A feasibility study of an artificial gravity system

70th International Astronautical Congress (IAC), Washington D.C., United States, 21-25 October 2019.

IAC-19-B3.7.13

A Feasibility Study of an Artificial Gravity System

Christopher Andrea Paissoni[a]*, Pier Carlo Berri[b], Dario Riccobono[c], Laura Mainini[d]

[a] *Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Corso Duca degli Abruzzi 129, Turin, Italy, 10040*, christopher.paissoni@polito.it
[b] *Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Corso Duca degli Abruzzi 129, Turin, Italy, 10040*, pier.berri@polito.it
[c] *Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Corso Duca degli Abruzzi 129, Turin, Italy, 10040*, dario.riccobono@polito.it
[d] *Visiting Professor, Department of Mechanical and Aerospace Engineering, Politecnico di Torino. Research Affiliate, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology.* laura.mainini@polito.it, *AIAA Senior Member*
* Corresponding Author

Abstract

Future crewed space exploration targets ambitious and distant destinations, requiring long duration missions that may largely affect the astronauts' health condition. To limit these effects, spacecraft will require additional solutions for the support of human safety, health and quality of life. Among those, artificial gravity might introduce a disruptive development to allow manned space exploration to achieve broader frontiers, reducing bone and muscle deterioration, motion sickness, and fluid redistribution. This work proposes the preliminary design of a rotating gravity system developed to support long-duration manned missions with a healthy living environment for human comfort. The design problem considers different aspects of the possible missions: it includes the identification of key design drivers and mission requirements, along with the exploration and assessment of possible system architectures accounting for deployment and operation constraints. The design process relies on the use of Multidisciplinary Design Optimization (MDO) methodologies to account for the interaction of multiple disciplines at the conceptual stage, and to benefit from this knowledge for the identification of the best design solutions for the rotating gravity system. This approach allows to evaluate the effect of several design choice at an early stage into the system development, to inform critical trade-off decisions and determine the feasibility of such a system with technology available today or in the near future.
Keywords: Artificial gravity, Multidisciplinary Design Optimization, Optimization.

1. Introduction

In the last years, the necessity to extend the human presence in space has gained increasing importance. In accordance with the Global Exploration Roadmap [1], several technological, physical, and ethical issues must be addressed before undertaking more challenging missions toward the other bodies of the Solar System. One of the main issues is related to the effects of a microgravity environment for long-term presence of the astronauts in space. The NASA Office of the Chief Health and Medical Officer (OCHMO) has intensively studied the negative effects on the human body in the space environment, considering the lessons learnt from the International Space Station (ISS) where the astronauts usually spent not more than six months. A number of side effects have been identified including bone and muscles deteriorations, motion sickness, fluid redistribution, disruption of senses, and immune function reduction. These side effects all relate to long time permanence and operations in microgravity environment and can deeply affect the health of the astronauts, not only during their permanency in space but also after their return on Earth. Artificially generating an Earth-like gravity acceleration is then highly desirable to counter the abovementioned issues.

In 1952, Dr. Wernher von Braun has been the first to describe the conceptual design of an orbiting space station (Figure 1) that would rotate to produce "synthetic" gravity [2].

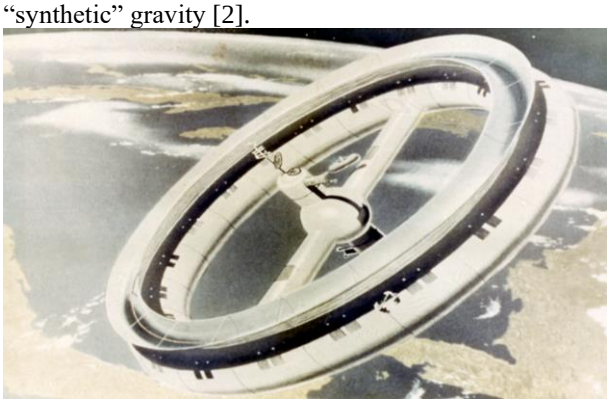

Figure 1: Braun's Space Station Concept (1952). Source: NASA/MSFC [3]. Image not subject to copyright as per NASA policy [4].

The Braun's design was a toroidal shaped space station of 250m diameter located in Low Earth Orbit (LEO). A gravity value of 1g was reached with a rotational motion of 4,9 Revolution Per Minute (RPM).
Several following concepts exploited the Braun's idea of a rotating system to generate gravity: the Discovery II (2005), a space transfer vehicle to Jupiter's orbit with three rotating habitable modules for the crew [5]; the Multi-Mission Space Exploration Vehicle (MMSEV), a space transport vehicle for long-duration, crewed mission with a rotating torus-ring habitat able to generate 0,69g [6], developed by NASA in 2011; the ISS Centrifuge DEMO (2011), a NASA's proposal for a rotating demonstration structure with the capability to become a sleep module of the ISS [7]; the Mars Gravity Biosatellite (2009), a LEO mission with the objective to study the effect of the Mars gravity on mammals [8].
Unfortunately, all the system concepts listed above have remained on paper due to several problems mainly related to their implementation. All the concepts rely on von Braun's idea to generate artificial gravity through rotating structures where astronauts are pushed to a relative floor by the resultant centrifugal force. The rotational speed is proportional to the radius of the toroid-like structure. For given spacecraft dimensions, a slow rotational speed is more advisable due to both a lower mass of the system and an easier controllability; however, to produce a given acceleration with a low angular rate, the system dimensions could raise dramatically and hinder launch operations, as well as manufacturing and maintenance. Then the spacecraft design will be a trade off between acceleration, size and angular rate.
Other variables, not directly related to the generated centripetal acceleration, strongly affect the design of such systems. For example, the system architecture in terms of number, shape and internal configuration of modules, deeply influence the physical model to be developed. Two possible opposite system architecture configurations could be identified: the first one is represented by a single toroidal module in which the structure is not interrupted by connection interface between modules. Alternatively, it is conceptually possible subdivide the toroidal structure in single modules, reducing the sectorial area considered for a single module.
In this paper, we address the design variables, regarding spacecraft geometry and kinematics, which mostly affect the design of a toroidal-like shape rotating space station. We leverage an optimization process aimed to identify sub-optimal solutions for this system architecture. We consider a modular toroidal spacecraft, and we aim at optimizing its geometrical and kinematic properties to find a design solution emerging from trade off studies aiming at maximizing the performance in terms of simulation of gravity (i.e. generating a uniform centripetal acceleration with low angular velocities) while containing the overall lifecyle costs.
In the Section II, there is description of the problem statement for the design of an artificial gravity system. Moreover, there is the presentation of the model developed in order to numerically assess the design parameters selected. The methodology followed for the optimization problem and the presentation of the optimization algorithms implemented in the MDO tool developed are reported in Section III. In Section IV, the main design results obtained from the optimization process are presented. At the end, the main conclusions of this work are drawn in Section V.

2. Problem Statement

This work is intended as a preliminary study to assess the feasibility of an artificial gravity environment for long duration human presence in space. The habitat will be placed in Low Earth Orbit, to ease cost effective and frequent resupply missions, crew turnaround and emergency evacuation in case of accidents.
We focus on a ring architecture, consisting of n interconnected modules. The radius of the structure R affects the number of modules, while the individual dimension and weight of each module are limited by the capabilities of the launch vehicle.
The architecture is optimized to achieve maximum pressurized volume at minimum manufacturing and deploying costs, while providing Earth-like gravity and a low angular velocity, in order to preserve the comfort of the crew and avoid motion sickness issues.
Table 1 summarizes the independent design variables considered for the study; most of them are related to the spacecraft geometry, and define the basic dimensions of ring radius R, radius of a single module r, while the centripetal acceleration g is related to the system performance in terms of comfort of the crew.

Table 1: Description of the design variables considered for the optimization.

Variable	symbol	unit	range
Ring radius	R	[m]	(0, 1000)
Number of modules	n	[ ]	(1, ∞(
Radius of a module	r	[m]	(0, R)
Centripetal acceleration	g	[m/s^2]	(3, 9.81)

The first approximation computational model employed is schematically shown in Figure 2; the following paragraphs briefly explain the assumptions adopted for each section of the model.

2.1. Computation of geometrical and kinematic properties

The set of design parameters listed in Table 1, namely R, n and r, define the simplified geometry of the spacecraft.

Table 2: Cost and capabilities of existing launch vehicles

Launcher	Max payload to LEO [t]	Fairing length l_F [m]	Fairing radius r_F [m]	Cost per launch (million USD)
Ariane 5 ES	21	17	2.7	150
Sojuz 2-1B	7.8	11.43	2.05	61
Falcon 9 v1.2	13.68	13.1	2.6	62
F9 expendable	22.8	13.1	2.6	92
Falcon Heavy	18.11	13.1	2.6	90
FH expendable	63.8	13.1	2.6	150
Delta IV	13.14	10.84	2	164
Delta IV Heavy	28.79	19.1	2.5	350
Atlas V	18.5	9	2	109

These variables affect the estimation of habitable volume and the angular rate required for a given level of centripetal acceleration. Additionally, the choice of a suitable launch vehicle is dictated by mass and volume limits of available launchers.

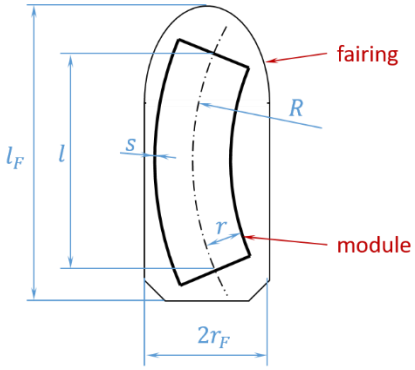

Figure 2: Simplified geometry of one spacecraft module inside the launcher payload fairing.

The geometrical relationships between the dimensions of the structure are the result of geometrical compatibility of the modules:

$$l = 2R\sin\frac{\pi}{n}$$
$$R > r$$
$$2(R+r)\sin\frac{\pi}{n} \leq l_F$$
$$(R+r) - (R-r)\cos\frac{\pi}{n} \leq 2r_F$$

where r_F and l_F are the radius and length available inside the launcher fairing. The dimension of the individual module is constrained by the available payload space in the fairing of the launch vehicle. Table 2 summarizes the capabilities of the considered launch vehicles.
Given the mass and dimensions of the individual module, the launch vehicle is chosen as the least expensive among those with compatible capabilities in terms of maximum payload to LEO and fairing dimensions, according to Table 2. The total volume of the spacecraft is approximated as the volume of a torus with the same radius, given by:

$$V_{tot} = 2\pi^2 R r^2$$

while the internal volume is estimated by multiplying the total by an empirical corrective coefficient k_{corr}, obtained by fitting available data regarding the habitation modules of the International Space Station:

$$V_{int} = k_{corr} V_{tot}$$

The centripetal acceleration generated by the rotating structure is given by:

$$g = \omega R^2$$

where ω is the angular rate of the spacecraft. To avoid sickness resulting from Coriolis forces experienced by the crew, this angular velocity shall be limited to a maximum value of 2 rpm [6].

2.2. Computation of pressurization and centrifugal stresses

To compute structural stresses experienced by the spacecraft, as a first approximation we assume a thin-wall torus shaped geometry, subject to pressure and centrifugal forces.
By considering a section of the torus and imposing the equilibrium between pressure load and tensile stress, the contribution to the latter given by pressurization is:

$$\sigma_{1,p} = \frac{p}{2}\frac{r}{s}$$

where s is the thickness of the wall. Similarly, the contribution of centrifugal forces on σ_1 is computed imposing the equilibrium between tensile stresses and centrifugal load for a section of the ring:

$$\sigma_{1,c} = \frac{\rho_l \omega^2 R}{4\pi r s}$$

where $\rho_l = \rho_{average} \pi r$ is the linear density of the module, estimated from empirical data and taking into account the structural mass increase needed to bear the centrifugal load itself.
The tangential component of stress σ_2 is computed only taking into account the pressure load, similarly to $\sigma_{1,p}$, yielding to:

$$\sigma_2 = \frac{r}{s} p$$

Then, the total stress σ_v is estimated with the von Mises criterion.

2.3. Mass estimation

The mass of each module is estimated by fitting available data regarding the habitable modules of the International Space Station, corrected with the additional mass needed to bear the centrifugal loads.
For this purpose, two von Mises stresses are computed, one neglecting centrifugal loads and the other considering them:

$$\sigma_{v1} = \sqrt{\left(\sigma_{1,p} + \sigma_{1,c}\right)^2 + \sigma_2^2 - \left(\sigma_{1,p} + \sigma_{1,c}\right)\sigma_2}$$

$$\sigma_{v2} = \sqrt{\sigma_{1,p}^2 + \sigma_2^2 - \sigma_{1,p}\sigma_2}$$

In both cases the model computes the wall thickness required for keeping the von Mises stress equal to the material yield stress divided by a safety factor k:

$$\frac{(s_{1,2})_{i+1}}{(s_{1,2})_i} = \frac{k\sigma_{v1,2}}{\sigma_{02}}$$

where $(s_{1,2})_i$ is the wall thickness used for the stress estimation and $(s_{1,2})_{i+1}$ is the updated wall thickness
Then, the total mass is updated by adding the structural mass resulting by the thickness increase. The stresses are updated and the process is repeated until convergence.

2.4. Cost estimation

The total cost C is estimated as the sum of manufacturing cost c_M for each module plus the total cost of launch $c_{L,T}$:

$$C = c_M + c_{L,T}$$

Cost of launch is obtained from available data related to most common launch vehicles (see Table 2), taking into account the total number of launches required. In particular, for a given module geometry and weight, we choose the cheapest launcher available, among those compatible in terms of fairing size and maximum payload to LEO. Then the cost of the selected launch vehicle c_{LV} is multiplied by the number of modules to be launched to estimate the total launch cost:

$$c_{L,T} = n\, c_{LV}$$

The manufacturing cost of the spacecraft modules is estimated by fitting data of ISS modules, which result to be approximately proportional to the spacecraft dry weight:

$$c_M = k\, m$$

where the coefficient k is found by a statistical study.

3. Methodology

The task of finding an optimal configuration for the artificial gravity system is a constrained multi-objective minimization problem.
We define a four element design vector $\boldsymbol{x} = \{g, r, R, n\}$ where g is the centripetal acceleration, r is the radius of the spacecraft modules, R is the radius of the entire spacecraft and n is the (integer) number of modules.
The structure of the problem is schematically explained in the diagram of Figure 3, which shows the connections between the various disciplinary blocks.

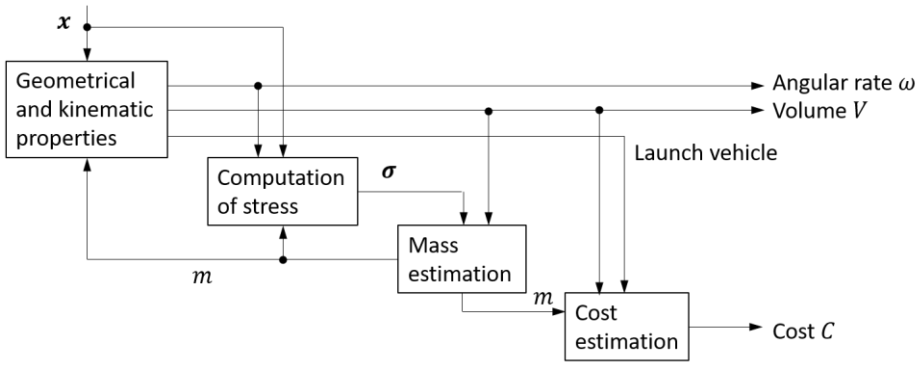

Figure 3: Structure of the problem.

The centripetal acceleration is bounded between 3 and 9.81 m/s^2 to lie inside the comfort zone described by [9]; additionally, the upper bound for the angular rate of the structure is set to 0.2 rad/s, approximately equivalent to the 2 rpm limit set by [9]. An upper bound for spacecraft radius R is set to 1000m: for values greater than this limit, the limited specific tensile strength of available materials starts to significantly increase the structural mass fraction, rapidly reaching impractical values.
As an additional constraint, for geometric compatibility reasons, the radius of a single module r shall be less than the overall spacecraft radius. However, the dimensions of the payload fairings of available launch vehicles likely set a much more restrictive upper limit for r. The optimization problem is defined as:

$$\begin{aligned} &\min_{\boldsymbol{x}} J(\boldsymbol{x}), \\ \text{s.t.} \quad &3m/s^2 \le g \le 9.81m/s^2 \\ &\omega(\boldsymbol{x}) \le 0.2rad/s \\ &R < 1000m \\ &r < R \end{aligned}$$

We propose two possible combinations of objectives to optimize:

1. Minimize estimated cost C and maximize internal habitable volume V, and
2. Minimize estimated cost C and angular rate ω.

For the first option, we adopt a weighted sum approach to define the following objective function:

$$J_1(\boldsymbol{x}) = -\alpha\hat{V} + \beta\hat{C}$$

where α and β are two scalar weights, with $\alpha + \beta = 1$. The coefficients α and β are chosen in the $(0,1)$ interval, and to compare homogeneous values the two objective functions are normalized as: $\hat{V} = V/1m^3$, $\hat{C} = C/10^6 USD$.
For the second option, the objective function is:

$$J_2(\boldsymbol{x}) = \alpha\hat{\omega} + \beta\hat{C}$$

Where $\hat{\omega} = \omega/10^{-6}\, rad/s$ is the normalized angular velocity. The objective function is nearly linear locally but has several step discontinuities caused by the discrete choice of one launch vehicle among a set of alternatives. We compare different optimization algorithms for the considered problem. In particular, we focus on two standard local search methods (Interior Point Algorithm and Sequential Quadratic Programming) to determine a baseline performance and two global search gradient free strategies (namely genetic algorithms and pattern search), which are likely more suitable to our inherently discontinuous problem.

3.1. Interior Point Algorithm

The Interior Point algorithm [10-14] combines the direct solution of the Karush-Kuhn-Tucker (KKT) equations and a Conjugate Gradient approach to solve a sequence of approximated minimization problems. This method is suited for large-scale nonlinear problems.

The inequality constraints are converted into equality constraints by adding to the objective function a logarithmic barrier function. Given the problem:

$$\min_{\boldsymbol{x}} J(\boldsymbol{x}), \text{ s.t. } h(\boldsymbol{x}) = 0 \text{ and } g(\boldsymbol{x}) \le 0$$

the approximate problem, for each $\mu > 0$, is:

$$\begin{aligned} &\min_{\boldsymbol{x},\boldsymbol{s}} J_\mu(\boldsymbol{x},\boldsymbol{s}) = \min_{\boldsymbol{x},\boldsymbol{s}} J(\boldsymbol{x}) - \mu \sum_i \ln(\boldsymbol{s}_i), \\ &\text{s.t. } h(\boldsymbol{x}) = 0 \text{ and } g(\boldsymbol{x}) + \boldsymbol{s} = 0 \end{aligned}$$

The solution of the approximate problem J_μ approaches the minimum of J as μ decreases to zero. At each iteration, the algorithm attempts a direct step to solve the KKT conditions for the approximate problem. If the direct step fails (e.g. if the approximate problem is not locally convex near the current point), a Conjugate Gradient step is performed to minimize a quadratic approximation to the approximate problem in a trust region.
For our application, the required starting point x_0 is chosen randomly among the bounds. Variables constrained by both a lower and an upper bound are chosen with uniform probability distribution, while those with only a lower bound are chosen by assuming a uniform distribution for their inverse. In order to reduce the dependence of the solution on the starting point, the optimization is performed several times for different $\boldsymbol{x}_0$, and the best solution in terms of value of the objective function J is kept.

3.2. Sequential Quadratic Programming (SQP)

A Sequential Quadratic Programming method [15-19] evaluates the local Jacobian and Hessian to obtain a local quadratic approximation of the objective function, in the neighborhood of the current point; equality and inequality constraints are locally linearized. The approximated Quadratic Problem at iteration q:

$$\begin{aligned} \min_s Q(s^q) = \min_s &J(x^q) + \nabla J(x^q)^\top s^q + \frac{1}{2} s^{q\top} B s^q, \\ \text{s.t.} \quad &\nabla g_j(x^q)^\top s^q + g^j(x^q) \le 0 \\ &\nabla h_j(x^q)^\top s^q + h^j(x^q) = 0 \end{aligned}$$

is solved to find the search direction s^q. Then, a 1D search is performed to compute the step length α^q, and the evaluation point is updated as:

$$x^{q+1} = x^q + \alpha^q s^q$$

For our implementation, we adopt the same starting point used for the Interior Point Algorithm (Section III.1).

3.3. Genetic Algorithm

Genetic algorithms were first introduced in 1970s as search methods used to find exact or approximate solutions to optimization and search problems [20-24]. Genetic algorithms emerge as a particular class of evolutionary algorithms that are adaptive heuristic search methods inspired by evolutionary biology, such as the natural selection and the natural genetics. They can be used to find optimal solutions in many applications of high complexity. Their heuristic features are used alternatively to other techniques such as simulated annealing, hill climbing, or tattoo search.

Genetic algorithms can explore the search space globally and are based on probabilistic strategies, in contrast with many local search methods. GAs simulate natural selection of best individuals, which will breed to generate the successive generations. The classical terminology calls “individual” a solution of the problem. All the considered individuals form a population. Each individual encodes its characteristics in one string called “chromosome”. According to the biology analogy, a chromosome is a sequence of alleles representing one quantum of information, such as bit, digit, and letter, etc. [21].

The genetic algorithm methodology starts with the initialization of the population, which is typically randomly generated. The population size is related with the nature of the problem and can reach several thousands of possible solutions. In general, the population is generated to cover the entire range of possible solutions, called the search space. In case of known areas where optimal solutions are likely to be found, the solutions may be “seeded” in such areas.

Within each generation, a portion of the population is selected to generate a new population which will become the next generation. The selection is based on the fitness (i.e. the value of the objective function) of each solution where fitter solutions (characterized by a lower value of the objective function) are more likely to be selected. A potential issue may be the premature convergence on a poor solution. For this reason, stochastic methods are designed to select a small portion of less fit solutions in order to keep the diversity of the population large.

The following step is to breed the selected population to generate the next generation. Two operations may be performed in this phase: crossover (also called recombination) and/or mutation. Each new solution is generated by breeding a pair or “parent” solutions selected from the pool previously selected. The “child” solution, generated by means of crossover and/or mutation, shares some characteristics of its “parents”. The process continues until a new population of the appropriate size is generated. The use of more than two "parents" may help with the obtaining a better "child" solution [22].

This process ends when a termination condition has been reached. Typical termination conditions may be a solution that satisfies the criteria is found, the fixed number of generations is reached, no better results can be obtained because the highest-ranking solution’s fitness is reaching or has reached a plateau.

For our application, to improve the repeatability of the optimizations, the Genetic Algorithm is executed ten times, and the better result is kept.

3.4. Pattern Search

In this paper we adopt the Generalized Pattern Search (GPS) method as implemented in Matlab. This method has been developed by Torzcov in 1997 [25] characterizing the pattern search methods, developed by Hooke and Jeeves [26] through a generalization of their method, resenting a global convergence theory [27-29]. These methods are very effective for some engineering problem in which the evaluation of the objective and constraint functions implies a high computational cost, as an alternative to using less expensive surrogates. The GPS algorithms compute a sequence of points that approach an optimal point. At each step
of the iteration, after the identification of an initial point, the algorithm searches a set of points, so-called “mesh” around that point. This mesh is created by adding a scalar multiple of a set of vectors called pattern to the current point identified. After that, the algorithm is able to find the best point in the mesh which improves the value of the objective function at the current point considered. Then, the point identified after this comparison becomes the new current point. The algorithm continues for a define number of iterations, eventually considering a tolerance value for which the iteration is terminated.

3.5. Simulated Annealing

The Simulated Annealing (SA) [30] is natural metaheuristic method based on an analogy between the quenching physical process for solids and the solution for a combinatorial optimization problem in order to find a good solution for such these problems [31-34]. This method derived from the Metropolis method. In particular, these methods were introduced for the local optimization or iterative improvement, in which an initial solution is repeatedly improved by making small local iterations until no such alteration yields a better solution. SA randomizes this procedure in a way that allows for occasional uphill moves (changes that worsen the solution), in an attempt to reduce the probability of a result, SA strength is that it avoids to getting caught at local maxima-minima solutions. This algorithms follow the following procedure: (i) Generation of the random solution for the define variable space; (ii) Calculation of

the cost using a pre-defined cost function; (iii) Generation of random solution in a sub-space close to the solution previously found; (iv) Calculation of the new value of the cost function; (v) Comparison of the new and old solution calculated. If the new solution is better with respect the problem defined, move on the new solution. If it is not, the algorithm calculates an "acceptance probability" which is a sort of recommendation on whether or not to jump to the new solution. (vi) repeating of steps from (iii) to (v) until an acceptable solution is found considering a certain tolerance value and a maximum number of iterations that could be performed.

4. Design Results

All the considered optimization algorithms identify a consistent minimum for the objective function. The optimizations are executed on a common laptop PC with an Intel i7 6500U processor, running Windows 10 and Matlab r2016a. The performance of the optimization algorithms in terms of computational time is shown in Table 3.

The two objective functions involve discrete variables and parameters, namely the number of modules and the discrete set of available launch vehicles. As a consequence, despite the used models being simple and nearly linear locally, the objective functions consist globally in a set of regular sections separated by step discontinuities. As an example, Figure 4 shows the dependency of the objective function J_2 upon the spacecraft radius and the centripetal acceleration. This particular situation poses some difficulties for the application of the gradient-based algorithms, since the gradient itself is not defined in the discontinuities. In this condition those search algorithms become strongly dependent on the choice of a suitable starting point.J_2 For this reason, to get a reliable result, the optimization needs to be repeated several times with different starting points, randomly sampled inside the search space. This greatly increases the global computational time, and partly compromises the fast convergence rate typical of gradient based methods. Conversely, the two global methods (and in particular the GA for its applicability to discrete variables) are more suitable for the problem, despite the higher computational cost.

For the first objective function J_1, the optimum point is always characterized by the maximum angular rate and the minimum centripetal acceleration allowed by the bounds. In this case, the Pareto front (Figure 5) is approximately a straight line: in fact, according to our simplified model, the cost of launch and construction results almost proportional to the pressurized volume of the spacecraft. Additionally, being the two objectives nearly proportional, there exist basically two solutions: one optimizes only the cost, regardless the internal volume, while the other considers only the volume. The transition between the two optimum solutions is quite sharp, and happens near $\alpha = 0.75$ and $\beta = 0.25$ (note that the Pareto front of Figure 4 is generated for $0.73 \leq \alpha \leq 0.77$ and $0.23 \leq \beta \leq 0.27$ **Errore. È prevista una cifra.**). In fact, it appears that our model can be locally approximated with the hyperplane $0.75\hat{V}(x) = 0.25\hat{C}(x)$**Errore. È prevista una cifra.**, in the region where the two optimum solutions are found. Then, deending on the weights of the objectives, the optimum will be either near the lower bound of cost or near the upper bound of volume. Clearly, the only solution characterized by reasonable costs is the smaller one, resulting in an already large volume of 3800 m^3 and a cost of about 11.8 billion USD.

The Pareto front of the second objective function J_2 is shown in Figure 6. In this case, the volume is not considered an objective since the lower bounds already gives a very large value. This graph highlights that a small reduction in angular rate can be achieved without affecting greatly the cost. Then, a point characterized by a slightly bigger radius and smaller angular rate can be chosen to increase the comfort of the crew or allow the simulation of higher accelerations.

Table 3: Average computational time for the considered optimization algorithms.

Algorithm	Computational time [s]	
	J_1	J_2
Interior Point	10.8502	8.4247
SQP	3.4208	3.1468
Genetic Algorithm	35.5161	29.6129
Pattern Search	18.2812	26.1764
Simulated Annealing	47.8688	54.8163

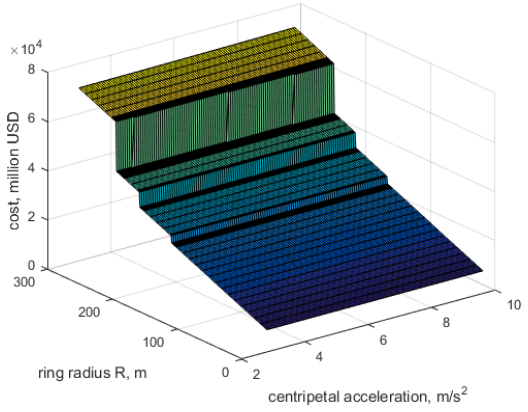

Figure 4: Objective function J_2 versus radius and centripetal acceleration.

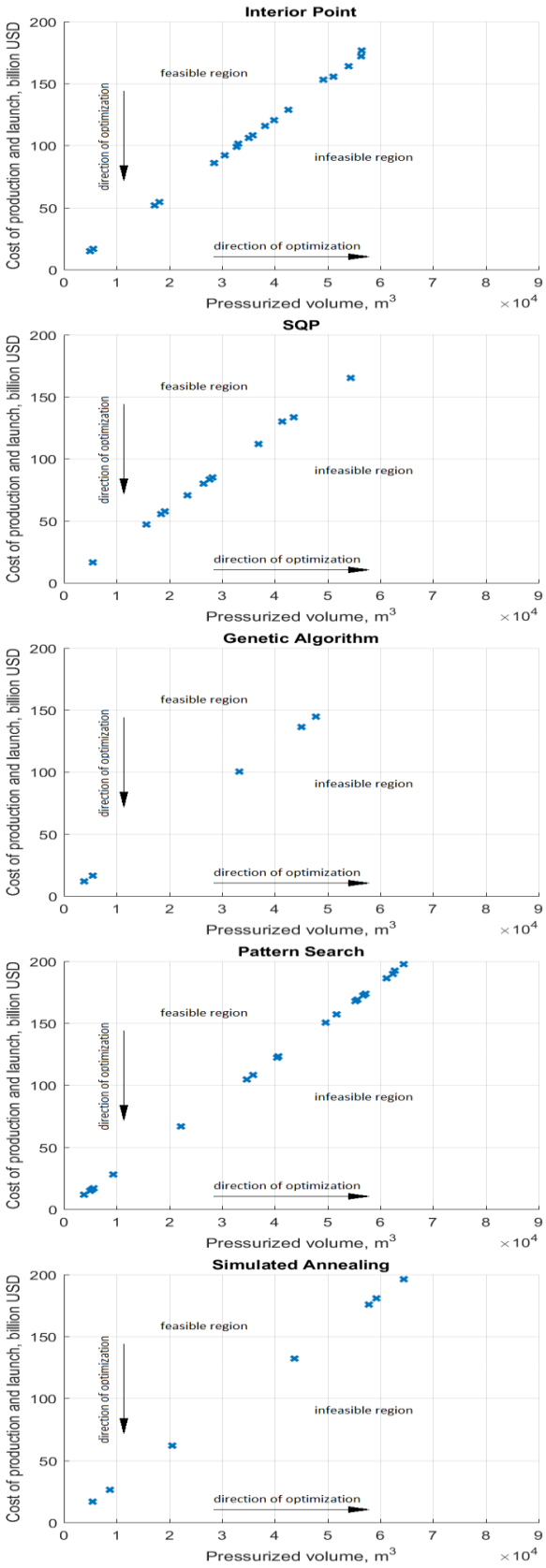

Figure 5: Pareto front of the first objective function J_1, with the considered search algorithms

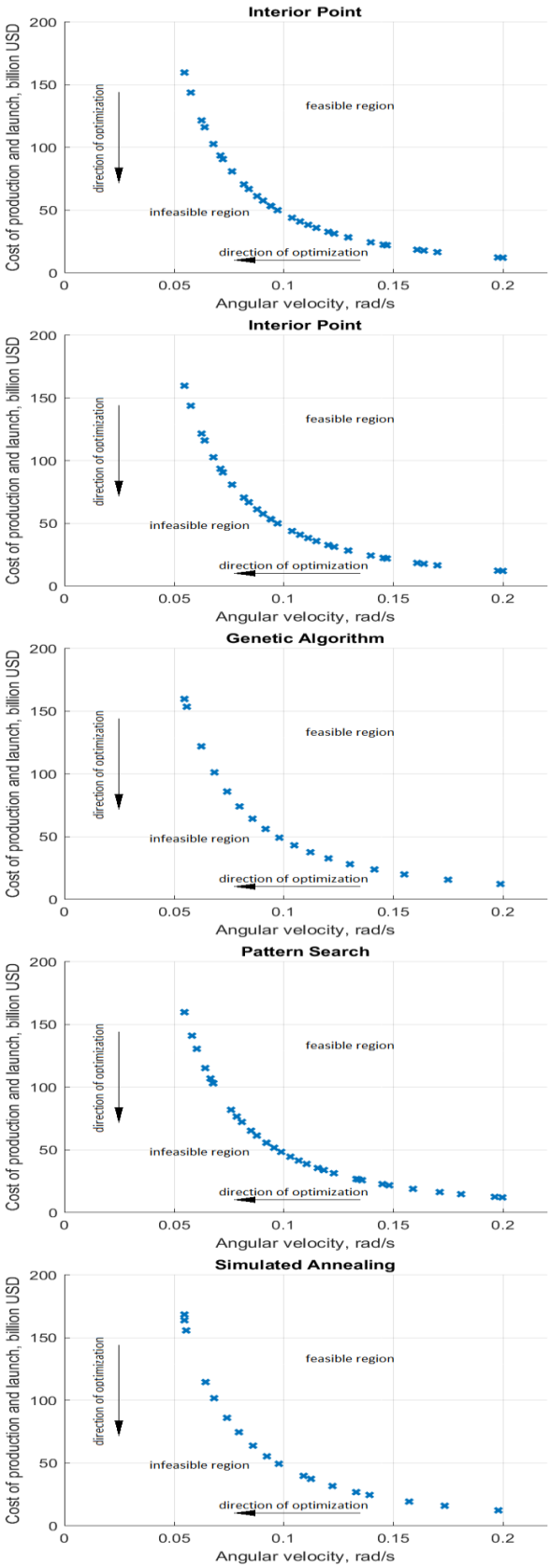

Figure 6: Pareto front of the second objective function J_2, with the considered search algorithms

5. Conclusions

A feasibility study on an artificial gravity system has been performed, taking into account the influence of a set of design variables on performance and cost. This preliminary work accounts for habitable modules. Costs of support systems, such as power generation, propulsion, attitude control, thermal control, are not yet considered.

The models employed for this work are very simple, yet they enable to account for the effect of several design choices on the manufacturing and deployment cost of the system, as well as on the expected performance. The use of a Multidisciplinary Design Optimization approach at this preliminary stage allows quantitatively evaluating different possible architectures of the system and informing the design decisions early in the development process. This way, it is possible to freeze the system architecture early, when radical design changes are not yet prohibitively costly.

However, our study determined how even considering the point of the Pareto front characterized by the smallest cost and volume, the resulting pressurized volume of the spacecraft is one order of magnitude greater than the largest spacecraft built to date, the International Space Station. The cost of production and launch of the Artificial Gravity System would be in excess of 10 billion USD, or more than 10% of the whole lifecycle cost of the ISS (which stands in the order of 100 billion USD [35]).

This very high cost is in fact the main reason why artificial gravity systems have not been usually considered as a feasible solution for long-term crewed space missions. Future developments in reusable launch vehicle will likely reduce launch related costs, thus helping to make artificial gravity systems affordable.

Acknowledgements

This work was supported by the PhD program at Politecnico di Torino. Additional acknowledgments to the Visiting Professor Program of Politecnico di Torino for the support to Dr. Laura Mainini.

References

1. *The Global Exploration Roadmap*, ISECG, January 2018.
2. W. Von Braun, *Crossing the Last Frontier*, Collier's, 22 Mar. 1952, pp. 24-30, 72-74
3. NASA image 9132079 https://images.nasa.gov/details-9132079
4. NASA Media Usage Guidelines, https://www.nasa.gov/multimedia/guidelines/index.html
5. C.H. Williams, L.A. Dudzinski, S.K. Borowski, A.J. Juhasz, *Realizing "2011: A Space Odyssey": Piloted Sperical Torus Nuclear Fusion Propulsion*, Proceedings 37th Joint Propulsion Conference and Exihibit cosponsored by the AIAA, SAE, AIChE, and ASME, Salt Lake City, Utah, July 2001
6. M.L. Holderman, *NAUTILUS-X, Multi-Mission Space Exploration Vehicle*, NASA internal presentation JSC/SSP, 2011
7. R. Zubrin, R. Wagner, *The Case for Mars: The Plan to Settle the Red Planet and Why We Must*, Free Press 0684827573/0684835509, 1997
8. A.M. Korzun, R.D. Braun, E.B. Wagner, T.R.F. Fulford-Jones, E.C. Deems, D.C. Judnick, J.E. Keese, *Mars Gravity Biosatellite: Engineering, science, and education*, Proceedings of 58th International Astronautical Congress, Hyderabad, India September 2007
9. T.W. Hall, *Artificial Gravity in theory and practice*, 46th International Conference on Environmental Systems, Vienna, Austria, July 2016
10. R.H. Byrd, J.C. Gilbert, J. Nocedal, *A Trust Region Method Based on Interior Point Techniques for Nonlinear Programming*, Mathematical Programming, 89(1):149–185, 2000
11. R.H. Byrd, M.E. Hribar, J. Nocedal, *An Interior Point Algorithm for Large-Scale Nonlinear Programming*, SIAM Journal on Optimization, 9(4):877–900, 1999
12. Waltz, R. A. , J. L. Morales, J. Nocedal, and D. Orban, *An interior algorithm for nonlinear optimization that combines line search and trust region steps*, Mathematical Programming, 107(3):391–408, 2006
13. F. A. Potra, S. J.Wright, Interior-point methods, Journal of Computational and Applied Mathematics, 124(1-2), 281-302, 2000.
14. M.H. Wright, Interior methods for constrained optimization, Acta Numerica, 1, 341–407, 1992.
15. M.C. Biggs, *Constrained Minimization Using Recursive Quadratic Programming*, Towards Global Optimization (L.C.W. Dixon and G.P. Szergo, eds.), North-Holland, pp 341–349, 1975
16. S.P. Han, *A Globally Convergent Method for Nonlinear Programming*, J. Optimization Theory and Applications, 22:297, 1977
17. M.J.D. Powell, *The Convergence of Variable Metric Methods for Nonlinearly Constrained Optimization Calculations*, Nonlinear Programming 3, (O.L. Mangasarian, R.R. Meyer and S.M. Robinson, eds.), Academic Press, 1978
18. M.J.D. Powell, *A Fast Algorithm for Nonlinearly Constrained Optimization Calculations*, Numerical Analysis, G.A.Watson ed., Lecture Notes in Mathematics, Springer Verlag, Vol. 630, 1978
19. V. Kungurtsev, M. Diehl, Sequential quadratic programming methods for parametric nonlinear

optimization, Computational Optimization and Applications, 59(3), 475-509, 2014.
20. L. Tsoukalas, R. Uhrig, *Fuzzy and Neural Approaches in Engineering*, Wiley, 1997
21. L. Lemarchand, A. Plantec, B. Pottier, S. Zanati, *An Object-Oriented Environment for Specification and Concurrent Execution of Genetic Algorithms*, 4(2):163-165, 1993
22. A. E. Eiben P. -E. Raue', Zs. Ruttkay, *Genetic Algorithms with Multi-parent Recombination*, Proceedings of the International Conference on Evolutionary Computation. The Third Conference on Parallel Problem Solving from Nature: pp.78-87. ISBN 3-540-58484-6, 1994
23. A. E. Eiben, J. E. Smith, Introduction to Evolutionary Computing, Springer, 2015.
24. A. Konak, D. W. Coit, A. E. Smith, Multi-objective optimization using genetic algorithms: A tutorial, Reliability Engineering and System Safety, 91(9), 2006.
25. V. Torczon, *On the Convergence of Pattern Search Algorithms*, SIAM Journal on Optimization 7(1):1-25, 1997
26. R. Hooke, T.A. Jeeves, "Direct search" solution of numerical and statistical problems, J. Assoc. Comput. Mach. 8(2):212-229, 1961
27. N. Findler, C. Lo, R. Lo, Pattern search for optimization, Mathematics and Computers in Simulation, 29(1), 1987.
28. Y. Sherif, B. A. Boice, Optimization by pattern search, European Journal of Operational Research, 78(3), 1994.
29. Torczon, On the convergence of pattern search algorithms, SIAM Journal on Optimization, 7(1), 1997.
30. D.S. Johnson, C.R. Aragon, L.A. McGeoch, C. Schevon, Optimization by Simulated Annealing: An Experimental Evaluation; Part I, Graph Partitioning, Operations Research 37(6):853-992, 1989
31. S. Kirkpatrick, C. D. Gelatt Jr, M. P. Vecchi, Optimization by simulated annealing, Science, 220(4598), 671–680, 1983.
32. K. Amine, Multiobjective simulated annealing: Principles and Algorithm Variants, Advances in Operations Research, 2019.
33. S. P. Brooks, B. J. T. Morgan, Optimization using simulated annealing, Journal of the Royal Statistical Society, 44(2), 241-257, 1995.
34. A. Dekkers, E. Aarts, Global optimization and simulated annealing, Mathematical Programming, 50(1-3), 367-393, 1991.
35. *How Much Does It Cost?*, European Space Agency (ESA), August 9, 2005 http://www.esa.int/Our_Activities/Human_Spaceflight/International_Space_Station/How_much_does_it_cost