 \boldsymbol{u_i} \\ K_1 \boldsymbol{r_i} \times \boldsymbol{u_i} - w_i K_2 \boldsymbol{u_i} \end{bmatrix}. \tag{22}$$

To combine the effect of N propellers, resulting force $\boldsymbol{F}$ and torque $\boldsymbol{M}$ are given by the sum of eq. 21, which can be written in matrix form as

$$\begin{bmatrix} \boldsymbol{F} \\ \boldsymbol{M} \end{bmatrix} = \begin{bmatrix} \boldsymbol{a_1} & \dots & \boldsymbol{a_6} \end{bmatrix} \begin{bmatrix} q_1 \\ \vdots \\ q_6 \end{bmatrix} = \boldsymbol{A q} \tag{23}$$

where $\boldsymbol{A}$ is the actuation matrix. If $\boldsymbol{A}$ is a square matrix, meaning the robot is neither under nor over actuated, and $\boldsymbol{A}$ is full rank, meaning all propellers are non-redundant, $\boldsymbol{A}$ is invertible. As a result, it is possible to establish the following relationship

$$\boldsymbol{q} = \boldsymbol{A}^{-1} \begin{bmatrix} \boldsymbol{F} \\ \boldsymbol{M} \end{bmatrix}. \tag{24}$$

Matrix $\boldsymbol{A}^{-1}$ can be rewritten as a composition of matrices b and c

$$\boldsymbol{A}^{-1} = \begin{bmatrix} \boldsymbol{b}_1^T & \boldsymbol{c}_1^T \\ \vdots & \vdots \\ \boldsymbol{b}_6^T & \boldsymbol{c}_6^T \end{bmatrix} \qquad \text{with } \boldsymbol{b_i}, \boldsymbol{c_i} \in \mathbb{R}^3, \tag{25}$$

being possible to decouple torque and force and write $\boldsymbol{q}$ as

$$\boldsymbol{q} = \boldsymbol{b^T F} + \boldsymbol{c^T M}. \tag{26}$$

B. Parameterization

Considering the targeted 6 propellers, the actuation matrix described in eq. 23 is a 6×6 matrix. $\boldsymbol{A}$ depends on propeller constants K_1 and K_2. To bypass this dependency, we divide eq. 22 by K_1 resulting in

$$\boldsymbol{a_i'} = \begin{bmatrix} \boldsymbol{u_i} \\ \boldsymbol{r_i} \times \boldsymbol{u_i} - w_i \frac{K_2}{K_1} \boldsymbol{u_i} \end{bmatrix}. \tag{27}$$

Considering the size of the propellers that are expected to be used, around 4", the ratio $\frac{K_2}{K_1}$, takes values of magnitude 10^{-2}. Considering this, and for design purposes we will make the approximation of this term ($\frac{K_2}{K_1} = 0$), resulting in

$$\boldsymbol{a_i'} = \begin{bmatrix} \boldsymbol{u_i} \\ \boldsymbol{r_i} \times \boldsymbol{u_i} \end{bmatrix}. \tag{28}$$

Despite this approximation, the methodology can be easily expanded to cover non-zero values of $\frac{K_2}{K_1}$.

The byproduct of eq. 28 is a dimensionless actuation matrix. Now, to fully define it, we need to specify each propeller

position relative to the CoM $\boldsymbol{r_i}$ and thrust direction $\boldsymbol{u_i}$. Without loss of generality, let us assume that each propeller is located on a sphere of unit radius ($\|\boldsymbol{r_i}\| = 1$). Like depicted in fig. 10, $\boldsymbol{r_i}$ and $\boldsymbol{u_i}$ can be written as

$$\boldsymbol{r_i} = \begin{bmatrix} \cos(\theta_i)\sin(\gamma_i) \\ \sin(\theta_i)\sin(\gamma_i) \\ \cos(\gamma_i) \end{bmatrix} \quad \text{with } \theta_i \in [0,\pi] \wedge \gamma_i \in [0,2\pi) \tag{29}$$

$$\boldsymbol{u_i} = \begin{bmatrix} \cos(\beta_i)\sin(\alpha_i) \\ \sin(\beta_i)\sin(\alpha_i) \\ \cos(\alpha_i) \end{bmatrix} \quad \text{with } \beta_i \in [0,\pi] \wedge \alpha_i \in [0,2\pi) \tag{30}$$

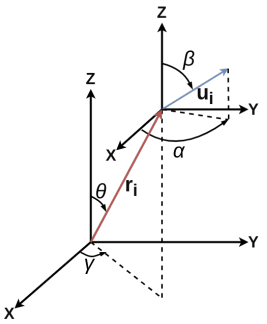

Fig. 10. $\boldsymbol{r_i}$ and $\boldsymbol{u_i}$ description relative to the body's CoM.

Considering eq. 29 and eq. 30 expanded for the six propellers, the resulting vector of optimization variables $\boldsymbol{s}$ is given by $\boldsymbol{s} = (\gamma_1, \ldots, \gamma_6, \theta_1, \ldots, \theta_6, \beta_1, \ldots, \beta_6, \alpha_1, \ldots, \alpha_6)$.

C. Evaluation

To choose the $\boldsymbol{s}$ which results in the most apt robot body, we must define a performance evaluation criteria. One approach is to use the maximum force and torque possible in any direction [4]. Let us consider that the actuation signal $\boldsymbol{q}$ is bounded between -1 and 1, such that[1]

$$\|\boldsymbol{q}\|_\infty \leq 1. \tag{31}$$

Let us also assume that $\boldsymbol{M} = 0$ and that $\boldsymbol{F} = F\boldsymbol{e}$ where $\boldsymbol{e}$ is the force direction. From eq. 26 and eq. 31 we get that

$$\|\boldsymbol{q}\|_\infty = \|F\boldsymbol{b}^T\boldsymbol{e}\|_\infty \leq 1, \tag{32}$$

resulting in the upper bound of F given by

$$F \leq \frac{1}{\|\boldsymbol{b}^T\boldsymbol{e}\|_\infty}. \tag{33}$$

in any direction $\boldsymbol{e}$. Likewise, the maximum force attainable along a given direction $\boldsymbol{e}$ is given by

$$F_{\boldsymbol{e}}^{max} = \frac{1}{\|\boldsymbol{b}^T\boldsymbol{e}\|_\infty}. \tag{34}$$

To obtain the maximum force attainable in any direction, we must minimize eq. 34 in all directions. Considering that $\|\boldsymbol{b}^T\boldsymbol{e}\|_\infty = \max_i |\boldsymbol{b}_i^T\boldsymbol{e}|$ and $|\boldsymbol{b}^T\boldsymbol{e}| \leq \|\boldsymbol{b_i}\|$, maximum force in any direction is given by

$$F^{max} = \frac{1}{\max_i \|\boldsymbol{b_i}\|} = \min_i \frac{1}{\|\boldsymbol{b_i}\|} \qquad i \in \{1,\ldots,6\}. \tag{35}$$

An analogous reasoning can be applied to obtain the maximum torque M^{max}, when $F = 0$, resulting in

$$M^{max} = \min_i \frac{1}{\|\boldsymbol{c}_i\|} \qquad i \in \{1,\ldots,6\}. \tag{36}$$

D. Design Space

The parameterization described in section III-B has the drawback of leading to many symmetric designs, meaning that a given configuration can be redundantly described by different $\boldsymbol{s}$. For instance, switching a $\boldsymbol{r_i}$, $\boldsymbol{u_i}$ pair with other pair, leads to a different $\boldsymbol{s}$ that represents that same design. To avoid such cases, we shall constraint the propeller position $\boldsymbol{r_i}$ so that γ_i is ordered by increasing values of i:

$$\gamma_1 \leq \gamma_2 \leq \gamma_3 \leq \gamma_4 \leq \gamma_5 \leq \gamma_6 \tag{37}$$

Other types of symmetric designs can be obtained by choosing an arbitrary parameter vector $\boldsymbol{s}$, and rotating every $\boldsymbol{u_i}$ and $\boldsymbol{r_i}$ by an arbitrary rotation, arriving at a $\boldsymbol{s}$ which represents the same design, but has a different $\boldsymbol{s}$. To try mitigate the symmetric designs caused by rotations, we apply a few constraints to the design space. First, we set a fixed $\boldsymbol{r_1}$ ($\gamma_1 = 0$ and $\theta_1 = \frac{\pi}{2}$). A fixed $\boldsymbol{r_1}$ constrains the possible set of rotations causing symmetric designs to rotations around $\boldsymbol{r_1}$. To further constraint the symmetric designs, we will also set $\boldsymbol{r_2}$ so that $\boldsymbol{r_1}$ and $\boldsymbol{r_2}$ always belong to the same plain, no matter the $\boldsymbol{s}$. An easy way to do this, given the current parameterization, is to set $\theta_2 = \frac{\pi}{2}$ and let γ_2 be a free optimization variable. Considering this, we rewrite the vector of optimization variables $\boldsymbol{s}$ as $\boldsymbol{s} = (\gamma_2, \ldots, \gamma_6, \theta_3, \ldots, \theta_6, \beta_1, \ldots, \beta_6, \alpha_1, \ldots, \alpha_6)$.

Finally, given that the propellers are assumed to be bi-directional, we can set $\boldsymbol{u}_i$ to only one hemisphere given by: $\beta_i \in [0,\pi] \wedge \alpha_i \in [0,\pi)$ for $i \in \{1,\ldots,6\}$. The resulting set of feasible designs $\mathcal{S}$ is a bounded box such that $\mathcal{S} = \{0 \leq \gamma_i \leq \pi, 0 \leq \theta_j \leq 2\pi, 0 \leq \beta_k \leq \pi, 0 \leq \alpha_k \leq \pi\}$ for $i \in \{2,\ldots,6\}$, for $j \in \{3,\ldots,6\}$, for $k \in \{1,\ldots,6\}$.

E. Optimization

We wish to maximize both F^{max} and M^{max}, which can be translated into a minimization problem by considering their negative counterpart ($-F^{max}$ and $-M^{max}$). With the discussed bounds and constraints in mind, we can formulate the following multi-objective optimization problem:

$$\begin{aligned} \text{Minimize} \quad & (-F^{max}, -M^{max}) \\ \text{w.r.t.} \quad & \boldsymbol{s} \in \mathcal{S} \\ \text{subject to} \quad & \gamma_2 \leq \gamma_3 \leq \gamma_4 \leq \gamma_5 \leq \gamma_6\,. \end{aligned} \tag{38}$$

Performance metrics F^{max} and M^{max} are calculated from $\boldsymbol{A^{-1}}$, which is numerically computed from $\boldsymbol{A}$, meaning they lack a closed-form solution. Consequently, their derivative is

[1]The infinity norm of $\boldsymbol{x}$ takes form $\|\boldsymbol{x}\|_\infty = \max\{|x_1|, \ldots, |x_n|\}$.

not available, making a derivative-free optimization methodology capable of addressing bounded problems with constraints necessary to estimate the Pareto front of the stated problem. Multiple algorithms have the capability of addressing this type of problem, namely evolutionary multi-objective approaches. However, based on our analysis we decided to use the Direct Multisearch for Multiobjective Optimization [16] algorithm.

F. Results

Point-wise approximation of the Pareto front can be found in fig. 11. The algorithm ran until a total of 4×10^5 objective function evaluations were generated starting from a random initialization. The Pareto front has a total of 447 non-dominated points.

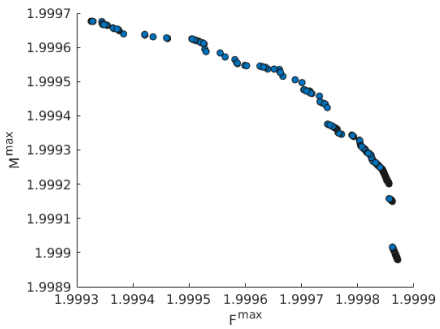

Fig. 11. Point wise pareto-front approximation.

To check how the $\boldsymbol{s}$ that make the Pareto front in fig 11 distribute over $\mathcal{S}$, we did the average and standard distribution of each optimization variable. Observing the order of magnitude of the standard deviation presented by the solutions, 10^{-3}, and considering the range of values taken by the performance metrics in the Pareto front, one can assume that the set of solution is well represented by it's own average. In fact, the data in fig. 11 suggests that the Pareto front is comprised of a single point, being the dispersion justified in the algorithm's numeric nature. If so, it shows that there are solutions to the problem in eq. 38 that are dominant in a way that maximize both F^{max} and M^{max}.

Figure 12 and 13 and shows a visual representation of propeller placement and thrust direction, between the design representative of the Pareto set and ACROBAT's first iteration (for comparison purposes, we will consider $\|\boldsymbol{r_i}\| = 1$ in both cases). The design obtained from the Pareto front has a performance vector of $(F^{max}, M^{max}) = (1.9996, 1.9994)$. For comparison, ACROBAT has a performance vector of $(F^{max}, M^{max}) = (2.0000, 2.0000)$. Performance-wise, both designs had similar results, which is not unexpected given that the problem defined in eq. 38 can hold multiple solutions (different $\boldsymbol{s}$ can have the same performance vector). Considering this, we believe the design presented in section I-B to be optimal, given that it has the same performance has an optimal design.

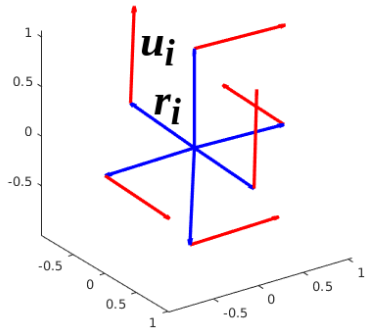

Fig. 12. Design obtained from the Pareto front.

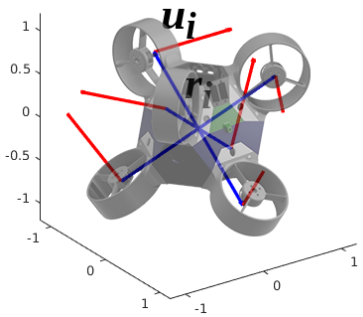

Fig. 13. ACROBAT's robot body.

IV. Simulation

To understand how the system of robot body + manipulator dynamically behaves and to create a platform for testing and validating control algorithms, ACROBAT was implemented in a realistic physics simulator. For this task, the Open Source Physics Simulator for Robotics Webots[2] was used. ACROBAT's first iteration implementation[3] in Webots is depicted in fig. 14. The implementation is modular, allowing for an easy change of geometric and/or physical parameters.

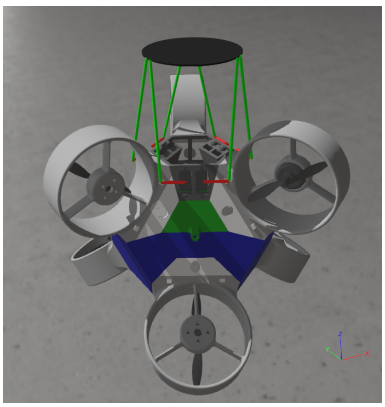

Fig. 14. ACROBAT's first iteration implemented in Webots.

[2]https://cyberbotics.com/
[3]https://github.com/CyberPoliceOfficer/Acrobat/tree/main/Webots

V. Prototype

To validate ACROBAT's design and help build a platform for the development of algorithms, namely for position control and cooperative AM, a prototype was aimed to be built. The prototype is expected to be tested in an air-bearing table, to emulate micro-gravity. Multiple experiments can be conceptualized, with fig. 15 showing one where two ACROBATs cooperatively 3D print a part, similar to the setup described in section I-B, with a robot dedicated to material deposition and other dedicated to extrusion.

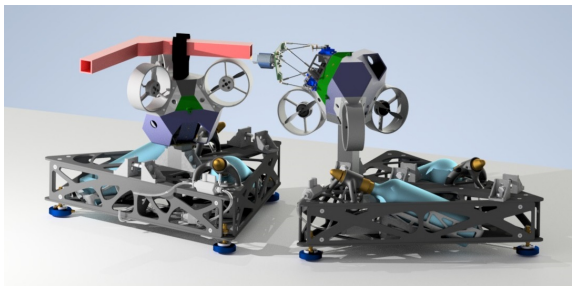

Fig. 15. CAD depiction of a proposed experiment.

ACROBAT's robot body was designed to be able to move with six DoF, but giving the limitations of the air-bearing table, it will only be able of movements within the table's plane. This means it will only be able to move in two translational directions and rotate in one, being reduced to three DoF. With this limitations in mind, we can safely remove three propellers from ACROBAT and mount it so that all the remaining propellers have their thrust vector parallel with the table's plane, like at depicted in fig. 15.

A. Manipulator

To validate the design and study the design options, ACROBAT's first iteration manipulator was built. The prototype with, an extrusion head mounted on the mobile platform, is depicted in fig. 16.

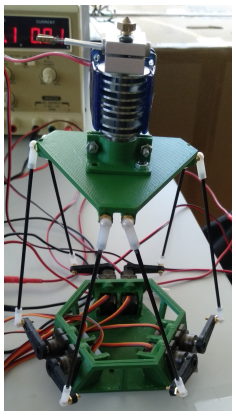

Fig. 16. Six-RSS Manipulator with a mounted extrusion head.

The manipulator's prototype had to rely on commercially available and 3D printable parts. Table II condensates the used material, and their source. The S-S joint was made from a carbon fiber rod. This rod had to be cut and iteratively filed to reach the specified length. To actuate the revolute joints six MG90S 180° servos were chosen.

TABLE II
Manipulator material.

Part	Source
Mobile Platform	3D printed using PLA
Fixed Platform	3D printed using PLA
R-S link	3D printed using PLA
S-S link	Carbon Fiber rod
joint at H_k	M2 Ball Socket Joint
joint at M_k	M2 Ball Socket Joint

B. Propulsion System

ACROBAT's ground prototype was projected to have a propulsion system composed of three propellers. Figure 17 depicts ACROBAT mounted with two ducted fans and their respective propellers and brushless DC motors. As previously stated, the propellers thrust vector is parallel to the plane in which the robot body is mounted.

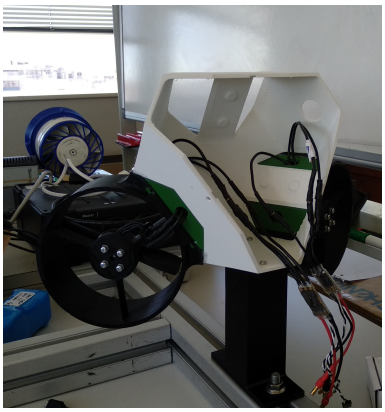

Fig. 17. ACROBAT's prototype mounted on a stand with two ducted fans installed.

1) Brush-less DC motors, ESC and propellers: To spin the propellers, Cobra CM-2204 2300 kv brush-less DC motors were used. This motors were chosen given their good price-performance, availability and experience in previous projects. A brush-less DC motors is electronically controlled by a ESC. The chosen ESC needed two important features: The ability to invert the spinning direction, while the robot is in operation and without rewiring, and having a low dead-band, meaning they must be able to command the motors do spin at relatively low RPM. With this considerations in mind, the ESC SEFM 30A from Aikon Electronics was chosen. Ideally, given ACROBAT's configuration, the propellers should have the same thrust coefficient, recall eq. (18), no matter the spinning direction. Unfortunately, given the use of commercially available parts, which follow a high demand for drone parts that clearly favor one direction of rotation, a fitting bi-directional propeller is not available. Consequently, the 4 inch propellers from HQProp were used.

2) Ducted fan: ACROBAT was idealized to have a ducted fan design, meaning that the propeller is cased by a tight duct, as depicted in fig. 18. In theory, and if properly built, this design choice would increase the propeller's performance while vastly improving the vehicle's safety. In practice there are a few challenges in the implementation of a ducted fan. It suffers from material compression during the printing, which creates a deviation between the desired diameter and the actual diameter. Other problem is that the propeller expands with a increasing rotating speed, creating a deviations in the propeller diameter. This challenges mean that manufacturing the must fit ducted fan would require a long iterative process of trial and error.

Fig. 18. Ducted fan with the propeller installed. d is the ducted fan diameter.

To evaluate the effect of the ducted fan diameter in the propeller's performance, the study shown in figs. 19 and 20 was conducted. The results were obtained by doing a RPM swipe for each ducted fan, while measuring the thrust and torque. With this data, the must fit K_1 and K_2, following eqs. (18) and (19) and in the least squares sense, was calculated and plotted in figs. 19 and 20. The experiment was conducted for a CW and CCW direction of rotation and each ducted fan was 3D printed using PLA. To conduct this experiment, the Series 1580 test stand from RCbenchmark[4] was used. This stand allows the user to send PWM signals to the ESC while measuring the propeller's RPM, thrust and torque.

Looking at figs. 19 and 20, we can clearly the uni-directional nature of the propeller, by analysing the values of K_1 for when there is no ducted fan installed. One assumption done in section III-C was that K_2 is negligible compared with K_1, assumption that is confirmed by the experimental data, given that K_1 is approximately two orders of magnitude bigger than K_2. But the most important take-away is that using a ducted fan improves the thrust coefficient K_1 in the direction of rotation that the propeller was designed to rotate, CW, but deteriorates in the opposite direction. A decreasing d leads to a increasing K_1 while rotating CW, but on the other hand K_1 does not decrease with a decreasing d while rotating CCW, with K_1 being unchanged. Looking at the torque coefficient K_2, we conclude that it is not as affected as the thrust coefficient, and remaining roughly the same.

[4]https://www.tytorobotics.com/products/thrust-stand-series-1580

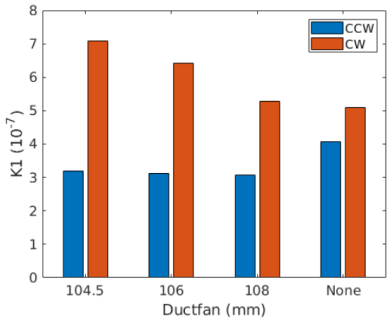

Fig. 19. Bar graph of K_1 for different ducted fan diameters d.

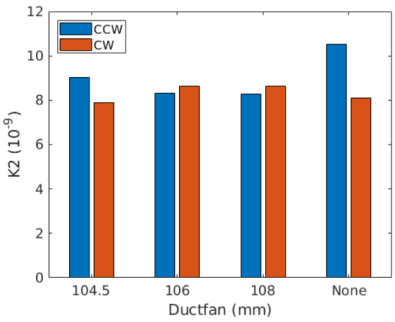

Fig. 20. Bar graph of K_2 for different ducted fan diameters d.

3) Calibration: One important step in the prototype's realization is the calibration of the propeller-motor-ESC system. The procedure and test-bench for data acquisition was similar to the one described in section V-B2. Ducted-fans with $d = 108mm$ were used. Even if their performance is inferior, having a ducted-dan is important for safety reasons and given that there were already three of this fans printed, they were used given that the time was constrained. A PWM sweap was conducted, with the average of a sample size of 10 taken in each PWM value. The experiment was repeated for the three propulsors, with results found in fig. 21. To minimize deviations, the used battery was fully charged in each experiment.

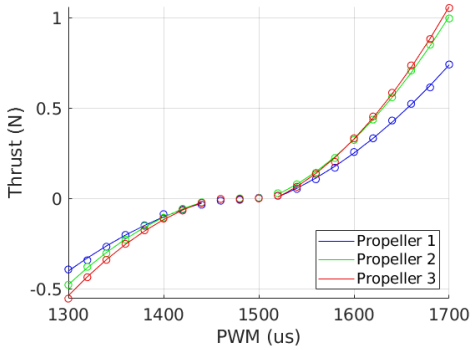

Fig. 21. Thrust data taken from each propeller. Each point presents the average of 10 samples, and the curve is a second order polynomial fit.

One thing to note, clearly visible in fig. 21, is that there's a actuation dead-band. The exact interval is different for each motor/ESC pairing, but empirical evidence suggest that we can safely actuate below $1440\mu s$ and above $1520\mu s$. To fit the acquired data, a second-order polynomial was used. Clearly the model in eq. (18) suggests a second order relationship. Considering the uni-directional nature of the propeller, a different fit was used for the two directions of rotation, with the corresponding data used accordingly. An interesting observation is the deviations between each propeller. Multiple explanations can be given, but the important lesson is that we must calibrate each propeller individually.

VI. Conclusions and Future Work

ACROBAT's parallel manipulator was studied: The inverse kinematic problem was solved, and the inverse kinematic Jacobian found. Both this results will be needed for the manipulator control and are need for the performance metric's computation. A design methodology was also presented, using a novel global accuracy metric. From this pareto-set, the relationship between the performance metrics was understood:

1) A higher work-space volume requires a loss in accuracy
2) The translational accuracy scales linearly with the rotational accuracy.

An approach to find an optimal version of ACROBAT's first iteration manipulator from the Pareto set was proposed, and the result illustrated, with the optimal version having a work-space roughly fifteen times larger, while maintaining the same levels of accuracy. A few improvements can be done to the current methodology, namely using a more general workspace and expanding the methodology to deal with un-actuated joints constraints.

A methodology for the design of ACROBAT's body was presented, with relevant metrics defined. The results revealed a interesting fact, that not only ACROBAT's body had the same performance as a solution found on the Pareto set, but also that it maximizes both trust and torque. Meaning that there are multiple solutions which maximize both trust and torque. With this in mind, we can safely guarantee that the current ACROBAT body is optimal.

Efforts towards the development of the ACROBAT's prototype were conducted. The parallel manipulator was assembled, with the accompanying software for kinematic control developed. The effect of having a ducted fan was studied, for different duct diameters. In the tested propeller, having a duct lead to a increase in the trust coefficient while rotating CW, with the largest gains being seen on the tighter ducts. While rotating CCW, having a duct worsened the trust coefficient. The relationship between the control PWM and the propeller trust and torque was found. To be ready for the targeted experiments, a few task still need to be completed. The manipulator's servos must be calibrated and the firmware to actuate the propellers must be created and tested.

Acknowledgments

I would like to thank Rodrigo Ventura, the supervisor of this thesis, for all the readiness and help granted. A big thanks to Alexandre Rocha, for designing ACROBAT's model in CAD, and providing assistance with any CAD related issue. I also would like to show my gratitude to Ana Custodio for kindly providing the code for the Direct Multisearch for Multiobjective Optimization algorithm.

References

[1] D. Miller, A. Saenz-Otero, J. Wertz, A. Chen, G. Berkowski, C. Brodel, S. Carlson, D. Carpenter, S. Chen, S. Cheng, D. Feller, S. Jackson, B. Pitts, F. Perez, and J. Szuminski, "Spheres: A testbed for long duration satellite formation flying in micro-gravity conditions," *Advances in the Astronautical Sciences*, vol. 105 I, pp. 167–179, 2000.

[2] M. Bualat, J. Barlow, T. Fong, C. Provencher, T. Smith, and A. Zuniga, "Astrobee: Developing a free-flying robot for the international space station," *AIAA SPACE 2015 Conference and Exposition*, pp. 1–10, 2015.

[3] S. Mitani, M. Goto, R. Konomura, Y. Shoji, K. Hagiwara, S. Shigeto, and N. Tanishima, "Int-Ball: Crew-Supportive Autonomous Mobile Camera Robot on ISS/JEM," *IEEE Aerospace Conference Proceedings*, vol. 2019-March, pp. 1–15, 2019.

[4] P. Roque and R. Ventura, "Space CoBot: Modular design of an holonomic aerial robot for indoor microgravity environments," *IEEE International Conference on Intelligent Robots and Systems*, vol. 2016-Novem, pp. 4383–4390, 2016.

[5] F. Pierrot, P. Dauchez, and A. Fournier, "HEXA: a fast six-DOF fully-parallel robot," in *Fifth International Conference on Advanced Robotics 'Robots in Unstructured Environments*. IEEE, 1991.

[6] R. T. Marler and J. S. Arora, "Survey of multi-objective optimization methods for engineering," *Structural and Multidisciplinary Optimization*, vol. 26, no. 6, pp. 369–395, 2004.

[7] R. E. Stamper, L. W. Tsai, and G. C. Walsh, "Optimization of a three DOF translational platform for well-conditioned workspace," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 4, no. April, pp. 3250–3255, 1997.

[8] Y. Lou, G. Liu, and Z. Li, "Randomized optimal design of parallel manipulators," *IEEE Transactions on Automation Science and Engineering*, vol. 5, no. 2, pp. 223–233, 2008.

[9] Z. Gao, D. Zhang, and Y. Ge, "Design optimization of a spatial six degree-of-freedom parallel manipulator based on artificial intelligence approaches," *Robotics and Computer-Integrated Manufacturing*, vol. 26, no. 2, pp. 180–189, 2010. [Online]. Available: http://dx.doi.org/10.1016/j.rcim.2009.07.002

[10] K. Tanakitkorn, "Design Optimisation of a 6-RSS Parallel Manipulator via Surrogate Modelling," *IOP Conference Series: Materials Science and Engineering*, vol. 501, no. 1, 2019.

[11] F. Hao and J. P. Merlet, "Multi-criteria optimal design of parallel manipulators based on interval analysis," *Mechanism and Machine Theory*, vol. 40, no. 2, pp. 157–171, 2005.

[12] R. Badora, "Stability properties of some functional equations," *Springer Optimization and Its Applications*, vol. 52, pp. 3–13, 2012.

[13] J. Vale, A. Rocha, M. Leite, and R. Ventura, "A multi-objective optimization approach to the design of a free-flyer space robot for in-orbit manufacturing and assembly," *International Conference on Multidisciplinary Design Optimization of Aerospace Systems - AeroBest 2021*, pp. 517–536, 2021.

[14] J. P. Merlet, *Parallel Robots*, 2nd ed. Springer Publishing Company, Incorporated, 2010.

[15] P. Cardou, S. Bouchard, and C. Gosselin, "Kinematic-sensitivity indices for dimensionally nonhomogeneous jacobian matrices," *IEEE Transactions on Robotics*, vol. 26, no. 1, pp. 166–173, 2010.

[16] C. P. Brás and A. L. Custódio, "On the use of polynomial models in multiobjective directional direct search," *Comput. Optim. Appl.*, vol. 77, pp. 897–918, 2020.

[17] B. McCormick, *Aerodynamics, Aeronautics, and Flight Mechanics*. Wiley, 1994.