DESIGN OF LQR WEIGHTING MATRICES FOR TIME VARYING OUTPUT COVARIANCE ASSIGNMENT

Conference Paper · August 2021

CITATIONS
0

READS
209

4 authors, including:

Vishala Arya
NASA JPL
18 PUBLICATIONS **97** CITATIONS
SEE PROFILE

Raman Goyal
Palo Alto Research Center
48 PUBLICATIONS **290** CITATIONS
SEE PROFILE

John Junkins
Texas A&M University
476 PUBLICATIONS **14,466** CITATIONS
SEE PROFILE

DESIGN OF LQR WEIGHTING MATRICES FOR TIME VARYING OUTPUT COVARIANCE ASSIGNMENT

Vishala Arya*, Raman Goyal†, Manoranjan Majji‡ and John L. Junkins§

A time-varying output covariance assignment problem in the presence of a stochastic disturbance is solved using finite-horizon optimal control formulation. It is shown that an assignment of time-varying output error covariance is possible in the presence of model error by utilizing a time varying linear quadratic regulator (LQR) controller with a class of output and control weighting sequences. The paper develops a systematic algorithm to calculate the sequence of such time varying output weights which are further shown to be the Lagrange multipliers associated with the covariance constraints. A short horizon attitude control problem with stringent covariance constraints is solved to demonstrate the utility of the proposed approach. Numerical results offer a degree of optimism about the broad applicability of the time varying covariance assignment approach to solve guidance and control problems associated with nonlinear dynamical systems.

Aerospace vehicle's guidance and control problems have been solved over recent decades for many space and aerial applications where they were largely restricted to spacecraft and aircraft attitude regulation. Recent advances have opened up the arena for various other applications like unmanned aerial vehicles (UAVs), underwater vehicles (AUVs), miniaturized satellites like CubeSats, and close mapping of irregular asteroids [1] such that the requirement of the maneuver efficiency and the class of constraints associated with them have become challenging to accommodate and crucial to solve for successful operation. Spacecraft control is accomplished either by open-loop control or closed-loop schemes. The ope loop approach is most often used to design a nominal trajectory and feedback closed-loop control is typically used to track the nominal trajectory. Using an open-loop approach, real time control works for a deterministic setting and is quite sensitive to model errors, parameter uncertainties and noise; all of which closed-loop schemes are more adept at handling [2].

Open-loop control mainly constitutes of using optimal control techniques or hybrid methods [3] to evaluate the pointing control, by solving a two point boundary value problem (TPBVP) assuming continuous and more recently even discrete control variables [4]. Closed-loop or feedback control techniques that have been traditionally used are Lyapunov functions to provide asymptotic stability

*PhD Student, Department of Aerospace Engineering, Texas A&M University, College Station, Texas 77843-3141, AIAA Member.

†Postdoctoral Researcher, Department of Aerospace Engineering, Texas A&M University, College Station, Texas 77843-3141.

‡Assistant Professor, Department of Aerospace Engineering, Texas A&M University, College Station, Texas 77843-3141, AIAA Member.

§Distinguished Professor, Department of Aerospace Engineering, Texas A&M University, College Station, Texas 77843-3141, Honorary Fellow AIAA.

characteristics, sliding mode control [5] that utilizes feedback linearization and an add-on term for uncertainty and robust control schemes like H_∞ control that provides spacecraft stability in the presence of disturbances by solving the Hamilton-Jacobi-Isaacs inequalities. Another class of controllers that have been shown to be very robust to model errors and disturbances are based on adaptive control techniques [6]. Adaptive techniques [7] estimate the external torques by tracking a Lyapunov function and updating the model in operation based on measured performances.

There has also been recent research in the area of covariance control, where its importance on conducting a stochastic mission analysis was first recognized due to its comparable accuracy to a Monte Carlo analysis with lesser computational effort [8]. Linear covariance techniques have been presented as an efficient methodology to design and statistically analyze the stability of the closed-loop systems and the performance of the desired outputs [9, 10]. Concurrently, covariance analysis has also been used specifically for bounding the output and state covariance at the final time for both continuous and discrete-time systems [10–12]. These works were limited to bounding the steady-state covariance for only the linear time-invariant systems [11–13].

The covariance control problem has recently been studied with applications in space mission designs (power descent guidance algorithm) and vehicle path planning [14–16]. These applications have created a thrust to move towards the finite-horizon covariance control of linear time-varying systems with different types of constraints [17–19]. In particular, the main focus has been on the steering of covariance from known Gaussian distribution to desired Gaussian distribution in finite-time for both discrete-time and continuous-time systems [20–22]. Usually, a fast and reliable solution can be obtained for these steering problems by posing them as a convex optimization problem [22, 23]. These finite-horizon covariance steering problems have been explored in the presence of both convex and non-convex state constraints [23], where the addition of convex constraints still allow the complete problem to be solved as a convex optimization problem and non-convex constraints are handled by dividing the constraint into a set of convex sets and then solving a mixed-integer convex program [23]. The above-mentioned convex and non-convex constraints can also be formulated as chance constraints which allow to satisfy the constraints with specified (high) probability [15, 24]. A solution for the nonlinear dynamical system was also recently proposed as an iterative covariance steering with convex state chance constraints [16].

In these previous works that have considered linear covariance analysis, the approach has been largely passive and not definitive in assigning a particular covariance profile to the linearized system. These approaches also restrict the placement of a bound on the output covariance only at the final time and attempt simply minimizing the covariance during the course of the maneuver [22, 23]. The main contribution of the present paper lies in establishing a systematic methodology that delivers the weighting matrices of the LQR such that the specified covariance bounds are satisfied at each time. The paper also allows to convert the output covariance assignment problem, which generally has a very specialized solution framework, to a very general and well-known LQR framework. Moreover, the traditional random guessing of the constant weighting matrices in LQR is replaced by a systematic iterative process that yields time-varying $\boldsymbol{Q}$ or $\boldsymbol{R}$ matrices to satisfy a pre-specified time-varying covariance constraint on the system output. In this work, we solve a finite-horizon optimal control problem for a discrete linear time-varying system subject to a stochastic disturbance assuming fully observable states. The novelty of the method lies in finding judicious time-varying values of $\boldsymbol{Q}$ which deliver controls that satisfy tighter covariance bounds to suit the natural dynamics of the system in the presence of stochastic disturbance, with unbounded control inputs. This is especially useful when higher precision is required in the maneuvers due to the enhanced capability

of balancing feedback control effort with the desired covariance at any time instant. While the procedure does not explicitly enforce actuator limits, it implicitly minimizes the control effort by modulating the $\boldsymbol{Q}$ matrix. The authors believe that this is the first time such a time-varying closed-loop covariance assignment technique for LTV systems has been proposed and demonstrated.

The formulation of this paper is as follows: Notation and system definitions are provided in Section II. Section II further formulates the problem statement to bound the output covariance at each time. Section III gives the main result of the paper to obtain the time-varying weighting matrices $\boldsymbol{Q}$ that minimizes the cost function associated with control input while bounding the output covariance. The associated "Covariance Assignment via Weight Selection Algorithm" is also presented in Section III. Section IV develops the specific framework to solve a spacecraft attitude control problem. The simulation results using the proposed algorithm and its comparison with LQR control for constant $\boldsymbol{Q}$ values are given to show the efficacy of the formulation in Section V.

BOUNDED OUTPUT COVARIANCE ASSIGNMENT

Notation

We denote by $\mathbb{R}^n$ the set of n-dimensional real vectors. The matrices are defined by bold uppercase letters as $\boldsymbol{Y}$. The expectation operator is defined by $\mathbb{E}[\cdot]$ and $\mathcal{N}(x, \boldsymbol{Y})$ denotes the Gaussian distribution with mean x and covariance $\boldsymbol{Y}$. The diagonal matrix generated from a vector $\boldsymbol{x}$ is denoted as $\textbf{diag}(\boldsymbol{x})$, the block diagonal matrix is denoted as $\textbf{blkdiag}(\boldsymbol{Y}_1, \boldsymbol{Y}_2, \cdots, \boldsymbol{Y}_3)$, and the transpose of a matrix $\boldsymbol{Y}$ is defined by $\boldsymbol{Y}^T$. The symbol $\mathbf{0}$ defines a zero matrix with suitable dimensions. The notations $\boldsymbol{X} \succ 0$ and $\boldsymbol{Y} \succeq 0$ denote the symmetric positive definite ($\mathbb{S}_n^{++}$) and symmetric positive semidefinite ($\mathbb{S}_n^{+}$) matrices, respectively.

System Definition

A discrete-time linear time-varying system is described by the following state-space representation:

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}_k \boldsymbol{x}_k + \boldsymbol{B}_k \boldsymbol{u}_k + \boldsymbol{D}_k \boldsymbol{\nu}_k, \tag{1}$$

$$\boldsymbol{y}_k = \boldsymbol{C}_k \boldsymbol{x}_k, \tag{2}$$

where $\boldsymbol{x}_k \in \mathbb{R}^n$ is the state of the system at time-step k, $\boldsymbol{u}_k \in \mathbb{R}^m$ is the control vector, $\boldsymbol{y}_k \in \mathbb{R}^p$ is the output of the system, and $k = \{0, 1, \cdots, N\}$. The noise in the system is added as process noise $\boldsymbol{\nu}_k$ which is modeled as independent zero mean white noises with covariance $\mathbb{W} \in \mathbb{S}^{++}$, i.e.:

$$\mathbb{E}[\boldsymbol{\nu}] = \mathbf{0},\ \mathbb{E}[\boldsymbol{\nu}\boldsymbol{\nu}^T] = \mathbb{W}, \tag{3}$$

where $\mathbb{E}[x]$ denotes the expected value of the random variable x. We assume the process noise covariance $\mathbb{W}$ to be known and fixed. Let us define the covariance of the state $\boldsymbol{x}_k$ and output $\boldsymbol{y}_k$ at any time k as:

$$\boldsymbol{P}_k \triangleq \mathbb{E}[\boldsymbol{x}_k \boldsymbol{x}_k^T], \tag{4}$$

$$\boldsymbol{Y}_k \triangleq \mathbb{E}[\boldsymbol{C}_k \boldsymbol{x}_k \boldsymbol{x}_k^T \boldsymbol{C}_k^T] = \boldsymbol{C}_k \boldsymbol{P}_k \boldsymbol{C}_k^T. \tag{5}$$

Problem Statement

The final problem statement is to find the feedback control $u_k = \boldsymbol{K}_k x_k$, such that the output covariance at all times can be assigned as:

$$\boldsymbol{Y}_k = \boldsymbol{C}_k \boldsymbol{P}_k \boldsymbol{C}_k^T = \bar{\boldsymbol{Y}}_k, \text{ for } k = 1, \cdots, N+1, \tag{6}$$

and the cost function J:

$$J = \mathbb{E}\left[\sum_{k=0}^{N} u_k^T \boldsymbol{R}_k u_k\right], \tag{7}$$

is minimized for a given sequence of $\boldsymbol{R}_k$ and commanded covariance profile ($\bar{\boldsymbol{Y}}_k$).

MAIN RESULT AND ALGORITHM

This section first provides the main theorem of the paper which gives the formulation to calculate the gain matrix $\boldsymbol{K}_k$ to solve the above mentioned problem statement for some chosen values of Lagrange Multipliers $\boldsymbol{Q}_k$. It is further observed that the solution for the problem statement has the same formulation as that of Linear Quadratic Regulator (LQR) problem where weighting matrix corresponding to state cost $\boldsymbol{Q}_k$ has taken the place of Lagrange Multipliers. The section further provides an iterative algorithm to calculate these weighting matrices $\boldsymbol{Q}_k$ to bound the output covariance while simultaneously minimizing the given cost function J.

Theorem: If there exists a feedback controller which minimizes the cost function:

$$J = \mathbb{E}\left[\sum_{k=0}^{N} u_k^T \boldsymbol{R}_k u_k\right], \tag{8}$$

and satisfies the constraints given as: $\boldsymbol{C}_k \boldsymbol{P}_k \boldsymbol{C}_k^T = \bar{\boldsymbol{Y}}_k$, for $k = 1, \cdots, N+1$, for the discrete-time time-varying linear system described by the system equation (Eq (1)) and the output equation ((Eq (2)), then the solution for the time varying feedback gain is given by

$$\boldsymbol{K}_k = -\left(\boldsymbol{R}_k + \boldsymbol{B}_k^T \boldsymbol{S}_{k+1} \boldsymbol{B}_k\right)^{-1} \boldsymbol{B}_k^T \boldsymbol{S}_{k+1} \boldsymbol{A}_k. \tag{9}$$

and the discrete time Riccatti equation for some choice of Lagrange multipliers $\boldsymbol{Q}_k$ for $k = 1, 2, \cdots, N+1$ is given as:

$$\boldsymbol{S}_k = \boldsymbol{C}_k^T \boldsymbol{Q}_k \boldsymbol{C}_k + \boldsymbol{A}_k^T \left(\boldsymbol{S}_{k+1} - \boldsymbol{S}_{k+1} \boldsymbol{B}_k \left(\boldsymbol{R}_k + \boldsymbol{B}_k^T \boldsymbol{S}_{k+1} \boldsymbol{B}_k\right)^{-1} \boldsymbol{B}_k^T \boldsymbol{S}_{k+1}\right) \boldsymbol{A}_k, \tag{10}$$

with the terminal condition $\boldsymbol{S}_{N+1} = \boldsymbol{C}_{N+1}^T \boldsymbol{Q}_{N+1} \boldsymbol{C}_{N+1}$.

Proof: Let us start by assuming the solution to be of the state-feedback form given as:

$$\boldsymbol{u}_k = \boldsymbol{K}_k \boldsymbol{x}_k. \tag{11}$$

The system dynamics equation for the above-mentioned control can be written as:

$$\boldsymbol{x}_{k+1} = (\boldsymbol{A}_k + \boldsymbol{B}_k \boldsymbol{K}_k) \boldsymbol{x}_k + \boldsymbol{D}_k \boldsymbol{\nu}_k, \tag{12}$$

and the corresponding covariance propagation equation for the state can be written as:

$$\boldsymbol{P}_{k+1} = \mathbb{E}[\boldsymbol{x}_{k+1}\boldsymbol{x}_{k+1}^T], \tag{13}$$

$$\boldsymbol{P}_{k+1} = (\boldsymbol{A}_k + \boldsymbol{B}_k\boldsymbol{K}_k)\boldsymbol{P}_k(\boldsymbol{A}_k + \boldsymbol{B}_k\boldsymbol{K}_k)^T + \boldsymbol{D}_k\mathbb{W}_k\boldsymbol{D}_k^T, \tag{14}$$

Now, let us write the cost function J in terms of state covariance $\boldsymbol{P}_k$ as:

$$J = \mathbb{E}\left[\sum_{k=0}^{N} \boldsymbol{x}_k^T\boldsymbol{K}_k^T\boldsymbol{R}_k\boldsymbol{K}_k\boldsymbol{x}_k\right] = \sum_{k=0}^{N} \mathrm{Tr}\left(\mathbb{E}\left[\boldsymbol{K}_k\boldsymbol{x}_k\boldsymbol{x}_k^T\boldsymbol{K}_k^T\boldsymbol{R}_k\right]\right), \tag{15}$$

$$J = \sum_{k=0}^{N} \mathrm{Tr}\left(\boldsymbol{R}_k\boldsymbol{K}_k\boldsymbol{P}_k\boldsymbol{K}_k^T\right), \tag{16}$$

where $\mathrm{Tr}(\cdot)$ is the trace operator. Using the cost function mentioned in (Eq (16), the Lagrangian function corresponding to the constraint optimization problem can be written as:

$$\mathcal{L} = \sum_{k=0}^{N} \mathrm{Tr}\left(\boldsymbol{R}_k\boldsymbol{K}_k\boldsymbol{P}_k\boldsymbol{K}_k^T\right) + \sum_{k=0}^{N} \mathrm{Tr}\left(\boldsymbol{S}_{k+1}(\boldsymbol{P}_{k+1} - \hat{\boldsymbol{A}}_k\boldsymbol{P}_k\hat{\boldsymbol{A}}_k^T - \boldsymbol{D}_k\mathbb{W}_k\boldsymbol{D}_k^T)\right) + \sum_{k=1}^{N+1} \mathrm{Tr}\left(\boldsymbol{Q}_k(\boldsymbol{C}_k\boldsymbol{P}_k\boldsymbol{C}_k^T - \bar{\boldsymbol{Y}}_k)\right), \tag{17}$$

where $\hat{\boldsymbol{A}}_k = \boldsymbol{A}_k + \boldsymbol{B}_k\boldsymbol{K}_k$, and $\boldsymbol{S}_{k+1}$ and $\boldsymbol{Q}_k$ are the Lagrange multipliers corresponding to covariance propagation equation (Eq (14)) and output covariance constraints (Eq (6)).

Now, the partial of the Lagrange function w.r.t. $\boldsymbol{P}_{N+1}$ gives:

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{P}_{N+1}} = -\boldsymbol{S}_{N+1} + \boldsymbol{C}_{N+1}^T\boldsymbol{Q}_{N+1}\boldsymbol{C}_{N+1}, \tag{18}$$

the partial of the Lagrange function w.r.t. $\boldsymbol{P}_k$ gives:

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{P}_k} = \boldsymbol{K}_k^T\boldsymbol{R}_k\boldsymbol{K}_k + \hat{\boldsymbol{A}}_k^T\boldsymbol{S}_{k+1}\hat{\boldsymbol{A}}_k - \boldsymbol{S}_k + \boldsymbol{C}_k^T\boldsymbol{Q}_k\boldsymbol{C}_k, \tag{19}$$

and the partial of the Lagrange function w.r.t. $\boldsymbol{K}_k$ gives:

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{K}_k} = 2\boldsymbol{R}_k\boldsymbol{K}_k\boldsymbol{P}_k + 2\boldsymbol{B}_k^T\boldsymbol{S}_{k+1}\boldsymbol{A}_k\boldsymbol{P}_k + 2\boldsymbol{B}_k^T\boldsymbol{S}_{k+1}\boldsymbol{B}_k\boldsymbol{K}_k\boldsymbol{P}_k. \tag{20}$$

The partials w.r.t. the Lagrange multipliers $\boldsymbol{S}_{k+1}$ and $\boldsymbol{Q}_k$ gives back the covariance propagation equation (Eq (14)) and output covariance constraints (Eq (6)).

Now, using the following necessary conditions for the minimization of the Lagrange function:

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{P}_{N+1}} = \boldsymbol{0}, \;\; \frac{\partial\mathcal{L}}{\partial\boldsymbol{P}_k} = \boldsymbol{0}, \;\; \frac{\partial\mathcal{L}}{\partial\boldsymbol{K}_k} = \boldsymbol{0}, \tag{21}$$

we obtain:

$$\boldsymbol{S}_{N+1} = \boldsymbol{C}_{N+1}^T\boldsymbol{Q}_{N+1}\boldsymbol{C}_{N+1}, \tag{22}$$

$$\boldsymbol{K}_k^T\boldsymbol{R}_k\boldsymbol{K}_k + \hat{\boldsymbol{A}}_k^T\boldsymbol{S}_{k+1}\hat{\boldsymbol{A}}_k + \boldsymbol{C}_k^T\boldsymbol{Q}_k\boldsymbol{C}_k = \boldsymbol{S}_k, \tag{23}$$

and

$$\boldsymbol{R}_k\boldsymbol{K}_k\boldsymbol{P}_k + \boldsymbol{B}_k^T\boldsymbol{S}_{k+1}\boldsymbol{B}_k\boldsymbol{K}_k\boldsymbol{P}_k = -\boldsymbol{B}_k^T\boldsymbol{S}_{k+1}\boldsymbol{A}_k\boldsymbol{P}_k. \tag{24}$$

Notice that the last equation can be used to solve for $\boldsymbol{K}_k$ in terms of $\boldsymbol{S}_{k+1}$ as:

$$\left(\boldsymbol{R}_k + \boldsymbol{B}_k^T\boldsymbol{S}_{k+1}\boldsymbol{B}_k\right)\boldsymbol{K}_k = -\boldsymbol{B}_k^T\boldsymbol{S}_{k+1}\boldsymbol{A}_k, \tag{25}$$

which yields the final equation (Eq (9)) given in the theorem. Let us write Eq (23) again as:

$$\begin{aligned}\boldsymbol{S}_k = \boldsymbol{A}_k^T\boldsymbol{S}_{k+1}\boldsymbol{A}_k + \boldsymbol{K}_k^T\left(\boldsymbol{R}_k + \boldsymbol{B}_k^T\boldsymbol{S}_{k+1}\boldsymbol{B}_k\right)\boldsymbol{K}_k + \boldsymbol{A}_k^T\boldsymbol{S}_{k+1}\boldsymbol{B}_k\boldsymbol{K}_k \\ + \boldsymbol{K}_k^T\boldsymbol{B}_k^T\boldsymbol{S}_{k+1}\boldsymbol{A}_k + \boldsymbol{C}_k^T\boldsymbol{Q}_k\boldsymbol{C}_k,\end{aligned} \tag{26}$$

which after substitution for $\boldsymbol{K}_k$ from Eq (9) in Eq (23) gives:

$$\boldsymbol{S}_k = \boldsymbol{C}_k^T\boldsymbol{Q}_k\boldsymbol{C}_k + \boldsymbol{A}_k^T\boldsymbol{S}_{k+1}\boldsymbol{A}_k - \boldsymbol{A}_k^T\boldsymbol{S}_{k+1}\boldsymbol{B}_k\left(\boldsymbol{R}_k + \boldsymbol{B}_k^T\boldsymbol{S}_{k+1}\boldsymbol{B}_k\right)^{-1}\boldsymbol{B}_k^T\boldsymbol{S}_{k+1}\boldsymbol{A}_k, \tag{27}$$

which again can be written to solve as a backward-recursive equation given as Eq (10) with terminal condition $\boldsymbol{S}_{N+1} = \boldsymbol{C}_{N+1}^T\boldsymbol{Q}_{N+1}\boldsymbol{C}_{N+1}$.

Remark: Relation with LQR

Notice that the feedback gain solution for the problem statement (Eq (9) and Eq (10)) with chosen Lagrange multipliers $\boldsymbol{Q}_k$ has the exact same solution obtained for the LQR problem with chosen weighting matrices $\boldsymbol{Q}_k$. This suggests that the time-varying output covariance problem can be solved by a substitute LQR control problem which is well understood and have well established algorithms with a judicious choice of $\boldsymbol{Q}$ to enforce the covariance constraint. The following subsection gives the iterative algorithm to calculate the corresponding weighting matrices $\boldsymbol{Q}_k$.

CAWS : Covariance Assignment via Weight Selection Algorithm

The iterative algorithm used for training the weighting matrix $\boldsymbol{Q}$ is given in **Algorithm**: CAWS. Assuming $\boldsymbol{R}_{0,..,N} = \boldsymbol{I}$ and $\boldsymbol{Q}^0_{1,..,N+1} = \boldsymbol{Q}^0\ \forall\ k = 1,..,N+1$, where $\boldsymbol{Q}^0$ is a chosen diagonal matrix, the algorithm first evaluates the output covariance ($\boldsymbol{Y}_k^0$) by solving the Riccatti equation. The total number of violations (n_{viol}) over all time steps and states in a single iteration of CAWS are then calculated using MATLAB's *find* function. For every iteration that carries a non-zero n_{viol} value, the algorithm alters all the elements of the $\boldsymbol{Q}$ matrix at all times as per their individual discrepancies expressed in terms of the ratio, $\boldsymbol{Y}_k^i/\bar{\boldsymbol{Y}}_k$. The algorithm continues until the weighting matrices $\boldsymbol{Q}^i_{1,..N+1}$ converges and the total number of violations go to zero.

The act of updating all elements of $\boldsymbol{Q}$ at all times in the i^{th} iteration ensures that any discrepancy in the covariance constraint at any time is collectively accredited by $\boldsymbol{Q}$ alterations throughout the trajectory. This procedure especially leads to obtaining a near-unique $\boldsymbol{Q}$ profile for all its elements. In other words, if the elements of $\boldsymbol{Q}^0$ are selected sufficiently small such that there are at least some violations at the beginning of the training algorithm, it is observed numerically that the converged $\boldsymbol{Q}$ is unique to some tolerance. Of course, training $\boldsymbol{Q}$ in this fashion to satisfy the time-varying covariance constraint alters $\boldsymbol{K}_k$ and therefore the control input sequence. As with all LQR applications, the feasibility of the controls if one considers actuator saturation bounds is practical issue one must address. In actual applications, the time-varying covariance commanded may not be physically realizable.

Algorithm: CAWS

Result: $\boldsymbol{Q}_{1,..,N+1}$
initialize $i = 0, \boldsymbol{Q}^i_{1,..,N+1} = \boldsymbol{Q}^0_{1,..,N+1}$,
$n_{\text{viol}} = 1$
while $n_{viol} \geq 1$ **do**
  initialize $k = N$, $\boldsymbol{S}^i_{N+1} = \boldsymbol{C}^T_{N+1}\boldsymbol{Q}^i_{N+1}\boldsymbol{C}_{N+1}$
  while $k \geq 1$ **do**
    $\boldsymbol{S}^i_k = \boldsymbol{C}^T_k\boldsymbol{Q}^i_k\boldsymbol{C}_k + \boldsymbol{A}^T_k\boldsymbol{S}^i_{k+1}\boldsymbol{A}_k - \boldsymbol{A}^T_k\boldsymbol{S}^i_{k+1}\boldsymbol{B}_k\left(\boldsymbol{R}_k + \boldsymbol{B}^T_k\boldsymbol{S}^i_{k+1}\boldsymbol{B}_k\right)^{-1}\boldsymbol{B}^T_k\boldsymbol{S}^i_{k+1}\boldsymbol{A}_k$
    $\boldsymbol{K}^i_k = -\left(\boldsymbol{R}_k + \boldsymbol{B}^T_k\boldsymbol{S}^i_{k+1}\boldsymbol{B}_k\right)^{-1}\boldsymbol{B}^T_k\boldsymbol{S}^i_{k+1}\boldsymbol{A}_k$
    $k = k - 1$
  end
  $\boldsymbol{K}^i_0 = -\left(\boldsymbol{R}_0 + \boldsymbol{B}^T_0\boldsymbol{S}^i_1\boldsymbol{B}_0\right)^{-1}\boldsymbol{B}^T_0\boldsymbol{S}^i_1\boldsymbol{A}_0$
  initialize $k = 0$, $\boldsymbol{P}^i_0 = \boldsymbol{P}_0$
  while $k \leq N$ **do**
    $\boldsymbol{P}^i_{k+1} = (\boldsymbol{A}_k + \boldsymbol{B}_k\boldsymbol{K}^i_k)\boldsymbol{P}^i_k(\boldsymbol{A}_k + \boldsymbol{B}_k\boldsymbol{K}^i_k)^T + \boldsymbol{D}_k\mathbb{W}_k\boldsymbol{D}^T_k$
    $\boldsymbol{Y}^i_k = \boldsymbol{C}_k\boldsymbol{P}^i_k\boldsymbol{C}^T_k$
    $k = k + 1$
  end
  $\boldsymbol{Y}^i_{N+1} = \boldsymbol{C}_{N+1}\boldsymbol{P}^i_{N+1}\boldsymbol{C}^T_{N+1}$
  n_{viol}= find($\boldsymbol{Y}^i_k - \bar{\boldsymbol{Y}} \geq \epsilon$)
  if $n_{viol} \geq 1$ **then**
    initialize $k = 1$,
    while $k \leq N + 1$ **do**
      $Q^{i+1}_k = \frac{Q^i_k(k)\boldsymbol{Y}^i_k}{\bar{\boldsymbol{Y}}_k}$
      $k = k + 1$
    end
  end
  $i = i + 1$
end

EXAMPLE APPLICATION

Telemetry, tracking and control subsystem provides vital communication that facilitates observance of the spacecraft's state from the ground and transmission of commands from the ground control to the satellite for guidance or conducting particular science operations, etc. Telemetry satellites provide essential data like spacecraft's attitude, operational state, and navigational data as well as timely down-link collected scientific data. This requires these satellites to constantly point to the receiving ground station for transmitting and receiving data. Therefore, along with the requirement of very low maneuver errors at the end of the orientation change, the pointing maneuvers of these satellites have stringent constraints on the angular errors throughout the maneuver to ensure a high quality data transmission. In the case that the orbit is eccentric, it is obvious that the required covariance of pointing errors will necessarily be time varying. Therefore, the attitude control problem of such satellites provides a good example to demonstrate the capability of the proposed algorithm.

Attitude dynamics

This sub-section establishes the attitude control dynamics of rigid spacecraft whose attitude is described by the Classical Rodrigues Parameters (CRPs). The CRPs constitute a symmetric stereographic set that exhibits uniqueness and is quasi-linear for large angular motions [25], earning them the position of being one of the favorable sets of parameters for attitude dynamics. Unlike the transcendental expressions used in the case of Euler angles, CRPs provide purely algebraic expressions for the governing equations of motion. The domain of linearization is also quite large as compared to the Euler angle parametrization [26].

The CRPs can be expressed in terms of Euler's principal rotation axis ($\hat{\mathbf{e}}$) and the principal angle (Φ) through the relation [27]:

$$\boldsymbol{q} = \hat{\boldsymbol{e}} \tan\left(\frac{\Phi}{2}\right) \tag{28}$$

Therefore, the singularity condition in the CRPs is encountered when the principal angle appoaches 180 deg. The kinematic relations are then given by [25]:

$$\begin{aligned} \dot{\boldsymbol{q}} &= \boldsymbol{A}\boldsymbol{\omega} \\ \boldsymbol{A} &= \frac{1}{2}\left(\boldsymbol{I}_{3\times 3} + [\tilde{\boldsymbol{q}}] + \boldsymbol{q}\boldsymbol{q}^T\right) \end{aligned} \tag{29}$$

where, $\tilde{\boldsymbol{q}} = \begin{pmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{pmatrix}$, $\boldsymbol{q} = [q_1, q_2, q_3]^T$ and ω is the angular velocity of the rigid body which can be conveniently evaluated by inverting Eq (29) as given in [25]:

$$\boldsymbol{\omega} = \boldsymbol{B}\dot{\boldsymbol{q}} \tag{30a}$$

$$\boldsymbol{B} = \frac{2}{(1 + \boldsymbol{q}^T\boldsymbol{q})}(\boldsymbol{I}_{3\times 3} - [\tilde{\boldsymbol{q}}]) \tag{30b}$$

Using Euler's equation for rigid body motion, the differential equation of angular velocity can be written as:

$$\dot{\boldsymbol{\omega}} = J_{\text{mat}}^{-1}(\boldsymbol{\omega} \times (J_{\text{mat}}\boldsymbol{\omega}) + \boldsymbol{u}) \tag{31}$$

where J_{mat} is a diagonal matrix (in the present example) with spacecraft's principal moment of inertia values put as its diagonal entries. Therefore, the dynamics assumes a cascade form of system where the control, $\boldsymbol{u}$ drives the angular velocities which in turn control the orientation using the first kinematic equation (Eq (29)). The system differential equations in the state space form can be expressed as:

$$\dot{\boldsymbol{x}} = \mathscr{F}(\boldsymbol{x}, \boldsymbol{u}, t) = \mathbb{A}(\boldsymbol{x}, t) + \mathbb{B}(\boldsymbol{x}, t)\boldsymbol{u} \tag{32}$$

where

$$\mathbb{A} = \begin{bmatrix} \frac{1}{2}(1+q_1^2)\omega_1 + (q_1q_2 - q_3)w_2 + (q_1q_3 + q_2)\omega_3 \\ \frac{1}{2}(1+q_2^2)\omega_2 + (q_1q_2 + q_3)w_1 + (q_2q_3 - q_1)\omega_3 \\ \frac{1}{2}(1+q_3^2)\omega_3 + (q_3q_1 - q_2)w_1 + (q_3q_2 + q_1)\omega_2 \\ \frac{(J_2-J_3)}{J_1}\omega_2\omega_3 \\ \frac{(J_3-J_1)}{J_2}\omega_1\omega_3 \\ \frac{(J_1-J_2)}{J_3}\omega_1\omega_2 \end{bmatrix}, \quad \mathbb{B} = \begin{bmatrix} \mathbf{0}_{3\times 3} \\ J_{\text{mat}}^{-1} \end{bmatrix}$$

and $\boldsymbol{x} = [q_1, q_2, q_3, \omega_1, \omega_2, \omega_3])$.

Design of the nominal trajectory using optimal control theory

Considering a finite horizon problem, the nominal trajectory for the maneuver is evaluated using optimal control theory with an objective to minimize the control effort. Therefore, the optimal control problem (OCP) can be stated as

$$\begin{aligned} \text{Minimize} \;\; J &= \int_{t_0}^{t_f} \frac{1}{2}\boldsymbol{u}^T\boldsymbol{u}dt, \\ \text{subject to :} \quad & \text{Equation (32)}, \\ & \boldsymbol{x}(t_f) - \tilde{\boldsymbol{x}}_d = \mathbf{0}, \\ & \boldsymbol{x}(t_0) = \boldsymbol{x}_0 \end{aligned} \tag{33}$$

where t_0 is the initial time, $\boldsymbol{x_0}$ are the the states at the initial time, t_f is the final time, $\Delta t = t_f - t_0$ is the time of flight, and $\tilde{\boldsymbol{x}}_d$ are the desired states at the final time.

The Hamiltonian associated with the defined OCP can be written as:

$$H = \frac{1}{2}\boldsymbol{u}^T\boldsymbol{u} + \boldsymbol{\lambda}^\top \left(\mathbb{A}(\boldsymbol{x}, t) + \mathbb{B}(\boldsymbol{x}, t)\boldsymbol{u}\right), \tag{34}$$

where $\boldsymbol{\lambda} = [\lambda_{q_1}, \lambda_{q_2}, \lambda_{q_3}, \lambda_{w_1}, \lambda_{w_2}\lambda_{w_3}]^\top$ are the costate of the system. Since the control variable $\boldsymbol{u}$ is unbounded and appears non-linearly in the Hamiltonian, strong form of optimality can be used to give:

$$\boldsymbol{u} = -\mathbb{B}(\boldsymbol{x}, t)^T\boldsymbol{\lambda}. \tag{35}$$

Here, the $\boldsymbol{u}$ is defined in the spacecraft's body frame. The co-states differential equation is derived using Euler-Lagrange equation,

$$\dot{\boldsymbol{\lambda}} = -\left[\frac{\partial H}{\partial \boldsymbol{x}}\right]^\top = \begin{bmatrix} \dot{\boldsymbol{\lambda}}_{\boldsymbol{q}} \\ \dot{\boldsymbol{\lambda}}_{\boldsymbol{\omega}} \end{bmatrix} \tag{36}$$

Since the value of co-states at both initial an final boundary are unknown, TPBVP is solved to satisfy the boundary conditions defined in Eq (33) using MATLAB's *fsolve* solver [28].

For conducting the linear covariance analysis for controller design, a linear discretized version of the system equations is obtained by piecewise linearizing about the nominal trajectory using the crude Euler's method with an integration step size of h seconds. Therefore, the resulting system matrices used for designing the controller are:

$$\begin{aligned} \mathscr{A}_c &= \left.\frac{\partial \mathscr{F}(\boldsymbol{x},\boldsymbol{u},t)}{\partial \boldsymbol{x}}\right|_{\bar{\boldsymbol{x}}_i,\bar{\boldsymbol{u}}_i}, \quad \mathscr{B}_c = \left.\frac{\partial \mathscr{F}(\boldsymbol{x},\boldsymbol{u},t)}{\partial \boldsymbol{u}}\right|_{\bar{\boldsymbol{x}}_i,\bar{\boldsymbol{u}}_i} \\ \mathscr{A}_d &= \boldsymbol{I}_{6\times 6} + h\mathscr{A}_c, \quad \mathscr{B}_d = h\mathscr{B}_c \end{aligned} \tag{37}$$

where, $\bar{\boldsymbol{x}}_i$ and $\bar{\boldsymbol{u}}_i$ are the state and controls on the nominal trajectory at the i^{th} time step and $[\cdot]_c$ and $[\cdot]_d$ are the linearized system matrices for continuous and discrete time dynamics equations.

RESULTS

Using the algorithm explained in Section II and dynamics provided in Section III, we solve a problem of orienting an asymmetric satellite from orientation A to B in fixed time. The boundary conditions of the maneuver are given in Table 1 where $\boldsymbol{x_A}$ and $\boldsymbol{x_B}$ define the states at orientation A and B respectively. Table 1 also lists the parameters used in the tracking problem where σ_0 defines the standard deviation of the uncertainty in the states at orientation A and $\tilde{\sigma}$ sets the upper bound on the output state error which is used to provide the desired output covariance profile. The σ_0 and $\tilde{\sigma}$ values given in Table 1 are constants and are given in degrees which are accommodated and transformed into CRPs using the 3-2-1 Euler angle sequence rotation by angles θ_1, θ_2 and θ_3, respectively. The non-linear transformation required to transform CRPs into Euler angles is obtained by using the Cayley transform identity:

$$\mathscr{C} = \frac{1}{(1+\boldsymbol{q}^T\boldsymbol{q})}\left((1-\boldsymbol{q}^T\boldsymbol{q})\boldsymbol{I}_{3\times 3} + 2(\boldsymbol{q}\boldsymbol{q}^T) - 2\tilde{\boldsymbol{q}}\right) \tag{38}$$

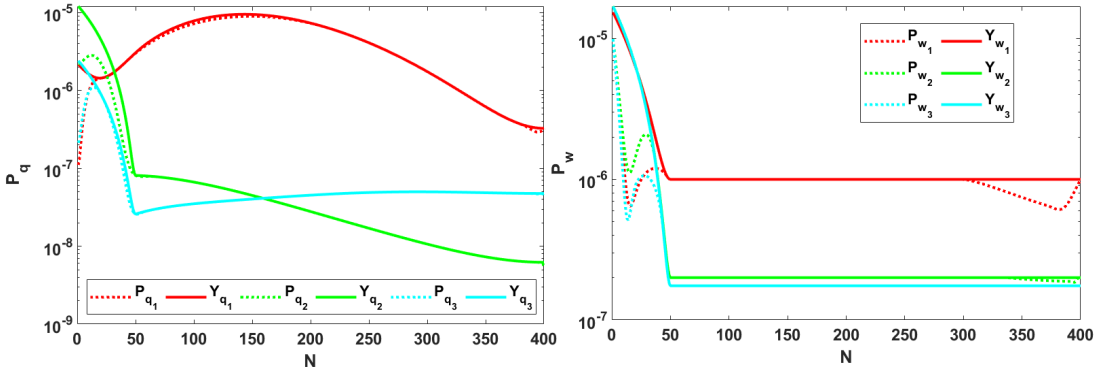

Figure 1: Time history of the diagonal entries of the output covariance for all states using the $\boldsymbol{Q}_d$ matrix designed by CAWS.

After performing the transformations discussed above at every discrete time step, the resulting covariance profiles (denoted by $\boldsymbol{Y}$) are plotted as solid lines in the left plot of Figure 1. The actuators

Table 1: Boundary conditions and parameters

$\boldsymbol{J}_{\text{mat}}$	diag([10,6,8])
$\boldsymbol{x_A}$ (deg., rad/sec)	$[-10^o, 20^o, 8^o, 0.1, -0.0011, 0.01]$
$\boldsymbol{x_B}$ (deg., rad/sec)	$[20^o, 5^o, 30^o, 0, 0, 0]$
T (sec)	40
N	400
σ_0 (deg, rad/sec)	$[2^o, 2^o, 2^o, 0.01, 0.01, 0.01]$
$\tilde{\sigma}$ (deg.,rad/sec)	$[0.5^o, 0.5^o, 0.5^o, 10^{-3}; 5 \times 10^{-4}; 5 \times 10^{-4}]$
$\mathbb{W}$ $(N^2 m^2)$	$[0, 0, 0, 10^{-7}, 10^{-7}, 10^{-7}]$
$\boldsymbol{Q_0}$	$10^{-10}\boldsymbol{I}_{6\times 6}$
$\boldsymbol{R}$	$\boldsymbol{I}_{6\times 6}$

are assumed to generate continuous control profile like in the case of momentum wheels or gas jet actuators with pulse width pulse frequency modulators. As given in Table 1, the time of the maneuver is set as $T = 40$ seconds which is discretized into $N = 400$ steps making the bandwidth of the controller to be 10Hz. The uncertainty in the system dynamics is modeled via a zero-mean Gaussian white noise, $\boldsymbol{\nu} : \mathcal{N}(0, \mathbb{W}) \in \mathbb{R}^6$. Since the kinematic relation (given in Eq (29)) is exact, no noise is assumed in the first three channels of the states as shown in Table 1.

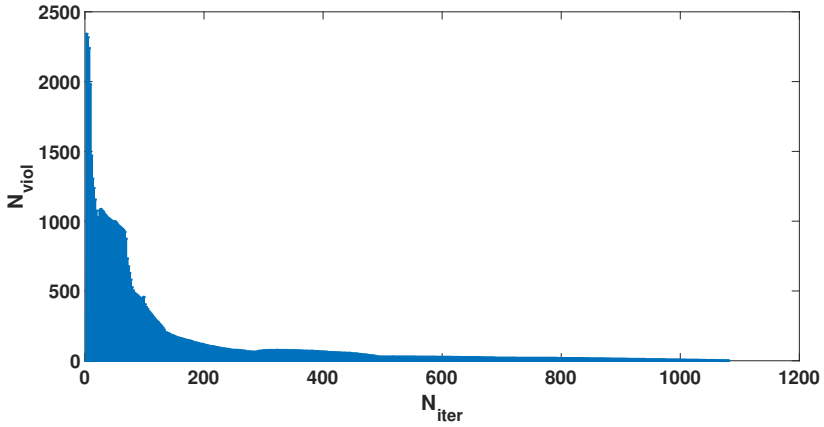

Figure 2: Rate of elimination of time points that violate the covariance bound constraint during CAWS.

The initial uncertainty and noise parameters used for training $\boldsymbol{Q}$ in CAWS are set to be a little higher than the one actually used in the non-linear simulation to design the feedback control on a more conservative system. Using the parameters for initial uncertainty and stochastic actuator errors defined in Table 1, CAWS algorithm is applied on the linearized system to obtain a time-varying profile of $\boldsymbol{Q}$ iteratively (N_{iter} iterations). Starting from some initial number of covariance violations for each of the states, the violations decrease gradually to converge with no violations towards the end as shown in Figure 2. As the number of violations decreases, the change in the magnitude of each of the diagonal entries of the $\boldsymbol{Q}$ matrix also converges as per the set tolerance. It is not necessary that the number of violations and the change in the norm of $\boldsymbol{Q}$ monotonically decreases throughout the training iterations as shown in Figure 2.

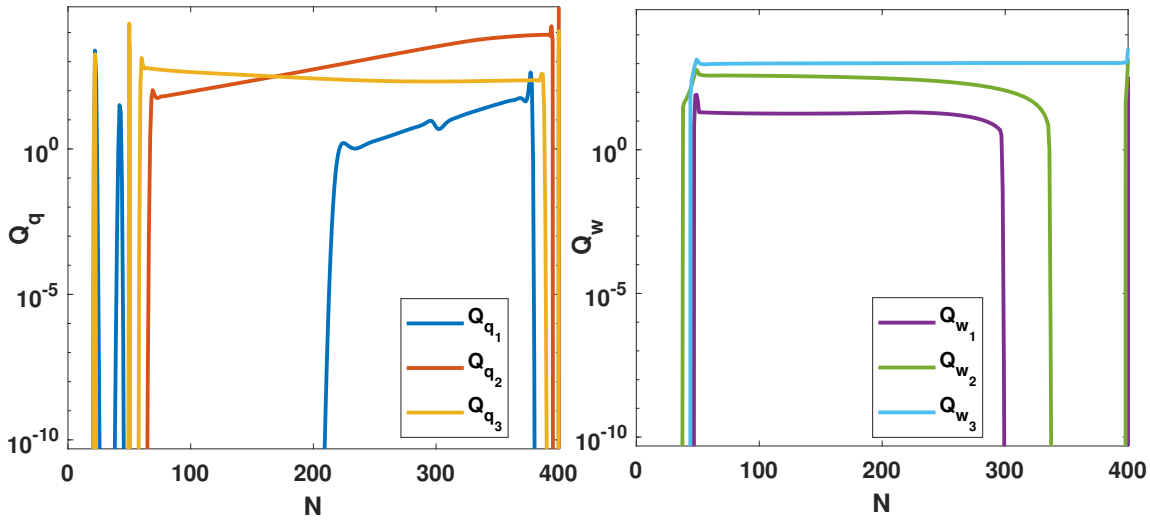

Figure 3: Diagonal entries for the designed $\boldsymbol{Q}_d$ matrix profile plotted against time-steps.

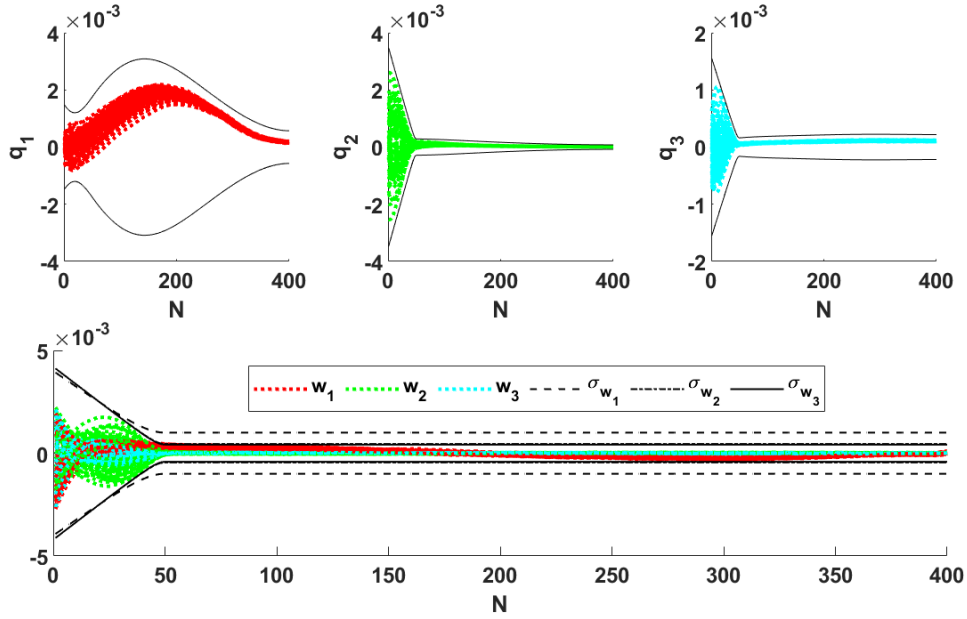

Figure 4: Time history of errors in CRPs and angular velocities plotted against time with the solid black lines denoting the set covariance bound for time varying $\boldsymbol{Q}$.

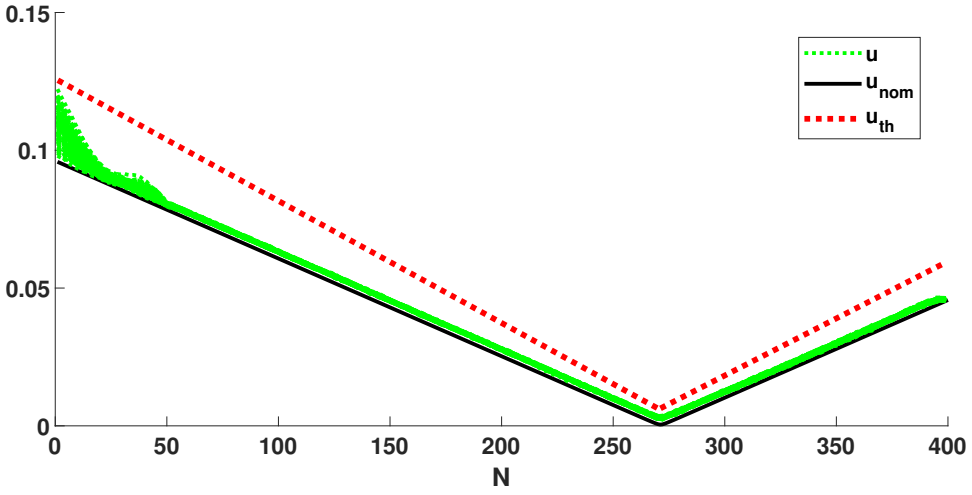

Figure 5: Time history of controls for 100 sample trajectories for time varying $\boldsymbol{Q}$.

Figure 1 shows the time history of the pre-specified covariance bounds and the diagonal entries of the achieved state covariance using the $\boldsymbol{Q}_d$ matrix designed by the iterative algorithm (CAWS). For the linearized system, one can observe that the actual covariances (solid lines) follow the specified covariance profile (dotted lines) in Figure 1, exactly except for few short time periods where it stays below the hard limit of the set covariance bound. This is owing to the cross-coupling of states in the natural dynamics of the system. In other words, it is possible that in order to follow the covariance profile set for q_1, tighter bounds are required for q_2 as per the natural dynamics of the system. It also aids in conserving on the extra control effort that would have been required to supersede the natural dynamics and match the covariance profile exactly. Figure 3 provides the diagonal entries for $\boldsymbol{Q}_d$ matrix for both CRPs and angular velocity errors plotted against time. Here, the values of $\boldsymbol{Q}_d$ are truncated at 10^{-12} by considering it essentially as 0 as per machine precision.

Post designing $\boldsymbol{Q}_d$ to solve the covariance assignment problem on the linearized system, the performance of the designed feedback control is tested by a Monte Carlo analysis for 100 trajectories. Since the attitude dynamics when designed using CRPs is not highly non-linear, one can notice that the trajectories in Figure 4 not only stay within the bounds (plotted as solid black lines) but the profile of the trajectories for the CRPs follow the shape of the covariance profile set in the linearized system as shown in Figure 1.

Figure 5 shows the time history of the magnitude of control torques applied for 100 sample trajectories. The black solid line featuring a very short "no control" arc near over the middle of the maneuver is the nominal control (u_{nom}) obtained using optimal control theory. The red dotted line denotes the upper bound (20% over the nominal value) on the total control effort allowed for the purpose of tracking the nominal trajectory. Even when no actuator constraints are formally imposed in our algorithm, Figure 5 shows that the required feedback control does not violate the upper bound set over the nominal control.

In order to demonstrate the advantage of our proposed algorithm over the classic constant $\boldsymbol{Q_c}$ strategy for solving traditional LQR, Monte Carlo runs for 100 trajectories are repeated keeping the $\boldsymbol{Q_c}$ values constant. The candidate values for the diagonal components of the constant $\boldsymbol{Q_c}$ is

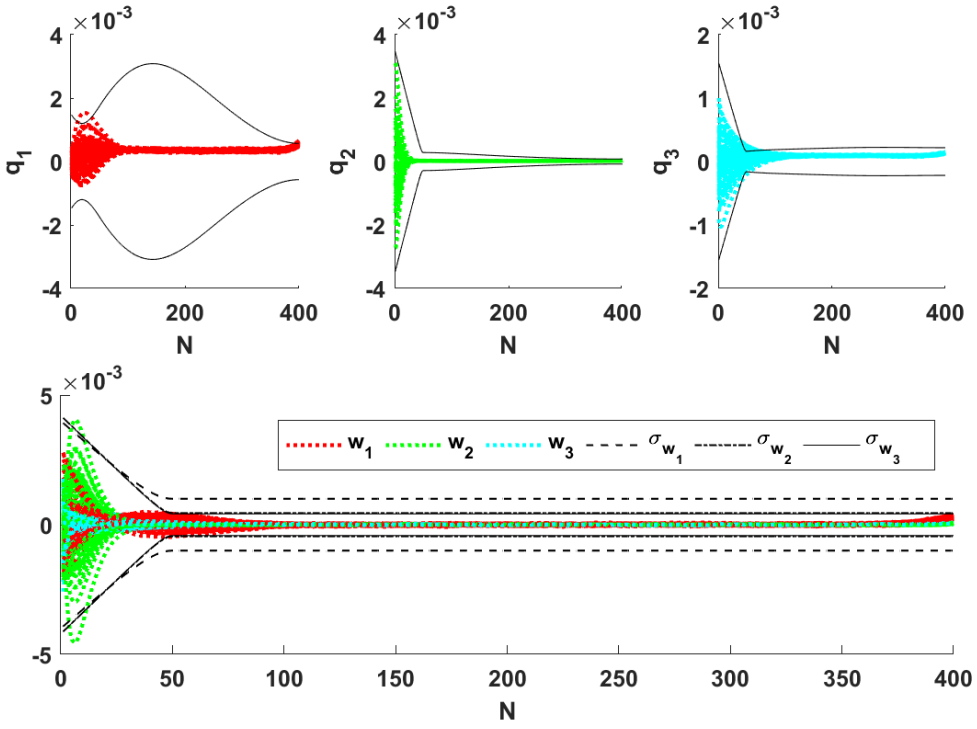

Figure 6: Time history of errors in CRPs and angular velocities plotted against time with the solid black lines denoting the set covariance bound for constant Q_c case.

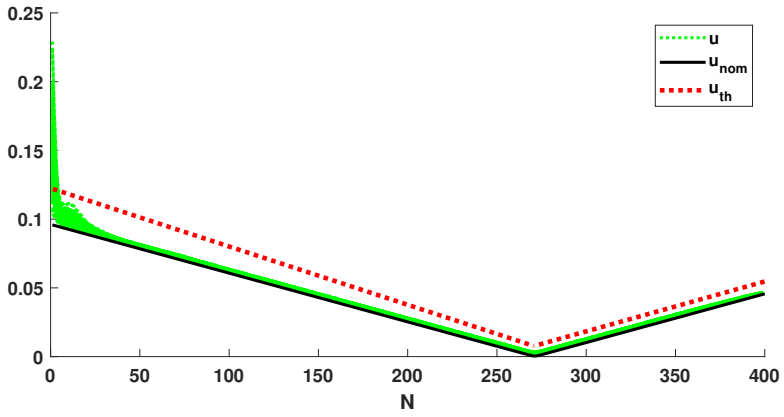

Figure 7: Time history of controls for 100 sample trajectories for constant Q_c case.

obtained by taking an average of the elements of $\boldsymbol{Q}_d$ profile such that:

$$\boldsymbol{Q_c} = [22, 2169, 332.3, 12.32, 210, 900]$$

The time history of state errors and actuator magnitude obtained from the Monte Carlo analysis is given in Figures 6 and 7. The results, though inferior to the results obtained by using CAWS, feature few violations, mainly at the beginning of the maneuver and towards the end. Traditionally, the constant $\boldsymbol{Q_c}$ values are iteratively composed by utilizing the experience and time of the control designer. One possible application of the proposed method is to provide a better starting iteration for the designer if constant $\boldsymbol{Q_c}$ solutions are ultimately desired.

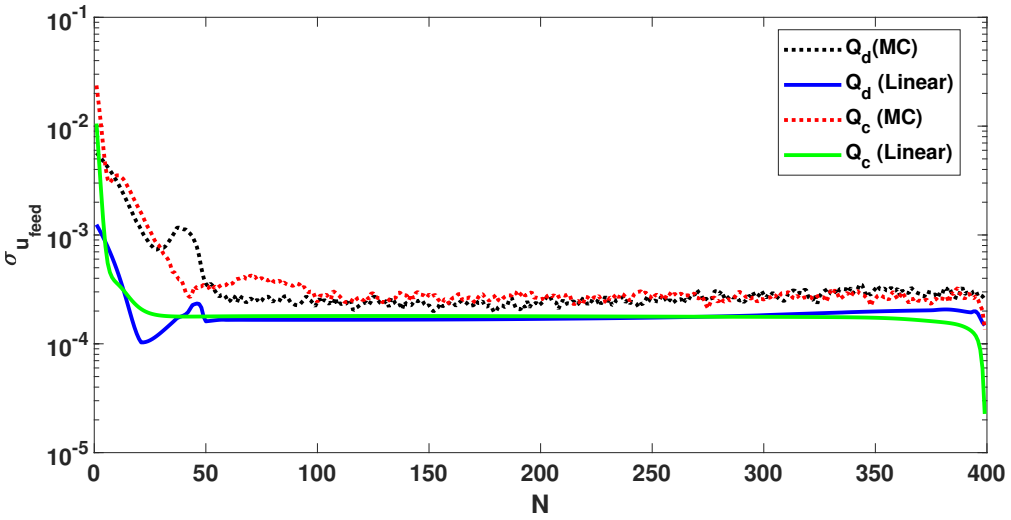

Figure 8: Comparison of standard deviation of feedback control history for designed $\boldsymbol{Q_d}$ and constant $\boldsymbol{Q_c}$ cases on linearized system and the non-linear Monte Carlo (MC) simulations.

The control effort required in the case of constant $\boldsymbol{Q_c}$ is almost twice the defined control threshold but only at the beginning of the maneuver as shown in Figure 7. This high value of controls is probably attributed to controller's act of tracking the nominal while accounting for large initial uncertainties as they occur. This does not happen in the case of variable $\boldsymbol{Q_d}$ designed by CAWS due to the facility of effectively distributing the control effort throughout the maneuver to counter any tracking errors owing to either initial uncertainty or disturbances.

Figure 8 shows the standard deviation of feedback control efforts for constant $\boldsymbol{Q_c}$ and designed $\boldsymbol{Q_d}$ weighting matrices over the complete time horizon. The figure shows the standard deviation for the linearized and nonlinear cases where linearized control efforts are calculated using square root of $\mathrm{Tr}(\boldsymbol{U}_k) = \mathrm{Tr}\left(\boldsymbol{K}_k \boldsymbol{P}_k \boldsymbol{K}_k^T\right)$ and the control efforts for the nonlinear case are calculated statistically by running 100 Monte Carlo simulations. Notice that the feedback control effort for the designed $\boldsymbol{Q_d}$ case is smaller than the constant $\boldsymbol{Q_c}$ case for the initial time-steps for both linear and nonlinear results owing to effectively distributing the control effort.

CONCLUSION AND FUTURE WORK

This work proposes a novel constitution that allows the control designer to assign time-varying output covariance in a guidance-type maneuver by systematically evaluating the time-varying weight-

ing sequences in a simple and elegant LQR framework. Though the weighting sequences $\boldsymbol{Q_d}$ are evaluated on a linearized system, the results from the Monte Carlo analysis show great promise in delivering equivalent results in the non-linear case. The problem is also solved using the traditional constant $\boldsymbol{Q_c}$ approach where the values of the constant $\boldsymbol{Q_c}$ are derived using the variable $\boldsymbol{Q_d}$ profile. The results obtained indicate the superiority of variable $\boldsymbol{Q_d}$ over the constant $\boldsymbol{Q_c}$ case as well as presents the usefulness of the proposed method in providing a potential starting point for iterating on the constant $\boldsymbol{Q}$'s diagonal entries.

It is not necessary to have $\boldsymbol{Q}$ as a diagonal matrix. In fact, the current algorithm can be expanded to use the full symmetric matrix $\boldsymbol{Q}$ to include the states cross-coupling weighting into the LQR formulation. The attitude control problem solved for demonstrating the efficacy of the algorithm has a large domain of linearization. Future work lies in extending this to a more nonlinear setting, such as solving the guidance problem for an interplanetary maneuver.

REFERENCES

[1] Xiangyu Li, Rakesh R Warier, Amit K Sanyal, and Dong Qiao. Trajectory tracking near small bodies using only attitude control. *Journal of Guidance, Control, and Dynamics*, 42(1):109–122, 2019.

[2] F Landis Markley and John L Crassidis. *Fundamentals of spacecraft attitude determination and control*. Springer, 2014.

[3] Vishala Arya, Ehsan Taheri, and John Junkins. Electric thruster mode-pruning strategies for trajectory-propulsion co-optimization. *Aerospace Science and Technology*, 116:106828, 2021.

[4] Vishala Arya, Ehsan Taheri, and John L Junkins. Low-thrust gravity-assist trajectory design using optimal multimode propulsion models. *Journal of Guidance, Control, and Dynamics*, pages 1–15, 2021.

[5] John L Crassidis and F Landis Markley. Sliding mode control using modified rodrigues parameters. *Journal of Guidance, Control, and Dynamics*, 19(6):1381–1383, 1996.

[6] Hanspeter Schaub, Maruthi R Akella, and John L Junkins. Adaptive control of nonlinear attitude motions realizing linear closed loop dynamics. *Journal of Guidance, Control, and Dynamics*, 24(1):95–100, 2001.

[7] Georgios Lymperopoulos and Petros Ioannou. Adaptive aircraft control in the presence of unstructured dynamic uncertainties. *Journal of Guidance, Control, and Dynamics*, 42(1):153–162, 2019.

[8] Randall S Christensen and David Geller. Linear covariance techniques for closed-loop guidance navigation and control system design and analysis. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 228(1):44–65, 2014.

[9] Raman Goyal, Manoranjan Majji, and Robert E Skelton. Integrating structure, information architecture and control design: Application to tensegrity systems. *Mechanical Systems and Signal Processing*, 161:107913, 2021.

[10] E. Collins and R. Skelton. A theory of state covariance assignment for discrete systems. *IEEE Transactions on Automatic Control*, 32(1):35–41, 1987.

[11] Anthony Hotz and Robert E Skelton. Covariance control theory. *International Journal of Control*, 46(1):13–32, 1987.

[12] R. E Skelton, T. Iwasaki, and K. Grigoriadis. *A Unified Algebraic Approach to Control Design.* Taylor & Francis, London, UK, 1998.

[13] Chen Hsieh and Robert E Skelton. All covariance controllers for linear discrete-time systems. *IEEE transactions on automatic control*, 35(8):908–915, 1990.

[14] Kazuhide Okamoto and Panagiotis Tsiotras. Input hard constrained optimal covariance steering. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 3497–3502. IEEE, 2019.

[15] Jack Ridderhof and Panagiotis Tsiotras. Minimum-fuel powered descent in the presence of random disturbances. In *AIAA Scitech 2019 Forum*, page 0646, 2019.

[16] Jack Ridderhof, Kazuhide Okamoto, and Panagiotis Tsiotras. Nonlinear uncertainty control with iterative covariance steering. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 3484–3490. IEEE, 2019.

[17] Efstathios Bakolas. Finite-horizon covariance control for discrete-time stochastic linear systems subject to input constraints. *Automatica*, 91:61 – 68, 2018.

[18] Abhishek Halder and Eric DB Wendel. Finite horizon linear quadratic gaussian density regulator with wasserstein terminal cost. In *2016 American Control Conference (ACC)*, pages 7249–7254. IEEE, 2016.

[19] Efstathios Bakolas. Optimal covariance control for stochastic linear systems subject to integral quadratic state constraints. In *2016 American Control Conference (ACC)*, pages 7231–7236. IEEE, 2016.

[20] Yongxin Chen, Tryphon T Georgiou, and Michele Pavon. Optimal steering of a linear stochastic system to a final probability distribution, part i. *IEEE Transactions on Automatic Control*, 61(5):1158–1169, 2015.

[21] Yongxin Chen, Tryphon T Georgiou, and Michele Pavon. Optimal steering of a linear stochastic system to a final probability distribution, part ii. *IEEE Transactions on Automatic Control*, 61(5):1170–1180, 2015.

[22] R. Goyal, R. E. Skelton, and M. Majji. Optimal actuator/sensor precision for covariance steering with soft convex constraints on state and control. In *2021 American Control Conference (ACC)*, May 25-28 2021.

[23] Kazuhide Okamoto and Panagiotis Tsiotras. Optimal stochastic vehicle path planning using covariance steering. *IEEE Robotics and Automation Letters*, 4(3):2276–2281, 2019.

[24] Lars Blackmore, Hui Li, and Brian Williams. A probabilistic approach to optimal robust path planning with obstacles. In *2006 American Control Conference*, pages 7–pp. IEEE, 2006.

[25] Hanspeter Schaub, John L Junkins, et al. Stereographic orientation parameters for attitude dynamics: A generalization of the rodrigues parameters. *Journal of the Astronautical Sciences*, 44(1):1–19, 1996.

[26] John L Junkins and Puneet Singla. How nonlinear is it? a tutorial on nonlinearity of orbit and attitude dynamics. *The Journal of the Astronautical Sciences*, 52(1):7–60, 2004.

[27] Hanspeter Schaub and John L Junkins. *Analytical mechanics of space systems*. Aiaa, 2003.

[28] Vishala Arya, Ehsan Taheri, and John L Junkins. A composite framework for co-optimization of spacecraft trajectory and propulsion system. *Acta Astronautica*, 178:773–782, 2021.