Proceedings of the 18th World Congress
The International Federation of Automatic Control
Milano (Italy) August 28 - September 2, 2011

Reconfigurable direct control allocation for overactuated systems

Max Demenkov *

* *Institute of Control Sciences, Russian Academy of Sciences,
Profsoyuznaya str. 65, Moscow 117 997, Russia
(phone/fax: +7(495)334 93 69, e-mail: demenkov@ipu.ru,
max.demenkov@gmail.com)*

Abstract: Control allocation deals with distributing the total virtual control input among the individual physical actuators in overactuated systems. We propose a new real-time solution for direct allocation problem, where a linear relationship exists between the virtual three-dimensional control vector and the actual constrained control inputs vector of a bigger size. The proposed procedure is based on zonotope properties and is suitable for on-line reconfiguration in case of actuators faults. The resulting vector of actual system inputs can be computed analogously to the determination of control in polyhedral Lyapunov function method.

Keywords: control allocation, control constraints, fault tolerant control, polyhedral functions

1. INTRODUCTION

Control allocation is a set of methods for control of overactuated (usually mechanical) systems, and deals with distributing the total control demand among the individual actuators. Using control allocation, the actuator selection task is separated from the regulation task in the control design.

The idea of control allocation allows to deal with control constraints and actuator faults separately from the design of the main regulator, which uses virtual unconstrained control input. In many cases, this input consists of three moments or angular rates used to control an aircraft, a car, etc. It is supposed that actual number of physical controls is greater than that of virtual ones. Also, an important issue in control allocation is the ability of fault tolerant reconfiguration. In case of fault in an actuator, instead of reconfiguring the main control law, we change only the distribution of the virtual control input among other vehicle actuators which are still in use.

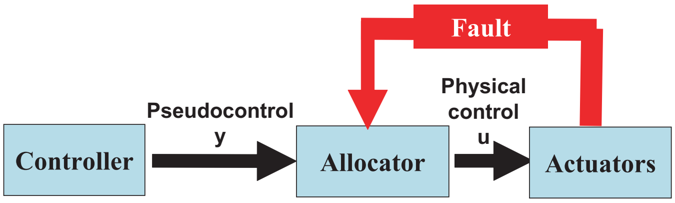

Fig. 1. Control allocation principle

Constrained control allocation problem has been attracting attention for around 17 years since the first method of this type - the so called *direct allocation* approach (Durham (1993, 1994)). For many years, control allocation has been studied almost solely within the aeronautical community (see e.g. Bodson (2002), Petersen and Bodson (2002), Doman and Oppenheimer (2002), Doman and Ngo (2002)), but in the last decade control allocation was applied to control of advanced cars (e.g. Tondel and Johansen (2005), Laine and Fredriksson (2008), Plumlee et al. (2004), Schofield and Hagglund (2008)) as well as marine vessels (e.g. Fossen et al. (2009), Spjtvold and Johansen (2009)), attitude control of satellites (e.g. Pulecchi and Lovera (2007), Thieuw and Marcille (2007), Lee et al. (2007)) and redundant robotic manipulators (e.g. Pechev (2008)). It is not now only application-specific, but can be considered as general control theory problem (Harkegard and Glad (2005), Tjonnas and Johansen (2008)).

The dependence between virtual control vector and the actual constrained control inputs can be considered as linear or nonlinear. In this paper, we consider only linear constrained problem. Moreover, we intend to improve the first method of its kind - i.e. direct allocation by Durham.

Despite the fact that control allocation started with Durham's method, it is frequently considered as too complex for application to real problems. In order to implement it, one have to construct a polytope that is a representation of attainable set of virtual control vectors. After that, to find actual control vector, one have to search among all facets of that polytope and then use some procedure to find actual controls. These procedures were not very effective from the beginning, despite some attempts to improve them in the following publications (Durham (1999), Petersen and Bodson (2002)). In this paper, we want to show that both the polytope construction and the search procedure can be done efficiently in real-time, and, moreover, suitable search and control generation method has been already proposed in control literature for the implementation of polyhedral Lyapunov functions method (Blanchini and Miani (2008), Blanchini (1995)).

Let us consider the following general model of a controlled dynamical system with 3-dimensional virtual input y:

$$\dot{x} = f(x,u), y = Bu + y_f, y = k(x). \qquad (1)$$

Here we use an assumption that a linear relationship exists between the virtual control vector $y \in R^3$ and the actual control vector $u \in R^m,\ m > 3$, while $B = [b_1|b_2|...|b_m]$ is the control effectiveness matrix having m columns.

978-3-902661-93-7/11/$20.00 © 2011 IFAC

10.3182/20110828-6-IT-1002.03790

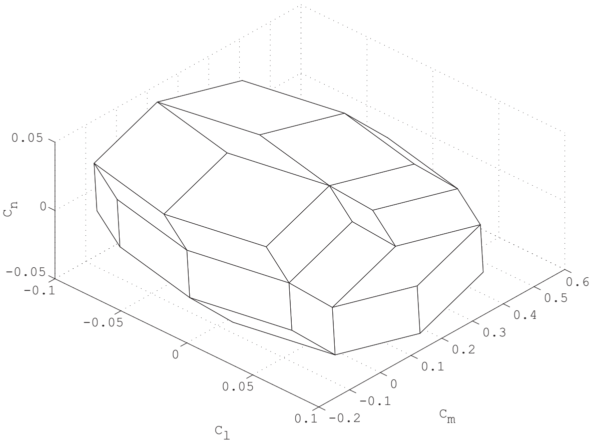

Fig. 2. An example of attainable virtual control set for the F-18 fighter aircraft

The fixed dimensionality of our problem stems from the fact that it usually represents vector of moments or moment rates, which are used to control air, ground or marine vehicles with redundant controls.

Note that feedback control law $y = k(x)$ is computed first and then we use it to determine actual control input u. Vector y_f contains faulty control inputs (we suppose that we know their actual positions) multiplied by the corresponding columns from the previous (non-faulty) B matrix. Therefore, in the sequel we can easily take $y_f = 0$ as we can always form $y_{new} = y - y_f$.

The problem is as follows: find u for a given y and B, which can change with every start of the algorithm. The simplest solution would be to apply Moore-Penrose pseudo-inverse matrix, but in this case not all possible control actions could be realized, because the control effectors are supposed to be limited by some minimal and maximal values:

$$u \in U,\ U = \{u \in R^m :\ u_{min}^{(i)} \le u^{(i)} \le u_{max}^{(i)}, i = \overline{1,m}\} \tag{2}$$

where $u^{(i)}$ denotes i-th component of the vector.

The achievable virtual control vectors are then confined to some *attainable virtual control set* (AVS) Y:

$$Y(U) = \{y : y = Bu, u \in U\} \tag{3}$$

The on-board implementation of a control allocation algorithm needs to be computationally effective and should always converge to a solution. Therefore, many optimal methods that might be easily applicable and reconfigurable off-line, like linear or quadratic programming (see e.g. Bodson (2002), Casavola and Garone (2007)), may not constitute a reasonable engineering solution to the problem, as pointed out e.g. in Cameron and Princen (2000). Recently discovered methods that are based on approximations or explicit representations of mathematical programming solutions (e.g. Johansen et al. (2005)) are indeed computationally effective, but lack in reconfiguration capability — in the case of an effector failure the amount of re-computations is too big.

Since Durham (1993, 1994) it is known that AVS in our case is a three-dimensional polytope (see Fig. 2). In case we have a representation of AVS in the form of a collection of non-overlapping convex cones, from Blanchini and Miani (2008), Blanchini (1995) one can establish a procedure for easy detection of the corresponding cone (i.e. the one in which virtual control vector lies) and following computation of vector u using a number of solutions of linear equations. This is done analogously to their polyhedral Lyapunov function method. The goal of this paper is to describe a relatively easy procedure of AVS construction in the form of facet normals from the given matrix B and control constraints (Demenkov (2008, 2007)) and the following determination of actual controls. The resulting procedure, based on the properties of special kind of polytopes called *zonotopes*, is of low complexity and can be recommended for on-line reconfiguration in fault-tolerant systems.

The drawback of the proposed algorithm is that it does not optimize the solution using any additional performance criteria (this is actually a direct consequence of direct allocation problem statement in general). In any case, it might be considered as a last resort in a system usually governed by some other (optimal) control allocation algorithm, to keep system functioning in situations where safety comes first after an actuator or optimal allocation algorithm failure.

2. PROPERTIES OF THE ATTAINABLE VIRTUAL CONTROL SET

In Ziegler (1995) we can find the definition of a *zonotope*: it is the image of a cube under an affine projection. Zonotopes are special polytopes. We recall that a polytope can be described both by the set of its vertices and by a system of linear inequalities.

From (3) we conclude that our AVS is exactly a zonotope. This fact has been first established in Durham (1994) and properties of the attainable set has been described there for 2D and 3D case.

Suppose that the barycentre of the control constraints hyperparallelepiped is computed:

$$u_0 = \frac{1}{2}(u_{min} + u_{max}),$$

then we can rewrite equation $y = Bu$ as follows:

$$y = y_0 + Zv, \tag{4}$$

where $v^{(i)} = (u^{(i)} - u_0^{(i)})/(u_{max}^{(i)} - u_{min}^{(i)})$- normalized actual control vector, $y_0 = Bu_0$ and $Z = [z_1|z_2|...|z_m]$ with columns $z_i = b_i(u_{max}^{(i)} - u_{min}^{(i)})$. Note that v is constrained now by m-cube: $||v||_\infty \le 1$.

Definition 1. A zonotope is the image of a cube under an affine projection, that is, a polytope $Y \subseteq R^n$ of the form

$$Y = \{y \in R^n : y = y_0 + \sum_{i=1}^{m} z_i v^{(i)}, -1 \le v^{(i)} \le 1\}$$

for some matrix $Z = [z_1|z_2|...|z_m] \in R^{n \times m}$.

Without loss of generality, in the sequel we suppose $y_0 = 0$ in the last definition.

The following three basic theorems can be extracted from literature on convex polytopes (see e.g. McMullen and Shepard (1971), Ziegler (1995)):

Theorem 1. Let y_i, $i = \overline{1,N}$ be vertices of a polytope Y in the Euclidean space. Any point $y \in Y$ can be written in the form

$$y = \sum_{i=1}^{N} \lambda_i y_i, \quad \lambda_i \ge 0, \quad \sum_{i=1}^{N} \lambda_i = 1,$$

that is, a convex combination of y_i.

This theorem has its dual one:

Theorem 2. Let $\{y_i\}_{i=\overline{1,N}}$ be any set of points in the Euclidean space and

$$y = \sum_{i=1}^{N} \lambda_i y_i, \quad \lambda_i \geq 0, \quad \sum_{i=1}^{N} \lambda_i = 1,$$

that is, a convex combination of y_i. Then $y \in Y$ where Y is a polytope with vertices from the set of points y_i.

This means that some but not all vertices of the polytope Y are from this set, while other y_i are situated inside Y or on its surface, being redundant for the polytope representation.

Consider now v as a convex combination of some vertices v_i, $i = \overline{1,M}$. Such representation always exists because v is limited to the hypercube H in R^m:

$$H = \{v \in R^m : ||v||_\infty \leq 1\}. \tag{5}$$

The vertices of this hypercube contain only extremal values in its components, i.e.

$$v_i^{(j)} = \pm 1, j = \overline{1,m}, i = \overline{1,M},$$

and we have

Theorem 3. The components of v corresponding to a vertex Zv of Y have only extremal values.

Now we are ready to introduce the main theorem of this section (Demenkov (2008)):

Theorem 4. Suppose $n = 3$. Then any normal vector of a facet of the polytope Y is a scaled cross product of some two columns taken from the matrix Z .

Proof: Consider the following optimization task:

$$d^T y \to \max, \\ y \in Y,$$

where d is a normal vector of the polytope facet. From linear programming theory, it is well known that the problem solution set will always include all vertices that are incident to this facet Dantzig (1966). Consider only these vertices as the solutions to the problem. We can rewrite the goal function as a function of the variable v :

$$f(v) = d^T y(v) = \sum_{j=1}^{m} d^T z_j v^{(j)},$$

and then our optimization task is

$$f(v) \to \max, \\ v \in H.$$

Note that

$$f(v) \leq \sum_{j=1}^{m} \sup_{v}(d^T z_j v^{(j)}) = \sum_{j=1}^{m} d^T z_j v_{ext}^{(j)} = f(v_{ext}),$$

where $v_{ext}^{(j)}$ has the form:

$$v_{ext}^{(j)} = \text{sign}(d^T z_j), \text{ if } d^T z_j \neq 0; \\ v_{ext}^{(j)} \in [-1, \ 1], \text{ if } d^T z_j = 0.$$

It is clear that since

$$f(v) \leq f(v_{ext}),$$

and varying any $v_{ext}^{(j)}$ in the case of $d^T z_j = 0$ does not change the value of $f(v_{ext})$, these v_{ext} are the solutions to our optimization task.

From Theorem 3, it is clear that since we consider only those v_{ext} that correspond to the vertices of Y, these v_{ext} have the form:

$$v_{ext}^{(j)} = \begin{cases} \text{sign}(d^T z_j), & d^T z_j \neq 0; \\ \pm 1, & d^T z_j = 0. \end{cases} \tag{6}$$

The number N_0 of zero entries in the $d^T Z$ row vector defines the number of vertices N_v, which are the solution to our optimization task:

$$N_v = 2^{N_0}.$$

Recall that a facet of a polytope in 3D space contains at least three vertices. It is clear that in the case of no zeros in the row vector $d^T Z$, only one vertex serves as a solution and therefore d is not a facet normal vector. One zero in $d^T Z$ corresponds to only two vertices and d still cannot be a normal vector of a facet. In the case of two zeros in $d^T Z$ we will have d, which is uniquely defined as the vector orthogonal to at least two columns of Z. And it is now a normal vector of a facet, because it corresponds to at least four vertices of Y.□

It is worth looking now at Fig. 3 in order to understand the principles of the proof.

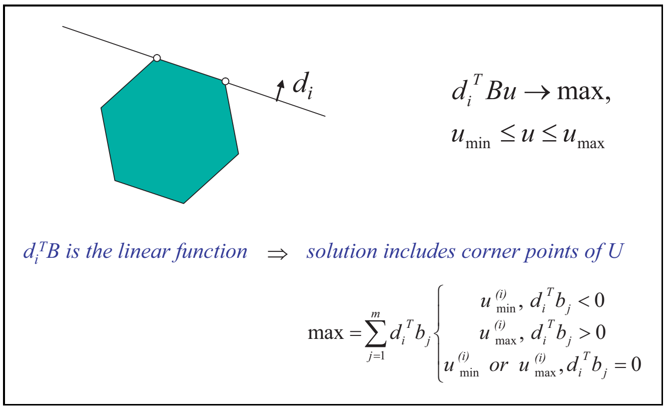
d_i
$d_i^T Bu \to \max,$
$u_{\min} \leq u \leq u_{\max}$
$d_i^T B$ is the linear function $\Rightarrow$ solution includes corner points of U
$\max = \sum_{j=1}^{m} d_i^T b_j \begin{cases} u_{\min}^{(i)}, d_i^T b_j < 0 \\ u_{\max}^{(i)}, d_i^T b_j > 0 \\ u_{\min}^{(i)} \ or \ u_{\max}^{(i)}, d_i^T b_j = 0 \end{cases}$

Fig. 3. Supporting picture for the proof of Theorem 4

3. DETERMINING ACTUAL CONTROLS

As it was established in the last section, any normal vector d of a polytope facet is computed as the cross product of two different columns z_i and z_j taken from the matrix Z:

$$d^{(1)} = z_i^{(2)} z_j^{(3)} - z_i^{(3)} z_j^{(2)}, \\ d^{(2)} = z_i^{(3)} z_j^{(1)} - z_i^{(1)} z_j^{(3)}, \\ d^{(3)} = z_i^{(1)} z_j^{(2)} - z_i^{(2)} z_j^{(1)}. \tag{7}$$

Note that for any facet normal vector d there exists its opposite in sign vector $-d$, which is defined as the normal vector of the opposite facet. Because of this, our polytope is a symmetric one.

Suppose that we have all columns of Z in a list and determine all pairs of one column z_i and any other column of Z from the list; the number of such pairs is $m-1$ and any pair gives us two valid facets of the polytope. In the next step, we have to remove this z_i from the list and repeat the procedure (now the number of pairs is $m-2$). Proceed the same way until we have at least two columns. The maximum number of facets is therefore less than $2m^2$.

Note that different couples of columns produce normal vectors in the same direction in space and for some d the solution of optimization task is achieved on a vertex. The determination of redundant hyperplanes (which are not facets) is easy - they have less than 3 incident vertices.

The procedure of determining the incident vertices for a given vector d can be easily extracted from the proof of Theorem 4 (look at equation (6)).

It can be seen in Fig. 4 that d_i is a true normal vector to a facet, while d_j is redundant.

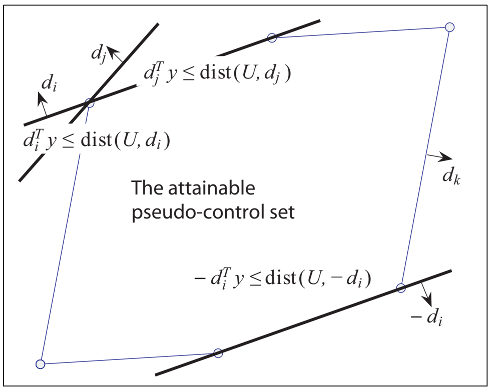
d_j
d_i
$d_j^T y \le \mathrm{dist}(U, d_j)$
$d_i^T y \le \mathrm{dist}(U, d_i)$
The attainable
pseudo-control set
d_k
$-d_i^T y \le \mathrm{dist}(U, -d_i)$
$-d_i$

Fig. 4. AVS represented by a set of linear inequalities.

Suppose we have a set $\{d_k\}_{k=\overline{1,M}}$ containing normals for each facet of Y. Suppose also that they are normalized, i.e. $d_k = d_k/dist(U, d_k)$, where

$$dist(U, d_k) = \sum_{j=1}^{m} d_k^T z_j v_{ext}^{(j)} \tag{8}$$

as in Theorem 4.

Each polytope facet represented by d_k has its convex cone (it is bounded by rays passing through vertices, which are incident to the corresponding facet). We can determine the number k of convex cone containing vector y as follows (Blanchini and Miani (2008), Blanchini (1995)):

$$k = \arg \max_{k=\overline{1,M}} d_k^T y. \tag{9}$$

If $d_k^T y \geq 1$, we need to scale y down to the surface of Y: $y = y/d_k^T y$. This is standard feature of Durham's direct allocation, as we cannot allocate outside AVS due to control constraints.

From the given d_k, we can easily determine all vertices v_i of the hypercube H, which correspond to points Zv_i lying in the particular facet of Y. All facet-incident vertices of the polytope Y are contained in this set of points, but not only those. Generally, some points Zv_i exist, which are *not* related to the vertices of Y but are placed on the same facet.

Nevertheless, for k-facet of Y, we can create a list $L_k = \{y_i\}_{i=\overline{1,N_k}}$ of those points $y_i = Zv_i$ that are incident to this facet.

We can split the convex cone corresponding to that facet into tetrahedra having exactly four vertices with one of them at the origin (we suppose that the origin lies not on the polytope boundary, this is always the case while we consider the polytope Y here). The number of such tetrahedra is clearly finite (see Fig. 5). We can establish all of them by simply enumerating all possible triples of points from our list.

Then, one can find a solution to the following linear equation system:

$$\lambda_1 y_i + \lambda_2 y_j + \lambda_3 y_k = y, \tag{10}$$

for each triple $\{v_i, v_j, v_k\}$ taken from the list L_k (note that $y_i = Zv_i$).

Analogously to Blanchini and Miani (2008), Blanchini (1995), if $\lambda_{1,2,3} \geq 0$ then y lies in the simplex defined by the triple $\{y_i, y_j, y_k\}$ and we can take

$$v = \lambda_1 v_i + \lambda_2 v_j + \lambda_3 v_k \tag{11}$$

as the corresponding actual control vector (normalized). Note that we can restore u from v: $u^{(i)} = v^{(i)}(u_{max}^{(i)} - u_{min}^{(i)}) + u_0^{(i)}$ (see the beginning of Section 2).

It is quite possible that we will have more than one tetrahedron constructed in this way and containing y. In this case, it would be possible to choose one of those tetrahedra that gives control vector which is closest to the previous one (we suppose that upper-level control values $y = k(x)$ vary as continuous functions).

Now, let us summarize the algorithm as follows:

Input: B, y and previous actual control vector v_{last}

- Compute Z in (4)
- Find all facet normals d_k from (7)
- Compute distances as in (8) and normalize all d_k
- Given y, find the corresponding cone number k as in (9)
- Scale down y if necessary
- Find vertices $y_i = Zv_i$ that lies on the k-facet from (6)
- Pick all triples from the list of y_i and determine corresponding solutions v from (10),(11) with nonnegative weights $\lambda_{1,2,3}$
- Pick the solution which is the closest to the previous one (v_{last}) in any norm
- Set $v_{last} = v$ and compute u from v

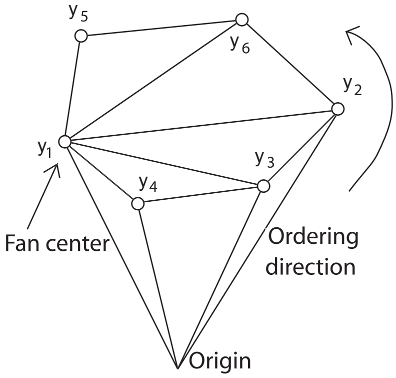

Fig. 5. An example of multiple tetrahedra within one convex cone defined by d_k ($L_k = \{y_1, ..., y_6\}$). Note that triples $\{y_1, y_4, y_5\}$ or $\{y_4, y_6, y_3\}$ (not shown on the picture) can form overlapping tetrahedra.

The main advantage of our procedure from the method of Durham is relatively (in comparison with the original method and its further modifications) simple computation of the solution. All operations can be done in real time without precomputation of any off-line tables (like in Petersen and Bodson (2002)) and with guarantee that the solution will be delivered for any possible input. In numerical values, a solution will be the same for our method and for Durham's for any possible example, which we therefore omit.

4. CONCLUSION

We revisited the direct control allocation method to propose an improved procedure of attainable set construction and control determination. Our procedure constructs representation of attainable set in the form of a collection of facet normals corresponding to non-overlapping convex cones. This allows the application of methods analogous to those associated with polyhedral Lyapunov functions for the solution of direct control allocation problem. The drawback of our solution is the absence of any optimization criteria while obtaining control vector.

5. ACKNOWLEDGMENTS

The author greatly appreciates the support from De Montfort University (UK) during his initial research on control allocation problem while preparing his PhD thesis there. He is also indebted for numerous discussions of control allocation algorithms with Eugene Kolesnikov (TsAGI, Russia).

REFERENCES

Blanchini, F. (1995). Non-quadratic lyapunov function for robust control. *Automatica*, 31, 451–461.

Blanchini, F. and Miani, S. (2008). *Set-theoretic methods in control*. Birkhäuser, Boston.

Bodson, M. (2002). Evaluation of optimization methods for control allocation. *Journal of Guidance, Control, and Dynamics*, 25(4), 703–711.

Cameron, D. and Princen, N. (2000). Control allocation challenges and requirements for the blended wing body. In *AIAA Guidance, Navigation and Control Conference Proceedings*. Denver, Colorado. AIAA Paper 2000-4539.

Casavola, A. and Garone, E. (2007). Adaptive fault tolerant actuator allocation for overactuated plants. In *Proceedings of the American Control Conference*. New York, USA.

Dantzig, G. (1966). *Linear programming and extensions*. Princeton University Press, Princeton.

Demenkov, M. (2007). Interval bisection method for control allocation. In *Proceedings of the 17th IFAC Symposium on Automatic Control in Aerospace*. Toulouse, France.

Demenkov, M. (2008). *Geometric algorithms for input constrained systems with application to flight control*. Ph.D. thesis, De Montfort University, UK.

Doman, D. and Ngo, A. (2002). Dynamic inversion-based adaptive/reconfigurable control of the x-33 on ascent. *Journal of Guidance, Control, and Dynamics*, 25(2), 275–284.

Doman, D. and Oppenheimer, M. (2002). Improving control allocation accuracy for nonlinear aircraft dynamics. In *AIAA Guidance, Navigation and Control Conference Proceedings*. Monterey, California. AIAA Paper 2002-4667.

Durham, W. (1993). Constrained control allocation. *Journal of Guidance, Control, and Dynamics*, 16(4), 717–725.

Durham, W. (1994). Constrained control allocation: three-moment problem. *Journal of Guidance, Control, and Dynamics*, 17(2), 330–336.

Durham, W. (1999). Efficient, near-optimal control allocation. *Journal of Guidance, Control, and Dynamics*, 22(2), 369–372.

Fossen, T., Johansen, T., and Perez, T. (2009). A survey of control allocation methods for underwater vehicles. In A. Inzartsev (ed.), *Intelligent Underwater Vehicles*. I-Tech Education and Publishing, Vienna.

Harkegard, O. and Glad, T. (2005). Resolving actuator redundancy - optimal control vs. control allocation. *Automatica*, 41, 137–144.

Johansen, T., Fossen, T., and Tondel, P. (2005). Efficient optimal constrained control allocation via multiparametric programming. *Journal of Guidance, Control, and Dynamics*, 28(3), 506–515.

Laine, L. and Fredriksson, J. (2008). Traction and braking of hybrid electric vehicles using control allocation. *International Journal of Vehicle Design*, 48(3–4), 271–298.

Lee, H., Lee, D., Bang, H., and Tahk, M. (2007). Singularity avoidance law for CMG using virtual actuator. In *Proceedings of 17th IFAC Symposium on Automatic Control in Aerospace*. Toulouse, France.

McMullen, P. and Shepard, G. (1971). *Convex polytopes and the upper bound conjecture*. Cambridge University Press, Cambridge.

Pechev, A. (2008). Inverse kinematics without matrix inversion. In *Proceedings of IEEE International Conference on Robotics and Automation*. Pasadena, CA, USA.

Petersen, J. and Bodson, M. (2002). Fast implementation of direct allocation with extension to coplanar controls. *Journal of Guidance, Control, and Dynamics*, 25(3), 464–473.

Plumlee, J., Bevly, D., and Hodel, A. (2004). Control of a ground vehicle using quadratic programming based control allocation techniques. In *Proceedings of the American Control Conference*. Boston, Massachusetts, USA.

Pulecchi, T. and Lovera, M. (2007). Attitude control of spacecraft with partially magnetic actuation. In *Proceedings of 17th IFAC Symposium on Automatic Control in Aerospace*. Toulouse, France.

Schofield, B. and Hagglund, T. (2008). Optimal control allocation in vehicle dynamics control for rollover mitigation. In *Proceedings of the American Control Conference*. Seattle, Washington, USA.

Spjtvold, J. and Johansen, T. (2009). Fault tolerant control allocation for a thruster-controlled floating platform using parametric programming. In *Proceedings of the 48th IEEE Conference on Decision and Control held jointly with the 2009 28th Chinese Control Conference*. Shanghai, China.

Thieuw, A. and Marcille, H. (2007). Pleiades-HR CMGs-based attitude control system design, development status and performances. In *Proceedings of 17th IFAC Symposium on Automatic Control in Aerospace*. Toulouse, France.

Tjonnas, J. and Johansen, T. (2008). Adaptive control allocation. *Automatica*, 44, 2754–2765.

Tondel, P. and Johansen, T. (2005). Control allocation for yaw stabilization in automotive vehicles using multiparametric nonlinear programming. In *Proceedings of the American Control Conference*. Portland, Oregon, USA.

Ziegler, G. (1995). *Lectures on Polytopes*, volume 152 of *Graduate Texts in Mathematics*. Springer-Verlag, New York.