Modified Cascading Generalized Inverse Control Allocation

Michael J. Acheson * and Irene Gregory†
NASA Langley Research Center, Hampton, VA, 23681

The current aviation revolution towards electric propulsion aircraft (e.g., electric vertical takeoff-and-landing) brings unique control challenges. These vehicles are typically over-actuated (more effectors than desired control outcomes), may require control strategies for the three phases of flight (hover, transition and cruise), and currently have limited electric power availability. These vehicle challenges bring the need for optimal control allocation to the forefront of research. A leading control allocation algorithm, used in current flight vehicles, is the Cascading Generalized Inverse (CGI). Unfortunately, the Cascading Generalized Inverse algorithm is unable to achieve some desired outcomes, it intermittently provides non-optimal allocations, and it may fail to preserve moment direction near maximal achievable outcomes. In this research, the shortcomings of the Cascading Generalized Inverse algorithm are addressed by augmenting the algorithm with Scalar Difference Quadratic unsaturation identification and location at each iteration. Rigorous theory is shown that the Modified Cascading Generalized Inverse performs better at obtaining optimal allocations for all attainable outcomes. Numerical case studies for over-actuated vehicles demonstrate resolution to the aforementioned deficiencies.

I. Introduction

The introduction of electric and hybrid-electric propulsion has brought on a genuine revolution to the entire aviation industry. Small electric propulsion vehicles (e.g., quadrotors, octorotors) are common place in the hobby community and university research, and are making their way into commercial activities (e.g., delivery services). Larger vehicles (e.g., four to six passengers) are foundational to the business case for advanced air mobility (AAM) and urban air mobility (UAM) segments of the industry. A wide array of commercial providers are racing towards certification to gain market share for the UAM market. Still larger vehicles are in the design build process (Airbus, Boeing, Embraer). Major transport aircraft manufacturers are designing and building short to long haul aircraft using batteries, hybrid (turbine electric), and even sustainable fuel-electric propulsion (e.g., hydrogen generation, and CO_2 capture to generate sustainable jet fuel). What is common to this wide range of electrified vehicles is a need to blend control of classical

*NASA Senior Researcher, Dynamic Systems and Control Branch, NASA Langley Research Center, Michael.J.Acheson@nasa.gov, AIAA member
†NASA Senior Technologist for Advanced Control Theory and Application, Dynamics Systems and Control Branch, NASA Langley Research Center, Irene.M.Gregory@nasa.gov, AIAA Fellow

aerodynamic effectors (e.g., rudder, ailerons) with electrical propulsion/control effectors (e.g., lifting rotors, pusher propeller, ducted fans). These effectors must be blended throughout all flight phases (hover or takeoff/landing, transition, and cruise). Moreover, whether due to electrical power limitations or due to a desire to substantially improve efficiency, optimal allocations of the aerodynamic and propulsor effectors is sought. These factors bring the need for research of control allocation to the forefront.

The rationale for use of control allocation is multi-faceted including: enlarging the set of available control outcomes, increasing aircraft performance [1], achieving control objectives while simultaneously optimizing secondary objectives [2] (e.g., minimal drag or minimal control), incorporating frequency apportioned control allocation [3, 4], incorporating structural load feedback [3, 5], and enhancing safety for control effector failures.

This research focuses on linear generalized inverse algorithms where initial work used a single generalized inverse providing optimal (minimal weighted quadratic norm) and feasible (i.e., attainable control) allocations over a strict subset of the attainable moments (also known as the desired outcomes in literature). Early work [6] and recent research [7, 8] detail metrics which quantify the moment subset for which a single generalized inverse yields feasible solutions. This prompted efforts to computationally determine the generalized inverse which maximizes the attainable subset [6, 8]. Other early algorithms used multiple generalized inverses to enlarge the attainable moments strict subset. Daisy-chaining [9] ganged controls together in two groups and used the second group only if the first group saturated. Redistributed generalized inverse algorithms utilize a sequence of generalized inverse iterations to enlarge the set of attainable moments. The Cascading Generalized Inverse (CGI) [10] is a preeminent iterative algorithm which uses a monotonic decreasing set of unsaturated control effectors to produce a cascade of generalized inverses. Limiting the CGI algorithm to one saturated effector per iteration ensures continuity of allocations which is essential for real-world physical systems [11]. The CGI algorithm fails to attain all desired outcomes, and does not always obtain the optimal (weighted quadratic norm) control allocation [3, 10]. In [12], saturating one effector per iteration yields the exact desired moment for those iterations where the number of unsaturated effectors is greater than or equal to the number of desired moments. For nearly maximal attainable moments, the CGI algorithm has fewer unsaturated effectors than moment components and cannot preserve moment direction for the desired moment [3].

A variety of algorithms augment the generalized inverse solution, which yields the exact desired moment, with a null space vector to achieve feasible allocations in the allowable control set (ACS). An early minimal control algorithm [13] searched all null space boundary objects (e.g., vertices, lines faces) for the one nearest the unconstrained pseudo inverse solution, but the large number of objects and numerical issues made implementation impractical. Recently, the Pseudo Inverse along the Null Space (PAN) algorithm [14, 15] augments the unconstrained minimal allocation by analytically iterating (until a feasible solution is found) for saturated effectors that minimize the magnitude of the null space component. This method was extended to the generalized weighted inverse and to allow for one-sided position limits [16]. A drawback is that PAN is unable to find optimal allocation for an infeasible solution case [16]. Moreover, [14–16] state that PAN obtains optimal allocations by iterating null space objects until a feasible allocation is found, but this research shows multiple null space objects might yield feasible allocations and therefore feasibility does not imply optimal allocation.

Closely related to CGI are the Prediction Method (PM) [17], the Affine Generalized Inverse (AGI) [18], and the Sequential Null Space Generalized Inverse (SNSGI) algorithms [19]. The PM uses a single weighted unconstrained generalized inverse augmented with a sequence of null space components analytically generated from a partition of the null space basis matrix for saturated effectors [17] similar to PAN. However, unlike PAN the PM solves for each piecewise linear and continuous allocation segment for all moments along the moment direction until the desired moment is obtained or the maximal boundary moment is achieved. The PM algorithm inherently generates the null space objects using a monotonically increasing set of saturated effectors and it achieves optimal allocation if the null space object is nearest the unconstrained generalized inverse component [17]. The AGI algorithm uses a sequence of generalized inverses formed from unsaturated effectors (equivalent to those in CGI) and a sequence of null space components generated from the monotonically increasing sequence of saturated effectors to solve for the piecewise linear and continuous allocations for all moments along the desired moment direction. The AGI was shown to yield equivalent solutions to the PM for identical sequences of effector sets for iterations when the number of unsaturated effectors is at least the number of moment components [18]. Similarly, the PM was proven to yield identical allocations to CGI for the same equivalent sets and limits on number of saturated effectors [20]. Therefore by transitivity, PM, AGI and CGI solutions yield equivalent allocations. Effector unsaturation or allowing a previously saturated effector to unsaturate in the current iteration, was identified as the cause of the PM's inability to obtain weighted optimal allocation solutions for some attainable moments [19]. The SNSGI algorithm incorporates the Scalar Difference Quadratic (SDQ) into the PM enabling effector unsaturation which aids SNSGI to achieve optimal allocations for all attainable moments [19, 20].

The focus of this research proposes the Modified Cascading Generalized Inverse (MCGI) which incorporates SDQ into the CGI algorithm to achieve optimal allocation solutions for all attainable moments. In this work, the use of affine generalized inverse subsets provides graphical insight into the three known issues of CGI and how MCGI resolves them. Moreover, the Appendix of this paper rigorously proves that MCGI achieves optimal allocations for all moments along the moment direction, up to and including the desired moment or maximal boundary moment.

II. Background

For systems with redundant controls (i.e., more effectors than outcomes), control allocation seeks the optimal control $\vec{u}$ to the under-determined system $B\vec{u} = \vec{m}_{des}$ for a given control effectiveness matrix $B \in \Re^{n \times m}$ and desired outcome $\vec{m}_{des}$ (e.g., accelerations, forces or moments) subject to position ($\vec{u}_{lwr}$ and $\vec{u}_{upr}$) and rate limitations ($\vec{r}_{lwr}$ and $\vec{r}_{upr}$).

The position limits define the Allowable Control Set (ACS or Ω) and thus the Attainable Moment Set (AMS or Φ):

$$\Omega = \left\{\vec{u} \in \Re^m \mid \vec{u}_{lwr_i} \leq \vec{u}_i \leq \vec{u}_{upr_i}, \forall i \in \{1, \ldots, m\}\right\} \tag{1}$$

$$\Phi = \{\vec{m} \in \Re^n \mid B\vec{u} = \vec{m}, \vec{u} \in \Omega\} \tag{2}$$

The optimal (minimal weighted quadratic norm) allocation problem for a given $\vec{m}_{des}$ is:

$$\vec{u}_{opt} = \underset{\vec{u}}{\operatorname{argmin}} \left\{\vec{u}^T W_{gi} \vec{u} \mid \vec{m}_{des} = B\vec{u},\ \vec{u} \in \Omega\right\} \tag{3}$$

for a positive (semi) definite weighting matrix $W_{gi} > 0$ ($W_{gi} \geq 0$). Use of $W_{gi} = I$ is the minimal control problem and the solutions to Eq. (3) for $W_{gi} > 0$ and the minimal control problem are:

$$P_{gi} = W_{gi}^{-1} B^T \left(B W_{gi}^{-1} B^T\right)^{-1} \tag{4}$$

$$P_{min} = B^T \left(B B^T\right)^{-1} \tag{5}$$

Given a linearized control effectiveness matrix $B \in \Re^{n\times m}$, then from linear algebra $\Re^m$ can be divided into two orthogonal subspaces such that $\Re^m = \Re^{m\parallel} + \Re^{m\perp}$ where $\Re^{m\parallel}$ is the subspace spanned by the columns of the Moore-Penrose generalized inverse P_{min} and $\Re^{m\perp} = \mathcal{N}(B)$. The allocation $\vec{u}^{\parallel} = P_{min}\vec{m}_{des} \in \Re^{m\parallel}$ is the smallest that yields $\vec{m}_{des}$, but $\vec{u}^{\parallel} \notin \Omega$, $\forall \vec{m}_{des} \in \Phi$. As the subspaces are orthogonal, we have $\left\|\vec{u}_{opt}\right\|_2^2 = \left\|\vec{u}^{\parallel}\right\|_2^2 + \|\vec{u}^{\perp}\|_2^2$ and therefore to minimize $\vec{u}_{opt}$ such that $B\vec{u}_{opt} = \vec{m}_{des}$, it is necessary and sufficient to minimize $\vec{u}^{\perp}$ such that $\vec{u}_{opt} = \left(\vec{u}^{\parallel} + \vec{u}^{\perp}\right) \in \Omega$.

For iterative generalized inverse algorithms, we define the i^{th} iteration set notation of unsaturated effector indices S_1^i, saturated effector indices S_2^i and the saturated effector position limit vector $\vec{s}_2^i$ as:

$$S_1^i = \{j \in 1, 2, \ldots, m \mid \vec{u}_{lwr}(j) < \vec{u}_{opt}(j) < \vec{u}_{upr}(j)\} \tag{6}$$

$$S_2^i = \{j \in 1, 2, \ldots, m \mid \vec{u}_{opt}(j) \in \{\vec{u}_{lwr}(j), \vec{u}_{upr}(j)\}\} \tag{7}$$

$$\vec{s}_2^i = \left[\ldots, \vec{u}_{opt}(j), \ldots\right]^T \forall j \in S_2^i \tag{8}$$

where $S_1^i \cup S_2^i = \{1, 2, \ldots, m\}$ and $S_1^i \cap S_2^i = \emptyset$. These sets reorder B and the null space of B or $\hat{B} = \mathcal{N}(B)$ as:

$$B = \begin{bmatrix} B_1^i & B_2^i \end{bmatrix}, \quad \hat{B}^T = \begin{bmatrix} \hat{B}_1^i \\ \hat{B}_2^i \end{bmatrix}^T \tag{9}$$

As the goal is to incorporate effector unsaturation into CGI, an overview of the PM and SDQ is warranted [17, 19],

[20]. Given a desired moment $\vec{m}_{des}$ and effector position limits, the PM starts at the origin ($\vec{0} \in \Phi$), proceeds along the moment direction $\hat{m}_{des} = \vec{m}_{des}/\|\vec{m}_{des}\|_2$ iteratively saturating one effector at a time resulting in piece-wise linear and continuous allocation segments until $\vec{m}_{des}$ or the maximal moment along $\hat{m}_{des}$ on the AMS boundary (i.e., $\delta(\Phi)$) is obtained. The unit vector gain a_1 at the corner of the first piecewise linear and continuous segment is:

$$a_1 = \underset{k}{\text{argmin}}\{k_j |\, j \in \{1, 2, \ldots, m\}\} \text{ where } k_j = \begin{cases} \vec{u}_{upr}(j)/\hat{u}_{des}(j), & \hat{u}_{des}(j) > 0 \\ \vec{u}_{lwr}(j)/\hat{u}_{des}(j), & \hat{u}_{des}(j) < 0 \end{cases} \tag{10}$$

$$\text{where} \quad \hat{u}_{des} = P_{min}\vec{m}_{des}/\|P_{min}\vec{m}_{des}\|_2 \tag{11}$$

Generalized inverse allocations are optimal if in the ACS, so the first segment's optimal allocation is:

$$\vec{u}_a = a\hat{u}_{des} = P_{min}\vec{m}_a \in \Omega, \quad \text{for } a \in [0, a_1] \tag{12}$$

The effector index j, which minimized Eq. (10), determines the effector sets $S_2^1 = \{j\}$, $S_1^1 = \{1, 2, \ldots, m\} \setminus \{j\}$ and

$$\vec{s}_2^{\,1} = \begin{cases} \vec{u}_{upr}(j), & \hat{u}_{des}(j) > 0 \\ \vec{u}_{lwr}(j), & \hat{u}_{des}(j) < 0 \end{cases} \tag{13}$$

The baseline Prediction Method iterates for $1 < i \le (m - n)$. For $i > 1$ we define the null space vectors (dropping the i superscript for $\hat{B}_2$):

$$\vec{c}_0^{\,i} = \hat{B}\hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \vec{s}_2^{\,i} \in \mathcal{N}(B) \tag{14}$$

$$\hat{c}_1^{\,i} = \hat{B}\hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \hat{u}_2^{\,i} \in \mathcal{N}(B) \tag{15}$$

where the vector $\hat{u}_2^i$ consists of only the components of $\hat{u}_{des} = P_{min}\vec{m}_{des}$ corresponding with the saturated effector indices in S_2^i. The control unit vector gain a_{i+1}, which saturates the next effector at the subsequent segment corner is:

$$a_{i+1} = \underset{k}{\text{argmin}}\{k_j |\, j \in S_1^i\} \text{ where } k_j = \begin{cases} \left(\frac{\vec{u}_{upr}(j) - \vec{c}_0^{\,i}(j)}{\hat{u}_{des}(j) - \hat{c}_1^{\,i}(j)}\right), & \left(\hat{u}_{des}(j) - \hat{c}_1^{\,i}(j)\right) > 0 \\ \left(\frac{\vec{u}_{lwr}(j) - \vec{c}_0^{\,i}(j)}{\hat{u}_{des}(j) - \hat{c}_1^{\,i}(j)}\right), & \left(\hat{u}_{des}(j) - \hat{c}_1^{\,i}(j)\right) < 0 \end{cases} \tag{16}$$

The resulting Prediction Method allocation solution [17] for any point on the i^{th} linear segment for $a \in [a_i, a_{a+1}]$ is:

$$\vec{u}_a = \vec{u}_{\perp_a} + \vec{u}_{\|_a} = \left(\vec{c}_0^{\,i} - a\hat{c}_1^{\,i}\right) + a\hat{u}_{des} \in \Omega \tag{17}$$

The PM was modified in [19] incorporating effector unsaturation defined as allowing an effector saturated in a previous iteration to unsaturate in the current one. Since $\vec{u}_{\|_a}$ in Eq. (17) is necessary and sufficient for $\vec{m}_a$, identifying effector unsaturation in the i^{th} iteration requires determining if a strict subset $S_{\underline{2}} \subset S_2^i$ yields a smaller $\vec{u}_{\perp_a}$ with the allocation feasible for some $a \in [a_i, a_{i+1}]$. For a given saturated indices set, the magnitude of the null space component is $f(a) = \|\vec{c}_0 + \vec{c}_1\|_2$, so we compare null space components for $S_{\underline{2}}$ and S_2^i with the Scalar Difference Quadratic (SDQ):

$$\mathcal{F}(a) = \left\|\vec{c}_{\underline{0}} + \vec{c}_{\underline{1}}\right\|_2^2 - \|\vec{c}_0 + \vec{c}_1\|_2^2 = c_a a^2 + c_b a + c_c \in \Re \tag{18}$$

$$c_a = \left(\hat{u}_{\underline{2}}^T \left(\hat{B}_{\underline{2}}\hat{B}_{\underline{2}}^T\right)^{-1} \hat{u}_{\underline{2}} - \hat{u}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \hat{u}_2\right) \tag{19}$$

$$c_b = -2\left(\vec{s}_{\underline{2}}^T \left(\hat{B}_{\underline{2}}\hat{B}_{\underline{2}}^T\right)^{-1} \hat{u}_{\underline{2}} - \vec{s}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \hat{u}_2\right) \tag{20}$$

$$c_c = \vec{s}_{\underline{2}}^T \left(\hat{B}_{\underline{2}}\hat{B}_{\underline{2}}^T\right)^{-1} \vec{s}_{\underline{2}} - \vec{s}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \vec{s}_2 \tag{21}$$

where the underbar denotes a strict subset where one or more effectors is removed from the set S_2^i. For the removal of one effector, the zero a_z of the SDQ coincides with the quadratic maximum where $\partial\mathcal{F}(a)/\partial a = 0 \implies a_z = -c_b/2c_a$.

In [19], the PM algorithm incorporating effector unsaturation was denoted the Sequential Null Space Generalized Inverse (SNSGI) algorithm. In this algorithm, for each iteration where less than or equal to $(m-n)$ effectors were saturated, all combinations of strict subsets $S_{\underline{2}}$ (with one effector removed from S_2^i) evaluated the SDQ. Effector unsaturation was identified and located if $a_z \in (a_i, a_{i+1}]$. Identified effector unsaturation required shortening the existing linear segment from $[a_i, a_{i+1}]$ to $[a_i, a_z]$ for S_2^i and starting the next iteration of the baseline PM at a_z using the strict subset $S_{\underline{2}}$ that identified unsaturation. Multiple instances of single effector unsaturation within the segment $[a_i, a_{i+1}]$ required use of the unsaturation occurring with the lowest a_z. Multiple effector unsaturation cases were performed as sequential single effector unsaturations over very small changes in moment magnitude. Since the first iteration with $(m-n)$ effectors saturated does not always yield maximal moments on the boundary of the AMS, unsaturation processing continued with $(m-n)$ effectors seeking unsaturation that increased the attainable moment. This was repeated until no increasing moment unsaturation was identified, thereby attaining the maximal moment on the AMS boundary.

III. Modified Cascading Generalized Inverse Algorithm

A. Legacy Cascading Generalized Inverse Algorithm

In this section, the Cascading Generalized Inverse Algorithm outlined in [3, 10] is presented. The following equations delineate the total moment, total control, delta control, and delta moment vectors respectively for the i^{th}

iteration for a given sequence of unsaturated and saturated effector indices sets [3].

$$\vec{m}_i = B\vec{u}_i \tag{22}$$

$$\vec{u}_i = \sum_{p=1}^{i} \Delta\vec{u}_p \tag{23}$$

$$\Delta\vec{u}_i = \left(k_i P_{T_i}\right)\Delta\vec{m}_i \tag{24}$$

$$\Delta\vec{m}_i = \vec{m}_{des} - \vec{m}_{i-1} \tag{25}$$

The vector $\vec{u}_i$ is the control allocation solution at the i^{th} corner of each piecewise allocation linear segment and $\vec{m}_i$ is the corresponding corner moment. P_{T_i} contains the generalized inverse P_{t_i} using the unsaturated effectors for the i^{th} iteration. The number of unsaturated effectors is $m - q$ for $B \in \Re^{n\times m}$ where q is the number of saturated effectors. If $m - q > n$ the P_{t_i} is the weighted generalized inverse computed using B_1. If $m - q = n$, then $P_{t_i} = B_1^{-1}$ and for $m - q < n$, then P_{t_i} is the weighted least squares:

$$P_{T_i} = \begin{bmatrix} P_{t_i} \in \Re^{(m-q)\times n} \\ 0 \in \Re^{q\times n} \end{bmatrix}, \quad P_{t_i} = \begin{cases} \left(W_P^T\right)^{-1} B_1^T \left(B_1 \left(W_P^T\right)^{-1} B_1^T\right)^{-1} & \text{for } (m-q) > n \\ B_1^{-1} & \text{for } (m-q) = n \\ \left(B_1^T W_d^T B_1\right)^{-1} B_1^T W_d^T & \text{for } (m-q) < n \end{cases} \tag{26}$$

The nominal version of the CGI algorithm applied to the minimal control problem starts using P_{min} to obtain $\Delta\vec{u}_1$. If any effectors in $\Delta\vec{u}_1$ exceed a effector position limit then the first scaling factor k_1 is determined, which brings the worst effector position to within limits. If no limits are exceeded, then $k_1 = 1$. In [3], the authors utilize the scaling on P_{t_1}, however the scaling could equally be considered as scaling $\Delta\vec{m}_1$ as $\Delta\vec{u}_i = \left(k_i P_{T_i}\right)\Delta\vec{m}_i = P_{T_i}\left(k_i \Delta\vec{m}_i\right)$. In this fashion, the first scaling factor k_1 yields the moment along $\hat{m}_{des}$ associated with the end (or corner) of the first piecewise linear segment. Subsequent corners are obtained in a similar fashion with $\vec{m}_i$ yielding the moment at the i^{th} segment corner. Expressed in terms of Eqs. (23, 24), the minimal scaling factor k_i for the i^{th} iteration is found using:

$$k_i = \underset{k}{\operatorname{argmin}}\{k_j \mid j \in S_1\} \text{ where} \tag{27}$$

$$k_j = \begin{cases} \left(\vec{u}_{upr}(j) - \vec{u}_{i-1}(j)\right) / \Delta\vec{u}_i(j) & \left(\vec{u}_{i-1}(j) + \Delta\vec{u}_i(j)\right) > \vec{u}_{upr}(j) \\ \left(\vec{u}_{lwr}(j) - \vec{u}_{i-1}(j)\right) / \Delta\vec{u}_i(j) & \left(\vec{u}_{i-1}(j) + \Delta\vec{u}_i(j)\right) < \vec{u}_{lwr}(j) \end{cases} \tag{28}$$

After each iteration, S_1^i and S_2^i are updated and iterations continue until the moment is achieved or $S_1^i = \emptyset$.

B. Modified Cascading Generalized Inverse

Incorporating effector unsaturation in the CGI algorithm is similar to that of the PM but uses a CGI specific SDQ. Specifically, identifying effector unsaturation in the i^{th} iteration requires determining if a strict subset $S_{\underline{2}} \subset S_2^i$ yields a smaller $\vec{u}_{\perp_a}$ such the allocation is feasible for some $a \in [a_i, a_{i+1}]$. From Lemma 0.1 of [20], the CGI form of $\vec{u}_{\perp_i}$ is:

$$\begin{aligned}\vec{u}_{\perp_i} &= \vec{c}_0^{\,i} + \vec{c}_1^{\,i} = \left(I - P_{T_i} B\right) \sum_{p=1}^{i-1} \Delta \vec{u}_p + \left(P_{T_i} - P_{min}\right) \vec{m}_{des} \\ &= \left(I - P_{T_i} B\right) \vec{u}_{i-1} + \left(P_{T_i} - P_{min}\right) \vec{m}_{des} \end{aligned} \tag{29}$$

Evaluating the null space component magnitudes for the sets $S_{\underline{2}}$ and S_2^i using Eq. (29) and simplifying we obtain:

$$\mathcal{F}(a) = \left\|\vec{c}_{\underline{0}} + \vec{c}_{\underline{1}}\right\|_2^2 - \|\vec{c}_0 + \vec{c}_1\|_2^2 = c_a a^2 + c_b a + c_c \in \Re, \ \text{ where } a_z = -c_b/2c_a \tag{30}$$

$$c_a = \hat{u}_{des}^T B^T \left(P_{\underline{T_i}}^T P_{\underline{T_i}} - P_{T_i}^T P_{T_i}\right) B \hat{u}_{des} \tag{31}$$

$$c_b = 2\hat{u}_{des}^T B^T \left(P_{\underline{T_i}}^T \left(I - P_{\underline{T_i}} B\right) - P_{T_i}^T \left(I - P_{T_i} B\right)\right) \vec{u}_{i-1} \tag{32}$$

$$c_c = \vec{u}_{i-1}^T \left(\left(P_{T_i} B\right)^T \left(I - P_{T_i} B\right) - \left(P_{\underline{T_i}} B\right)^T \left(I - P_{\underline{T_i}} B\right) + \left(P_{T_i} - P_{\underline{T_i}}\right) B\right) \vec{u}_{i-1} \tag{33}$$

For each corner (starting at the $i = 2$), the MCGI algorithm creates all combinations of saturation subsets $S_{\underline{2}}$ which differ by one effector from S_2^i and performs each SDQ evaluation. Effector unsaturation is identified if $a_z \in (a_i, a_{i+1}]$ and requires shortening the existing linear segment from $[a_i, a_{i+1}]$ to $[a_i, a_z]$ for S_2^i and then starting the next iteration of the baseline CGI at a_z using the strict subset $S_{\underline{2}}$ that identified unsaturation. Multiple instances of single effector unsaturation and multiple effector unsaturation cases are processed the same as with PM. Additionally, iterations after the first instance of $(m - n)$ saturated effectors continue with unsaturation processing until the desired moment or boundary moment are achieved.

C. Graphic Understanding of the Unsaturation Modification Impact

In this section, affine generalized inverse subsets are used to provide insight into CGI and the MCGI. For the non-affine generalized inverse P_{gi} (with $\vec{u} = P_{gi}\vec{m}$), then the non-affine set Φ_{gi} (which denotes feasible allocations), is bounded by the intersection of the ACS with the subspace spanned by the columns of P_{gi} [7]. This boundary is obtained by checking the allocation feasibility for each combination of n saturated effectors for all position limit combinations. Defining the set A with n saturated effectors for one particular combination and the unsaturated indices set B, we write:

$$\vec{u}_A = \left[P_{gi}\right]_A \vec{m} \tag{34}$$

$$\vec{u}_B = \left[P_{gi}\right]_B \vec{m} \tag{35}$$

where $\vec{u}_A$ is one of the prescribed combinations of saturated effector positions. Since P_{gi} is invertible, then $\vec{u}_B = \left[P_{gi}\right]_B \left[P_{gi}\right]_A^{-1} \vec{u}_A$. We know $\vec{u}_A$ is feasible, so if $\vec{u}_B$ is feasible then we retain this particular combination's solution. The convex hull (in moment space) of all retained feasible solutions is the boundary of the feasible subset Φ_{gi}.

The CGI (and AGI) specific generalized inverse for the i^{th} iteration is $P_{T_i}^T = \begin{bmatrix} P_{t_i}^T & 0 \end{bmatrix}$ with $P_{t_i} = B_1^T \left(B_1 B_1^T\right)^{-1}$ prescribed by the unsaturated effectors in S_1^i and the zero rows of P_{T_i} corresponding with the saturated effectors of S_2^i. Thus for the saturated effector position limits $\vec{s}_2^i$ we divide the affine solution into unsaturated and saturated components:

$$\vec{u}_1^i = \left[\vec{c}_0^i\right]_1 + P_{t_i}\vec{m} \tag{36}$$

$$\vec{u}_2^i = \vec{s}_2^i \tag{37}$$

Applying the feasible subset process of [7] to the affine generalized inverse of Eqs. (36, 37), we examine combinations of n saturated effectors from the unsaturated indices set S_1^i and obtain:

$$\vec{u}_{1A}^i = \left[\vec{c}_0^i\right]_{1A} + \left[P_{t_i}\right]_A \vec{m} \tag{38}$$

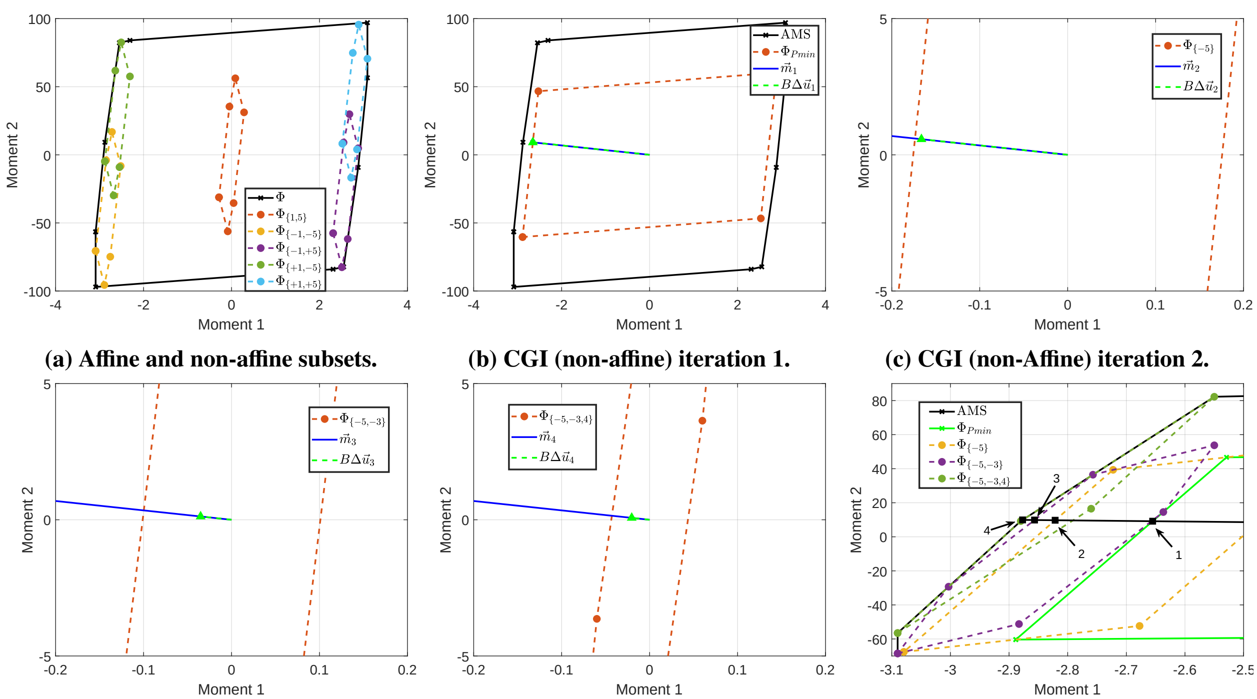
$$\vec{u}_{1B}^i = \left[\vec{c}_0^i\right]_{1B} + \left[P_{t_i}\right]_B \vec{m} \tag{39}$$

$$\vec{u}_2^i = \vec{s}_2^i \tag{40}$$

Now since $\vec{u}_{1A}^i$ and $\vec{u}_2^i$ are feasible, we require the feasibility of $\vec{u}_{1B}^i = \left[\vec{c}_0^i\right]_{1B} + \left[P_{t_i}\right]_B \left[P_{t_i}\right]_A^{-1} \left(\vec{u}_{1A}^i - \left[\vec{c}_0^i\right]_{1A}\right)$. The convex hull of combinations yielding feasible allocations for $\vec{u}_{1B}^i$ bounds the affine generalized inverse subset. For notation, we denote the non-affine feasible set using the saturated effectors in S_2^i. For example, $\Phi_{\{2,5\}} \subset \Phi$ consists of moments for which $P_{\{2,5\}}$ yields feasible allocations. We distinguish the affine subsets by signing the saturated effectors. For example the affine subset $\Phi_{\{+2,-5\}} \subset \Phi$ is generated by $P_{\{2,5\}}$ where the null space vector $\vec{c}_0^i$ has the second effector saturated at its upper limit and the fifth effector saturated at the lower limit.

Figure 1a shows an example of a non-affine subset $\Phi_{\{1,5\}}$ and corresponding affine attainable moment subsets $\Phi_{\{\pm 1,\pm 5\}}$. In this figure, moments within the interior of an affine subset yield allocation solutions with only the effectors $\{1,5\}$ saturated at their corresponding position limits using Eq. (37). Conversely, an affine subset boundary edge moment saturates an additional effector and a corner moment saturates two additional effectors from Eq. (36) shown in this 2-D case. Figures 1b -1e show an example of the nominal CGI solutions of Eqs. (22 -25) and the non-affine subsets generated from the saturated effectors at each iteration. While an additional effector saturates during each CGI iteration, only in the first iteration does the current iteration moment contribution $B\Delta\vec{u}_i$ (shown in green) reach the non-affine subset boundary as might be expected. Figure 1f shows the same CGI solution but now shows the affine subsets generated by the saturated effectors at each iteration. This figure shows that the total moment for each iteration terminates on the affine subset boundary. For example, the first iteration reaches the boundary of Φ_{Pmin} saturating the fifth effector at the lower position limit. The second iteration starts within the interior of $\Phi_{\{-5\}}$ (where only the fifth effector is saturated at the lower limit) and terminates at an edge of $\Phi_{\{-5\}}$ which saturates the third effector at its lower position limit. This process continues for all iterations. The use of affine subsets in Figure 1f provides graphical insight into the effector saturation allocation process during legacy CGI iterations.

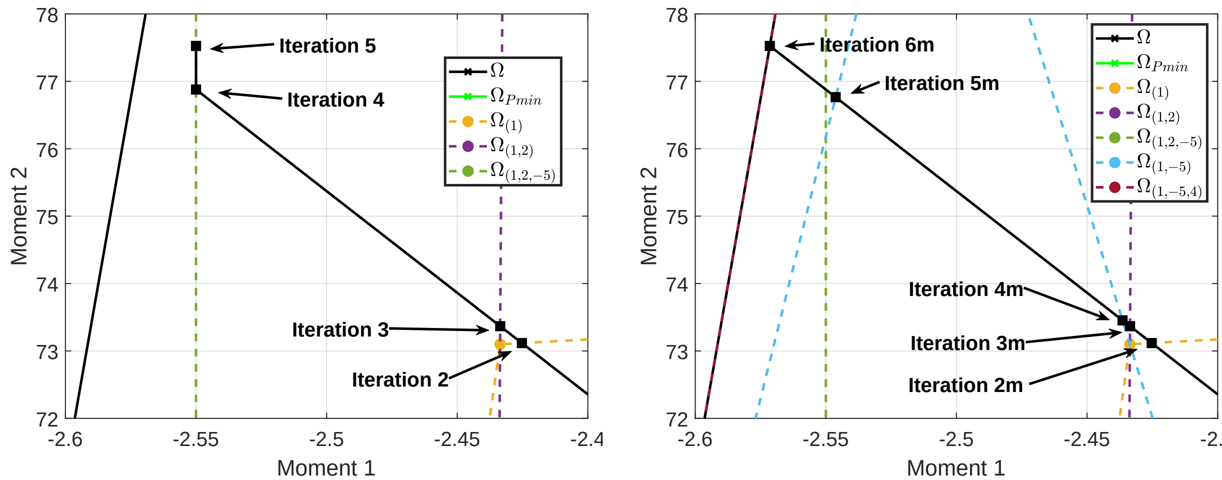
We will use affine subsets to aid understanding of the known issues with CGI and the improvements provided by MCGI. Figures 2a and 2b show the CGI and MCGI iterations using affine subsets for an example moment on the AMS boundary. Figure 2a clearly shows a failure to preserve moment direction after iteration 4, and an inability to obtain some moments (between iteration 4 and the AMS boundary). These CGI failures occur after iteration four as the algorithm only has one available non-saturated effector to yield two desired moment components. Examining the MCGI iterations in Figure 2b, the algorithm preserves moment direction and attains moments which were previously unobtainable. Moreover, after the moment at 3m, the MCGI algorithm yields allocation in the interior of $\Phi_{\{1,2,-5\}}$ for a short distance before switching to $\Phi_{\{1,-5\}}$ by unsaturating effector 2. The point of effector unsaturation occurs at iteration 4m on the boundary of $\Phi_{\{1,-5\}}$ ensuring continuity of allocations (i.e., effectors $\{1, 2, -5\}$ are saturated at iteration 4m moment on the boundary of $\Phi_{\{1,-5\}}$.) However, after the iteration 4m moment, effector two unsaturates in the interior of $\Phi_{\{1,-5\}}$ until the iteration terminates at moment 5m. Comparing this to Figure 2a shows how MCGI utilizes effector unsaturation to attain smaller allocations for identical moments after the moment at 4m.



(a) Affine and non-affine subsets. **(b) CGI (non-affine) iteration 1.** **(c) CGI (non-Affine) iteration 2.**

(d) CGI (non-affine) iteration 3. **(e) CGI (non-affine) iteration 4.** **(f) CGI (affine) iterations 1-4.**

Fig. 1 Affine and Non-Affine CGI Iterations



(a) Nominal CGI **(b) Modified CGI**

Fig. 2 CGI and MCGI Comparison

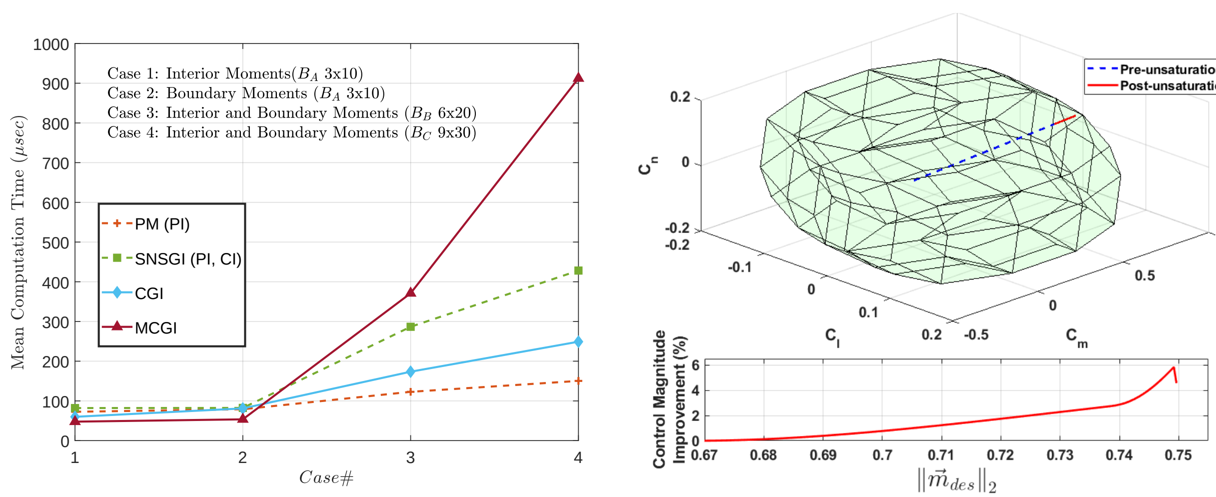
IV. Results

Four numerical cases studies evaluated the performance of the MCGI against the PM, the SNSGI, and the nominal CGI for the minimal control allocation problem. These studies examined algorithm performance for three different constant control effectiveness matrices to evaluate computational speed, and each algorithm's ability to produce the desired moments and associated minimal allocation. The control effectiveness matrices are denoted $B_A \in \Re^{3\times10}$, $B_B \in \Re^{6\times20}$, and $B_C \in \Re^{9\times30}$. Equation (41) is the linearized control effectiveness matrix B_A for the NASA F-18 High Alpha Research Vehicle (HARV) [21] at an altitude of 10,000 feet, Mach = 0.3 and AoA = 12.5 degrees. B_B and B_C were randomly generated matrices and effector position limit vectors with B_B representative of a modern aircraft with six desired outcomes (e.g., three moments and three forces) for 20 effectors. The increased dimensions of B_C challenged the algorithms as increasing m generally increases the number of iterations required for a given $\vec{m}_{des}$ and increasing n increases the matrix inversion cost at each iteration. Case studies one and two utilized B_A with one thousand interior and boundary moments respectively, which were generated by random combinations of all effectors saturated at some effector position limit from Eq. (42) separated into boundary and interior cases. Similarly cases three and four consisted of a mixture of one thousand random interior and boundary moments generated from random combinations of saturated effectors for B_B and B_C respectively.

$$B = \begin{bmatrix} -4.382 & +4.382 & -5.841 & +5.841 & +1.674 & -6.280 & +6.280 & +2.920 & +1.000 & +1.000 \\ -0.5330 & -0.5330 & -6.486 & -6.486 & +0.000 & +6.234 & +6.234 & +0.001 & +33.53 & +0.001 \\ +1.100 & -1.100 & +3.911 & -3.911 & -7.428 & +0.000 & +0.000 & +.030 & +0.001 & +14.85 \end{bmatrix} \times 10^{-2} \tag{41}$$

$$\begin{aligned} \vec{u}_{upr} &= \begin{bmatrix} +0.183 & +0.183 & +0.524 & +0.524 & +0.524 & +0.785 & +0.785 & +0.524 & +0.524 & +0.524 \end{bmatrix} \\ \vec{u}_{lwr} &= \begin{bmatrix} -0.183 & -0.183 & -0.524 & -0.524 & -0.524 & -0.785 & -0.785 & -0.524 & -0.524 & -0.524 \end{bmatrix} \end{aligned} \tag{42}$$

Figure 3a compares computational performance for the non-unsaturation capable algorithms (PM and CGI) and the effector unsaturation capable algorithms (SNSGI and MCGI) using MATLAB®* m-files auto-coded mex files on a standard PC with an Intel I7 processor. Both PM and SNSGI utilized partitioned inverses to improve algorithm efficiency [19] . The additional computational burden of unsaturation identification and location is evident when comparing PM to SNSGI and similarly when comparing CGI to MCGI. All algorithms have good speed for B_A but the partitioned inverse algorithms outperform their counterparts by approximately a factor of two in case four using the largest matrix B_C.



(a) Algorithm Mean Computation Time **(b) Single Unsaturation Case MCGI to CGI Comparison**

Fig. 3 Algorithm Comparisons

Next the algorithms were evaluated for moment and optimal allocation capability for cases 1-4. The results for case two are shown in Table 1 where one thousand boundary moments were validated against the unique Direct Allocation (DA) solution using the online supplement [7]. Results for cases one, three and four are omitted for brevity as they

*The use of trademarks or names of manufacturers in this report is for accurate reporting and does not constitute an official endorsement either expressed or implied, of such products or manufacturers by the National Aeronautics and Space Administration.

yielded similar performance shown in Table 1. A moment error metric E_{m_i} in Eq. (43), compares an algorithm's moment output $\vec{m}_{alg_i}$ to desired moment $\vec{m}_{des_i}$ for each moment $i \in [1, 2, \ldots, 1000]$ of case two. Similarly, a signed control error metric E_{c_i} of Eq. (44), compares each algorithm's $\vec{u}_{alg_i}$ to the DA algorithm's unique boundary allocation reference $\vec{u}_{ref_i}$ [6].

$$E_{m_i} = \left\|\vec{m}_{des_i} - \vec{m}_{alg_i}\right\|_2 / \left\|\vec{m}_{des_i}\right\|_2 \tag{43}$$

$$E_{c_i} = \left(\left\|\vec{u}_{alg_i}\right\|_2 - \left\|\vec{u}_{ref_i}\right\|_2\right) / \left\|\vec{u}_{ref_i}\right\|_2 \tag{44}$$

The mean and standard deviation for both the moment (μ_m and σ_m) and control (μ_u and σ_u) error metrics are also provided using all one thousand moments of case two. Additionally, a binary (i.e., success or fail) moment validation metric V_{m_i} determined if all components of an algorithm's moment solution $\vec{m}_{alg_i}$ was within a component-wise tolerance of 1×10^{-10} to $\vec{m}_{des_i}$. Similarly, a control validation metric V_{u_i} compared $\vec{u}_{alg_i}$ to $\vec{u}_{ref_i}$ with the same component-wise tolerance. The values of V_m and V_u (in red) denote the percentage of failed validation cases whereas V_m and V_u (in black) are successful cases for algorithms. Table 1 shows that MCGI always achieves the correct boundary moment and unique allocation exhibiting an improvement over CGI which fails in 34% of cases.

Table 1 Boundary Validation Statistics, $B_A \in \Re^{3\times10}$, 1000 Cases

	$V_m(\%)$	$\mu_m(\%)$	$\sigma_m(\%)$	$V_u(\%)$	$\mu_u(\%)$	$\sigma_u(\%)$
PM (PI)	35.5	+0.447	0.749	39.8	-3.77	5.16
SNSGI (PI, IC)	100	$+1.83 \times 10^{-11}$	3.75×10^{-11}	4.0	-5.92×10^{-07}	1.32×10^{-06}
CGI	34.3	+0.319	0.633	34.3	-8.07	7.21
MCGI	100	$+1.31 \times 10^{-12}$	8.32×10^{-12}	100	-1.31×10^{-10}	5.62×10^{-10}
DA	100	REF SOURCE *	-	100	REF SOURCE *	-

1) V_m, V_u: % validation cases. Failure (red) and success (black)
2) Mean (μ_m) and Std Dev (σ_m) of moment error metric E_m
3) Mean (μ_u) and Std Dev (σ_u) of control error metric E_u
* Unique DA solution used as reference control

Finally, an unsaturation example using B_A is shown in Fig. 3b for a series of desired non-dimensionalized aerodynamic moment coefficients from the origin to $\vec{m}_{des}^T = \begin{bmatrix} 0.053 & 0.761 & 0.1 \end{bmatrix}$ on the boundary of the AMS. The red portion of this line delineates those moments after unsaturation of effector six occurs. The lower subplot shows the control magnitude (%) reduction $(\|\vec{u}_{CGI}\|_2 - \|\vec{u}_{MCGI}\|_2) / \|\vec{u}_{MCGI}\|_2$ achieved for post-unsaturation moments only. In this example, MCGI is seen to have achieved an $\approx 6\%$ maximum control magnitude reduction over CGI analyzed at identical moments after unsaturation occurs.

V. Conclusions

Analysis of the legacy CGI algorithm has demonstrated that it intermittently yields non-optimal control allocation solutions, fails to preserve moment direction, and/or is unable to obtain desired moments. In this research, a proposed Modified Cascading Generalized Inverse algorithm incorporated an effector unsaturation capability and was shown to resolve these deficiencies. The use of affine generalized inverse subsets provides graphical insight into the CGI algorithm and physical meaning to effector unsaturation. Moreover, rigorous proofs are included that show the theoretical basis for effector unsaturation ensuring optimal allocations, which addresses a substantial gap in the existing body of generalized inverse literature.

Appendix

This appendix documents Scalar Difference Quadratic (SDQ) proofs discussed in [19] and [20]. These proofs show that use of the SDQ in the PM enables it to achieve minimal control allocations for all desired moments in the AMS. Similarly, incorporation of SDQ into CGI achieves optimal control allocations for all moments due to the equivalence of PM and CGI allocations at each iteration [20] and the equivalence of the PM and CGI SDQ formulations [19]. The SDQ is a scalar function used to compare magnitudes of the null space components of control solutions using two different sets of saturated effector indices $S_{\underline{2}}$, S_2 where $S_{\underline{2}} \subset S_2$.

The aggregated proofs in this appendix show that if the discriminant of the SDQ function $\mathcal{F}(a)$ is identically zero, then the real repeated root a_z where $\mathcal{F}(a_z) = 0$ coincides with the maximum value of the negative semi-definite SDQ. Additionally, it is shown that the respective optimal allocation solutions $\vec{\underline{u}}_{\text{opt}}$ and $\vec{u}_{\text{opt}}$ yield equivalent moments as a function of the scalar gain a. Moreover, all components of the optimal allocation solutions $\vec{\underline{u}}_{\text{opt}}$, and $\vec{u}_{\text{opt}}$ are identical at the root a_z. When taken together, these proofs show that if the SDQ root is within the current linear segment, then unsaturating the effectors $S_2 \setminus S_{\underline{2}}$ ensures continuity of allocations at a_z and yields smaller total allocation magnitudes for moments after a_z. Thus remarkably, in any arbitrary linear solution segment, if the SDQ discriminant is zero (which is always true when unsaturating one effector) then simply determining if $a_z \in (a_i, a_{i+1}]$ identifies that effector unsaturation must occur at a_z to achieve smaller subsequent allocations.

To facilitate other proofs comparing the saturation indices subsets $S_{\underline{2}}$, and S_2 with $S_{\underline{2}} \subset S_2$ where the saturation indices sets differ by one or more effectors, we establish the following reordered partition notation for B and $\mathcal{N}(B) = \hat{B}$:

$$B = \begin{bmatrix} B_1 & B_R & B_2 \end{bmatrix}, \quad \hat{B}^T = \begin{bmatrix} \hat{B}_1^T & \hat{R}^T & \hat{B}_2^T \end{bmatrix} \tag{45}$$

where the quantities B_R and $\hat{R}$ are the portions of B and $\hat{B}$ corresponding to the effectors which are removed from S_2 to create the strict subset $S_{\underline{2}}$. The following (reordered) partitioned inverse from [19] is used in subsequent proofs:

$$\left(\begin{bmatrix}\hat{R}\\ \hat{B}_2\end{bmatrix}\begin{bmatrix}\hat{R}^T & \hat{B}_2^T\end{bmatrix}\right)^{-1} = \begin{bmatrix}\Delta^{-1} & -\Delta^{-1}\hat{R}\hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\\ -\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\left(\hat{B}_2\hat{R}^T\right)\Delta^{-1} & \left(\hat{B}_2\hat{B}_2^T\right)^{-1} + \left(\hat{B}_2\hat{B}_2^T\right)^{-1}\left(\hat{B}_2\hat{R}^T\right)\Delta^{-1}\left(\hat{B}_2\hat{R}^T\right)^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\end{bmatrix} \tag{46}$$

$$\text{where } \Delta = \hat{R}\hat{R}^T - \hat{R}\hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{B}_2\hat{R}^T \tag{47}$$

Additionally, it is useful to note that $\left(\hat{B}_2\hat{B}_2^T\right)$ is symmetric and positive semidefinite where $B \in \Re^{n\times m}$. Moreover, $\left(\hat{B}_2\hat{B}_2^T\right)$ is invertible for $\hat{B}_2 \in \Re^{d\times(m-n)}$ where $d \leq (m-n)$ is the number of saturated effectors, and therefore $\left(\hat{B}_2\hat{B}_2^T\right)$ and $\left(\hat{B}_2\hat{B}_2^T\right)^{-1}$ are positive definite for $d \leq (m-n)$. Now we establish the total allocation solutions $\vec{u}_A$ and $\vec{u}_B$ as:

$$\vec{u}_A = \begin{bmatrix}\hat{B}_1\\ \hat{R}\\ \hat{B}_2\end{bmatrix}\hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\vec{s}_2 + \begin{bmatrix}B_1^T\\ B_R^T\\ 0\end{bmatrix}\left(B_1B_1^T + B_RB_R^T\right)^{-1}\vec{m} \tag{48}$$

$$\vec{u}_B = \begin{bmatrix}\hat{B}_1\\ \hat{R}\\ \hat{B}_2\end{bmatrix}\begin{bmatrix}\hat{R}^T & \hat{B}_2^T\end{bmatrix}\left(\begin{bmatrix}\hat{R}\\ \hat{B}_2\end{bmatrix}\begin{bmatrix}\hat{R}^T & \hat{B}_2^T\end{bmatrix}\right)^{-1}\begin{bmatrix}\vec{s}_R\\ \vec{s}_2\end{bmatrix} + \begin{bmatrix}B_1^T\left(B_1B_1^T\right)^{-1}\\ 0\\ 0\end{bmatrix}\vec{m} \tag{49}$$

which correspond with $S_{\underline{2}}$ and S_2 respectively. Equations (48, 49) are Affine Generalized Inverse total allocation solutions, which were shown in [18] to be equivalent to those of the Prediction Method. Using $\vec{u}_B$ from Eq. (49) and the partitioned inverse from Eqs. (46, 47), then after simplification we obtain:

$$\vec{u}_A = \begin{bmatrix}\hat{B}_1\\ \hat{R}\\ \hat{B}_2\end{bmatrix}\hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\vec{s}_2 + \begin{bmatrix}B_1^T\\ B_R^T\\ 0\end{bmatrix}\left(B_1B_1^T + B_RB_R^T\right)^{-1}\vec{m} \tag{50}$$

$$\vec{u}_B = \begin{bmatrix}\hat{B}_1\left(I - \hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{B}_2\right)\left(\hat{R}^T\Delta^{-1}\vec{s}_R - \hat{R}^T\Delta^{-1}\hat{R}\hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\vec{s}_2\right) + \hat{B}_1\hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\vec{s}_2\\ \vec{s}_R\\ \vec{s}_2\end{bmatrix} + \begin{bmatrix}B_1^T\left(B_1B_1^T\right)^{-1}\\ 0\\ 0\end{bmatrix}\vec{m} \tag{51}$$

Lemma 0.1. *Given the control effectiveness matrix $B \in \Re^{n\times m}$, and the optimal allocation solutions of Eqs. (50, 51), then $B\vec{u}_A = B\vec{u}_B = \vec{m}$.*

Proof. First expanding $B\vec{u}_A$ and using $B_1\hat{B}_1 + B_R\hat{R} = -B_2\hat{B}_2$ (i.e., $B\hat{B} = 0$), then:

$$
\begin{aligned}
B\vec{u}_A &= \begin{bmatrix} B_1 & B_R & B_2 \end{bmatrix} \begin{bmatrix} \hat{B}_1 \\ \hat{R} \\ \hat{B}_2 \end{bmatrix} \hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \vec{s}_2 + \begin{bmatrix} B_1 & B_R & B_2 \end{bmatrix} \begin{bmatrix} B_1^T \\ B_R^T \\ 0 \end{bmatrix} \left(B_1B_1^T + B_RB_R^T\right)^{-1} \vec{m} \\
&= \left(B_1\hat{B}_1 + B_R\hat{R}\right) \hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \vec{s}_2 + B_2\hat{B}_2\hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \vec{s}_2 + \left(B_1B_1^T + B_RB_R^T\right)\left(B_1B_1^T + B_RB_R^T\right)^{-1} \vec{m} \\
&= \left(-B_2\hat{B}_2\right) \hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \vec{s}_2 + B_2\vec{s}_2 + I\vec{m} = -B_2\vec{s}_2 + B_2\vec{s}_2 + \vec{m} = \vec{m}
\end{aligned}
\tag{52}
$$

Similarly, expanding $B\vec{u}_B$ and using $B_1\hat{B}_1 = \left(-B_R\hat{R} - B_2\hat{B}_2\right)$, and $\Delta = \hat{R}\hat{R}^T - \hat{R}\hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \hat{B}_2\hat{R}^T$, then:

$$
\begin{aligned}
B\vec{u}_B =& B_1\hat{B}_1 \left(I - \hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \hat{B}_2\right) \left(\hat{R}^T\Delta^{-1}\vec{s}_R - \left(\hat{R}^T\Delta^{-1}\hat{R}\hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1}\right)\vec{s}_2\right) + B_1\hat{B}_1\hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \vec{s}_2 + B_R\vec{s}_R \\
&+ B_2\vec{s}_2 + B_1B_1^T \left(B_1B_1^T\right)^{-1} \vec{m} \\
=& \left(-B_R\hat{R} - B_2\hat{B}_2\right) \left(I - \hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \hat{B}_2\right) \hat{R}^T\Delta^{-1}\vec{s}_R - \left(B_R\hat{R} + B_2\hat{B}_2\right) \hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \vec{s}_2 + B_R\vec{s}_R + B_2\vec{s}_2 + I\vec{m} \\
&+ \left(B_R\hat{R} + B_2\hat{B}_2\right) \left(I - \hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \hat{B}_2\right) \left(\hat{R}^T\Delta^{-1}\hat{R}\hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1}\right)\vec{s}_2 \\
=& -B_R \left(\hat{R}\hat{R}^T - \hat{R}\hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \hat{B}_2\hat{R}^T\right) \Delta^{-1}\vec{s}_R + B_R \left(\hat{R}\hat{R}^T - \hat{R}\hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \hat{B}_2\hat{R}^T\right) \Delta^{-1}\hat{R}\hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \vec{s}_2 \\
&- B_R\hat{R}\hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \vec{s}_2 - B_2\vec{s}_2 + B_R\vec{s}_R + B_2\vec{s}_2 + \vec{m} \\
=& -B_R\vec{s}_R + B_R\hat{R}\hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \vec{s}_2 - B_R\hat{R}\hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \vec{s}_2 - B_2\vec{s}_2 + B_R\vec{s}_R + B_2\vec{s}_2 + \vec{m} = \vec{m}
\end{aligned}
\tag{53}
$$

Therefore $B\vec{u}_A = B\vec{u}_B = \vec{m}$ as claimed. □

Lemma 0.2. *Given the control effectiveness matrix $B \in \Re^{m\times n}$, the partitioned inverse of Eq. (46), and saturated effector SDQ defined in Eqs. (30 -33), then the curvature of the SDQ $c_a \leq 0$ and similarly the discriminant $c_b^2 - 4c_ac_c \leq 0$.*

Proof. Expanding the partitioned inverse term from Eq. (31) we obtain:

$$\begin{bmatrix}\hat{u}_2^T & \hat{u}_R^T\end{bmatrix}\left(\begin{bmatrix}\hat{B}_2\\ \hat{R}\end{bmatrix}\begin{bmatrix}\hat{B}_2^T & \hat{R}^T\end{bmatrix}\right)^{-1}\begin{bmatrix}\hat{u}_2\\ \hat{u}_R\end{bmatrix} =$$

$$\begin{bmatrix}\hat{u}_2^T & \hat{u}_R^T\end{bmatrix}\begin{bmatrix}(\hat{B}_2\hat{B}_2^T)^{-1} + (\hat{B}_2\hat{B}_2^T)^{-1}(\hat{B}_2\hat{R}^T)\Delta^{-1}(\hat{B}_2\hat{R}^T)^T(\hat{B}_2\hat{B}_2^T)^{-1} & -(\hat{B}_2\hat{B}_2^T)^{-1}(\hat{B}_2\hat{R}^T)\Delta^{-1}\\ -\Delta^{-1}\hat{R}\hat{B}_2^T(\hat{B}_2\hat{B}_2^T)^{-1} & \Delta^{-1}\end{bmatrix}\begin{bmatrix}\hat{u}_2\\ \hat{u}_R\end{bmatrix} \tag{54}$$

$$\begin{aligned} &= \hat{u}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{u}_2 + \hat{u}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\left(\hat{B}_2\hat{R}^T\right)\Delta^{-1}\left(\hat{B}_2\hat{R}^T\right)^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{u}_2 \\ &- \hat{u}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\left(\hat{B}_2\hat{R}^T\right)\Delta^{-1}\hat{u}_R - \hat{u}_R^T\Delta^{-1}\hat{R}\hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{u}_2 + \hat{u}_R^T\Delta^{-1}\hat{u}_R \end{aligned} \tag{55}$$

Now expanding the inverse term from Eq. (31) we obtain:

$$\begin{aligned}\begin{bmatrix}\vec{s}_2^T & \vec{s}_R^T\end{bmatrix}\left(\begin{bmatrix}\hat{B}_2\\ \hat{R}\end{bmatrix}\begin{bmatrix}\hat{B}_2^T & \hat{R}^T\end{bmatrix}\right)^{-1}\begin{bmatrix}\hat{u}_2\\ \hat{u}_R\end{bmatrix} &= \vec{s}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{u}_2 + \vec{s}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\left(\hat{B}_2\hat{R}^T\right)\Delta^{-1}\left(\hat{B}_2\hat{R}^T\right)^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{u}_2 \\ &- \vec{s}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\left(\hat{B}_2\hat{R}^T\right)\Delta^{-1}\hat{u}_R - \vec{s}_R^T\Delta^{-1}\hat{R}\hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{u}_2 + \vec{s}_R^T\Delta^{-1}\hat{u}_R\end{aligned} \tag{56}$$

Similarly expanding the term from Eq. (33) yields:

$$\begin{aligned}\begin{bmatrix}\vec{s}_2^T & \vec{s}_R^T\end{bmatrix}\left(\begin{bmatrix}\hat{B}_2\\ \hat{R}\end{bmatrix}\begin{bmatrix}\hat{B}_2^T & \hat{R}^T\end{bmatrix}\right)^{-1}\begin{bmatrix}\vec{s}_2\\ \vec{s}_R\end{bmatrix} &= \vec{s}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\vec{s}_2 + \vec{s}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\left(\hat{B}_2\hat{R}^T\right)\Delta^{-1}\left(\hat{B}_2\hat{R}^T\right)^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\vec{s}_2 \\ &- \vec{s}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\left(\hat{B}_2\hat{R}^T\right)\Delta^{-1}\vec{s}_R - \vec{s}_R^T\Delta^{-1}\hat{R}\hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\vec{s}_2 + \vec{s}_R^T\Delta^{-1}\vec{s}_R\end{aligned} \tag{57}$$

So then substituting Eqs. (55 -57) into Eqs. (31 -33) and simplifying:

$$\begin{aligned} c_a &= -\hat{u}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\left(\hat{B}_2\hat{R}^T\right)\Delta^{-1}\left(\hat{B}_2\hat{R}^T\right)^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{u}_2 + \hat{u}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\left(\hat{B}_2\hat{R}^T\right)\Delta^{-1}\hat{u}_R \\ &+ \hat{u}_R^T\Delta^{-1}\hat{R}\hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{u}_2 - \hat{u}_R^T\Delta^{-1}\hat{u}_R \end{aligned} \tag{58}$$

$$\begin{aligned} c_b &= -\vec{s}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\left(\hat{B}_2\hat{R}^T\right)\Delta^{-1}\left(\hat{B}_2\hat{R}^T\right)^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{u}_2 + \vec{s}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\left(\hat{B}_2\hat{R}^T\right)\Delta^{-1}\hat{u}_R \\ &+ \vec{s}_R^T\Delta^{-1}\vec{R}\hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{u}_2 - \vec{s}_R^T\Delta^{-1}\hat{u}_R \end{aligned} \tag{59}$$

$$\begin{aligned} c_c &= -\vec{s}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\left(\hat{B}_2\hat{R}^T\right)\Delta^{-1}\left(\hat{B}_2\hat{R}^T\right)^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\vec{s}_2 + \vec{s}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\left(\hat{B}_2\hat{R}^T\right)\Delta^{-1}\vec{s}_R \\ &+ \vec{s}_R^T\Delta^{-1}\hat{R}\hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\vec{s}_2 - \vec{s}_R^T\Delta^{-1}\vec{s}_R \end{aligned} \tag{60}$$

The inverse $\left(\hat{B}_2\hat{B}_2^T\right)^{-1} > 0$ for $k \leq (m-n)$ where $\left(\hat{B}_2\hat{B}_2^T\right)^{-1} \in \Re^{(k\times k)}$. Since every submatrix of a positive definite matrix is positive definite, then the submatrix Δ^{-1} is positive definite and thus we can factor such that $\Delta^{-1} = \Delta^{-1/2}\Delta^{-1/2}$. Now let d be the number of saturated effectors to remove (i.e., $\vec{s}_R$, $\vec{u}_R \in \Re^d$), then we define the following terms:

$$\xi = \Delta^{-1/2}\hat{R}\hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \vec{s}_2 \in \Re^d \tag{61}$$

$$\tau = \Delta^{-1/2}\hat{R}\hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \hat{u}_2 \in \Re^d \tag{62}$$

$$\eta = \Delta^{-1/2}\vec{s}_R \in \Re^d \tag{63}$$

$$\zeta = \Delta^{-1/2}\hat{u}_R \in \Re^d \tag{64}$$

Using these definitions, we can rewrite Eq.(58 -60) as:

$$c_a = -1\left(\tau^T\tau - 2\tau^T\zeta + \zeta^T\zeta\right) = -\left(\tau^T(\tau-\zeta) - \zeta^T(\tau-\zeta)\right) = -\left((\tau-\zeta)^T(\tau-\zeta)\right) \leq 0, \text{ where } c_a \in \Re \tag{65}$$

$$c_b = 2\left(\xi^T\tau - \xi^T\zeta - \eta^T\tau + \eta^T\zeta\right) = 2\left((\xi-\eta)^T\tau - (\xi-\eta)^T\zeta\right) = 2\left((\xi-\eta)^T(\tau-\zeta)\right) \in \Re \tag{66}$$

$$c_c = -1\left(\xi^T\xi - 2\xi^T\eta + \eta^T\eta\right) = -\left((\xi-\eta)^T(\xi-\eta)\right) \leq 0, \text{ where } c_c \in \Re \tag{67}$$

From Eq. (65), we have $c_a \leq 0$. We can express the discriminant as:

$$\begin{aligned} c_b^2 - 4c_ac_c &= 4\left((\xi-\eta)^T(\tau-\zeta)\right)\left((\xi-\eta)^T(\tau-\zeta)\right) - 4\left((\tau-\zeta)^T(\tau-\zeta)\right)\left((\xi-\eta)^T(\xi-\eta)\right) \\ &= 4\|\xi-\eta\|_2^2\|\tau-\zeta\|_2^2\cos^2\theta - 4\|\xi-\eta\|_2^2\|\tau-\zeta\|_2^2 \end{aligned} \tag{68}$$

where θ is the angle between the vectors $(\xi-\eta)$ and $(\tau-\zeta)$ and thus $c_b^2 - c_ac_c \leq 0$ as claimed. □

In the sequel, we utilize the roots a_z of the SDQ to locate unsaturation. If the discriminant of the scalar quadratic SDQ is negative, then the two roots are imaginary and not of interest. For case where the discriminant is identically zero, then the quadratic roots are real and repeated. However, if $c_a = 0$ and $c_c \neq 0$, then $c_b = 0$ and the quadratic degenerates to the linear constant case $\mathcal{F}(a) = c_c < 0$. Similarly, if $c_c = 0$ and $c_a \neq 0$, then again $c_b = 0$ and the quadratic form is $\mathcal{F}(a) = c_a a^2 < 0$ whose roots are $a_z = 0$. Also, from Eq. (68), we see that the discriminant is always zero in the scalar case where $(\xi-\eta)$, $(\tau-\zeta) \in \Re$. However, when more than one effector is unsaturated help only select combinations of $\vec{s}_2$ and $\vec{s}_R$ with $\hat{u}_{\underline{2}}$ and $\hat{u}_R$ yield parallel vectors that produce a zero discriminant and thus real solutions.

Lemma 0.3. *Given the control effectiveness matrix $B \in \Re^{m\times n}$, then the total control allocations of Eqs. (50, 51) using $S_{\underline{2}}$ and S_2 respectively are identical if $\vec{m}$ is scaled such that the components $[\vec{u}_A]_R$ saturate identically to $[\vec{u}_B]_R = \vec{s}_R$.*

Proof. By inspection, we can see that the saturated components $[\vec{u}_A]_2$ of Eqs. (50) are identical to the saturated components of Eq. (51) where $[\vec{u}_B]_2 = \vec{s}_2$. The components of $[\vec{u}_A]_R$ are assumed to scale with $\vec{m}$ to be identically equivalent to $\vec{s}_R$, so specifically $\vec{s}_R = \hat{R}\hat{B}_2\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\vec{s}_2 + B_R^T\left(B_1B_1^T + B_RB_R^T\right)^{-1}\vec{m}$. Therefore, it remains to show that the unsaturated components of $[\vec{u}_A]_1$ and $[\vec{u}_B]_1$ are identical. Expanding these components of $\vec{u}_A$ and $\vec{u}_B$ yields:

$$[\vec{u}_A]_1 = \hat{B}_1\hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\vec{s}_2 + B_1^T\left(B_1B_1^T + B_RB_R^T\right)^{-1}\vec{m} \tag{69}$$

$$[\vec{u}_B]_1 = \hat{B}_1\left(I - \hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{B}_2\right)\left(\hat{R}^T\Delta^{-1}\vec{s}_R - \hat{R}^T\Delta^{-1}\hat{R}\hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\vec{s}_2\right) + \hat{B}_1\hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\vec{s}_2 + B_1^T\left(B_1B_1^T\right)^{-1}\vec{m} \tag{70}$$

Now substituting $\vec{s}_R = \hat{R}\hat{B}_2\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\vec{s}_2 + B_R^T\left(B_1B_1^T + B_RB_R^T\right)^{-1}\vec{m}$ into Eq. (70) yields:

$$[\vec{u}_B]_1 = \hat{B}_1\left(I - \hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{B}_2\right)\hat{R}^T\Delta^{-1}B_R^T\left(B_1B_1^T + B_RB_R^T\right)^{-1}\vec{m} + \hat{B}_1\hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\vec{s}_2 + B_1^T\left(B_1B_1^T\right)^{-1}\vec{m} \tag{71}$$

Now comparison of Eq.(69) to Eq. (71) shows that we only need show that:

$$B_1^T\left(B_1B_1^T + B_RB_R^T\right)^{-1} = \hat{B}_1\left(I - \hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{B}_2\right)\hat{R}^T\Delta^{-1}B_R^T\left(B_1B_1^T + B_RB_R^T\right)^{-1} + B_1^T\left(B_1B_1^T\right)^{-1} \tag{72}$$

Multiplying both sides by $\left(B_1B_1^T + B_RB_R^T\right)$ yields:

$$0 = \left(\hat{B}_1\left(I - \hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{B}_2\right)\hat{R}^T\Delta^{-1} + B_1^T\left(B_1B_1^T\right)^{-1}B_R\right)B_R^T \tag{73}$$

It is sufficient that $0 = \hat{B}_1\left(I - \hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{B}_2\right)\hat{R}^T\Delta^{-1} + B_1^T\left(B_1B_1^T\right)^{-1}B_R$ so multiplying by $\Delta > 0$ of Eq. (47) yields:

$$\hat{B}_1\left(I - \hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{B}_2\right)\hat{R}^TB_R^T + B_1^T\left(B_1B_1^T\right)^{-1}B_R\hat{R}\left(I - \hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{B}_2\right)\hat{R}^TB_R^T \tag{74}$$

$$= \left(\hat{B}_1 + B_1^T\left(B_1B_1^T\right)^{-1}B_R\hat{R}\right)\left(I - \hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{B}_2\right)\hat{R}^TB_R^T \tag{75}$$

Since $B\hat{B} = 0$, then $B_1\hat{B} + B_R\hat{R} + B_2\hat{B}_2 = 0$ and thus substituting for $B_R\hat{R}$ in Eq. (75) yields:

$$\left(I - B_1^T\left(B_1B_1^T\right)^{-1}B_1\right)\hat{B}_1\left(I - \hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{B}_2\right)\hat{R}^TB_R^T \tag{76}$$

Now recalling from linear algebra that $(\mathcal{R}(A))^{\perp} = \mathcal{N}\left(A^T\right)$, then from Eq. (76) we have

$$\mathcal{P}_{\left(B_1^T\right)^{\perp}} = \left(I - B_1^T\left(B_1B_1^T\right)^{-1}B_1\right) = \mathcal{P}_{\mathcal{N}(B_1)} = \hat{N}_1\left(\hat{N}_1^T\hat{N}_1\right)^{-1}\hat{N}_1^T \tag{77}$$

so Eq. (76) is identically:

$$\hat{N}_1 \left(\hat{N}_1^T \hat{N}_1\right)^{-1} \hat{N}_1^T \hat{B}_1 \left(I - \hat{B}_2^T \left(\hat{B}_2 \hat{B}_2^T\right)^{-1} \hat{B}_2\right) \vec{r}^T \vec{b}^T \tag{78}$$

$$= \hat{N}_1 \left(\hat{N}_1^T \hat{N}_1\right)^{-1} \hat{N}_1^T \hat{B}_1 \left(\hat{R}^T B_R^T - \hat{B}_2^T \left(\hat{B}_2 \hat{B}_2^T\right)^{-1} \hat{B}_2 \hat{R}^T B_R^T\right) \tag{79}$$

Also from linear algebra we can decompose B using Singular Value Decomposition (SVD) such that $B = U\Lambda V^T$ where $V = \begin{bmatrix} V_a & \hat{B} \end{bmatrix}$ where V_a is an orthonormal basis of the rowspace of B, with $VV^T = I$. Let $\hat{N}_1 = \mathcal{N}(B_1)$ then:

$$VV^T \begin{bmatrix} \hat{N}_1 \\ \vec{0} \end{bmatrix} = \begin{bmatrix} \hat{N}_1 \\ \vec{0} \end{bmatrix} = \left(V_a V_a^T + \hat{B}\hat{B}^T\right) \begin{bmatrix} \hat{N}_1 \\ \vec{0} \end{bmatrix} \tag{80}$$

Now from the SVD we know that $V_a^T \vec{x} = 0, \forall \vec{x} \in \mathcal{N}(B)$ and as $\begin{bmatrix} \hat{N}_1 \\ \vec{0} \end{bmatrix} \in \mathcal{N}(B)$, then Eq. (80) simplifies to:

$$\begin{bmatrix} \hat{N}_1 \\ \vec{0} \\ \vec{0} \end{bmatrix} = \hat{B}\hat{B}^T \begin{bmatrix} \hat{N}_1 \\ \vec{0} \end{bmatrix} = \begin{bmatrix} \hat{B}_1 \hat{B}_1^T & \hat{B}_1 \hat{R}^T & \hat{B}_1 \hat{B}_2^T \\ \hat{R}\hat{B}_1^T & \hat{R}\hat{R}^T & \hat{R}\hat{B}_2^T \\ \hat{B}_2 \hat{B}_1^T & \hat{B}_2 \hat{R}^T & \hat{B}_2 \hat{B}_2^T \end{bmatrix} \begin{bmatrix} \hat{N}_1 \\ \vec{0} \\ \vec{0} \end{bmatrix} \tag{81}$$

Examination of Eq. (81) shows that $\hat{R}\hat{B}_1^T \hat{N}_1 = \vec{0}$ and thus $\hat{N}_1^T \hat{B}_1 \hat{R}^T = \vec{0}$. Similarly $\hat{N}_1^T \hat{B}_1 \hat{B}_2^T = \vec{0}$ which show that Eq. (79) is identically zero, so $[\vec{u}_A]_1 = [\vec{u}_B]_1$ and thus $\vec{u}_A = \vec{u}_B$ as claimed. □

Lemma 0.4. *Given the control effectiveness matrix $B \in \Re^{m\times n}$, the total allocation solutions for the strict subset in Eq. (50) and the nominal set in Eq. (51), and the Scalar Difference Quadratic Eqs. (18 -21), then $\exists k \in \Re$ (scaling factor) such that scaling $\vec{m}$ results in $[\vec{u}_A]_R = [\vec{u}_B]_R$ is equivalent with the discriminant $c_b^2 - 4c_a c_c = 0$ with $c_a, c_c \neq 0$.*

Proof. First to show that if $\exists k \in \Re$ (scaling factor) such that scaling $\vec{m}$ results in $[\vec{u}_A]_R = [\vec{u}_B]_R$, then the discriminant $c_b^2 - 4c_a c_c = 0$ with $c_a, c_c \neq 0$. Given the strict subset $S_{\underline{2}}$ and the current saturated effector set S_2^i where $S_{\underline{2}} \subset S_2^i$ with the saturated limits identical such that $\vec{s}_{\underline{2}}(j) = \vec{s}_2^i(j)$, $\forall j \in S_{\underline{2}}$ then Eqs. (50, 51) show that $[\vec{u}_A]_2 = [\vec{u}_B]_2$. Now by assumption we have a scaling factor k on $\vec{m}$ such that $[\vec{u}_A]_R = [\vec{u}_B]_R$ which by Lemma 0.3 shows equality for all total allocation solution components such that $\vec{u}_A = \vec{u}_B$. Since we have a strict subset that yields identical moments at one point, then these solutions have identical null-space components and therefore there exists a real zero to the SDQ. However, Lemma 0.2 shows that the discriminant of the SDQ is negative definite. Since a negative discriminate yields two imaginary solutions, then the real solution implies that $c_b^2 - 4c_a c_c = 0$ as claimed.

Now to show that $c_b^2 - 4c_a c_c = 0$ with $c_a, c_c \neq 0$ implies $\exists k \in \Re$ (scaling factor) such that scaling $\vec{m}$ results in $[\vec{u}_A]_R = [\vec{u}_B]_R$. Assume that $c_b^2 - 4c_a c_c = 0$, then for the scalar quadratic function we have two repeated roots at $a_z = -c_b/2c_a$. We now show that a_z is the scalar factor we seek and moreover it is identically the scaling factor found during the Prediction Method iteration from Eq. (28). In order to determine the PM scaling factor for the saturated effector indices in S_2^i but not in $S_{\underline{2}}$, then in Eq. (28) we replace the upper and lower position limits with the limits sought $\vec{s}_R$ for those components and assembling them as a vector:

$$\vec{k}_R = \left(\frac{\vec{s}_R - \left[\vec{c}_0^i\right]_R}{\hat{u}_R - \left[\hat{c}_1^i\right]_R} \right),\ R = S_2^i \setminus S_{\underline{2}} \tag{82}$$

Then we use $\Delta^{-1/2}$ so we obtain:

$$\vec{k}_R = \frac{\Delta^{-1/2}}{\Delta^{-1/2}} \vec{k}_R = \left(\frac{\Delta^{-1/2}\left[\vec{c}_0^i\right]_R - \Delta^{-1/2}\vec{s}_R}{\Delta^{-1/2}\left[\hat{c}_1^i\right]_R - \Delta^{-1/2}\hat{u}_R} \right),\ R = S_2^i \setminus S_{\underline{2}} \tag{83}$$

Now noting from Eqs. (14, 15) that $\left[\vec{c}_0^i\right]_R = \hat{R}\hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \vec{s}_2$ and $\left[\hat{c}_1^i\right]_R = \hat{R}\hat{B}_2^T \left(\hat{B}_2\hat{B}_2^T\right)^{-1} \hat{u}_2$, then from Eqs.(61 -64):

$$\vec{k}_R = \left(\frac{\xi - \eta}{\tau - \zeta} \right),\ R = S_2^i \setminus S_{\underline{2}} \tag{84}$$

Now for the discriminant identically zero then from Lemma 0.2 we know that the vectors $(\xi - \eta)$ and $(\tau - \zeta)$ are parallel. Therefore we can utilize vector inner product to obtain the scalar gain k_R:

$$k_R = \frac{(\tau - \zeta)^T \Delta^{-1/2}}{(\tau - \zeta)^T \Delta^{-1/2}} \vec{k}_R = \frac{(\tau - \zeta)^T (\xi - \eta)}{(\tau - \zeta)^T (\tau - \zeta)} = \frac{-c_b}{2c_a} \tag{85}$$

As the scaling we seek is non-zero, then we require $(\xi - \eta)$, $(\tau - \zeta) \neq 0$ which by Eqs. (65, 67) implies $c_a, c_c \neq 0$. Equation (85) shows that $k_R = a_z = -c_b/(2c_a)$, but the PM scaling factor k_R saturates the R effectors at their position limits and therefore from Eqs. (50, 51) we have that $k_R\hat{u}_{des} = P_{min}\vec{m}$, which implies that $\vec{m} = k_R B\hat{u}_{des}$ such that $[\vec{u}_A]_R = [\vec{u}_B]_R$. Thus $c_b^2 - 4c_a c_c = 0$ with $c_a, c_c \neq 0$ implies $\exists k \in \Re$ (scaling factor) such that scaling $\vec{m}$ results in $[\vec{u}_A]_R = [\vec{u}_B]_R$. Therefore $c_b^2 - 4c_a c_c = 0$ with $c_a, c_c \neq 0$ is equivalent to $\exists k \in \Re$ where scaling $\vec{m}$ results in $[\vec{u}_A]_R = [\vec{u}_B]_R$ as claimed. □

Theorem 1. *Given a control effectiveness matrix $B \in \Re^{n\times m}$, the Moore-Penrose generalized inverse $P_{min} = B^T \left(BB^T\right)^{-1}$, a desired moment $\vec{m}_{des} \in \Phi$, and the Scalar Difference Quadratic of Eqs. (30 -32), then the moment direction preserving optimal (2-norm minimal) allocation solution:*

$$\vec{u}_{opt} = \underset{\vec{u}}{\text{argmin}} \left\{ \vec{u}^T \vec{u} \mid \vec{m}_{des} = B\vec{u},\ \vec{u} \in \Omega \right\}$$

is achieved for all moments continuously along $\hat{m}_{des} = \vec{m}_{des}/\|\vec{m}_{des}\|_2$ *up to and including* $\vec{m}_{des}$ *using the Prediction Method augmented with unsaturation identification and location for all iterations where greater than one but less than or equal to* $(m-n)$ *effectors are saturated. Unsaturation is identified in the current linear segment* $\left[\vec{m}_{a_i}, \vec{m}_{a_{i+1}}\right]$ *along* $\vec{m}_{des}$ *given the current set of saturated effector indices* S_2^i *if*

$$a' = \underset{a_z}{\operatorname{argmin}} \left\{ a_z = -c_b/2c_a \in (a_i, a_{i+1}] \mid c_b^2 - 4c_a c_c = 0,\ c_b \neq 0,\ S_{\underline{2}} \subset S_2^i \right\} \tag{86}$$

is a non-empty set and the unsaturation occurs at a'*. Unsaturation detection requires updating the current iteration's linear segment from* $[a_i, a_{i+1}]$ *to* $[a_i, a']$ *where* $a\hat{u}_{des} = P_{min}\vec{m}_a$ *with* $\hat{u}_{des} = (P_{min}\vec{m}_{des})\,\|P_{min}\vec{m}_{des}\|_2$.

Proof. From linear algebra, $\Re^m$ can be divided into two orthogonal subspaces such that $\Re^m = \Re^{m\|} + \Re^{m\perp}$ where $\Re^{m\|}$ is the span of the columns of the Moore-Penrose generalized inverse $P_{min} = B^T \left(BB^T\right)^{-1}$ and $\Re^{m\perp} = \mathcal{N}(B)$. The control allocation $\vec{u}^{\|} = P_{min}\vec{m}_{des} \in \Re^{m\|}$ minimally yields $\vec{m}_{des}$, but $\vec{u}^{\|} \notin \Omega$, $\forall \vec{m}_{des} \in \Phi$. As the subspaces are orthogonal, we have $\left\|\vec{u}_{opt}\right\|_2^2 = \left\|\vec{u}^{\|}\right\|_2^2 + \|\vec{u}^{\perp}\|_2^2$ and therefore to minimize $\vec{u}_{opt}$ such that $B\vec{u}_{opt} = \vec{m}_{des}$, it is necessary and sufficient to minimize $\vec{u}^{\perp}$ such that $\vec{u}_{opt} = \left(\vec{u}^{\|} + \vec{u}^{\perp}\right) \in \Omega$

In [17], the PM finds piecewise linear and continuous control allocation segments (not necessarily optimal) for moments $\vec{m}_a$ along $\hat{m}_{des}$ up to an including $\vec{m}_{des}$. For rectilinear effector position constraints, the PM yields optimal solutions $\vec{u}_a$ for the linear segment $a \in [a_i, a_{i+1}]$ if the boundary object S_2^i gives the minimal $\vec{u}_a^{\perp}$ that returns $\vec{u}_a^{\|}$ to the boundary of Ω [17]. An inductive proof is utilized for the PM augmented with the SDQ to ensure that this correct boundary object is used. We show that given the known minimal set S_2^i at the start of the i^{th} segment $a \in [a_i, a_{i+1}]$, then use of the SDQ on a small subset of possible boundary transition objects ensures optimal allocation $\vec{u}_a = \vec{u}_a^{\|} + \vec{u}_a^{\perp} \in \Omega$ for the entire segment, and thus the next iteration starts with the known minimal set S_2^{i+1} at a_{i+1}. The inductive proof is completed as the first iteration gives the minimal boundary set at a_1 from the minimal total allocation $\vec{u}_a = \vec{u}_a^{\|}$ and $\vec{u}_a^{\perp} = \vec{0}$, $\forall a \in [0, a_1]$. For the inductive proof, we seek the transition set (if it exists) that maintains the minimal total allocation magnitude in the current segment $a \in (a_i, a_{i+1}]$. This necessitates two requirements that all candidate minimal transition sets must satisfy. First for continuity, the current set S_2^i and the transition set S_2 must have identical allocation solutions at some point $a' \in (a_i, a_{i+1}]$. Second, it requires the total allocation solution for S_2 is smaller than that obtained with S_2^i after transition (i.e., $a > a'$).

We note that the saturated components of the total allocation $\left[\vec{u}_{S_2^i}\right]_2 = \vec{s}_2^{\,i}$ as seen from Eqs. (14, 15, 17), where $[]_2$ denotes the saturated effector vector components:

$$\left[\vec{u}_{S_2^i}\right]_2 = \left(\left[\vec{c}_0^{\,i}\right]_2 + a\left[\hat{c}_1^{\,i}\right]_2\right) + a\left[\hat{u}_{des}\right]_2 = \left(\hat{B}_2\hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\vec{s}_2^{\,i} - a\hat{B}_2\hat{B}_2^T\left(\hat{B}_2\hat{B}_2^T\right)^{-1}\hat{u}_2\right) + a\hat{u}_2 = \vec{s}_2^{\,i} \tag{87}$$

Equation (87) also holds for transition sets, which implies either $S_{\underline{2}} \subset S_2^i$ or $S_2^i \subset S_{\overline{2}}$ (underbar and overbar denote strict subsets and supersets respectively), as transition subsets and supersets satisfy a necessary continuity condition that:

$$\vec{s}_{\underline{2}}(j) = \vec{s}^{i}_{2}(j),\ \forall j \in S_{\underline{2}} \text{ and } S_{\underline{2}} \subset S^{i}_{2} \text{ or} \tag{88}$$

$$\vec{s}^{i}_{2}(j) = \vec{s}_{\overline{2}}(j),\ \forall j \in S^{i}_{2} \text{ and } S^{i}_{2} \subset S_{\overline{2}} \tag{89}$$

We can eliminate supersets as viable transition sets by restating the SDQ $\mathcal{F}(a) = \|\vec{c}_0 + \vec{c}_1\|_2^2 - \left\|\vec{c}_{\overline{0}} + \vec{c}_{\overline{1}}\right\|_2^2$. Lemma 0.2 shows quadratic curvature $c_a \leq 0$ and thus the superset null space components $\left\|\vec{c}_{\overline{0}} + \vec{c}_{\overline{1}}\right\|_2^2$ are identical or larger than the baseline components $\|\vec{c}_0 + \vec{c}_1\|_2^2$ from S^i_2, which fails to obtain smaller total allocations within the segment.

Restricting our candidate transition set search to strict subsets $S_{\underline{2}}$, we define the total control allocation solution $\vec{u}_A$ for the strict subset $S_{\underline{2}}$ in Eq. (48) and $\vec{u}_B$ for the current set S^i_2 in Eq. (49). We claim that for the current iteration with S^i_2, then the transition set $S_{\underline{2}} \subset S^i_2$ satisfies both unsaturation requirements if the zero of the SDQ $a_z = -c_b/(2c_a) \in (a_i, a_{i+1}]$ and $c_b^2 - ac_ac_c = 0$ with $c_a, c_c \neq 0$. By Lemma 0.1, the strict subset allocation $\vec{u}_A$ achieves the identical desired moment $\vec{m}_a$, $\forall a \in [a_i, a_{i+1}]$ as $\vec{u}_B$ and therefore both sets have identical $\vec{u}^{\parallel}_a$ allocations. Now comparing null-space allocation components, then by Lemma 0.4 we have a scaling factor such that $[\vec{u}_A]_R = [\vec{u}_B]_R$, $R = S^i_2 \setminus S_{\underline{2}}$. Now by Lemma 0.3, we also have $[\vec{u}_A]_1 = [\vec{u}_B]_1$ and therefore we have continuity of allocations at $a_z \in (a_i, a_{i+1}]$. Moreover, since Lemma 0.2 shows that $c_a \leq 0$, which together with $c_a \neq 0$ implies that $c_a < 0$. As the quadratic curvature is negative, then $\left\|\vec{u}^{\perp}_{A_a}\right\|_2 < \left\|\vec{u}^{\perp}_{B_a}\right\|_2$, $\forall a > (a_z, a_{i+1}]$ and thus the transition set achieves a smaller total magnitude for $a > (a_z, a_{i+1}]$, so indeed both transition requirements are met. We seek the strict subset that yields the smallest null space allocation component or equivalently unsaturates first corresponding with the subset that satisfies Eq. (86). If a' from Eq. (86) does not exist, then the set S^i_2 yields minimal total allocation solutions for $a \in [a_i, a_{i+1}]$ and the PM resumes iterating with the known minimal set S^{i+1}_2 at $a = a_{i+1}$ using the nominal PM method. If the subset is found satisfying Eq. (86), then we shorten the current linear segment to $[a_i, a']$ for S^i_2. This ensures optimal allocation solutions for $a \in [a_i, a']$ and restarts the PM iteration using the known minimal set $S'_2 \subset S^i_2$, which minimized Eq. (86) at a'. We know the first PM iteration obtains optimal allocation solutions for $a \in [0, a_1]$ and the minimal set S^1_2 from Eq. (10) with a null space component identically zero at $a = a_1$. Thus the inductive proof holds for iterations $i \leq (m - n)$. Therefore we are guaranteed optimal allocation solutions for all moments along $\hat{m}_{des}$ from the origin to the moment generated by the largest unit vector gain a obtained. Iterations are performed until the desired moment $\vec{m}_{des}$ is obtained in one of the resulting optimal allocation segments.

However, $\vec{m}_{des}$ might be near or identically the maximal moment on the boundary of the AMS. While it is known that all facets on the boundary of the AMS consist of objects with $(m - n)$ saturated effectors, that does not imply that the final object found using the SDQ augmented PM yielded the object on the boundary of the AMS. Therefore, after the final iteration and unsaturation detection was performed for $(m - n)$ saturated effectors, we continue to run SDQ unsaturation detection in accordance with Eq. (86) which checks other boundary objects with $(m - n)$ effectors saturated that preserve moment direction. If unsaturation is detected where $a' \in (a_i, a_{i+1}]$, then we compute the new segment $a \in \left[a', a^{'+1}\right]$ where $a^{'+1}$ is obtained using the PM method but we retain the new segment only if it yields a larger moment magnitude or equivalently $a_{i+1} < a^{'+1}$. This unsaturation check (which also provides for increasing moment magnitude) is performed until moment magnitude cannot be increased which ensures this object yields optimal allocation up to the maximum moment $\vec{m}_{des} \in \delta(\Omega)$. □

Acknowledgements

This work was supported with funding from NASA Transformation Tools and Technologies (TTT), Intelligent Contingency Management (ICM) subproject.

References

[1] Denham, J., "Project MAGIC CARPET: Advanced Controls and Displays for Precision Carrier Landings," *AIAA SciTech Forum*, AIAA Paper 2016-1770, 2016. doi:10.2514/6.2016-1770.

[2] Durham, W., Bolling, J., and Bordignon, K., "Minimum drag control allocation," AIAA Paper 1996-3410, 1996. doi:10.2514/6.1996-3410.

[3] Bordignon, K., and Bessolo, J., "Control Allocation For The X-35B," *2002 Biennial International Powered Lift Conference and Exhibit*, AIAA Paper 2002-6020, 2002. doi:10.2514/6.2002-6020.

[4] Davidson, J., Lallman, F., and Bundick, W. T., "Integrated Reconfigurable Control Allocation," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, AIAA Paper 2001-4083, 2001. doi:10.2514/6.2001-4083.

[5] Frost, S., Bodson, M., Burken, J., Jutte, C., Taylor, B., and Trinh, K., "Flight Control with Optimal Control Allocation Incorporating Structural Load Feedback," *Journal of Aerospace Information Systems*, Vol. 12, No. 12, 2015, pp. 825–834. doi:10.2514/1.I010278.

[6] Durham, W., "Constrained Control Allocation," *Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 4, 1994, pp. 717, 725. doi:10.2514/3.21201.

[7] Durham, W., Bordignon, K., and Beck, R., *Aircraft Control Allocation*, AIAA Aerospace Series, Wiley, West Sussex, 2017.

[8] Shi, J., Zhang, W., Li, G., and Liu, X., "Research on allocation efficiency of the redistributed pseudo inverse algorithm," *Science China Information Sciences*, Vol. 53, 2010, pp. 271–277. doi:10.1007/s11432-010-0032-x.

[9] Buffington, J., and Enns, D., "Lyapunov Stability Analysis of Daisy Chain Control Allocation," *Journal of Guidance, Control, and Dynamics*, Vol. 19, 1996, pp. 1226–1230. doi:10.2514/3.21776.

[10] Bordignon, K., *Constrained Control Allocation for Systems with Redundant Control Effectors*, Ph.D. Dissertation, Virginia Polytechnic Institute and State University, Blacksburg, VA, 1996.

[11] Baratcart, T., Slvucci, V., and Koseki, T., "On the Continuity of Cascaded Generalized Inverse Redundancy Resolution, with Application to Kinematically Redundant Manipulators," *39th Annual Conference of the IEEE Industrial Electronics Society*, IEEE, 2013. doi:10.1109/IECON.2013.6700208.

[12] Johannes, S., and Fichter, W., "Fast Exact Redistributed Pseudoinverse Method for Linear Actuation Systems," *Transactions on Control Systems Technology*, Vol. 47, 2019, pp. 451–458. doi:10.1109/TCST.2017.2765622.

[13] Bordignon, K., and Durham, W., "Null-Space Augmented Solutions to Constrained Control Allocation Problems," *Guidance, Navigation, and Control Conference*, AIAA Paper 1995-3209, 1995. doi:10.2514/6.1995-3209.

[14] Bozorgnia, D., and Sedigh, A. K., "A new control allocation methodology based on the pseudo inverse along the null space," *21st Iranian Conference on Electrical Engineering*, ICEE, 2013. doi:10.1109/IranianCEE.2013.6599534.

[15] Tohidi, S., Sedigh, A. K., and Buzorgnia, D., "Fault Tolerant Control Design using adaptive control allocation based on the pseudo inverse along the null space," *International Journal of Robust and Nonlinear Control*, Vol. 26, 2016, pp. 3541–3557. doi:10.1002/rnc.3518.

[16] Xiuwei, H., and Guangren, D., "A developed constrained control allocation approach based on pseudo inverse," *2017 Chinese Automation Congress(CAC)*, IEEE, 2017. doi:10.1109/CAC.2017.8242762.

[17] Acheson, M., "Optimal Control Prediction Method for Control Allocation," *AIAA Aviation and Aeronautics Forum and Exposition (AIAA AVIATION 2018)*, AIAA Paper 2018-3897, 2018. doi:10.2514/6.2018-3897.

[18] Acheson, M., "Affine Generalized Inverse for Optimal Control Allocation," *AIAA Aviation and Aeronautics Forum and Exposition (AIAA AVIATION 2018)*, AIAA Paper 2018-3902, 2018. doi:10.2514/6.2018-3902.

[19] Acheson, M., "Computational Efficiency Enhancements for Optimal Control Allocation," *AIAA Aviation and Aeronautics Forum and Exposition (AIAA AVIATION 2019)*, AIAA Paper 2019-3462, 2019. doi:10.2514/6.2019-3462.

[20] Acheson, M., "Control Effector Unsaturation Modification to the Cascading Generalized Inverse Control Allocation Algorithm," *AIAA Science and Technology Forum and Exposition (AIAA SCITECH 2020)*, AIAA Paper 2020-0853, 2020. doi:10.2514/6.2020-0853.

[21] Glaze, M., and Durham, W., "Design and Implementation of a Control Allocation Toolbox for Matlab V5.0," AIAA Paper 1998-4436, 1998. doi:10.2514/6.1998-4436.