Query-based Attitude Control of a Low Altitude Communications Satellites

Technical Report of the ISIS Group
at the University of Notre Dame
ISIS-94-012
September, 1994

C. Lucisano, M.D. Lemmon
Department of Electrical Engineering
University of Notre Dame
Notre Dame, IN 46556

Interdisciplinary Studies of Intelligent Systems

Query-based Attitude Control of a Low Altitude Communications Satellites

C. Lucisano, M.D. Lemmon
Department of Electrical Engineering
University of Notre Dame
Notre Dame, IN 46556 USA

Sept. 15, 1994

Abstract

In August 1993, heavy meteor activity caused the Olympus telecommunications satellite (S/C) to lose Earth pointing attitude. Automatic emergency reacquisition maneuvers consumed so much propellant that the mission had to be terminated one year ahead of schedule. This paper re-examines the Olympus scenario using a recently proposed adaptive control scheme. The proposed method is a "query-based" adaptive control scheme which uses a cutting-plane algorithm known as the ellipsoid method to identify stabilizing controller gains. This paper demonstrates the query-based adaptation scheme in a detailed simulation of the Olympus satellite. The results show that the proposed method is able to autonomously identify nominal controller gains for the Olympus scenario outlined above.

1 Introduction

On 11 August 1993 during an abnormal Perseid meteor shower, the European telecommunications satellite Olympus lost Earth pointing attitude. The control mode switched automatically to the thruster based Emergency Sun Acquisition mode, which provided control along the pitch and roll axis. The emergency maneuver failed to re-acquire sun pointing and introduced high body rates as a side effect. The ground station disabled the emergency mode and the spacecraft was driven again into a stable spinning state. Subsequent analysis of post anomaly telemetry showed that there was insufficient propellant left to reestablish normal station keeping. The Olympus mission was therefore terminated one year ahead of schedule.

The Olympus scenario clearly demonstrates the limited autonomy of existing communication satellites. The high body rates induced by the emergency manuever perturbed the system's linearized dynamics so that the original attitude controller worked poorly. Had Olympus been able to autonomously reset its controller, then it may have been possible to re-establish nominal operation without depleting the spacecraft's limited propellant reserves. Such autonomous resetting of controller gains could have been accomplished using an adaptive control scheme. This paper examines this conjecture using a recently proposed adaptation scheme [Lemmon 1993] based on the ellipsoid method.

The approach proposed in this paper orignally appeared in [Lemmon 1993] and has been applied in [Lemmon 1995] with regard to stabilizing hybrid control systems. The method is an on-line learning algorithm based on an inductive inference protocol. Inductive inference is a machine learning protocol which learns functional mappings from examples of the function's input/output pairs. The algorithm uses an algorithmic construction known as the *oracle* to guide the learning process. Because of its reliance on oracle constructions, these algorithms are also called *query-based* learning methods. The direct adaptive control scheme presented in [Lemmon 1993] is a query-based learning method which identifies a set of quadratically stabilizing control gains for a linear time-invariant plant assuming full state feedback. The interesting aspect of this procedure is that it can be proven to converge after a finite number of updates. This means that the learning method would "stabilize" the system after a finite number of "stumbles". In addition to this, it is well known that upper bounds on this finite convergence time scale in a polynomial manner with problem size, thereby recommending the method's use in large-scale systems.

Early work [Lemmon 1992] suggested query-based adaptation had promise as a means for vehicular stabilization. It was therefore suggested that this technique might be useful in situations like the Olympus scenario. This paper shows how the query-based methods in [Lemmon 1993] were implemented in a detailed digital simulation of the Olympus satellite. The simulation experiments were based on the Olympus scenario outlined above, in an attempt to see if query-based adaptation would have been able to significantly reduce the spacecraft's fuel consumption in reacquiring its Earth pointing attitude. These results were very encouraging and suggested that these learning methods are feasible methods for insuring higher degrees of spacecraft autonomy.

The remainder of this paper is organized as follows. Section 2 discusses the simulated Olympus spacecraft model which was used in this paper. Section 3 discusses query-based adaptation and discusses a novel implementation of the algorithm which has not been previously presented. Section 4 discusses the simulation results. Section 5 summarizes the significant results of this paper.

2 The Olympus Spacecraft

The Olympus spacecraft was a telecommunications satellite built by British Aerospace for the European Space Agency (ESA). The three-axis-stabilized satellite had a payload consisting of two TV channels, one scientific package for research on radio propagation and a system to assist with video conferencing within Europe. It orbited about 1000 kilometers above the Earth's surface. Because of this low orbit, there were significant disturbance torques associated with aerodynamic drag on the solar panels. There were also additional disturbance torques due to solar radiation pressure, gravity gradient effects, and bending modes. The on-board attitude control subsystem included an analog sun sensor, 3 digital sun senors, 2 radio frequency (RF) sensors, 2 digital Infra-Red sensors. Body rates were measured by 3 orthogonal gyros and a redundant skewed one [Wadey 1991]. During normal on-station operations optical sensors were used to control the roll and the pitch axes, while the yaw axis was controlled via one component of the yaw gyro pack. In all the other cases, which imply the loss of the Earth pointing mode, the attitude measurement was provided by the gyros. Normal attitude control was accomplished through the use of reaction wheels. There were also thrusters which could be used in emergency maneuvers and in off-loading the reaction wheels.

The Olympus spacecraft's equations of motion are shown below. These equations are a standard represention of the spacecraft's dynamics assuming that inertial variations due to bending are negligible. These equations represent the system as modeled in the Olympus simulation used by this paper. The simulation was once used by British Aerospace in its validation and design of the Olympus satellite. The simulation model was therefore felt to be a very accurate representation of the Olympus spacecraft.

The spacecraft dynamical equation is

$$J\dot{\omega} + \omega \times (J\omega + \mathbf{h}) + \dot{\mathbf{h}} = M_{\text{ext}} + \psi\ddot{\mathbf{u}} \tag{1}$$

where $\omega \in \Re^3$ is the body angular velocity and $\mathbf{h} \in \Re^3$ is the reaction wheel momentum (S/C reference frame). In this context ω is one of the system states. The other system states are the attitude angles which are represented by the 3-vector $\alpha \in \Re^3$. The dynamics of the attitude angles are given by the standard kinematic equations.

$$\dot{\alpha} = \omega - \omega_o \tag{2}$$

where ω_o is the orbital angular rate (pitch). The other terms in equation 1 are $J \in \Re^{3\times3}$, the spacecraft's inertial matrix, $\psi \in \Re^{3\times n}$, the flexure moment of the modal shapes, $\mathbf{u} \in \Re^n$, the bending deflections, and $M_{\text{ext}} \in \Re^3$ a vector of disturbance torques accounting for aerodynamic drag, gravity gradients, and solar pressure. Bending mode dynamics are described by a standard set of n second order differential equations.

$$\ddot{\mathbf{u}} + M_b\ddot{\mathbf{u}} + 2\zeta\Omega\dot{\mathbf{u}} + \Omega^2\mathbf{u} = \psi^T\dot{\omega} \tag{3}$$

where $M_b \in \Re^{n\times n}$ is a square matrix of normalized integrals of the bending mode shapes, $\zeta \in \Re^{nxn}$ is a diagonal matrix of modal damping coefficients, and $\Omega \in \Re^{n\times n}$ is a diagonal matrix of modal frequencies.

As noted above, the nominal behaviour points the spacecraft at nadir. The control problem is therefore a regulation problem in which the body angles need to be kept near zero with as little variation as possible. One simple control system is a proportional derivative (PD) controller. Let $\tilde{\alpha} = \alpha - \alpha_d$, where α_d is the desired angular body coordinates then the control torque is

$$\tau = -\dot{\mathbf{h}} = -K_p\tilde{\alpha} - K_d\omega \tag{4}$$

where $K_p \in \Re^{3\times 3}$ and $K_d \in \Re^{3\times 3}$ are diagonal matrices. The actual spacecraft controller is a sophisticated digital implementation of a PID controller. Our simulations used a PD controller whose gains, K_p and K_d, were made equivalent to the proportional and derivative control gains in the actual spacecraft's digital control system. The nominal set of controller gains for this simulation are shown below.

$$K_p = \begin{pmatrix} 450 & 0 & 0 \\ 0 & 56 & 0 \\ 0 & 0 & 470 \end{pmatrix} \quad K_d = \begin{pmatrix} 490 & 0 & 0 \\ 0 & 310 & 0 \\ 0 & 0 & 510 \end{pmatrix} \tag{5}$$

3 Query-based Attitude Stabilization

The problem to be solved is as follows. Consider the controlled linear system

$$\begin{aligned} \dot{x} &= Ax + BKx & (6) \\ y &= x & (7) \end{aligned}$$

where $x \in \Re^n$, $A \in \Re^{n\times n}$, $B \in \Re^{n\times m}$, $K \in \Re^{m\times n}$. The matrix K is a matrix of controller gains which are assumed to stabilize the system matrix A. If the system matrix A changes in an unexpected manner, then the gain matrix K may no longer stabilize the system. In the context of the Olympus scenario, the spin disturbances introduced by the emergency maneuver would have changed the plant's linearized system matrix. The objective is to re-identify the stabilizing gains as quickly as possible.

In [Lemmon 1993], it was suggested that this problem might be solved by using an inductive inference protocol built around two things; a Lyapunov equation for the system, and an algorithm known as the ellipsoid method. The resulting procedure was an inductive inference protocol which can also be refered to as a query-based adaptation algorithm. On the basis of observed system states and state derivatives, the query-based algorithm constructs a sequence of gain matrices which converge to a stabilizing gain if one exists. The advantages of this method are its convergence after a finite number of steps and the fact that this convergence time scales in a polynomial manner with the number of control gains to be determined. This implies that such algorithms yield direct adaptive controllers which are able to quickly learn how to stabilize the plant when critical destabilizing events arise during the system's lifetime.

A block diagram of the algorithm is shown in figure 1. Upon detection of the plant's instability via a stability-oracle, the algorithm initializes itself. The initialization procedure

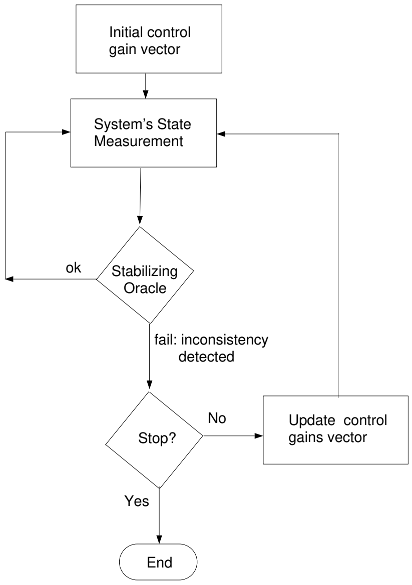

Figure 1: Flowchart for Query-based Stabilization Algorithm.

constructs an ellipsoid bounding the set of stabilizing gains. The system then begins observing plant states using the current set of control gains. If the stability oracle declares that an observed state and state derivative are inconsistent with the conjecture that these current gains stabilize the system, then a single iteration of the ellipsoid algorithm is used to update the gains. This process of observing plant states, querying the stability-oracle, and updating the gain matrices is then repeated until no further inconsistencies are declared by the stability-oracle.

Figure 1 shows that the algorithm is based upon two fundamental components; the stability oracle and the update algorithm. The following subsections discuss these two algorithm components in more detail.

3.1 Stability-Oracle

For autonomous LTI systems (see equation 6 with $K = 0$), a necessary and sufficient condition that the system be stable is that there exist positive definite matrices P and Q satisfying the Lyapunov equation,

$$A^T P + PA = -Q \tag{8}$$

Once P is known, then the Lie derivative of the functional $V = x^t Px$ with respect to the vector field Ax is negative semi-definite. Therefore for all $x \in \Re^n$ and $\dot{x} = Ax$, the following inequality must hold

$$\begin{aligned} \dot{V} &= 2\dot{x}^T Px && (9)\\ &= 2x^t(A^t P + PA)x && (10)\\ &< 0 && (11) \end{aligned}$$

Any system for which inequality 11 holds for all x is said to be quadratically stabilizable with respect to P. Such systems are also said to be P-stable.

If there is any x for which inequality 11 does not hold then, the system is not P-stable. Inequality 11 therefore provides a sufficient test for a P-stable system. This condition can therefore be used as the basis for the stability oracle. In particular, the stability-oracle is a Boolean functional of the form

$$\mathcal{O}(x, \dot{x}) = \begin{cases} 0 & \text{if } \dot{x}_{i-1}^T P x_{i-1} - \dot{x}_i^T P x_i < 0 \\ 1 & \text{otherwise} \end{cases} \tag{12}$$

The stability oracle therefore provides a declaration (0,1) about whether or not the current plant gains are consistent with the hypothesis that the controlled plant is P-stable.

A key issue in the design of query-based adaptive control systems concerns the selection of P. Prior work [Lemmon 1993] assumed that P was known and an example was used in which P was the identity matrix. Unfortunately, the set of plants for which $P = I$ is extremely small. A better way to determine P is suggested in [Boyd 1989]. In this approach, it is assumed that the system matrix, A, lies in a matrix interval, $[A_1, A_2]$. Letting M_i ($i = 1, \ldots, N$) be a finite set of square symmetric matrices, where $N = \frac{n(n+1)}{2}$ and n here indicates the number of gains to be determined. we define a basis for a matrix G,

$$G_i = -M_i \oplus (A_1^T M_i + M_i A_1) \oplus (A_2^T M_i + M_i A_2) \tag{13}$$

where $A \oplus B$ denotes the block diagonal matrix with A and B on its diagonals. With these definitions Boyd [Boyd 1989] has shown that there exists a Lyapunov matrix P for which all matrices in $[A_1, A_2]$ are P-stable if and only if there exist positive coefficients a_i ($i = 1, \ldots, N$) such that

$$G = \sum_{i=1}^{N} a_i G_i < 0 \tag{14}$$

Note that since equation 14 is affine in the coefficients a_i, linear programming algorithms can be used to find a_i. The associated programming problem seeks to find those a_i which minimize the largest eigenvalue of G. In particular let $\lambda_0(a_1, \ldots, a_N)$ denote the largest eigenvalue of G, then our the coefficients yielding the desired P matrix exist if and only if $\lambda_0^* < 0$ where

$$\lambda_0^* = \min_{|a_i| \leq 1} \lambda_0(a_1, \ldots, a_N) \tag{15}$$

This is a constrained optimization problem which is linear in the coefficients a_i.

To obtain the P matrix for the Olympus scenario, we first need to derive the nominal linearized system matrix A for the spacecraft. Equation 1 can be rewritten as

$$(J - \psi\psi^T)\dot{\omega} - [(J\omega + \mathbf{h})\times]\omega = -\dot{\mathbf{h}} + M_{\text{ext}} - 2\psi\zeta\Omega\dot{\mathbf{u}} - \psi\Omega^2\mathbf{u} \tag{16}$$

where $[(J\omega + \mathbf{h})\times]$ is the skew symmetric matrix associated with vector cross product. This equation can be rewritten as

$$J^*(\omega, \alpha)\ddot{\alpha} + \zeta^*(\omega, \alpha, \dot{\omega})\dot{\alpha} = -\dot{\mathbf{h}} + \mathbf{d} \tag{17}$$

where

$$J^*(\omega,\alpha) = (J - \psi\psi^T) \tag{18}$$
$$\zeta^*(\omega,\alpha,\dot\omega) = -[(J\omega + \mathbf{h})\times] \tag{19}$$
$$\mathbf{d} = M_{\text{ext}} - 2\psi\zeta\Omega\dot{\mathbf{u}} - \psi\Omega^2\mathbf{u} + [(J\omega+\mathbf{h})\times]\omega_o \tag{20}$$

These equations can be rewritten as a set of coupled linearized first order differential equation.

$$\frac{d}{dt}\begin{pmatrix}\alpha\\ \dot\alpha\end{pmatrix} = \begin{pmatrix}0 & I\\ 0 & -(J^*)^{-1}\zeta^*\end{pmatrix}\begin{pmatrix}\alpha\\ \dot\alpha\end{pmatrix} + \begin{pmatrix}-(J^*)^{-1}(\dot{\mathbf{h}}+\mathbf{d})\\ 0\end{pmatrix} \tag{21}$$

The preceding equation does not include the controller gain matrices. Using equation 4, the system matrix A becomes

$$A(K_p,K_d) = \begin{bmatrix}0 & I\\ (J^*)^{-1}K_p & (J^*)^{-1}(\zeta^* + K_d)\end{bmatrix} \tag{22}$$

For the nominal gains used in the Olympus satellite and the nominal system parameters, this yields a linearized system matrix of the form

$$\mathcal{A} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 0 & 0 & 1\\ -0.0684 & -0.0002 & 0.0014 & -0.0745 & -0.0010 & 0.0017\\ -0.0014 & -0.0304 & 0.0002 & -0.0015 & -0.1661 & 0.0003\\ 0.0013 & 0.0000 & -0.0829 & 0.0012 & 0.0001 & -0.0900 \end{bmatrix} \tag{23}$$

This system matrix was computed under the assumptions that the system was operating close to its desired setpoint.

Now that the linearized system matrix, $A(K_p, K_d)$,has been determined, the Olympus S/C's P matrix can be computed. In this case, we wanted to determine the P matrix so that all plants with system matrix $A(K_p \pm \Delta, K_d \pm \Delta)$ where $\delta = .2I$ would be P-stable. An interior point optimization algorithm [Gonzaga 1992] was used to solve the associated linear program. The resulting P matrix is shown below.

$$P = \begin{pmatrix} 0.0593 & 0.0000 & -0.0003 & 0.0258 & 0.0011 & 0.0005\\ 0.0000 & 0.0062 & 0.0000 & -0.0011 & 0.0151 & 0.0000\\ -0.0003 & 0.0000 & 0.0576 & 0.0000 & 0.0001 & 0.0231\\ 0.0258 & -0.0011 & 0.0000 & 1.0000 & -0.0069 & -0.0155\\ 0.0011 & 0.0151 & 0.0001 & -0.0069 & 0.2475 & 0.0006\\ 0.0005 & 0.0000 & 0.0231 & -0.0155 & 0.0006 & 0.8612 \end{pmatrix} \tag{24}$$

Figure 2 shows the evolution of the Lyapunov function V assuming the P matrix computed above and assuming $P = I$. Note that if $P = I$, then the function is not monotone decreasing and hence is not a Lyapunov function for this system. On the otherhand, for the P matrix computed above, the function V is clearly monotone decreasing as would be expected of a valid Lyapunov functional. Furthermore, this Lyapunov functional is not only valid for the nominal plant, but it is also nominal for any system within the matrix interval.

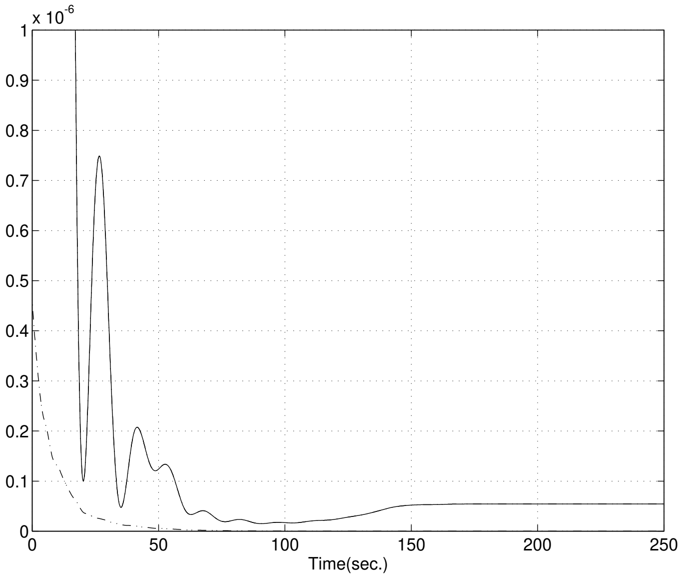

Figure 2: Time history of candidate Lyapunov functionals for the Olympus S/C, for the case P=I (–) and our P (-.).

3.2 Ellipsoid Update

The oracle is used to declare whether or not the system's current state is consistent with the hypothesis that the system is P-stable. If the declaration is false, then the update algorithm needs to compute a new set of controller gains which are consistent with the previously observed states. The ellipsoid method [Groetschel 1988] provides just such an algorithm. The use of this algorithm was the primary innovation suggested in [Lemmon 1993]. In this section we summarize that algorithm and previous results on its convergence rate.

Let the set of gains K_p and K_d which stabilize a given plant be denoted as $\mathcal{K}$. The ellipsoid method starts with an initial ellipsoid whose radius is large enough to include $\mathcal{K}$. Let K denote a vector consisting of the plant's current control gains, K_p and K_d. Let x and $\dot{x}$ denote recent observations of the state and its time rate of change. Assume that the stability oracle declares this data, $(x,\dot{x})$ to be inconsistent with the hypothesis that K renders the plant P-stable. In this case, the inequality (11) does not hold and we can quickly infer that any control gain K^* such that

$$x^T(A^T(K^*)P + PA^T(K^*))x > x^T(A^T(K)P + PA^T(K))x \tag{25}$$

will not P-stabilize the plant either. Since this inequality is affine in $K^* - K$, it can be rewritten as a linear inequality constraint on the infeasible control gains. This inequality constraint is denoted as

$$c^T K^* > c^T K \tag{26}$$

By removing this halfspace of gains from the initial ellipsoid, a convex body is obtained which is not only smaller than the original ellipsoid but still bounds the set of P-stabilizing gains $\mathcal{K}$. A minimal volume ellipsoid containing this convex body can be computed using the central-cut ellipsoid algorithm [Groetschel 1988]. This then represents the basic operation of the update algorithm shown in figure 1.

Previous papers have shown that this update algorithm will converge in finite time to a set of P-stabilizing gains.

Theorem 1 *[Lemmon 1993] Let E denote the ellipsoid of maximal volume v contained within the set of P-stabilizing gains $\mathcal{K}$. Let V denote the volume of the initial ellipsoid, containing $\mathcal{K}$, used by the query-based adaptation algorithm. Let n be the number of control gains. The query-based adaptation algorithm will identify a stabilizing gain vector (if one exists) after no more than $2n \ln \frac{V}{v}$ failed oracle queries.*

The proof is found in [Lemmon 1993]. It is a consequence of the following relation between the volumes of two subsequent ellipsoids computed by the central-cut ellipsoid method,

$$\frac{Vol(E_{i+1})}{Vol(E_i)} < e^{-\frac{1}{2n}} \tag{27}$$

Using this relation at the N-th iteration and recognizing that the volume of Nth ellipsoid cannot be less than v yields the desired result.

4 Simulation Results

The Query-Based learning protocol described in the previous section was applied to the Olympus scenario discussed above. The control objective was to keep the three attitude coordinates; i.e. the roll (ϕ), the pitch (θ) and the yaw (ψ) angles close to the set point zero. In the Olympus scenario, the spacecraft enters into an anomalous spinning motion. The objective is to return the spacecraft to its non-spinning Earth pointing attitude. The problem is that increased spin rates perturb the system matrix (23) so that the original control gains no longer work well. In these simulations, the spin disturbance occurs at time $t = 100$ seconds.

Tracking performance of the fixed PD controller and the Query-Based adaptive controller are shown in figure 3. In this example, the spin disturbance produces an angular velocity vector of [0.001,0.016,0.0005] rad/sec. Comparing the two cases shows that the Query-Based controller eliminated large pitch oscillations that were present when the conventional PD controller was used.

Figure 4 shows the torque signal commands to the roll reach wheel. This figure gives a measure of the control effort expended in controlling the perturbed system. Note that the query-based controller appears to exert less effort than the conventional PD controller. Integrating the squared torque command shows that the control effort for the roll actuator has been reduced from 0.252 to 0.11 through the use of the query-based method.

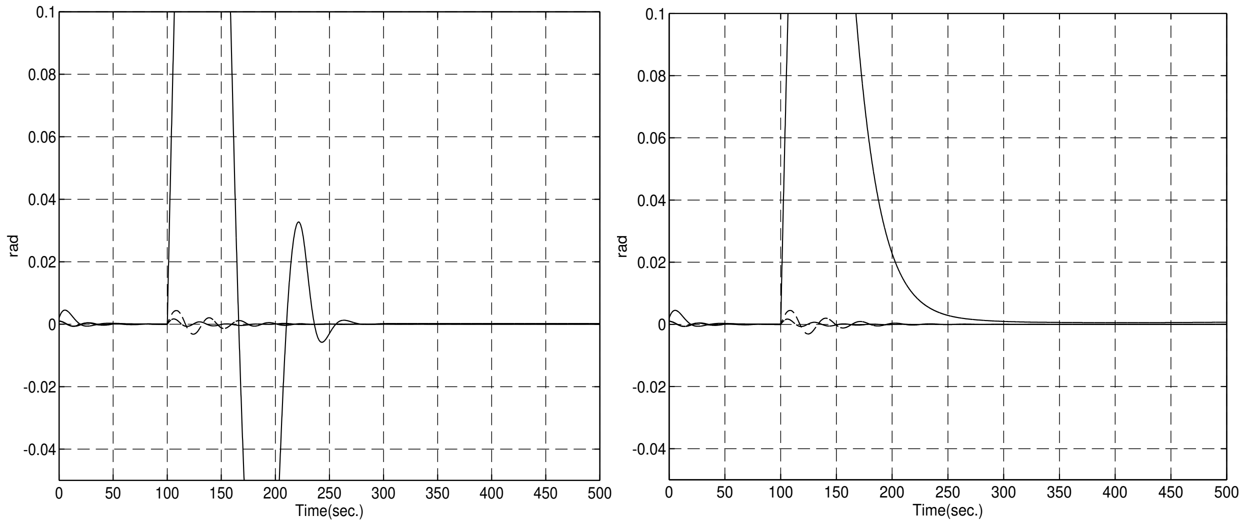

Figure 3: Roll(-.), Pitch (–), Yaw angles(- -) in the case of conventional fixed PD controller (left) and Query-Based adaptive controller (right).

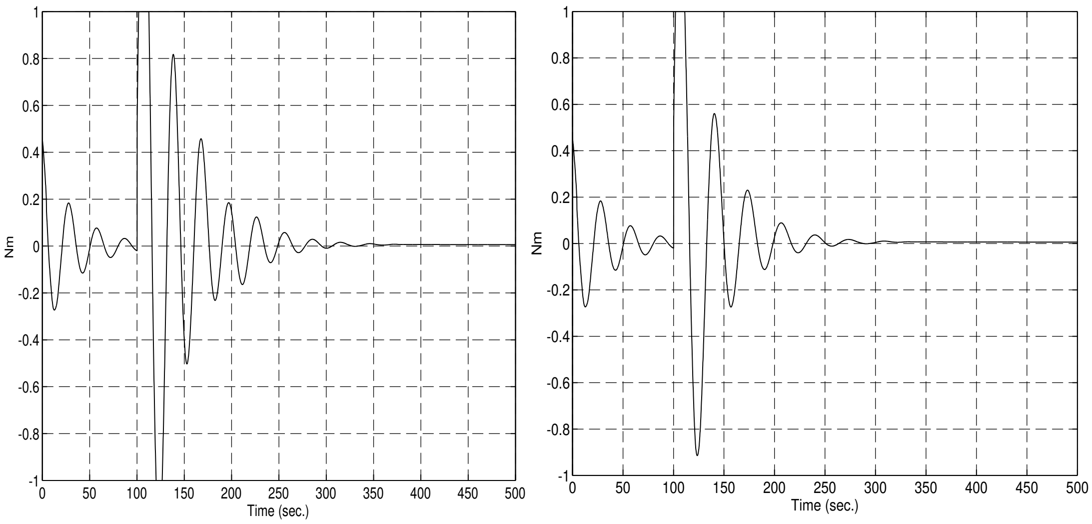

Figure 4: Torque along the roll axes reaction wheel, in the case of fixed PD controller (left) and Query Based one (rigth).

More extensive simulation results examined the query-based algorithm's performance for a range of initial spin disturbances. In these simulation experiments, the spin rate was varied between 0.5 deg/sec and 3 deg/sec. Figure 5 compares the squared integrated torque command exerted by the query-based and conventional PD controllers. The results show that for small spin rates, the two approaches are comparable. For the moderate range of spin rates, the query-based method appears to recover the setpoint with significantly less energy being expended. As the disturbance rate increases the energy savings reduce because of the reaction wheel's limited ability to absorb higher spin rates.

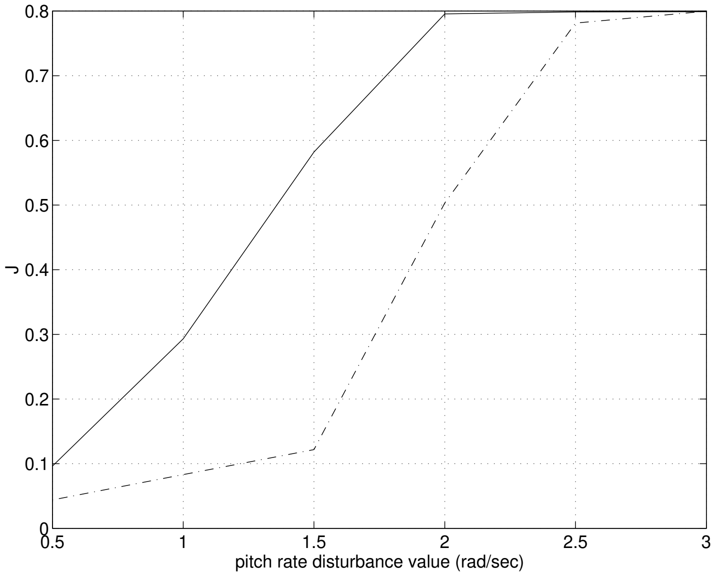

Figure 5: Fixed PD (–) and Query Based (- -) controller energy consumption.

Recall that the stability-oracle determined a P matrix such that $x^t P x$ is a Lyapunov functional for any plant within a specified matrix interval. In our example, this matrix interval was obtained by perturbing the controller gains. We therefore know the volume of the set of "target" gains our query-based algorithm is trying to identify. Inserting this information into theorem 1, an upper bound on the query-based method's convergence time can be quickly computed. For this particular example, the stabilizing control gain must be acquired after no more than 330 updates. Figure 6 shows the number of updates used by the query-based algorithm in stabilizing the plant. As can be seen, increasing the spin disturbances makes the stabilizing gains more difficult to identify and hence more updates are needed. In all cases, however, the actual number of updates is bounded above by the computed theoretical bound.

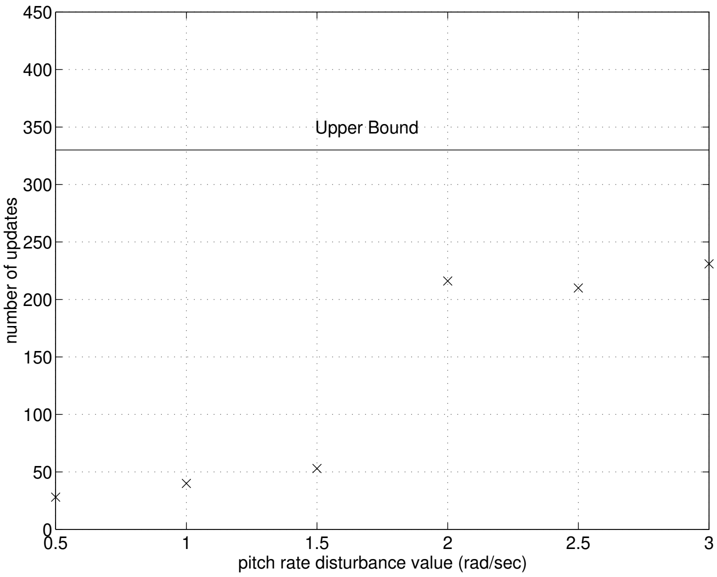

Figure 6: Updates versus disturbance spin rate. The symbol $\times$ indicates the number of updates required by the Query Based algorithm.

5 Summary

This paper evaluated the feasibility of using query-based adaptation schemes in spacecraft attitude control. The principal results of this paper are two fold. First, the paper showed how stability oracles could be obtained by finding the P matrix for which an interval of nominal plants were P-stable. Based on the formalisms developed in [Boyd 1989], it was demonstrated how the P matrix could be computed by solving a linear programming problem. Second, the paper applied the resulting algorithm to a highly detailed simulation of the Olympus telecommunication satellite. The simulation experiments evaluated the query-based algorithm's performance on a scenario which was responsible for the early termination of the actual satellite's mission. In this scenario, efforts at controlling large spin rate disturbances depleted propellant reserves to a level where the mission had to be terminated a year early. Simulation results showed that the query-based method was capable of controlling spin disturbances with lower control effort than the conventional PD controller used in the system. This result therefore suggests that the use of query-based methods may have been able to extend mission life by conserving limited fuel reserves.

References

[Boyd 1989] S. Boyd, Q. Yang. *Structured and Simultaneous Lyapunov Functions for System Stability Problems.* Int. J. Control, 1989, vol. 49, No. 6, 2215-2240.

[Groetschel 1988] M. Groetschel, L. Lovasz and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization.* Springer-Verlag, 1988. MacGraw-Hill, 1983.

[Wadey 1991] P. Wadey, P. Miles and R. Lowes. *Some Aspects of the Attitude and Orbit Control Subsystem of the Olympus Spacecraft.* Proc. First ESA Internat. Conf. on 'Spacecraft Guidance, Navigation and Control Systems', ESTEC, Noordwijk, The Nederlands, pp. 545-549, June 1991.

[Gonzaga 1992] Clovis C. Gonzaga. *Path-Following Methods for Linear Programming.* SIAM REVIEW, Vol. 34, No. 2, pp. 167-224, June 1992

[Lemmon 1993] M. D. Lemmon, C. J. Bett. *Direct Adaptive Stabilization of Linear Systems Using Query-Based Protocols.* Proc. of the 32th Conference on Decision and Control, San Antonio, Texas- December 1993, vol. 4, 3086-3091.

[Lemmon 1995] M. D. Lemmon, P.J. Antsaklis, *Inductively inferring valid logical models of continuous-state dynamical systems*, Theoretical COmputer Science B, to appear in Vol 137, Jan 1994.

[Lemmon 1992] M. D. Lemmon, *Ellipsoidal Method for the estimation of sliding mode domains of Variable Structure Systems*, Conference on Information and Systems (CISS), pp. 1018-1023, Princeton, NJ 1992.