POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

Optimal control allocation problem analysis and implementation for in-orbit activities

Laurea Magistrale in Space Engineering - Ingegneria Spaziale

Author: Emanuele Tagnin

Advisor: Prof. Pierluigi Di Lizia

Co-advisor: Gioacchino Scirè

Academic year: 2022-2023

1. Introduction

Control allocation studies how a control action shall be distributed among a set of redundant actuators, subject to hardware limitations on their produced output, called effort. The main feature of control allocation is to split the loop into two independent blocks. The low level control takes the action generated by the high level one, represented by a certain control logic, and distributes it among the actuators effort. This division simplifies the control definition, as it enables the user to separate the controller from the actuators architecture, thus giving more flexibility in the scheme design, as multiple iterations and corrections may change it. It also highlights the physical features of the actuators in exam, namely minimum and maximum outputs, as control logic may not manage them [14].

1.1. Work innovations

Spacecrafts for in-orbit applications are equipped with redundant actuators because they can cope with the potential failure of some of them (reliability) and perform accurately a wide span of different manoeuvres, while minimizing key cost functions (flexibility). These actions must be performed with low computational effort. However, present literature consists of individual papers that describe a specific strategy tailored to a theoretical mission [10, 15, 16] that focuses more on the cost minimization than CPU time. In addition, the literature regarding space control allocation is thinner and less organized with respect to the aeronautical field. The innovations that the presented work brings pertain to two different aspects. From a theoretical point of view, a more rigorous and plain organization of different allocating strategies is outlined, along with the clear selection of performance assessing parameters, which are accuracy (error between produced and required actions), cost minimization, CPU time and ability to cope with unattainable input actions. From a technical view, an overall transposition of simple, reliable, computationally efficient algorithms from aircraft attitude control [9, 11, 13] into space environment is conducted with a reformulation of all the involved quantities. Moreover a comparison between strategies is carried out in order to define a discerning tool that can help an AOCS designer to select the optimal method. Then, an allocating scheme is determined for a developing mission conducted with

the Company, that proves not only the wide span of manoeuvres the allocation can undergo with mild modifications that are completely decoupled from the controller, but how the high number DOFs optimum problem, that is not treated as rigorously as in aeronautics, that involves actuators limitations, is simplified into a robust and efficient element that completely disregards the controller and whose test on a real mission development gives a great and prompt response.

2. Control Allocation theory

Control allocation studies how the vector of required action $\boldsymbol{m} \in \mathbb{R}^n$, output of the high level control, can be mapped over a set of actuator DOFs, represented by $\boldsymbol{u} \in \mathbb{R}^m$. The two spaces where $\boldsymbol{m}$ and $\boldsymbol{u}$ lie, the moment space $\mathbb{R}^n$ and control space $\mathbb{R}^m$, are linked together through a function called effectiveness $B : \mathbb{R}^m \to \mathbb{R}^n$. In the analyzed framework, B is assumed to be linear and time-invariant. The first hypothesis states that the effectiveness can be seen as a set of gains, that defines how "effective" is a specific actuator in the action generation. The second aspect, instead, means that the actuators disposition does not change over time. Hence, B can be represented as an $n \times m$ matrix. The studied case will consider $m > n$, which implies that the linear system $\boldsymbol{m} = B\boldsymbol{u}$ has an infinite number of solutions. The problem can be stated as reported in Equation (1).

$$find\ \boldsymbol{u}\ s.t. \begin{cases} residual\ \boldsymbol{m} - B\boldsymbol{u} = 0 \\ p\ is\ minimum \\ \boldsymbol{u_{min}} \leq \boldsymbol{u} \leq \boldsymbol{u_{max}} \end{cases} \quad (1)$$

Where p may represent a key mission parameter to minimize, such as propellant consumption. $\boldsymbol{u_{min}}$ and $\boldsymbol{u_{max}}$ define the feasible enclosed subspace of the control space named Ω, admissible control set. The subspace is closed and bounded and represents a geometrical m-D polyhedron, envelope of feasible control DOFs. If a linear mapping is considered, it is possible to define the n-D subspace Φ which corresponds to Ω, called attainable moment set, which is the action envelope the system can generate through its actuator configuration. Being B linear time-invariant, also Φ is an enclosed and bounded subspace. In the next section, the studied methodologies are reported and evaluated in terms of accuracy, which is the ability to fulfill the equation $\boldsymbol{m} - B\boldsymbol{u} = 0$, optimality (ability to minimize p) and computational time, for attitude control application, thus $\boldsymbol{m} \in \mathbb{R}^3$ is the high level control required torque.

2.1. Allocation methods description

The investigated algorithms can be divided into methodologies that compute the solution focusing on the accuracy and indirectly influencing the cost minimization, and vice versa. Belonging to the first group, the Generalized Inverse (GI) [4] involves the generation of a matrix P which satisfies $BP = eye(n)$, identity matrix. P forms a basis for an n-D subspace of $\mathbb{R}^m$, called P_s. Since this basis contains the origin, and if it is contained within Ω, these two elements are guaranteed to intersect. A simple and effective way to specify the Generalized Inverse is through the implementation of the pseudo-inverse, which computes $\boldsymbol{u}$ by minimizing its l_2 norm. The solution found takes the form $P = B^T(BB^T)^{-1}$. If the algorithm is able to rapidly compute a solution, the pseudo-inverse does not include directly cost minimization and feasible ranges. Therefore if a GI would solve the allocation with unfeasible output, it is required a post process to enforce the actuators ranges on the solution. To cope with these issue the Cascade Generalized Inverse (CGI) [4] exploits the intrinsic redundancy of the system, computing sequential allocating problems. The CGI calculates the first solution through a GI, then if u_i exceeds the valid range, it is brought to saturation, and the residual $\boldsymbol{m} - B\boldsymbol{u}$ is computed. This residual is fed again in the allocation problem, while omitting the i^{th} column of saturated B, and the process is repeated as long as $m - n \geq 0$. To take into consideration feasible ranges, Direct Allocation (DA) [4] is implemented. It is able to compute an optimal solution, which indirectly influences the cost minimization. DA method considers the direction of the required torque in Φ, it projects the unitary vector onto the control space in Ω and scales the magnitude to match $||\boldsymbol{m}||$ or to intersect the boundary $\partial\Omega$. To apply the method Φ must contain the origin, be convex and points on $\partial\Phi$ correspond to unique points on $\partial\Omega$. Instead of inefficiently computing the whole envelope Φ, it is possible to exploit the

geometry of the problem. Considering $n = 3$, $\boldsymbol{m}$ direction intersects $\partial\Phi$ on a plain. The same applies for $\boldsymbol{u}$ and $\partial\Omega$, where each plain is defined by $m-2$ fixed variables and 2 varying. It is possible to search through each couple and detect the one that generates in Φ the plain of intersection with $\boldsymbol{m}$. Once the correct direction is found, it is just necessary to scale the magnitude to meet $||\boldsymbol{m}||$ or, if unattainable, $\partial\Phi$. The method is able to compute the exact allocating solution with feasibility constraints, but is characterized by tight assumptions. The aforementioned hypothesis can be loosen for a general case in expense of accuracy loss, as the required control direction is going to be approximated.
The second family of methods selects as a driver the minimization of the scalar function p (e.g. fuel consumption) while taking into consideration the feasible inequality constraints and the allocating equality ones. The considered algorithm, the Linear Programming (LP) [3], solves the following problem:

$$minimize_{\boldsymbol{u}} \ \boldsymbol{f}^T\boldsymbol{u} \ s.t \begin{cases} \boldsymbol{u_{min}} \leq \boldsymbol{u} \leq \boldsymbol{u}_{max} \\ B\boldsymbol{u} = \boldsymbol{m} \end{cases} \quad (2)$$

In this case, p is represented by a weighted sum of actuators efforts. For example, if RCT are considered, the thrust is linked to the propellant mass consumption. The problem is solved through the `Matlab` function `linprog.m`. The main advantage of this approach is that an optimal solution is found while considering multiple constraints on $\boldsymbol{u}$. On the other hand, the method is computationally extremely slow, and the output may be imprecise when dealing with unattainable moments. Once the two families have been analyzed, it is possible to define a method such that both aspects, accuracy and optimization, are considered at the same time. The Weighted Least Square algorithm [11], in fact, merges the two drivers in a unique cost definition as:

$$\begin{aligned} \boldsymbol{u} = arg\ min_{\boldsymbol{u_{min}} \leq \boldsymbol{u} \leq \boldsymbol{u_{max}}} : \\ \gamma||W_a(B\boldsymbol{u} - \boldsymbol{m})||^2 + \\ +||W_p(\boldsymbol{u} - \boldsymbol{u_p})||^2 \end{aligned} \quad (3)$$

The above problem statement declares that the solution will be a trade-off between the two 2-norm cost functions. The first part represents the accuracy of the solution. The second part, instead, represents the cost minimization, stated as the difference between the control command set $\boldsymbol{u}$ and the desired one $\boldsymbol{u_p} = \boldsymbol{0}$. Concerning space applications, the actuators effort can be generally linked to the parameter that is necessary to minimize, such as power or propellant consumption. By reducing the overall intensity of the variables in $\boldsymbol{u}$, it will eventually reduce the cost. The combination of the two contributions shows the great utility of the method: the freedom the user has to select, depending on the mission requirements, the characterization of the allocator. Each quantity in Equation (3) is then multiplied by a weighting factor, respectively $W_a \in \mathbb{R}^{n\times n}$ and $W_p \in \mathbb{R}^{m\times m}$. Each states how a single variable is to be prioritized with respect to the other. Generally, these matrices are diagonal and have the same values if no prioritization is required. γ is an additional constant that highlights the accuracy total cost with respect the other one, as the output may approach the null, but feasible solution. To solve the WLS, an Active Set Algorithm is used [11].

2.2. Strategies comparison

All the described methods have their own points of strength and weakness. It is necessary to highlight them through an actual implementation. The simulation consists of an attitude tracking problem for a spacecraft equipped with the set of redundant actuator at disposal. The simulation is ran for each algorithm, considering different working orbits and, as evaluating parameters, the cost optimization, accuracy, computational time and the behaviour when dealing with unattainable moments. The methodologies are applied onto RW and RCT. For the sake of brevity, only the reaction wheels case is here illustrated, specifically in a circular SSO Dawn-Dusk orbit at 500 km of altitude. In Table 1, the optimization of each algorithm, represented by the final storage is reported. It is clear to see that the DA and WLS are more capable of minimizing the cost. Concerning the accuracy, the residual $||\boldsymbol{m} - B\boldsymbol{u}||$ is practically 0 for each method, thus only attainable torques are encountered. The last quantity represents the ratio between the methods computational time and the maximum of these (WLS for 48.92s for 50000s of simulating time). Therefore, it is possible to state that GI and CGI calculate a fast rough solution, while the other two are able to

refine it in expense of more computational cost. LP was implemented instead on the RCT case, and it emerged that the solution, even though optimized, required an extremely high computational cost.

	GI	CGI	DA	WLS
h_{t_f} [Nms]	20.79	20.79	20.57	19.17
m_{res} [Nm]	∼0.00	∼0.00	∼0.00	∼0.00
t_{com} [-]	0.90	0.90	0.99	1.00

Table 1: Performance evaluation for SSO case

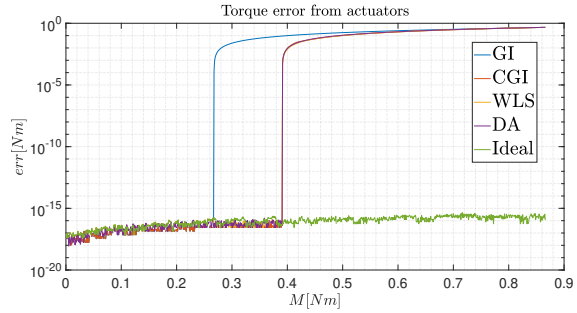

Figure 1: Residual as the required torque grows

In conclusion, in Figure 1 the ability to cope with unattainable moments is shown. The residual is plotted with respect an increasing torque. GI performs the worse, while the other are more capable of delaying the failure condition with respect the ideal, unbounded case.

3. Mission simulation

In this section a real developing in-orbit servicing mission is presented. The application is used to demonstrate the effectiveness of the control allocation scheme, its robustness and flexibility to cope with different types of manoeuvres. Moreover it shows how to navigate in the allocator selection depending on the mission requirements. The spacecraft *Module A (MA)* shall carry the client *Module B (MB)* to target position. The two, then, separate. After the detachment, a movable thrust vector, mounted on *MB*, is activated and its effects have to be counter acted by a pose control system. The overall spacecraft, excluding the thrust appendage on *MB* is modelled as shown in Figure 2.
Both parts are equipped with the same sets of redundant RW ($m = 4$) and RCT ($m = 24$).

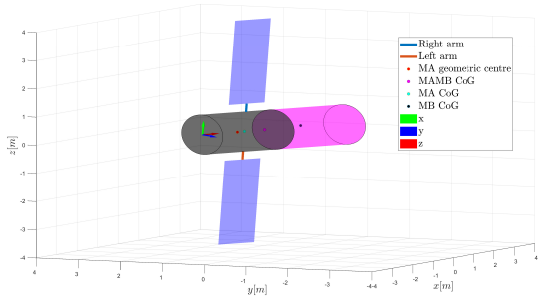

Figure 2: *MAMB* attached spacecraft model

In addition, the actuators effort is assumed to be continuous and WLS strategy is implemented due to its inherent flexibility required as the spacecraft perform different operations. The working orbit is an equatorial GEO. For computational efficiency, the control action is generated through a PID controller, since it is completely decoupled from the actuators architecture. Concerning environmental elements, SRP and gravity gradient are implemented. Additionally, the state vector for Phase IV is represented by the pose, while for the other by the angular velocity.

3.1. Phase I - *MAMB* phasing

The satellite shall bring the client spacecraft to a target orbit. This operation translates into a tracking problem by means of reaction wheels effort, to save propellant. The characteristic inertia is rather consistent, so the control effort is distributed among *MA* and *MB* wheels. To keep *MB* storage as low as possible to avoid pre-loading before the operations in Phase II, a weighting vector is defined. It ensures a greater percentage of the total required torque to be assigned to *MA*. However, if the wheels on *MA* will reach torque or storage saturation, also *MB* will fully contribute to the control effort definition. In Figure 3 it is evident the different loading between the two sets of RW.

3.2. Phase II - *MB* desaturation

Before the separation, it is necessary to empty the momentum storage accumulated during the previous phase in *MB*. First, the control authority is shifted to *MA*, while *MB* takes, as an input, a virtual torque that is necessary to allow the wheels to deplete their stored energy. Of course it is clear to understand that, if the re-

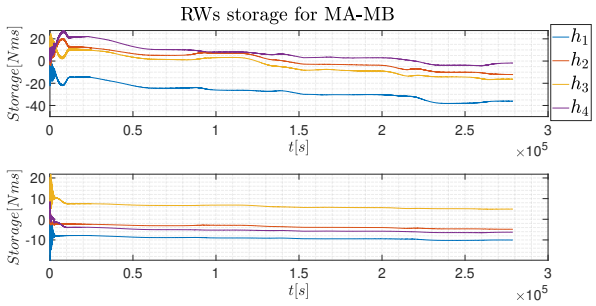

Figure 3: *MA* and *MB* reaction wheel storage

action wheels are damping out the storage, they are transmitting an undesired torque that has to be counter acted by the other available actuators. As the manoeuvre occurs, the control authority is shifted from the *MA* reaction wheels to *MB* RCT, linearly over the course of 100 s. This transfer is executed because the thrusters are able to provide a higher net torque compared to the RW set. Moreover, it is the first procedure to decouple the two satellites in terms of control authority and distribution. At this stage and in the next ones, the spacecraft is required to point inertially a fixed orientation, with the target attitude matching the last targeted orientation in the previous phase.

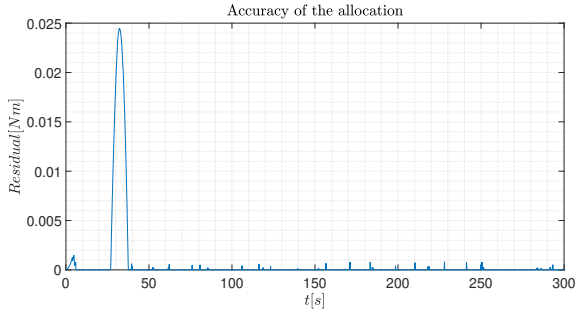

Figure 4: Required-provided torque residual

In Figure 4, $||\boldsymbol{m} - B\boldsymbol{u}||$ is shown, demonstrating the necessity of the authority shift, as the RW are not able to cope with the required torque.

3.3. Phase III - *MAMB* separation

Phase III consists in the separation between the two spacecrafts. Once the storage in *MB* is emptied, it detaches from *MA*, following a straight trajectory parallel to the z axis. This manoeuvre is modelled as an instantaneous division between *MA* and *MB*, in which an impulsive variation of the mass and inertia properties is introduced. Here, the attention is brought onto the first spacecraft due to possible impact between solar wings and *MB*. To assess the performance, an impulsive perturbing torque is applied to *MA* to simulate the separation load, instantaneously at 50s during the simulation, drawn randomly from an interval of $\pm 50N$ in the three directions. Regarding control, the high-level block is disabled for a brief period as the separation occurs. As Figure 5 reports, the impulse load introduction is clearly visible at $t = 50s$, where the pointing error is subject to an almost discontinuous variation.

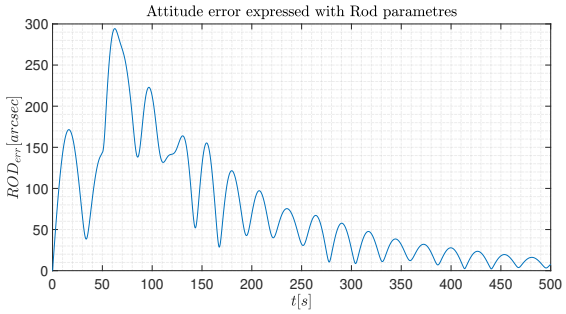

Figure 5: *MA* attitude error

3.4. Phase IV - *MB* thrust vectoring compensation

The client spacecraft *MB* has to execute a set of operations that require a thrust vectoring action. The resultant effects on the spacecraft are a reaction torque and force applied to *MB* centre of gravity. In order to maintain fixed position and attitude, a pose control system is implemented, thus $n = 6$ and $B_{RCT} \in \mathbb{R}^{6\times 24}$. At low level control, the required action is split into torque and force. The first quantity is distributed among the RW and, if unattainable moment is detected, RCT will receive the residual. Concerning the control force, it is managed entirely by the thrusters. In Figure 6 the daisy chain approach is represented: the RCT are active when the load is unbearable for the RW. While in Figure 7 the CoG drift is illustrated, which is in reasonable range, thus proving that the allocation can cope even with pose control with simple modification in the allocator.

3.5. Robustness analysis

The next step is to assess the robustness of the allocator when uncertainties and noise are present during the simulation. Therefore, dis-

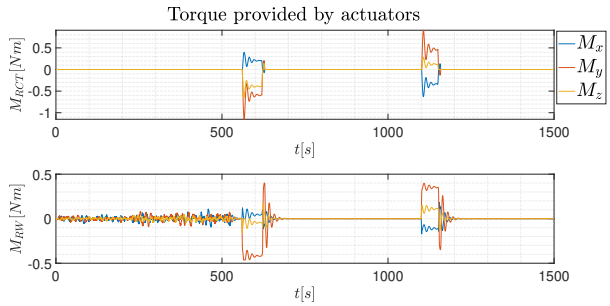

Figure 6: RW and RCT provided torque

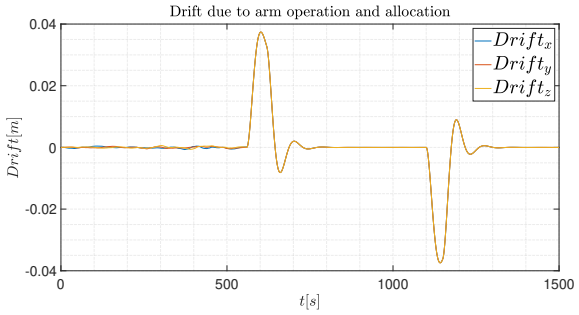

Figure 7: CoG drift from nominal position

turbances are introduced in the angular velocity measurements and hardware parameters, such as CoG position, are scattered. A Monte Carlo simulation is performed to study the control allocation capabilities in this frame work. Clearly, it is expected a general performance decrement in terms of fuel consumption and pointing accuracy. From the simulation, scalar key elements are plotted with a CDF, while arrays are studied considering the envelope, defined by the worst case trend, the best one and the average evolution. These quantities are analyzed to assess AOCS mission requirements compliance. In Figure 8 an example of CDF for the consumed propellant in Phase IV is shown. In Figure 9, the envelope of the accuracy in Phase II is presented.

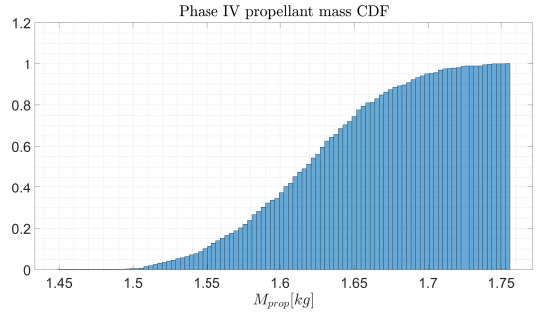

Figure 8: CDF for the consumed propellant mass in Phase IV

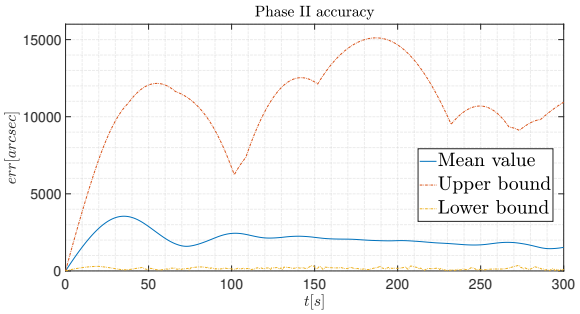

Figure 9: Accuracy in Phase II

4. Conclusions

Through the investigation, it has become apparent that there is no one "perfect" strategy among those proposed, as each has its own features that prioritize a specific aspect of the allocation. The optimal solution depends on the mission scenario, architecture, environment and requirements. Straightforward solutions such has the GI compute a fast, rough solution, while DA spends more time to refine the output. The presented work did not limit in the simple description and evaluation of the different methods, but also showed that a clear re-definition of the allocation statement generated a powerful tool that extremely simplifies the problem resolution with a flexible and robust characterization. This is demonstrated in the application of the control allocation in real mission scenario. In Ph. I computational efficiency for long time span simulation is demonstrated, in Ph. II the flexibility of the allocation, in Ph. III the ability to manage sharp state variation and in Ph. IV the allocation for pose control. In the design, it was necessary to apply mild modification to perform multiple and different operations, which is the innovative and strong aspect of the work: the complete definition of a simple block, decoupled from the control logic selection, that can solve a multi-dimensional optimum problem with feasible ranges in a prompt and effective way. Naturally, the next steps involve improving control allocation performance, discovering new customized solutions, and incorporating time-varying effectiveness matrix into the analysis. To sum up, the proposed inquiry establishes the initial phase for the AOCS subsystem design and constructs a useful instrument to effectively handle and enhance the control allocation in the closed loop control system.

5. Acknowledgements

I express my gratitude to everyone who supported and guided me through my academic journey. Without your enduring patience and support through my highs and lows, this work would not have been possible. I would like to extend special thanks to my family, friends, and my advisor P. Di Lizia, whose guidance was instrumental in the development of this thesis, G. Scirè, who not only answered my questions and addressed my doubts on a weekly basis for several months, but introduced me in the company and to a lot of intelligent, caring people.

References

[1] Franco Bernelli. Slides on attitude definition. *Spacecraft Attitude Dynamics Master Course*, 2020.

[2] Ake Björck. Numerical methods for least squares problems. *Choice Reviews Online*, 34(03):34–1602, 11 1996.

[3] Marc Bodson. Evaluation of Optimization Methods for Control Allocation. *Journal of Guidance Control and Dynamics*, 25(4):703–711, 7 2002.

[4] Kenneth A. Bordingnon and Wayne Durham. Closed-form solutions to constrained control allocation problem. *Journal of Guidance Control and Dynamics*, 18(5):1000–1007, 9 1995.

[5] Pierluigi Di Lizia. Notes on Covariance Matrix. *Spacecraft Guidance and Navigation Master Course*, 2021.

[6] Wayne Durham. Attainable moments for the constrained control allocation problem. *Journal of Guidance Control and Dynamics*, 17(6):1371–1373, 11 1994.

[7] Wayne Durham. Constrained control allocation - Three-moment problem. *Journal of Guidance Control and Dynamics*, 17(2):330–336, 3 1994.

[8] Wayne Durham. Computationally Efficient Control Allocation. *Journal of Guidance Control and Dynamics*, 24(3):519–524, 5 2001.

[9] W.C. Durham. Constrained Control Allocation. *Journal of Guidance, Control, and Dynamics,*, Vol. 16, No. 4:717–725, 8 1993.

[10] Chang Haitao, Panfeng Huang, Yizhai Zhang, Zhongjie Meng, and Zhengxiong Liu. Distributed Control Allocation for Spacecraft Attitude Takeover Control via Cellular Space Robot. *Journal of Guidance Control and Dynamics*, 41(11):2499–2506, 9 2018.

[11] Ola Härkegård. Efficient active set algorithms for solving constrained least squares problems in aircraft control allocation. *Conference on Decision and Control*, 12 2002.

[12] Ola Härkegård. Backstepping and control allocation with applications to flight control. *Digitala Vetenskapliga Arkivet*, 1 2003.

[13] Ola Härkegård. Dynamic Control Allocation Using Constrained Quadratic Programming. *Journal of Guidance Control and Dynamics*, 27(6):1028–1034, 11 2004.

[14] Ola Härkegård and S. Torkel Glad. Resolving actuator redundancy—optimal control vs. control allocation. *Automatica*, 41(1):137–144, 1 2005.

[15] Xiaoyu Lang and Anton H. J. De Ruiter. Distributed optimal control allocation for 6-dof spacecraft with redundant thrusters. *Aerospace Science and Technology*, 118:106971, 11 2021.

[16] Weiqu Liu, Yunhai Geng, Baolin Wu, and James D. Biggs. Distributed Constrained Control Allocation for Cellularized Spacecraft Attitude Control System. *Journal of Guidance Control and Dynamics*, 45(2):385–393, 2 2022.

[17] John A. M. Petersen and Marc Bodson. Constrained quadratic programming techniques for control allocation. *IEEE Transactions on Control Systems and Technology*, 14(1):91–98, 1 2006.