

Constrained Quadratic Programming Techniques for Control Allocation

John A. M. Petersen^{*}
Raytheon Missile Systems
Tucson, AZ 85706

Marc Bodson[†]
University of Utah
Salt Lake City, UT 84112

Abstract—The paper considers the objective of optimally specifying redundant control effectors under constraints, a problem commonly referred to as control allocation. The problem is posed as a mixed ℓ_2 -norm optimization objective and converted to a quadratic programming formulation. The implementation of an interior-point algorithm is presented. Alternative methods including fixed-point and active set methods are used to evaluate the reliability, accuracy and efficiency of the primal-dual interior-point method. While the computational load of the interior-point method is found to be greater for problems of small size, convergence to the optimal solution is also more uniform and predictable. In addition, the properties of the algorithm scale favorably with problem size.

Index Terms—Control allocation, flight control, quadratic programming, interior-point methods.

^{*} Principal Systems Eng., GNC Center, Raytheon Missile Systems, 1151 E. Hermans Rd, Tucson, AZ, 85706.

[†] Professor, Electrical & Computer Engineering, University of Utah, 50 S Central Campus Dr Rm 3280, Salt Lake City, UT 84112, U.S.A.

INTRODUCTION

Control allocation is the problem of managing the redundant control effectors such as found in modern aircraft. Two classes of algorithms for solving the problem are linear programming techniques and least-squares (LS) methods, corresponding to ℓ_1 -norm and ℓ_2 -norm objectives, respectively. Perhaps the most notable advantage of the ℓ_2 -norm formulation is that it yields solutions that use all the surfaces, instead of relying on a few, as with the ℓ_1 -norm. This characteristic limits the degradation of performance during an actuator failure, as more control surfaces are utilized to compensate for the loss of a specific control [1]. Benefits from this behavior also include a lower sensitivity to numerical data, and smoother flight trajectories.

Current methods for solving the ℓ_2 -norm formulation include LS with clipping, redistributed pseudo-inverse [2,3], an approximate quadratic programming method [4], and the fixed-point method [5]. These methods are simple and easy to implement, and they work well for a restricted range of commands. However, they typically find approximate solutions. Recently, active set methods, which find the exact optimal solution in a finite number of steps, have been proposed for solving control allocation problems [6].

A powerful class of algorithms for solving optimization problems is the set of interior-point (IP) methods. IP methods have been applied to ℓ_1 -norm formulations of control allocation problems [7], but are also capable of minimizing non-linear objective functions constrained by linear equality and inequality constraints. Primal-dual IP methods travel along the proximity of a central path that leads to an optimal point. A benefit of such methods is that the relative distance from the optimum is always known,

so that one can exit the algorithm when a solution is reached within a specified tolerance. Convergence is also uniform towards the optimal solution. The IP method that is described and implemented in this paper is adapted from Vanderbei [8,9]. Test results with two aircraft models enable comparisons of the fixed-point, active set, and primal-dual methods, in terms of accuracy, convergence, and computational efficiency.

PROBLEM STATEMENT

The objective of control allocation in flight control is to determine the n components of the control vector, u , that result in the desired roll, pitch, and yaw components of the acceleration vector, a_d . These components are assumed to be related to the control vector by the controls effectiveness matrix, CB , as follows:

$$\{a_d = CBu \mid u_{\min} \leq u \leq u_{\max}\} \quad (1)$$

where $CB \in \mathfrak{R}^{3 \times n}$. Model reference control laws [10] and dynamic inversion control laws [11] allow one to specify the trajectories of the output of the system by selecting the value of the term CBu due to the control input. The existence of a "control allocator" enables one to divide the control problem into a small size dynamic compensator, and a large size static redundancy management system. We expect that other applications with redundant effectors will emerge in the future.

In general, there is no guarantee that a_d is attainable or that the solution is unique. If the solution is not unique, a secondary objective is to minimize the magnitude of the control vector, or its distance from a preferred control value, u_0 . Combining two

objectives is known as mixed optimization, and can be expressed with the quadratic programming problem

$$\begin{aligned} \min_u \quad & J = \|CBu - a_d\|_2^2 + h\|u - u_0\|_2^2 \\ \text{subject to} \quad & u_{\min} \leq u \leq u_{\max} \end{aligned} \quad (2)$$

where $h > 0$. The factor h is used to adjust the relative weighting of the secondary criteria and is usually chosen to be small.

Note that an optimization problem

$$\begin{aligned} \min_x \quad & f(x) \\ \text{subject to} \quad & G(x) = 0, \quad H(x) \geq 0 \end{aligned} \quad (3)$$

has a global solution if G and H are linear functions and $f(x)$ is convex [12]. Equation (2) will be shown to fall in this category meaning that a control allocation problem with mixed ℓ_2 -norm terms has a unique solution.

CURRENT METHODS

Current methods for solving quadratic control allocation problems are based on pseudo-inverse techniques. LS with clipping (LSC) is simply a truncated LS solution. Redistributed pseudo-inverse [2,3] successively computes a LSC solution using the remaining unsaturated variables after each iteration to reduce the error until all variables are saturated or the objective is achieved. Enns' [4] proposed a quadratic programming method that iterates to find a solution that lies on an ellipsoid that encircles the feasible space. This solution is clipped to give controls that satisfy their limits. To solve dual purpose objectives these methods must be broken into a two-step process, essentially

solving two problems sequentially. The main attraction of these methods is that they are simple to implement. However, they typically do not yield an optimum solution, especially for unattainable commands.

Burken, et al [5], used a fixed-point method to solve a mixed optimization problem equivalent to (2). This method is very easy to code and is fast for most achievable commands. Although Lu [13] proved global convergence for the method, in practice it converges quite slowly for large commands. Therefore, the fixed-point method is usually implemented with a fixed number of iterations such as 50 [6,14].

Härkegård [6] has recently proposed active set methods for solving control allocation problems. Among these the weighted least-squares (WLSQ) method was the most efficient. Active set solutions are much like simplex solutions except they can be applied to quadratic cost functions. Convergence is fast, but an upper bound on the number of iterations can be very large.

QUADRATIC PROGRAMMING FORMULATIONS OF CONTROL ALLOCATION

Equation (2) can be converted to a standard quadratic problem formulation. Let

$$\begin{aligned} x &= u - u_{\min}, & x_0 &= u_0 - u_{\min} \\ x_{\max} &= u_{\max} - u_{\min}, & a_0 &= a_d - CBu_{\min} \end{aligned} \quad (4)$$

resulting in the constraint set

$$x + w = x_{\max}, \quad x \geq 0, w \geq 0 \quad (5)$$

w is a slack variable used to guarantee the upper bound on x . J can be expanded to

$$\begin{aligned}
J &= (CBx - a_0)^T (CBx - a_0) + h(x - x_0)^T (x - x_0) \\
&= \frac{1}{2} x^T Hx + c^T x + k
\end{aligned} \tag{6}$$

where $H = 2(CB^T CB + hI)$, $c^T = -2(a_0^T CB + hx_0^T)$, and $k = a_0^T a_0 + hx_0^T x_0$. Since a constant in the objective function does not affect the optimal solution, k is dropped and the final form is

$$\begin{aligned}
\min_x \quad & J = \frac{1}{2} x^T Hx + c^T x \\
\text{subject to} \quad & x + w = x_{\max}, x \geq 0, w \geq 0
\end{aligned} \tag{7}$$

INTERIOR-POINT ALGORITHMS

If the weighting factor, h , is greater than zero, or if CB has full row rank, H will be positive definite. Under this condition, the objective function of (7) is convex and the Karush-Kuhn-Tucker (KKT) [15,16] optimality conditions apply globally. Making use of logarithmic barrier functions to satisfy the lower bound constraints, the Lagrangian of (7) is expressed as

$$\begin{aligned}
L &= \frac{1}{2} x^T Hx + c^T x + z^T (x + w - x_{\max}) \\
&\quad - \mu \sum_{i=1}^n \log(x_i) - \mu \sum_{i=1}^n \log(w_i)
\end{aligned} \tag{8}$$

where $\mu > 0$. From the Lagrangian, the first order optimality conditions are derived as

$$\begin{aligned}
Hx + c + z - s &= 0, & Xs - \mu e &= 0 \\
x + w - x_{\max} &= 0, & Wz - \mu e &= 0 \\
x > 0, w > 0, z > 0, s > 0
\end{aligned} \tag{9}$$

where X and W are diagonal matrices whose diagonal elements are x and w , respectively. e is defined as a column vector of ones. To satisfy the KKT conditions, (9) must hold with $\mu = 0$. In this case, $Xs = 0$ and $Wz = 0$, which are known as the

complementarity conditions. The parameter, μ , is referred to as the complementarity gap and is used to guide the solution along a trajectory called the central path. The central path is a sequence of solutions that leads to the optimal point. Path-following methods attempt to travel in the neighborhood of the central path until a solution is near the optimum. There is an abundant variety of mechanisms and theorems in operations research literature for achieving this result. The following presentation is an attempt to evaluate the implementation of the applicable methods to the control allocation problem.

Primal-Dual Interior-Point Algorithm

You will have the greatest control over the appearance of your figures if you are able to prepare electronic image files. If you do not have the required computer skills, just submit paper prints as described above and skip this section.

1) *Step Direction:* To get the step direction, we substitute $\{s + \Delta s, w + \Delta w, x + \Delta x, z + \Delta z\}$ for $\{s, w, x, z\}$ in (9) and drop the non-linear terms to arrive at

$$\begin{aligned}\Delta x &= D(r_c + W^{-1}r_{wz} - X^{-1}r_{xs} - W^{-1}Zr_u) \\ \Delta w &= -\Delta x - r_u \\ \Delta s &= -X^{-1}r_{xs} - X^{-1}S\Delta x \\ \Delta z &= -W^{-1}r_{wz} - W^{-1}Z\Delta w\end{aligned}\tag{10}$$

where $D = (H + X^{-1}S + W^{-1}Z)^{-1}$ and where the residuals are defined as

$$\begin{aligned}r_c &= Hx + c + z - s, & r_{xs} &= Xs - \mu e \\ r_u &= x + w - x_{\max}, & r_{wz} &= Wz - \mu e\end{aligned}\tag{11}$$

These residuals make initialization a simple matter by allowing infeasible starting points, *i.e.* points that do not satisfy the equality constraints, which can be difficult to determine.

2) *Step Size*: Since the variables are coupled through the equation $Hx + c + z - s = r_c$, a common step size must be used in the updates of all variables. To satisfy the inequality constraints, the maximum allowable step size, α , must be determined. The update law is

$$\begin{aligned} x &= x + \rho\alpha\Delta x, & s &= s + \rho\alpha\Delta s \\ w &= w + \rho\alpha\Delta w, & z &= z + \rho\alpha\Delta z \end{aligned} \quad (12)$$

where $\alpha = \min\{\alpha_x, \alpha_w, \alpha_s, \alpha_z\}$ and

$$\alpha_p = \min\left\{\left[\frac{-p_i}{\Delta p_i}, 1\right] \Delta p_i < 0, i = 1..n\right\} \quad (13)$$

for $p \in \{x, w, s, z\}$. The term ρ must be in the range $0 < \rho < 1$, but is usually chosen above 0.9 for fast convergence. We use $\rho = 0.9995$ in our implementation.

3) *Computation of μ* : As μ goes to zero, the iterates converge to an optimal point. In an attempt to keep the variables in the proximity of the central path, the elements of Xs and Wz are reduced to zero at a similar rate. This can be accomplished by computing μ using the average of the complementarity conditions, such as

$$\gamma = \frac{x^T s + w^T z}{2n}, \quad \mu = \sigma\gamma \quad (14)$$

where $0 < \sigma < 1$. σ can be chosen dynamically to improve convergence as suggested by Vanderbei [8] and Zhang [17]. We have experienced good results employing Zhang's method, $\sigma = \min(0.1, k\gamma)$, with $k = 100$.

4) *Stopping Criteria:* From the KKT criterion, the optimal solution occurs when all the residuals and the complementarity gap are zero. The residuals, r_c and r_u , can be forced to be zero at initialization. Interestingly, we have found that this strategy worked well, although it does not with ℓ_1 -norm optimization[7]. With r_c and r_u both zero, the only errors left in the system are directly related to μ . Therefore, when μ has converged sufficiently close to zero, the algorithm is terminated.

5) *Starting Point:* The presence of H in the matrix D has a stabilizing effect on the conditioning of the system, as well as adding robustness to the starting point. Likewise, the starting point is of little consequence as virtually any interior starting point converges in similar time. One may take advantage of this property to reduce computations by eliminating some of the residuals. Setting initial values of $x = w = 0.5x_{\max}$ and $s = z > 0$, forces $r_c = r_u = 0$ so that the equations of (10) may be simplified.

Predictor-Corrector Path-Following

The vast majority of commercial software is based on Mehrotra's predictor-corrector method [18]. Mehrotra split the correction into two steps, a predictor and a corrector, by exploiting 2^{nd} order terms of the primal-dual update. The predictor step advances the iterate toward the optimal solution while reducing infeasibility. The corrector step keeps the updated point near the central path and utilizes the 2^{nd} order terms to make a

correction toward the optimal point. Zhang and Zhang [19] further extended Mehrotra's method to quadratic programming, including a quadratic line search for α . The predictor-corrector method has been applied to ℓ_1 -norm objectives and was not advantageous for control allocation problems of small size [7]. Therefore, it was not implemented for the ℓ_2 -norm objective.

Numerical Stability

In the experiments tested, the primal-dual IP algorithm consistently converged to extremely small values ($<1e-15$). This benefit stems from the multiple positive terms that add to the diagonal elements of D , resulting in a well-conditioned matrix. However, the cost of this robustness is increased computational load, namely due to the computation of D^{-1} . One method available to partially alleviate this problem is to compute D^{-1} using the Sherman-Morrison-Woodbury formula

$$(P + QR^T)^{-1} = P^{-1} - \left[P^{-1}Q(I + R^T P^{-1}Q)^{-1} R^T P^{-1} \right] \quad (15)$$

where $P \in \mathfrak{R}^{n \times n}$, $Q \in \mathfrak{R}^{m \times n}$, and $R \in \mathfrak{R}^{m \times n}$.

TEST RESULTS AND COMPARISONS

The three algorithms, fixed-point, WLSQ active set, and primal-dual IP, were tested using linear models of a Boeing C-17 cargo jet and a version of Lockheed-Martin's advanced tailless fighter design. Comparisons of the algorithms are made regarding accuracy, efficiency, and convergence properties.

Measure of Accuracy

The following metric is used to gauge acceleration accuracy

$$\varepsilon_a = \frac{\|a_d - CBu\|_2 - \|a_d - CBu_{opt}\|_2}{\|CB\|_2} \quad (16)$$

Normalizing the acceleration error by CB has the effect of turning ε_a into a measurement similar to actuator angle error. This is convenient for determining an acceptable range for ε_a , since the resolution of actuators is easily obtained. u_{opt} is determined from the WLSQ active set method because it converges to the exact solution in a finite number of steps. For IP algorithms, accuracy is determined by the stopping tolerance. For the simulations that follow, a stopping tolerance of $\varepsilon_s \leq 0.0001$ is used to yield an acceptable error of $\varepsilon_a \leq 0.1$. The weighting parameter, chosen as $h = 0.0001$, is sufficient to guarantee uniqueness while providing minimum control effort.

Fixed-Point vs. Primal-Dual IP

The fixed-point method is theoretically capable of finding an optimal point, but may require many iterations to converge. Therefore, a typical implementation will use a fixed number of iterations. Increasing the number of iterations improves the result, but eventually becomes impractical. The initial cost of the fixed-point method is large, but it requires few computations per iteration. The convergence is highly dependent on the attainability of the command. We measure attainability by the magnitude of a command, a_d , normalized by the magnitude of the boundary point, a_b , that lies in the direction of the command

$$\eta = \frac{\|a_d\|}{\|a_b\|} \quad (17)$$

Fig. 1 demonstrates the great variability in convergence rate over the attainability axis for the tailless model (with similar results for the C-17 model). Convergence is fast for small commands ($\eta < 0.5$). When the boundary of the attainable set is reached, convergence can become very slow. Plots for only one command are given but they are typical of the fixed-point method.

Fig. 2 shows a plot of acceleration error of the fixed-point and primal-dual IP methods as functions of floating point operations (flops). In the plot, convergence behavior over three values of η demonstrate that the IP method exhibits much more consistency between models than the fixed-point method. Also note that the initial cost of the fixed-point method is almost as high as the stopping point ($\varepsilon_a \leq 0.1$) for the worst IP curve.

The sharp changes in the slope of the error curves of the fixed-point method are due to actuator saturations. Fig. 3 shows the error together with the actuator positions as a function of the number of iterations. There are three distinct changes in the convergence rate shown. Each corresponds to at least one actuator reaching a position limit.

Active Set vs Primal-Dual IP

A test set Ψ is created, comprised of three subsets of vectors, namely, achievable, exactly achievable, and unachievable acceleration commands corresponding to $\{\frac{1}{2}\Omega, \Omega, 2\Omega\}$, respectively. The subset, $\Omega \subset \mathfrak{R}^3$, consists of 1000 vectors in random directions that are scaled so that each vector just touches the boundary of the attainable

set. Two more subsets of vectors are then created by halving and doubling the magnitudes. Most control allocation algorithms behave differently for these three subsets of commands. The test set was therefore created this way to ensure that each case was adequately represented.

As with the fixed-point method, the WLSQ active set method converges quickly for small commands. For larger commands, convergence is initially quite slow. The final iteration usually drops the error from about 0.1 to about 0.00001. This makes it difficult to predict if the algorithm is near an optimal solution. Fig. 4 is a plot of the active set acceleration errors using the C-17 model. The convergence rates for each subset of Ψ are different enough to show them separately. Outlines are drawn to represent the envelope of acceleration errors for the subsets. Solutions to small commands are quickly found while larger commands require a wider range of iterations. Similar results, but fewer iterations overall, are found for the tailless model.

The primal-dual IP method exhibits at least exponential convergence for both models. The results for the C-17 are shown in Fig. 5 and are representative of both models. The number of computations per iteration for an active set method is variable. During the progression of a solution, the computational load of an active set method can often decrease, whereas the IP method requires the same number of computations at each iteration. Although active sets may require more iterations, the active-set method is more efficient than the IP method for the models tested.

Another interesting difference between the active set and IP methods is the evolution of the estimate toward the optimal solution. With the active set method,

several variables are typically at their extremal values. Variables can be changed from inactive to active, preventing them from being stuck at non-optimal values. However, it can also lead to multiple switchings in values which forces one to wait until the algorithm has fully completed its decisions. In contrast, IP methods tend to exhibit more gradual behavior. Fig. 6 and Fig. 7 show that actuator values converge much smoother when computed using the primal-dual IP method than with the active set method.

Another advantage of the interior-point methods is that they scale very favorably with problem size. Active set methods generally require increasing numbers of iterations with increasing numbers of control effectors. This relationship is shown to be essentially linear in Fig. 8. For a given problem size (defined by the number of controls), 1000 random CB 's and a_d 's are created and solved using both the active set and primal-dual IP methods. The maximum number of iterations for each method is plotted. The convergence of the IP method is shown to be independent of problem size. The computational advantage of the IP method is even more pronounced with regard to the number of flops required for convergence. Fig. 9 reveals an exponential increase in computations with number of controls for the active set method, whereas the IP method has more of a linear relationship. The crossover point for which IP methods begin to outperform active sets in computational efficiency is around $n=15$.

CONCLUSIONS

Traditional methods of control allocation do not always provide an optimal solution, especially for unattainable commands. To address this problem, several optimization algorithms have recently been proposed that obtain exactly optimal solution. A primal-

dual interior-point method was described in this paper, including the details needed for a successful implementation. Linear models of a C-17 and an advanced tailless aircraft were used to test accuracy and efficiency of the method. The results for the interior-point method were also compared to those obtained with a fixed-point method and an active set method. The fixed-point algorithm is the simplest of the three algorithms considered. It provides good results for commands that are well within the attainable set. However, convergence can become very slow at the boundary and for unattainable commands. The active set algorithm converges exactly to the solution in a finite number of steps, and is computationally very efficient for control allocation problems of small size. Interior-point algorithms compete favorably with active set methods for problems with a large number of control effectors, but require somewhat more computations for current, typical applications. Nevertheless, their uniform convergence properties are a significant advantage, since real-time control requires a fixed bound on the number of iterations. In addition, predicting when the solution is near optimal is not possible with active set methods, but both feasible and accurate for interior-point methods. The computational load of interior-point methods is also tractable with modern computers, although significant.

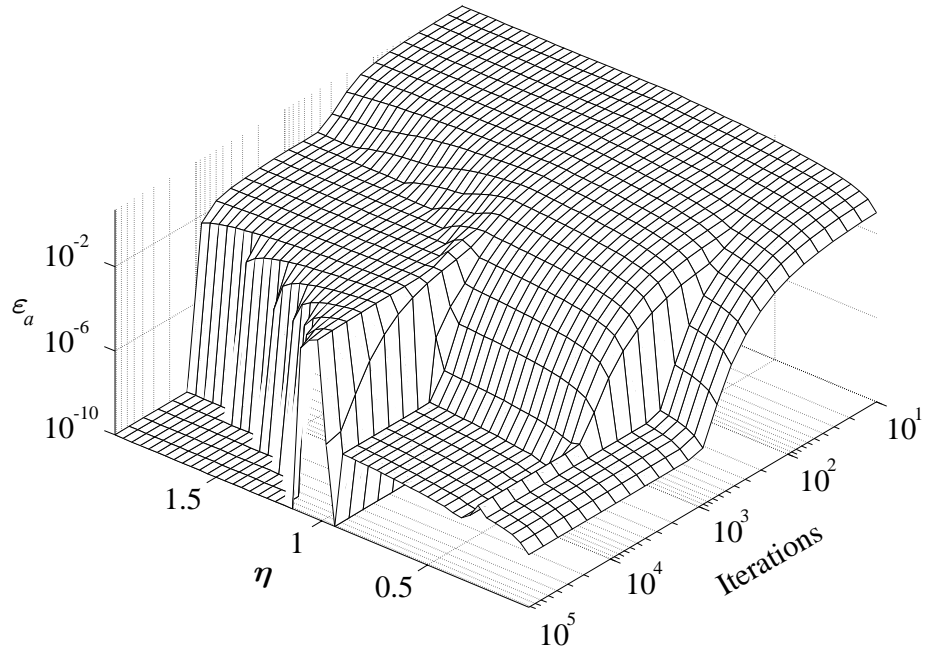


Fig. 1. Acceleration error of fixed-point method over a wide range of attainability and number of iterations. (Tailless Fighter Model)

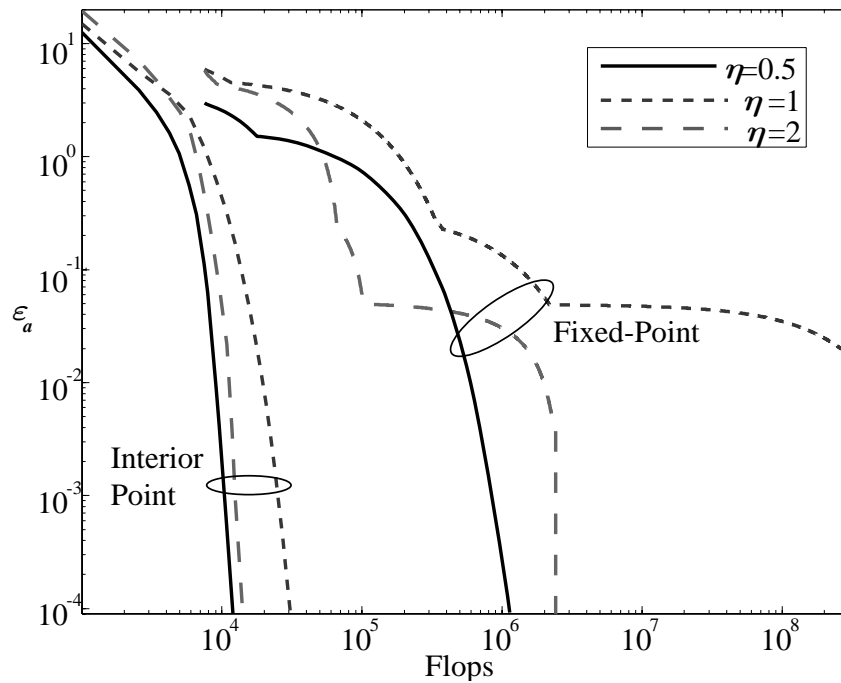


Fig. 2. Acceleration error as functions of the number of iterations using fixed-point and IP methods for different values of η . (Tailless Fighter Model)

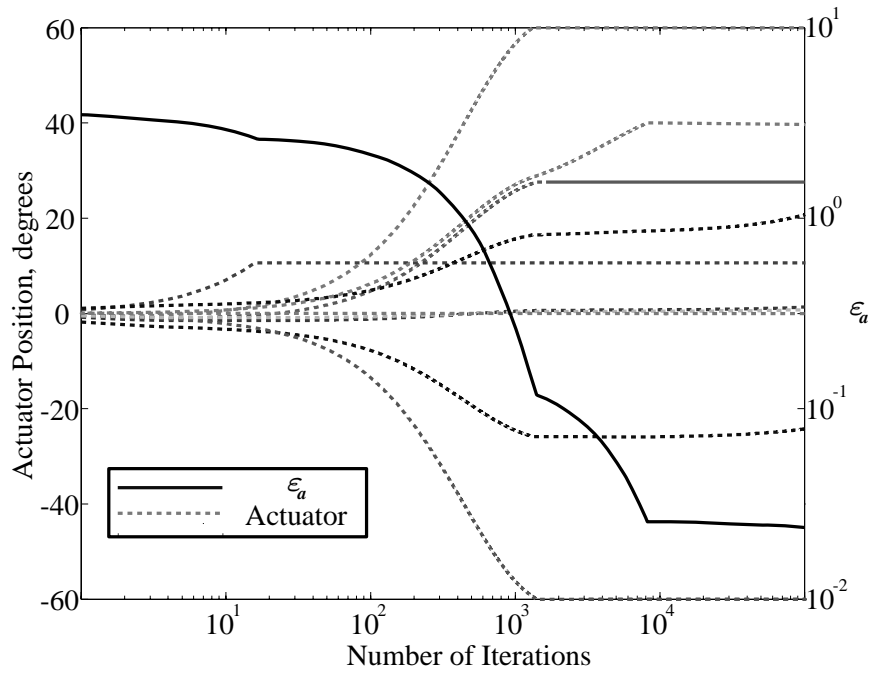


Fig. 3. Relationship of convergence of ε_a to actuator saturations.

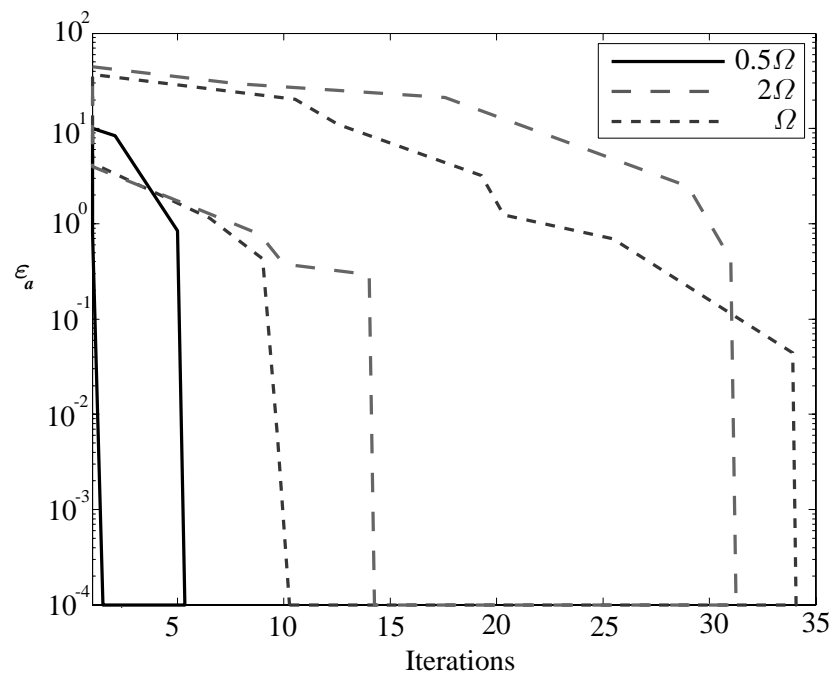


Fig. 4. Envelope of acceleration errors using an active set method for different magnitudes of commands.(C-17 Model)

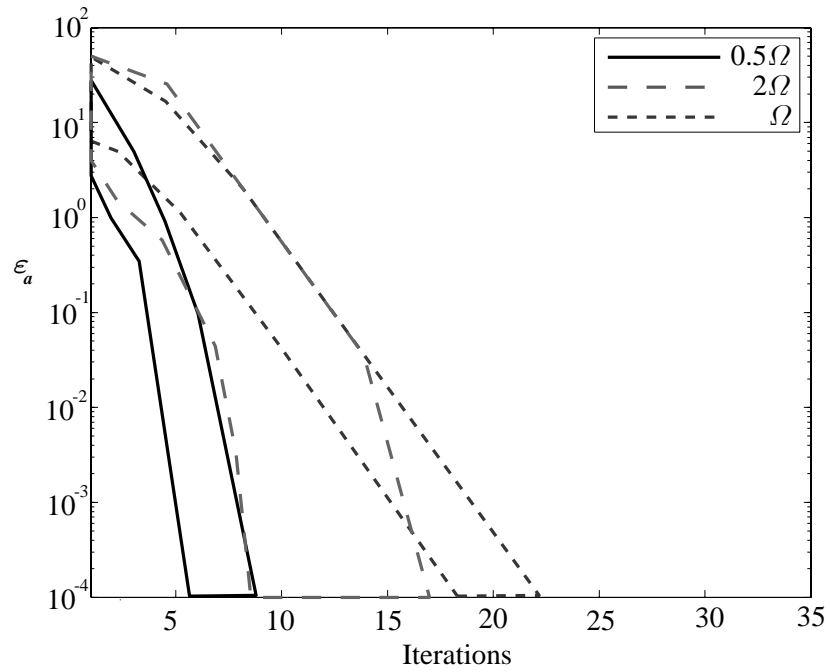


Fig. 5. Envelopes of primal-dual IP convergence rates for different magnitudes of commands for the C-17 model.

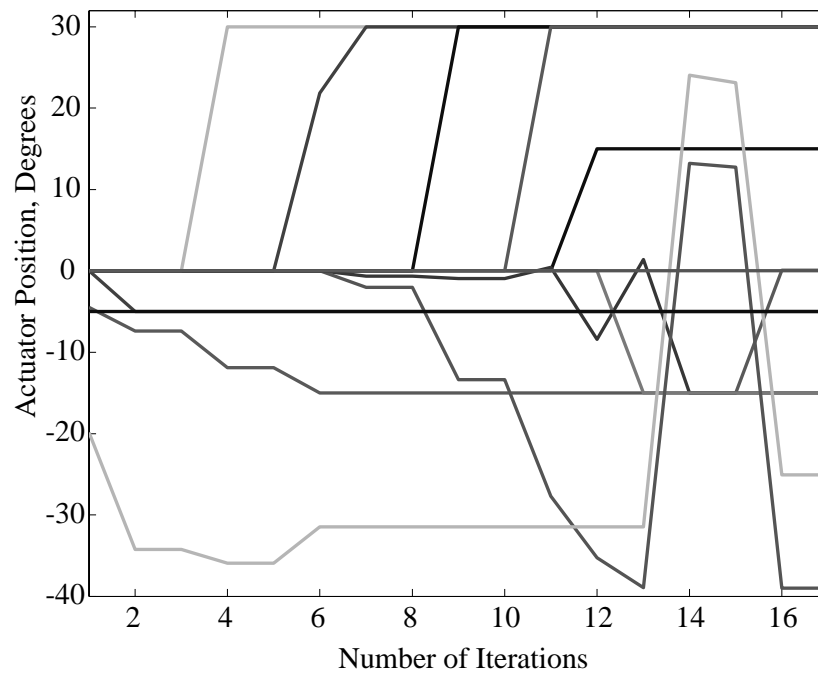


Fig. 6. Actuator values during active set convergence. (C-17 Model)

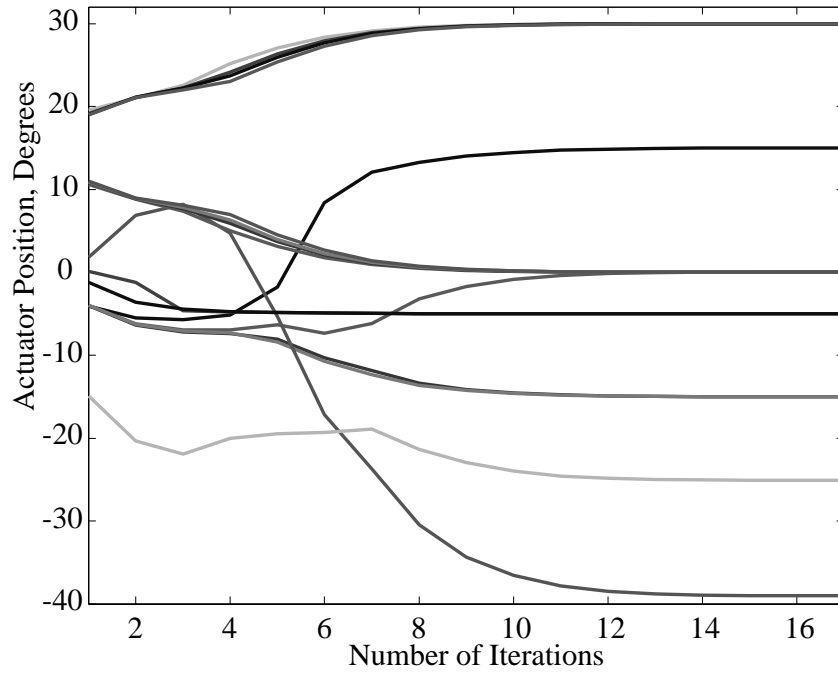


Fig. 7. Actuator values during primal-dual IP convergence. (C-17 Model)

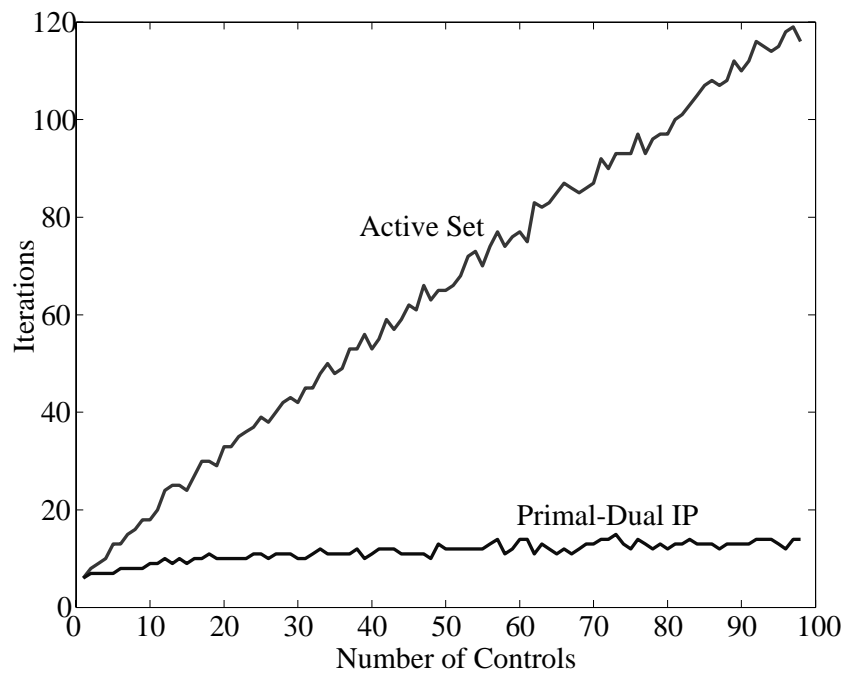


Fig. 8. Maximum number of iterations for convergence.

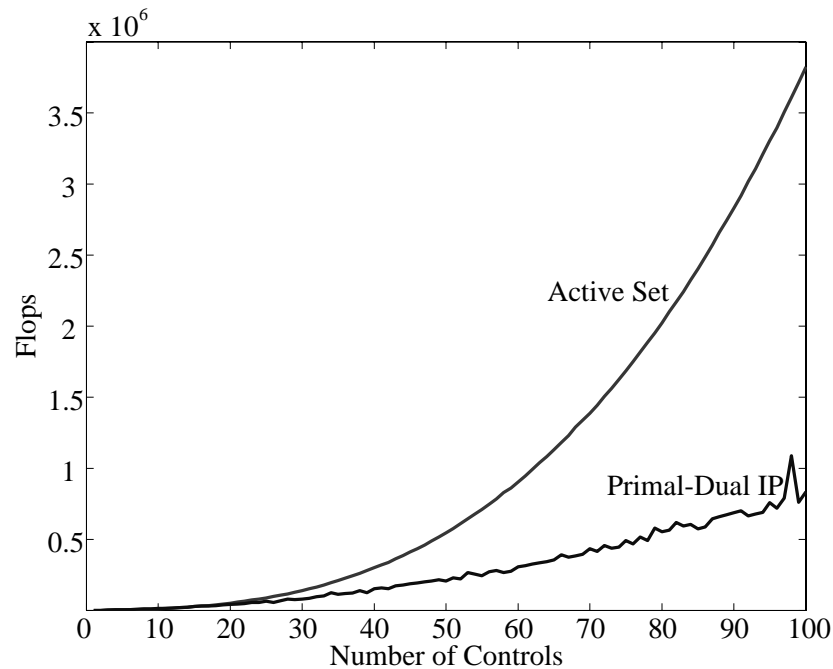


Fig. 9. Maximum number of flops for convergence.

REFERENCES

- [1] A. B. Page, M. L. Steinberg, "High-Fidelity Simulation Testing of Control Allocation Methods," *Proc. AIAA Guidance, Navigation, and Control Conf.*, Monterey, CA, 2002, Paper AIAA 2002-4547.
- [2] J. Virnig, D. Bodden, "Multivariable Control Allocation and Control Law Conditioning when Control Effectors Limit," *Proc. AIAA Guidance, Navigation, and Control Conf.*, Scottsdale, AZ, 1994, Paper AIAA 94-3609.
- [3] K.A. Bordignon, "Constrained Control Allocation for Systems with Redundant Control Effectors," Ph.D. Thesis, Dept. Aerospace & Ocean Eng., Virginia Polytechnic Institute & State Univ., Blacksburg, Virginia, 1996.

- [4] D. Enns, "Control Allocation Approaches," *Proc. AIAA Guidance, Navigation, and Control Conf.*, Boston, MA, Aug. 1998, Paper AIAA 98-4109.
- [5] J. J. Burken, P. Lu and Z. Wu, "Reconfigurable Flight Control Designs with Application to the X-33 Vehicle," *Proc. AIAA Guidance, Navigation, and Control Conf.*, Portland, OR, 1999, Paper AIAA 99-4134.
- [6] O. Härkegård, "Efficient Active Set Algorithms for Solving Constrained Least Squares Problems in Aircraft Control Allocation," Dept of Electrical Engineering, Linköpings University, Sweden, Tech. Rep. LiTH-ISY-R-2426, 2002.
- [7] J. Petersen and M. Bodson, "Interior Point Algorithms for Control Allocation," *J. of Guidance, Control, and Dynamics*, submitted for publication.
- [8] R. Vanderbei, "LOQO: An Interior Point Code for Quadratic Programming," Princeton Univ., Princeton, N.J., Tech. Rep. SOR 94-15, 1994, revised 1998.
- [9] R. Vanderbei, *Linear Programming: Foundations and Extensions*, Boston, MA, Kluwer Academic Publishers, 2001, ch. 23.
- [10] M. Bodson and W. Pohlchuck, "Command Limiting in Reconfigurable Flight Control," *J. of Guidance, Control, and Dynamics*, Vol. 21, No. 4, pp. 639-646, 1998.
- [11] J. Buffington, P. Chandler and M. Pachter, "Integration of On-line System Identification and Optimization-based Control Allocation," *Proc. AIAA Guidance, Navigation, and Control Conf.*, Boston, MA, Aug. 1998, Paper AIAA 98-4487.

- [12] S. Wright, *Primal-Dual Interior-point Algorithms*, Argonne National Laboratory, Argonne, Ill.: SIAM Publications, 1997, Appendix A.
- [13] P. Lu, "Constrained Tracking Control of Nonlinear Systems," *Systems & Control Letters*, Vol. 27, pp. 305-314, 1996.
- [14] M. Bodson, "Evaluation of Optimization Methods for Control Allocation," *J. of Guidance, Control, and Dynamics*, Vol. 25, No. 4, pp. 703-711, 2002.
- [15] W. Karush, "Minima of functions of several variables with inequalities as side constraints," MS thesis, University of Chicago, Chicago, Ill., 1939.
- [16] H. Kuhn and A.W. Tucker, "Nonlinear programming," *Proc. 2nd Berkeley Symp. on Mathematical Statistics and Probability*, edited by J. Neyman, Univ. of California, Berkley, CA, 1951.
- [17] Y. Zhang, "Solving Large-Scale Linear Programs by Interior-point Methods Under the MATLAB Environment," Dept of Mathematics and Statistics, Univ. of Maryland, Baltimore, MD, Tech. Rep. TR96-01, 1996.
- [18] S. Mehrotra, "On the Implementation of a Primal-Dual Interior-point Method," *SIAM J. of Optimization*, Vol. 2, No. 4, pp. 575-601, 1992.
- [19] Y. Zhang and D Zhang, "On Polynomiality of the Mehrotra-type Predictor-Corrector Interior-Point Algorithms," *Mathematical Programming*, vol. 68, pp. 303-318, 1995.