Stochastic Receding Horizon Control with Output feedback and bounded controls

Authors : Peter Hokayem, Eugenio Cinquemani, Debasish Chatterjee, Federico Ramponi, John Lygeros
ECE 555 Course Project : Karthik Manoj & Girish Madhavan

1 Introduction

A considerable amount of research has been done on deterministic receding horizon control, providing strong proofs for recursive feasibility and stability of receding horizon control laws in a noise-free deterministic setting. These techniques can be extended to cases when bounded exogenous noise or parametric uncertainty enters the system. However, the case when a stochastic system is subjected to process noise, imperfect state measurements, and bounded control inputs is still lacking. This problem is harder to solve since it may not be possible to determine an a priori bound on the values taken by noise, as this complicates the stability and feasibility proofs. The problem being the noise (at least in the additive case) eventually drives the state outside any bounded set, regardless of its size. Moreover, using any standard linear state feedback would end up violating the hard bounds on the control input.

This paper proposes a solution to the general Receding Horizon control problem for linear systems with Noisy process dynamics, imperfect state information, and bounded control inputs. Both the process and measurement noise are assumed to enter the system in an additive manner, and we require the designed control inputs to satisfy the hard bounds. A certain finite-horizon optimal control problem is solved over a prediction horizon $N \geq N_c$, periodically at $t = 0, N_c, 2N_c, \ldots$, where N_c is the control horizon (*number of steps taken by the system to reach a desired state*). Apart from the bounds on control inputs, it's also possible to impose some variance-like bounds on the predicted future states and inputs, using them as a soft state constraint, that are similar to integrated chance constraints.

Key challenges to our problem setup :

- Since the state information is imperfect, one needs a filter to estimate the state
- In the presence of unbounded noise, it is not possible to ensure any bound on the control values generated by linear state feedback, this is because of additive noise driving the states outside the stable region. To overcome this problem, nonlinear control policies will be needed. This problem is made worse by the fact that only incomplete state information is available
- It's unclear if the application of bounded control policies stabilizes the system in any reasonable sense. Achieving asymptotic stability for a linear system in the presence of stochastic process noise is not realistic.

This paper relaxes the notion of stability to mean-square boundedness of the state and imposes extra conditions so that the system matrix (A) is Lyapunov stable and the pair (A, B) is stabilizable. In a nutshell, this paper tries to solve our problem as follows: Given a suitable subclass of bounded causal feedback policies, it describes how to augment the finite-horizon optimal control problem to be solved periodically every N_c steps with a stability constraint and the resulting optimization problem can be approximated to a globally feasible second-order cone program (SOCP). Under the assumption that all noises are Gaussian, an algorithm similar to the Kalman filtering technique can be utilized for updating the conditional density of the state given the history of previous outputs and report tractable solutions for the off-line computation of the time-dependent variance and covariance matrices in the optimization program. Finally, it proves that the recursive application of generated control policies renders the state of the overall system mean-square bounded. The uniqueness of this paper lies in its ability to deal with recursive feasibility and stability, which was not possible with earlier formulations.

Related Work : The research on stochastic horizon stochastic control is broadly subdivided into two regions :

- Multiplicative noise, the case when noise enters the state equation multiplicatively, attempts to solve this problem include relaxing mixed hard state-input constraints into expectation constraints. Along with suitable terminal costs, that would render the overall MPC problem stable under full state feedback. Similarly, in the case when the output matrix (C) is uncertain, the MPC problem is solved under probabilistic constraints on the outputs and full state feedback. Further research exists in controlling the effects of other sources of uncertainty or noise. [1, 2, 3]
- Additive noise, the approach in this paper is based on the idea of using affine parameterization of control policies for finite-horizon linear quadratic problems, utilized within the robust MPC framework for full state feedback or output feedback with Gaussian state and measurement noise inputs. One approach treats this stochastic programming problem as a deterministic one with bounded noise values and solves a robust optimization problem over a finite horizon, followed by estimating the performance when the noise can take unbounded values (considering low probability for high noise values). Despite all these approaches, the ability to deal with unbounded noise with bounded control inputs and suitable proofs for stability and recursive feasibility are still absent. [4, 5, 6]

2 Problem Setup

First, we consider the following system dynamics :

$$x_{t+1} = Ax_t + Bu_t + w_t \tag{1}$$

$$y_t = Cx_t + v_t \tag{2}$$

Where,

- ⋆ $t \in \mathbb{N}$
- ⋆ $x_t \in \mathbb{R}^n$ is the state
- ⋆ $u_t \in \mathbb{R}^m$ is control input
- ⋆ $y_t \in \mathbb{R}^p$ is the output
- ⋆ $w_t \in \mathbb{R}^n$ is the process noise
- ⋆ $v_t \in \mathbb{R}^p$ is the measurement noise
- ⋆ A, B & C are known matrices

Now certain assumptions are made about this process :

i The pair (A, B) is stabilizable : All uncontrollable states are stable

ii The matrix A is Lyapunov stable: eigenvalues of A $\lambda(A)$ lie inside the unit circle about othe rigin, the ones that lie on the circle have equal algebraic and geometric multiplicities

iii $x_0 \sim \mathcal{N}(0, \Sigma_0),\ w_t \sim \mathcal{N}(0, \Sigma_w),\ \&\ v_t \sim \mathcal{N}(0, \Sigma_v)$ with Σ_w and Σ_v being positive definite

iv $(A, \Sigma_w^{0.5})$ is controllable and (A, C) is observable

v The control inputs are bounded, $\|u_t\|_\infty \leq U_{max} \quad \forall\ t \in \mathbb{N}$

The system dynamics is assumed to be given in the Jordan canonical form, because it's possible to find an appropriate coordinate transformation to bring a system representation from any arbitrary form to the Jordan canonical form. Now, on suitably reordering the Jordan block, we get:

$$(A, B) \equiv \left(\begin{bmatrix} A_s & 0 \\ 0 & A_o \end{bmatrix}, \begin{bmatrix} B_s \\ B_o \end{bmatrix} \right),\ \text{where } A_s \in \mathbb{R}^{n_s \times n_S} \text{ is Schur stable, and } \|\lambda(A_o)\| = 1,\ A_o \in \mathbb{R}^{n_o \times n_o}$$

By Assumption (ii), A_o is a block-diagonal matrix with the diagonal blocks being either ± 1 or 2×2 rotation matrices, as the eigenvalues can be located anywhere on the unit circle. As a consequence, A_o is orthogonal. Since (A, B) is stabilizable, the pair (A_o, B_o) must be reachable in a finite number of steps $\kappa \leq n_o$ based on the dimension of A_o. This reachability index k is fixed throughout the rest of the paper. Therefore, our modified state equation is as follows :

$$\begin{bmatrix} x^s_{t+1} \\ x^o_{t+1} \end{bmatrix} = \begin{bmatrix} A_s x^s_t \\ A_o x^o_t \end{bmatrix} + \begin{bmatrix} B_s \\ B_o \end{bmatrix} u_t + \begin{bmatrix} w^s_t \\ w^o_t \end{bmatrix}$$

Let $\mathcal{Y}_t := \{y_0, \dots, y_t\}$ denote the set of output observations till time t. We now fix a prediction horizon $N \in \mathbb{N}_+$, with $N \geq \kappa$ [Giving enough steps for the system to reach the desired state] and define the cost function $\mathcal{J}_t$ as follows :

$$\mathcal{J}_t = \mathbb{E}\left(\|x_{t+N}\|^2_{Q_N} + \sum_{k=0}^{N-1} (\|x_{t+k}\|^2_{Q_k} + \|u_{t+k}\|^2_{R_k}) \right)$$

Where Q_k and R_k are symmetric positive semi-definite matrices, for $k = 0, \dots, N-1$ and Q_N is symmetric positive semi-definite as well. Compact notation for system evolution starting at t over a single prediction horizon N is as follows

$$X_t = \mathcal{A}x_t + \mathcal{B}U_t + \mathcal{D}W_t, \quad Y_t = \mathcal{C}X_t + V_t$$

Where,

$X_t = \begin{bmatrix} x_t^T & x_{t+1}^T & \dots & x_{t+N}^T \end{bmatrix}^T$ $\quad Y_t = \begin{bmatrix} y_t^T & y_{t+1}^T & \dots & y_{t+N}^T \end{bmatrix}^T$ $\quad \mathcal{C} = diag\{C, \dots, C\}$: NxN matrix

$U_t = \begin{bmatrix} u_t^T & u_{t+1}^T & \dots & u_{t+N-1}^T \end{bmatrix}^T$ $\quad V_t = \begin{bmatrix} v_t^T & v_{t+1}^T & \dots & v_{t+N}^T \end{bmatrix}^T$

$W_t = \begin{bmatrix} w_t^T & w_{t+1}^T & \dots & w_{t+N-1}^T \end{bmatrix}^T$ $\quad \mathcal{A} = \begin{bmatrix} I & A & \dots & A^N \end{bmatrix}^T$

$$\mathcal{B} = \begin{bmatrix} 0 & \dots & \dots & 0 \\ B & \ddots & & \vdots \\ AB & B & \ddots & \vdots \\ \vdots & & \ddots & 0 \\ A^{N-1}B & \dots & AB & B \end{bmatrix} \qquad \mathcal{D} = \begin{bmatrix} 0 & \dots & \dots & 0 \\ I & \ddots & & \vdots \\ A & I & \ddots & \vdots \\ \vdots & & \ddots & 0 \\ A^{N-1} & \dots & A & I \end{bmatrix}$$

Moreover, the cost function at t can also be rewritten as :

$$\mathcal{J}_t = \mathbb{E}_{\mathcal{Y}_t}\left[\|X_t\|^2_{\mathcal{Q}} + \|U_t\|^2_{\mathcal{R}} \right]$$

Where, $\mathcal{Q} = diag\{Q_0, \dots, Q_N\}$ and $\mathcal{R} = diag\{R_0, \dots, R_{N-1}\}$. Now our cost function ($\mathcal{J}_t$) is a conditional expectation given observations till t, computing this would require $f(x_t|\mathcal{Y}_t)$, the conditional density of state given past and current measurements. For $t, s \in \mathbb{N}$, define $\hat{x}_{t|s} = \mathbb{E}_{\mathcal{Y}_t}[x_t]$: the Estimate of state at t, given s observations and $P_{t|s} = \mathbb{E}_{\mathcal{Y}_t}[(x_t - \hat{x}_{t|s})(x_t - \hat{x}_{t|s})^T]$: Covariance matrix of the estimate

Proposition 2 : Using Assumption (iii) and further assuming that u_t is a deterministic function of $\mathcal{Y}_t$, we can say $f(x_t|\mathcal{Y}_t)$ and $f(x_{t+1}|\mathcal{Y}_t)$ are the probability densities of Gaussians : $\mathcal{N}(\hat{x}_{t|t}, P_{t|t})$ and $\mathcal{N}(\hat{x}_{t+1|t}, P_{t+1|t})$, respectively, with $P_{t|t} \geq 0$ and $P_{t+1|t} \geq 0$ for $t = -1, 0, 1, 2 \dots$. Moreover, their conditional means and covariances are computed iteratively starting at $(\hat{x}_{0|-1}, P_{0|-1}) := (0, \Sigma_{x_0})$ as follows [7] :

$$\begin{aligned}
\hat{x}_{t+1|t+1} &= \hat{x}_{t+1|t} + P_{t+1|t}C^T(CP_{t+1|t}C^T + \Sigma_v)^{-1}(y_{t+1} - C\hat{x}_{t+1|t}) \\
P_{t+1|t+1} &= P_{t+1|t} - P_{t+1|t}C^T(CP_{t+1|t}C^T + \Sigma_v)^{-1}CP_{t+1|t} \\
\hat{x}_{t+1|t} &= A\hat{x}_{t|t} + Bu_t \\
P_{t+1|t} &= AP_{t|t}A^T + \Sigma_w
\end{aligned}$$

From proposition 2, we can observe that the conditional mean and covariance propagate similarly to a Kalman filter. Using this iterative method, we can characterize the conditional density at any time, which is then used in evaluating the cost function. In this paper, the control input (u_t) for the receding horizon control case will be a non-linear function of $\mathcal{Y}_t$, therefore, we cannot assume that all probability distributions in the problem are Gaussian like LQG. We make a slight change in notation of the estimate $\hat{x}_{t|t}$ to $\hat{x}_t$ for convenience :

$$\hat{x}_t = \begin{bmatrix} (\hat{x}_t^s)^T & (\hat{x}_t^o)^T \end{bmatrix}$$

Corresponding to the Jordan decomposition form of the modified state equation. Let $K_t = (AP_{t|t}A^T + \Sigma_w)C^T(C(AP_{t|t}A^T + \Sigma_w)C^T + \Sigma_v)^{-1}$ and define $\Gamma_t = I - K_tC$ & $\Phi_t = \Gamma_t A$, using this we can write the estimation error vector over a single prediction horizon N as :

$$E_t = X_t - \hat{X}_t = \mathcal{F}_t e_t + \mathcal{G}_t W_t - \mathcal{H}_t V_t$$

Where,

$$e_t = x_t - \hat{x}_t, \quad \hat{X}_t = \begin{bmatrix} \hat{x}_t^T & \dots & \hat{x}_{t+N}^T \end{bmatrix}^T$$

$$\mathcal{G}_t = \begin{bmatrix} 0 & \dots & 0 & 0 \\ \Gamma_t & \dots & 0 & 0 \\ \Phi_{t+1}\Gamma_t & \dots & 0 & 0 \\ \vdots & & \vdots & \vdots \\ \Phi_{t+N-2}\dots\Phi_{t+1}\Gamma_t & \dots & \Gamma_{t+N-2} & 0 \\ \Phi_{t+N-1}\dots\Phi_{t+1}\Gamma_t & \dots & \Phi_{t+N-1}\Gamma_{t+N-2} & \Gamma_{t+N-1} \end{bmatrix} \quad \mathcal{F}_t = \begin{bmatrix} I \\ \Phi_t \\ \Phi_{t+1}\Phi_t \\ \vdots \\ \Phi_{t+N-1}\dots\Phi_t \end{bmatrix}$$

$$\mathcal{H}_t = \begin{bmatrix} 0 & 0 & \dots & 0 & 0 \\ 0 & K_t & \dots & 0 & 0 \\ 0 & \Phi_{t+1}K_t & \dots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & \Phi_{t+N-2}\dots\Phi_{t+1}K_t & \dots & K_{t+N-2} & 0 \\ 0 & \Phi_{t+N-1}\dots\Phi_{t+1}K_t & \dots & \Phi_{t+N-1}K_{t+N-2} & K_{t+N-1} \end{bmatrix}$$

Similarly, the innovations sequence can be written as :

$$Y_t - \hat{Y}_t = \mathcal{C}\mathcal{F}_t e_t + \mathcal{C}\mathcal{G}_t W_t + (I - \mathcal{C}\mathcal{H}_t)V_t, \text{ Where } \hat{Y}_t := \mathcal{C}\hat{X}_t$$

We can observe that the Innovation sequence over the prediction horizon is independent of the control input U_t. Moreover, under our previous assumptions, the error vector (e_t) is a Gaussian random variable with zero mean and $P_{t|t}$ variance

2.1 Optimization problem and Control Policies :

Ideally, the cost function would have to be minimized over the class of all causal feedback functions. But this problem is extremely difficult to solve, therefore, we restrict attention to a subclass of causal feedback policies for which the optimization problem is tractable. Hence the suggested approach is, given a control horizon $N_c \geq 1$ and a prediction horizon $N \geq N_c$ we would like to periodically minimize the cost function at $t = 0, N_c, 2N_c, \dots,$ over the class of following control policies :

$$u_{t+\ell} = \eta_{t+\ell} + \sum_{i=0}^{\ell} \theta_{t+\ell,t+i}\varphi_i(y_{t+i} - \hat{y}_{t+i})$$

Where,

$$\ell = 0, 1, \dots, N-1$$

$\hat{y}_i = C\hat{x}_i$ is the output of the estimator

$\varphi_i(z) = (\varphi_{i,1}(z_1), \dots, \varphi_{i,p}(z_p))$, for any vector$z = (z_1, \dots, z_p) \in \mathbb{R}^p$

where $\varphi_{i,j} : \mathbb{R} \to \mathbb{R}$ is any function with $\sup_{s\in\mathbb{R}} |\varphi_{i,j}(s)| \leq \varphi_{max} \leq \infty$, for some $\varphi_{max} > 0$

the feedback gains $\theta_{l,i} \in \mathbb{R}^{m\times p}$ and the affine terms $\eta_\ell \in \mathbb{R}^m$ are the decision variables

In our policy class, we can observe that computing $u_{t+\ell}$ requires the measured outputs starting from t up to $t + \ell$ only, which needs finite memory. Additionally, the difference in the measured outputs is saturated before being used in our control policy, this is done to consider unbounded noise and still end up with bounded policies. Moreover, we can choose any saturation function $\varphi_i(\cdot)$, e.g., piecewise linear, sigmoidal, etc. Again, the control policy can be compactly written over the entire prediction horizon as

$$U_t = \boldsymbol{\eta}_t + \Theta_t\varphi(Y_t - \hat{Y}_t)$$

Where,

$$\Theta_t = \begin{bmatrix} \theta_{t,t} & 0 & \cdots & 0 \\ \theta_{t+1,t} & \theta_{t+1,t+1} & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ \theta_{t+N-1,t} & \theta_{t+N-1,t+1} & \cdots & \theta_{t+N-1,t+N-1} \end{bmatrix}$$

$$\varphi(Y_t - \hat{Y}_t) = \begin{bmatrix} \varphi_0(y_t - \hat{y}_t) \\ \vdots \\ \varphi_{N-1}(y_{t+N-1} - \hat{y}_{t+N-1}) \end{bmatrix}, \quad \boldsymbol{\eta} = \begin{bmatrix} \eta_t \\ \eta_{t+1} \\ \vdots \\ \eta_{t+N-1} \end{bmatrix}$$

Since the innovation vector $Y_t - \hat{Y}_t$ is not a function of $\boldsymbol{\eta}_t$ and Θ_t, the control inputs U_t are affine in the decision variables; this will be used to show the convexity of the optimization problem in the next section. Finally, the bounds on the control input can be rewritten as: $\|U_t\|_\infty \leq U_{max}\ \forall\ t = 0, N_c, 2N_c, \ldots$
Hence, the optimization problem to be solved periodically at times $t = 0, N_c, 2Nc, \ldots$ as :

$$\min_{(\boldsymbol{\eta}_t, \Theta_t)} \mathcal{J}_t \tag{3}$$

Under constraints:

$$X_t = \mathcal{A}x_t + \mathcal{B}_t U_t + \mathcal{D}W_t \tag{4}$$
$$Y_t = \mathcal{C}X_t + V_t \tag{5}$$
$$U_t = \boldsymbol{\eta}_t + \Theta_t \varphi(Y_t - \hat{Y}_t) \tag{6}$$
$$\|U_t\|_\infty \leq U_{max} \tag{7}$$

3 Main Results

Now that the optimization problem is formulated and is feasible recursively every N_c steps, there is no stability guarantee for the receding horizon controller on using the control policies. Since the noise is unbounded, we cannot assume a compact, robust, positively invariant terminal region [terminal states could end up anywhere depending on the noise, instead of reaching the neighborhood of a desired state]. To tackle this problem, this paper introduces a stability constraint that, if recursively feasible, would render the closed-loop system mean-square bounded. [39] shows that this constraint is indeed recursively feasible.
We can see that the state estimate at the end of each control horizon (that is, at $t + N_c$, for $t = 0, N_c, 2N_c, \ldots$ is

$$\hat{x}_{t+N_c} = A^{N_c}\hat{x}_t + \mathcal{R}_{N_c}(A, B) \begin{bmatrix} u_t \\ \vdots \\ u_{t+N_c-1} \end{bmatrix} + \Xi_t \tag{8}$$

where $\mathcal{R}_{N_c}(A, B) = \begin{bmatrix} A^{N_c-1}B & \cdots & AB & B \end{bmatrix}$ is the reachability matrix of (A, B) at step N_c and

$$\Xi_t := \begin{bmatrix} A^{N_c-1}K_t & A^{N_c-2}K_{t+1} & \cdots & K_{t+N_c-1} \end{bmatrix} \left(CA \begin{bmatrix} e_t \\ \vdots \\ e_{t+N_c-1} \end{bmatrix} + C \begin{bmatrix} w_t \\ \vdots \\ w_{t+N_c-1} \end{bmatrix} + \begin{bmatrix} v_t \\ \vdots \\ v_{t+N_c-1} \end{bmatrix} \right)$$

This Ξ_t indicates the noise entering the system and has a fourth-order bounded moment. For boundedness of state variance, the variance of x_{t+N_c} needs to be bounded. However, since it contains Ξ_t, we need to bound its moments. Proposition 3 shows that at least after some time T', there is a uniform bound on its first moment.

Proposition 3 : There exists an integer T' and a positive constant ζ,depending on the given problem parameters, such that

$$\mathbb{E}\left[\|\Xi_t\| \big| \{y_0, \ldots, y_t\}\right] \leq \zeta, \quad \forall t \geq T' \tag{9}$$

Using the constant $zeta$, the following "drift condition" on the orthogonal part of the system is enforced: $\forall \varepsilon > 0$ and for every $t = 0, N_c, 2N_c, \dots,$ the control inputs $U_t \in \mathbb{U}$ is designed so that

$$\left\| A_o^{N_c} \hat{x}_t^o + \mathcal{R}_{N_c}(A_o, B_o) \begin{bmatrix} u_t \\ \vdots \\ u_{t+N_c-1} \end{bmatrix} \right\| \leq \|\hat{x}_t^o\| - \left(\zeta + \frac{\varepsilon}{2}\right), \quad \text{whenever } \|\hat{x}_t^o\| \geq \zeta + \varepsilon \tag{10}$$

This condition guarantees that, on average, the state norm contracts every N_c steps, which is then used to show that the closed-loop system is mean-square bounded. N_c must be chosen (depending on reachability index κ such that the constraint in eqn. 10 is feasible.
The constraint in eqn. 10 is augmented to the optimization problem in eqn. 3 to obtain the optimization problem for the Basic Stochastic Receding Horizon Algorithm.

$$\min_{(\boldsymbol{\eta}_t, \Theta_t)} \mathcal{J}_t \tag{11}$$

Under constraints:

$$\begin{aligned} X_t &= \mathcal{A}x_t + \mathcal{B}_t U_t + \mathcal{D}W_t \\ Y_t &= \mathcal{C}X_t + V_t \\ U_t &= \boldsymbol{\eta}_t + \Theta_t \varphi(Y_t - \hat{Y}_t) \\ \|U_t\|_\infty &\leq U_{max} \end{aligned}$$
$$\left\| A_o^{N_c} \hat{x}_t^o + \mathcal{R}_{N_c}(A_o, B_o) \begin{bmatrix} u_t \\ \vdots \\ u_{t+N_c-1} \end{bmatrix} \right\| \leq \|\hat{x}_t^o\| - \left(\zeta + \frac{\varepsilon}{2}\right), \quad \text{whenever } \|\hat{x}_t^o\| \geq \zeta + \varepsilon$$

A few further assumptions are made to turn the above optimization into a second-order cone program (SOCP). They are:

i The control and prediction horizons satisfy $N \geq N_c = \kappa$,where κ is the reachability index of the orthogonal subsystem (A_o, B_o).

ii The control authority $U_{\max} \geq U^*_{\max}$, where $U^*_{\max} := \sigma_{\min}(\mathcal{R}_{N_c}(A_o, B_o))^{-1} \left(\zeta + \frac{\varepsilon}{2}\right)$

The paper presents the main theorem that says that the optimization is complex and can be approximated by an SOCP.

Theorem 5: Considering the system described in the start and supposing that all the assumptions stated so far are satisfied, we have:

i For every time $t = 0, N_c, 2N_c, \dots,$ the optimization problem eqn. 11 is convex and can be conservatively approximated and solved via the following globally (hence recursively) feasible second order cone program (SOCP):

$$\min_{z_1, z_2, z_3, \boldsymbol{\eta}_t, \Theta_t} z_1 \tag{12}$$

subject to

$$\|\boldsymbol{\eta}_t + \Theta_t \Lambda_t^\varphi\|^2_{\mathcal{M}} + \mathrm{tr}\left(\Theta_t^{\mathrm{T}} \mathcal{M} \Theta_t (\Lambda_t^{\varphi\varphi} - \Lambda_t^\varphi \Lambda_t^{\varphi \mathrm{T}})\right) + 2\hat{x}_t^{\mathrm{T}} \mathcal{A}^{\mathrm{T}} \mathcal{Q} \mathcal{B} \boldsymbol{\eta}_t + 2\mathrm{tr}\left(\Theta_t^{\mathrm{T}} \mathcal{B}^{\mathrm{T}} \mathcal{Q}(\mathcal{D}\Lambda_t^{w\varphi} + \mathcal{A}\Lambda_t^{x\varphi})\right) \leq z_1 \tag{13}$$

$$|(\boldsymbol{\eta}_t)_i| + \|(\Theta_t)_i\|_1 \varphi_{\max} \leq U_{\max} \quad \forall i = 1, \dots, Nm \tag{14}$$

Whenever $\|\hat{x}_t^o\| \geq \zeta + \epsilon$:

$$\begin{aligned} \left\| A_o^{N_c} \hat{x}_t^o + \mathcal{R}_{N_c}(A_o, B_o)(\boldsymbol{\eta}_t)_{1:N_c m} \right\| &\leq z_2 \\ \|\mathcal{R}_{N_c}(A_o, B_o)(\Theta_t)_{1:N_c m}\|_\infty &\leq z_3 \\ z_2 + \sqrt{n_0} \varphi_{\max} z_3 &\leq \|\hat{x}_t^o\| - \left(\zeta + \frac{\epsilon}{2}\right) \end{aligned} \tag{15}$$

where $\mathcal{M} := \mathcal{R} + \mathcal{B}^{\mathrm{T}}\mathcal{Q}\mathcal{B}$ and

$$\begin{aligned}
\Lambda_t^{\varphi} &:= \mathbb{E}[\varphi(Y_t - \hat{Y}_t)\big|\{y_0, \ldots, y_t\}] \\
\Lambda_t^{x\varphi} &:= \mathbb{E}[x_t\varphi(Y_t - \hat{Y}_t)^{\mathrm{T}}\big|\{y_0, \ldots, y_t\}] \\
\Lambda_t^{w\varphi} &:= \mathbb{E}[W\varphi(Y_t - \hat{Y}_t)^{\mathrm{T}}\big|\{y_0, \ldots, y_t\}] \\
\Lambda_t^{\varphi\varphi} &:= \mathbb{E}[\varphi(Y_t - \hat{Y}_t)\varphi(Y_t - \hat{Y}_t)^{\mathrm{T}}\big|\{y_0, \ldots, y_t\}]
\end{aligned} \tag{16}$$

ii The optimization in eqn. 11 done via the SOCP approximation in part (i) renders the closed-loop system mean-square bounded,i.e., for any initial y_0, there exists a (computable) finite constant $\gamma > 0$, depending on the given problem parameters, such that

$$\sup_{t\in\mathbb{N}} \mathbb{E}[\|x_t\|^2\big|\mathcal{Y}_0] \leq \gamma \tag{17}$$

Algorithm 1 Stochastic Receding Horizon Algorithm (Without additional state and energy bounds)

Require: density $f(x_0|\mathcal{Y}_{-1}) := \mathcal{N}(0, \Sigma_{x_0})$ (Can be arbitrary mean as well)

set $t \leftarrow 0$, $\hat{x}_{0|-1} \leftarrow 0$ (becomes arbitrary mean if used), and $P_{0|-1} \leftarrow \Sigma_{x_0}$
loop
    for $i = 0$ to $N_c - 1$ **do**
        measure y_{t+i}
        calculate $\hat{x}_{t+i}(= \hat{x}_{t+i|t+i})$ and $P_{t+i|t+i}$ using **Proposition 2**
        if $i = 0$ **then**
            solve the optimization problem (11) for the optimal policy $\{u_t^*, \cdots, u_{t+N-1}^*\}$
        end if
        using the obtained control policy above, compute and apply u_{t+i}^*
        calculate $\hat{x}_{t+i+1|t+i}$ and $P_{t+i+1|t+i}$ using **Proposition 2**
    end for
    set $t \leftarrow t + N_c$
end loop

Till now, the problem formulation was capable of finding a solution for unbounded process noise with bounded control inputs while keeping the state mean square bounded (γ - computed from system dynamics), but in practice, we would like to impose further constraints on state and input vectors. One such example for a class of constraints would be :

$$\mathbb{E}_{\mathcal{Y}_t}\left[||X_t||_{\mathcal{S}}^2 + \mathcal{L}^T X_t\right] \leq \alpha_t \tag{18}$$

$$\mathbb{E}_{\mathcal{Y}_t}\left[||U_t||_{\tilde{\mathcal{S}}}^2\right] \leq \beta_t \tag{19}$$

Where, $\mathcal{S}$ and $\tilde{\mathcal{S}}$ are symmetric positive semi-definite matrices, α_t and β_t are positive values. It is to be noted that these are expectation-type constraints, and tighter constraints are not possible as noise is unbounded. Upon the above constraints, we reformulate our optimization as follows :

$$\min_{\eta_t, \Theta_t} \{\mathcal{J}_t|(4), (5), (6), (7), (10), (18), (19)\} \tag{20}$$

We run into an issue when we try to perform this optimization using algorithm 1; the constraints (18) and (19) are not feasible at time t. To have feasible optimization at every time step, algorithm 1 is modified to perform a bisection search on α_t and β_t values if they aren't feasible initially. The search is performed in the intervals $[\alpha_t, \alpha_t^*]$ and $[\beta_t^*, \beta_t]$, with α_t^* and β_t^* defined as follows :

$$\alpha_t^* = 3tr(\mathcal{A}^T\mathcal{S}\mathcal{A}\mathbb{E}_{\mathcal{Y}_t}[x_t x_t^T] + \mathcal{D}^T\mathcal{S}\mathcal{D}\Sigma_w) + \mathcal{L}^T\mathcal{A}\hat{x}_t + 3Nm\sigma_{max}(\mathcal{B}^T\mathcal{S}\mathcal{B})U_{max}^2 + ||\mathcal{L}^T\mathcal{B}||_1 U_{max}$$

$$\beta_t^* = Nm\sigma_{max}(\tilde{\mathcal{S}})U_{max}^2$$

The search is performed iteratively till the change in α_t and β_t falls below a threshold (δ) or a maximum number of iterations is reached ($\bar{\nu}$), in order to maintain reasonable computational costs.

We now expand the constraints such that the entire optimization problem becomes convex at every time $t = 0, N_c, 2N_c, ...$ and it can be solved using SOCP, like algorithm 1, the formulation from Theorem 5 is maintained, and the following conditions are appended to the optimization :

$$
\begin{aligned}
&\|\eta_t + \Theta_t \Lambda_t^{\varphi}\|^2_{\mathcal{B}^T \mathcal{S} \mathcal{B}} + \operatorname{tr}\left(\Theta_t^T \mathcal{B}^T \mathcal{S} \mathcal{B} \Theta_t (\Lambda_t^{\varphi\varphi} - \Lambda_t^{\varphi} \Lambda_t^{\varphi T})\right) \\
&\quad + 2\hat{x}_t^T \mathcal{A}^T \mathcal{S} \mathcal{B} \eta_t + 2 \operatorname{tr}\left(\Theta_t^T \mathcal{B}^T \mathcal{S}(\mathcal{D}\Lambda_t^{w\varphi} + \mathcal{A}\Lambda_t^{x\varphi})\right) \\
&\quad + \mathcal{L}^T \mathcal{B}(\eta_t + \Theta_t \Lambda_t^{\varphi}) + \operatorname{tr}\left(\mathcal{A}^T \mathcal{S} \mathcal{A} E_{\mathcal{Y}_t}[x_t x_t^T]\right) + \operatorname{tr}(\mathcal{D}^T \mathcal{S} \mathcal{D} \Sigma_w) + \mathcal{L}^T \mathcal{A} \hat{x}_t \le \alpha_t \\
&\|\eta_t + \Theta_t \Lambda_t^{\varphi}\|^2_{\tilde{\mathcal{S}}} + \operatorname{tr}\left(\Theta_t^T \tilde{\mathcal{S}} \Theta_t (\Lambda_t^{\varphi\varphi} - \Lambda_t^{\varphi} \Lambda_t^{\varphi T})\right) \le \beta_t
\end{aligned}
$$

Moreover, using algorithm 2 with the above Cone Approximations would render the closed-loop system mean-square bounded, similar to algorithm 1.

Algorithm 2 Modified Stochastic Receding Horizon Algorithm (With state and energy bounds)

Require: density $f(x_0|\mathcal{Y}_{-1}) := \mathcal{N}(0, \Sigma_{x_0})$ (Can be arbitrary mean as well)
1: set $t \leftarrow 0$, $\hat{x}_{0|-1} \leftarrow 0$(becomes arbitrary mean if used), and $P_{0|-1} \leftarrow \Sigma_{x_0}$
2: **loop**
3:   **for** $i = 0$ to $N_c - 1$ **do**
4:     measure y_{t+i}
5:     calculate $\hat{x}_{t+i}(= \hat{x}_{t+i|t+i})$ and $P_{t+i|t+i}$ using **Proposition 2**
6:     **if** $i = 0$ **then**
7:       solve the optimization problem (20) using the given α_t and β_t
8:       **if** step 7 is feasible **then**
9:         save the optimal sequence $\{u_t^*, u_{t+1}^*, \cdots, u_{t+N-1}^*\}$
10:         **goto** step 28
11:       **else**
12:         set $\bar{\alpha} \leftarrow \alpha_t^*$, $\alpha \leftarrow \alpha_t$, $\bar{\beta} \leftarrow \beta_t^*$, and $\beta \leftarrow \beta_t$
13:         solve the optimization problem (20) using $\bar{\alpha}$ and $\bar{\beta}$ to obtain $\{u_t^*, u_{t+1}^*, \cdots, u_{t+N-1}^*\}$
14:         set $\nu \leftarrow 1$
15:         **repeat**
16:           set $\alpha_t \leftarrow (\bar{\alpha} + \alpha)/2$ and $\beta_t \leftarrow (\bar{\beta} + \beta)/2$
17:           solve the optimization problem (20) using the new α_t and β_t
18:           **if** step 17 is feasible **then**
19:             set $\bar{\alpha} \leftarrow \alpha_t$ and $\bar{\beta} \leftarrow \beta_t$
20:             save the new optimal sequence $\{u_t^*, u_{t+1}^*, \cdots, u_{t+N-1}^*\}$
21:           **else**
22:             set $\alpha \leftarrow \alpha_t$ and $\beta \leftarrow \beta_t$
23:           **end if**
24:           set $\nu \leftarrow \nu + 1$
25:         **until** $(|\bar{\alpha} - \alpha| \le \delta$ and $|\bar{\beta} - \beta| \le \delta)$ or $(\nu > \bar{\nu})$
26:       **end if**
27:     **end if**
28:     apply u_{t+i}^*
29:     calculate $\hat{x}_{t+i+1|t+i}$ and $P_{t+i+1|t+i}$ using **Proposition 2**
30:   **end for**
31:   set $t \leftarrow t + N_c$
32: **end loop**

The proofs for the Propositions, Theorems and Corollaries in sections 2 and 3 are available in the appendix of Hokayem et.al.[8].

4 Discussion

4.1 Recursive Feasibility and Mean Squared Boundedness

The SOCPs solved above are globally feasible, independently of the initial conditions of the plant and the estimator, and hence, there is no need for an initially feasible and invariant set of initial conditions. This notion of stability is weaker than that of asymptotic stability or input-to-state stability (Nominal methods), as this only bounds the average state norm. But this is the best performance metric that can be used in this case because of unbounded process and measurement noise, it is impossible to guarantee that the states would converge to the origin or that it would be ultimately bounded. Using the assumptions made previously, we can derive constant γ, such that $\mathbb{E}[||x_t||^2] \leq \gamma$.

4.2 Offline Computation of Λ Matrices

The objective function along with the constraints for algorithms 1 and 2 have been proven to be recursively feasible and are Second Order Cone Programming problems for which different numerical solvers are available, like fmincon, sdpt3[9, 10], etc. But for its constraints to be tractable for optimization during simulations, we need to compute $\Lambda_t^{\varphi}, \Lambda_t^{w\varphi}, \Lambda_t^{\varphi\varphi}$, and $\Lambda_t^{x\varphi}$. The brute force approach to compute these matrices would be to perform numerical integration w.r.t the Gaussian measures $w_t, \ldots, w_{t+N-1}$, $v_t, \ldots, v_{t+N-1}$ and $(x_t - \hat{x}_t)$ given $\mathcal{Y}_t$. The large dimensions of the integration space would hinder this method from being used for online computations.

Proposition : Under earlier assumptions *iii* and *iv*, it can be proven that the discrete time Algebraic Riccati Equation: $P = A[P - PC^T(CPC^T + \Sigma_v)^{-1}CP]A^T + \Sigma_w$, has a unique solution $P^* \geq 0$. Moreover $P_{t+1|t}$ converges to P^* as $t \to \infty$, for any initial condition $P_{0|-1} \geq 0$. As a consequence of this convergence, we can see from **Proposition 2**, that $P_{t|t}$ converges to $P^o = P^* - P^*C^T(CP^*C^T + \Sigma_v)^{-1}CP^*$, i.e. the asymptotic error covariance matrix of the estimator $(\hat{x}_t)$ converges to P^o.

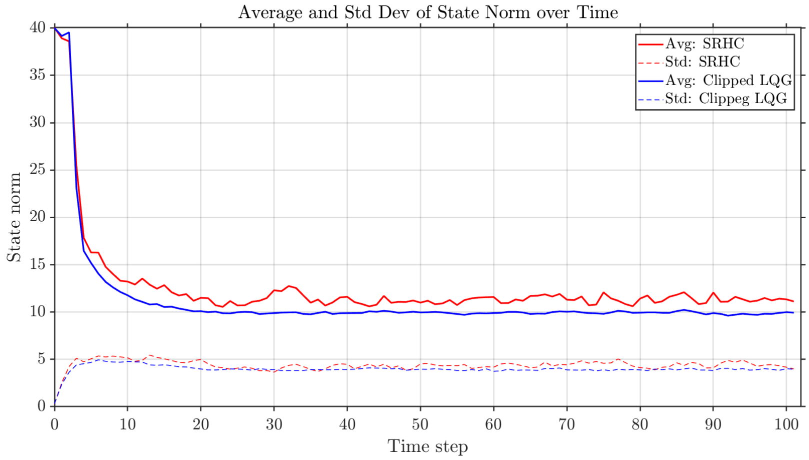
Using this we can compute $\Lambda_t^{\varphi}, \Lambda_t^{w\varphi}$, and $\Lambda_t^{\varphi\varphi}$, as they only depend on $P_{t|t}$ and they can be computed offline ignoring their transience, because the error covariance matrix in principle converges in few iterations. Now we by interpolating $\Lambda_t^{x\varphi}$ we can reduce its requirements for online computation, $\Lambda_t^{x\varphi}(\hat{x}_t, P_{t|t}) = \Lambda_t^{e\varphi}(P_{t|t}) + \hat{x}_t\Lambda_t^{\varphi}(P_{t|t})^T$, Where $\Lambda_t^{e\varphi} := \mathbb{E}[(x_t - \hat{x}_t)\varphi(Y_t - \hat{Y}_t)^T]$. Moreover, since $\Lambda_t^{e\varphi}$ is dependent on $x_t - \hat{x}_t$ and $x_t - \hat{x}_t$ is conditionally zero mean given observations, it can be computed offline and stored, thereby allowing easier computation of $\Lambda_t^{x\varphi}$ on the fly. Similarly, the other Λ_t can be computed offline by performing Monte Carlo simulations of the state space and averaging them out to compute the expectation. [*In our case, we computed the expectation over* 10^5 *sampled trajectories*]

5 Simulations & Results

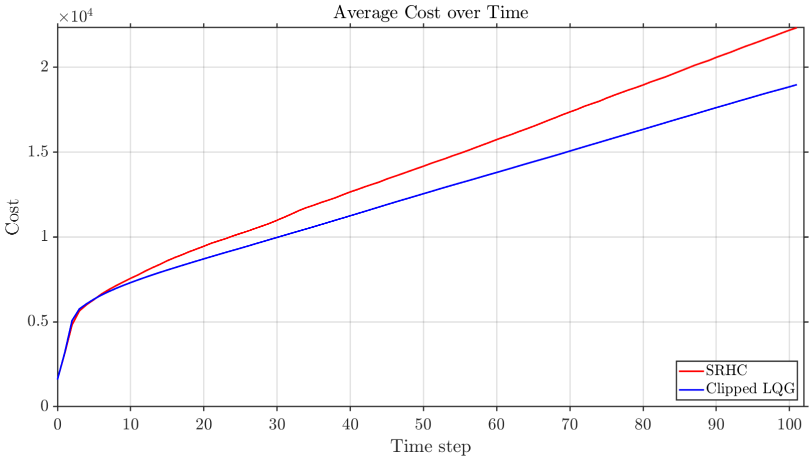
Simulations for algorithms were performed using the following example :

$$A = \begin{bmatrix} 0.9 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \cos(\psi) & -\sin(\psi) \\ 0 & 0 & \sin(\psi) & \cos(\psi) \end{bmatrix}, B = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}, \text{ and } C = I$$

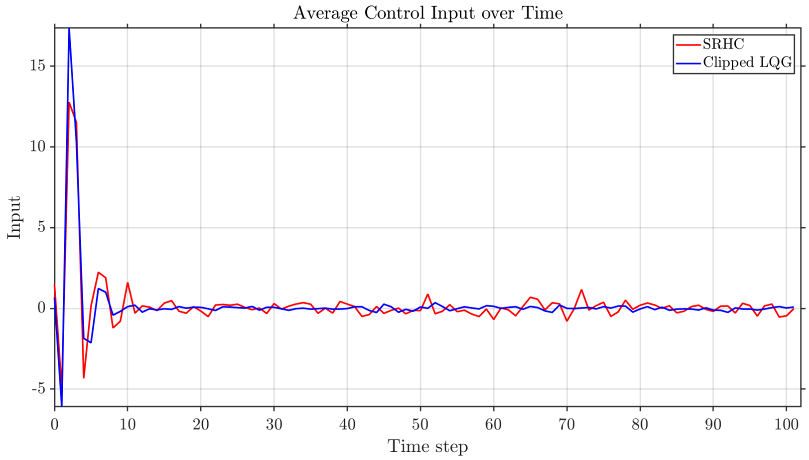
Where $\psi = \frac{\pi}{2}$, the orthogonal part is 3-dimensional and its controllability index (κ) is 3. Moreover, we consider the following during simulation, $x_0 \sim \mathcal{N}(\begin{bmatrix} 20 & -20 & 20 & -20 \end{bmatrix}^T, I), w_t \sim \mathcal{N}(0, 10I), v_t \sim \mathcal{N}(0, 10I), Q = I, R = 1, N = 5, N_c = \kappa = 3$ and φ is the piecewise linear saturation function with $\varphi_{max} = 1$. Also, the theoretical bound on control input $(U_{\max})$ is approx. 453 for $\varepsilon = 10$ and $\zeta = 391.31$. The simulations were run for t = 0 to t = 100 for 100 randomly initialized trajectories. The Λ_t values were precomputed and stored by performing Monte-Carlo simulations on 10^5 trajectories as suggested by the paper. Initially, the FMINCON solver available as default in MATLAB was used. This however was very slow since FMINCON is a general nonlinear programming solver. Hence, we chose the solver SDPT3 [9, 10] as mentioned in [8] which can solve SOCPs efficiently. YALMIP [11] was used to model the problem in a quick and concise manner. The solver, on average, used about 0.1 seconds for each optimization at N_c.



(a) State norm



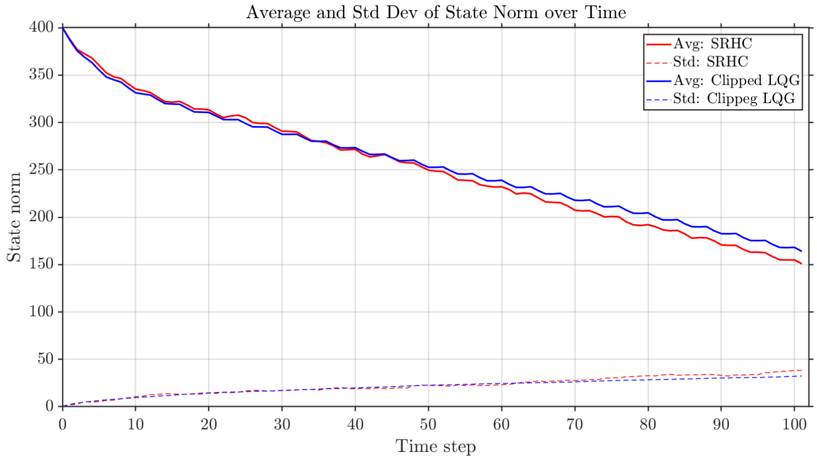
(b) Cost



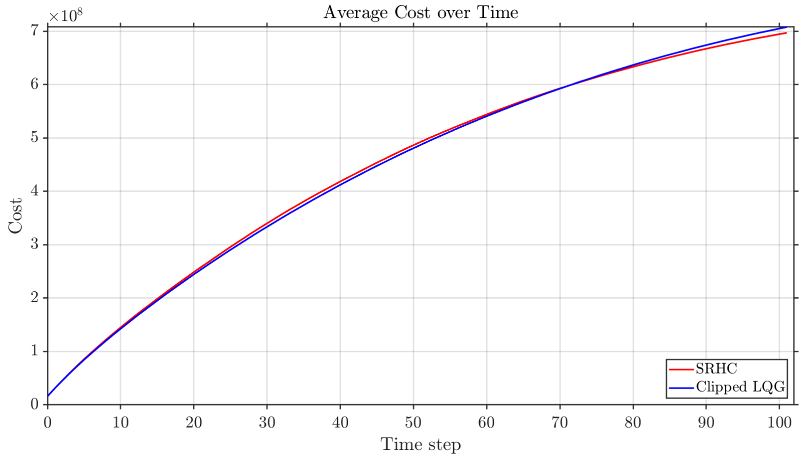
(c) Input

Figure 1: Comparison of State norm, Cost, and Input over time for example 1

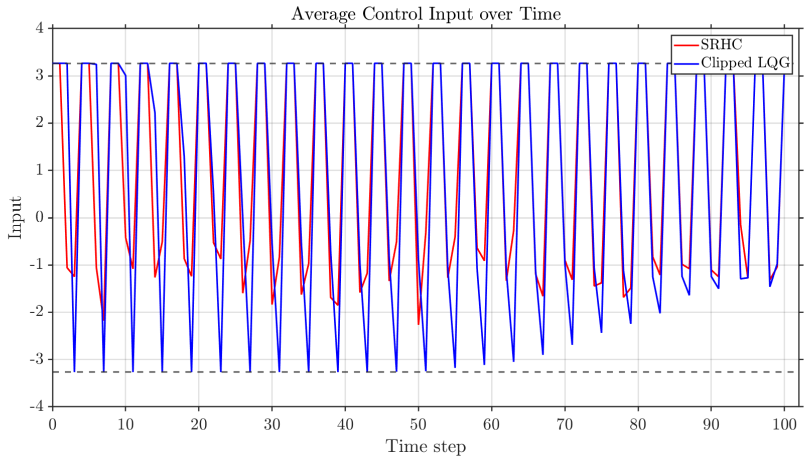
For the above system, Algorithm 1 has been simulated and has been compared with clipped LQG, which is just the standard LQG with saturation added to the computed inputs. The comparison is given in fig. 1. We can observe from fig. 1b that SHRC cost, albeit higher, is close to Clipped LQG. This is because, with such a high bound on the control input, the Clipped LQG behaves as standard LQG, which is the optimal solution in this case as the control input does not reach its theoretical bounds.



(a) State norm



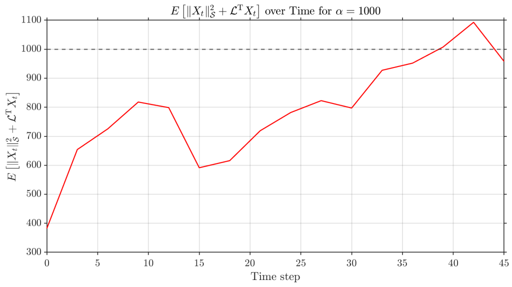
(b) Cost



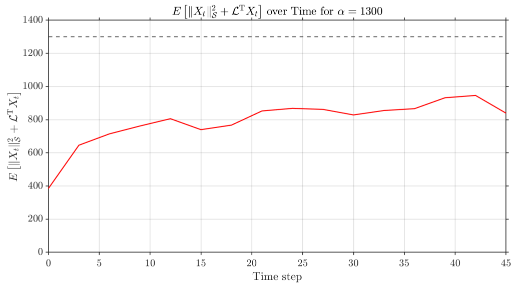
(c) Input

Figure 2: Comparison of State norm, Cost, and Input over time for example 2

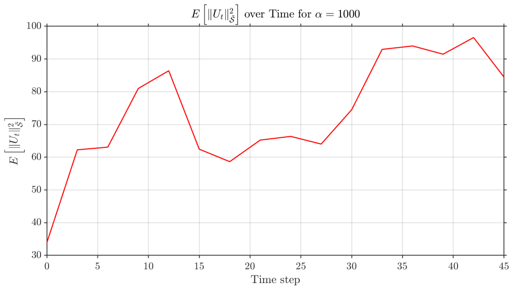
For the next example, we have $Q = 100I$, $R = I$, $\zeta = 2$, $\varepsilon = 0.5$, which would give $U_{max} = 3.2664$. According to the paper, this value of ζ is far below the required theoretical bound and hence, there is no theoretical guarantee that the closed-loop system is mean-square bounded. But, as seen in fig. 2a, Algorithm 1 is still stabilizing and matches the performance of clipped LQG pretty closely. We can see from fig. 2c that the clipped LQG hits saturation way more than Algorithm 1. Despite Algotithm 1 not taking full advantage of the control authority, it is able to outperform clipped LQG in the long run in terms of overall cost (fig. 2b).



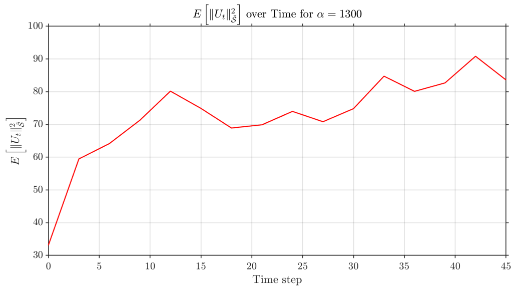
(a) State Constraint, $\alpha = 1000$



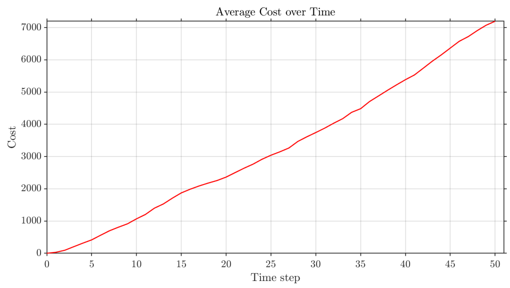
(b) State Constraint, $\alpha = 1300$



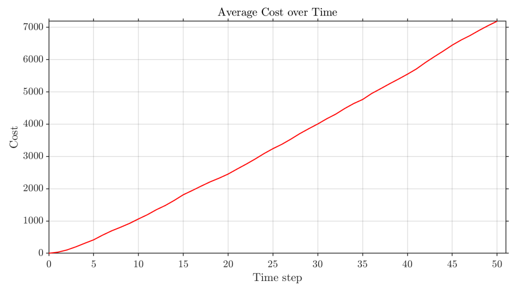
(c) Energy Constraint, $\alpha = 1000$



(d) Energy Constraint, $\alpha = 1300$



(e) Cost, $\alpha = 1000$



(f) Cost, $\alpha = 1300$

Figure 3: Algorithm 2: State and Energy constraints

In the above figure (fig. 3), we have simulated algorithm 2 for example 1, with $x_0 \sim \mathcal{N}(0, I)$, $\beta_t = 1026045 = 5 \cdot U_{\max}^2$, $\mathcal{S} = I = \tilde{\mathcal{S}}$, $\mathcal{L} = 0$, and two values of α_t, 1000 and 1300, α_t and β_t being state and energy constraints as defined in equations 18 and 19 respectively. We can see from figs. 3a and 3b that for $\alpha_t = 1000$, the state constraint is violated at one point, while for $\alpha_t = 1300$, there is no violation. This means that there are times when $\alpha_t = 1000$ is infeasible, meanwhile $\alpha_t = 1300$ is feasible always. Comparing the costs of algorithms 1 and 2, (figs. 1b, 3e and 3f), we can see that adding these constraints reduce the overall cost of the policy (≈ 1000 v.s. 7000).

6 Conclusions

This paper presents a method for stochastic receding horizon control of discrete-time linear systems affected by process and measurement noise, under bounded input policies. It demonstrates that the optimization problem solved periodically is both recursively feasible and convex. Additionally, the paper illustrates how a specific stability condition can be employed to ensure that the application of the receding horizon controller results in the system state being mean-square bounded. The authors also discuss methods to precompute Λ_t matrices offline for faster computation, highlighting the efficiency of their algorithm as compared to existing methods. Moreover the paper extends its theoretical proofs for a more practical case like state and energy constraints.

Link to code : SRHC Implementation GitHub Repository

References

[1] James Primbs and Chang Sung. "Stochastic Receding Horizon Control of Constrained Linear Systems With State and Control Multiplicative Noise". In: *Automatic Control, IEEE Transactions on* 54 (Mar. 2009), pp. 221–230. DOI: 10.1109/TAC.2008.2010886.

[2] Mark Cannon, Basil Kouvaritakis, and Xingjian Wu. "Model predictive control for systems with stochastic multiplicative uncertainty and probabilistic constraints". In: *Automatica* 45.1 (2009), pp. 167–172. ISSN: 0005-1098. DOI: https://doi.org/10.1016/j.automatica.2008.06.017. URL: https://www.sciencedirect.com/science/article/pii/S0005109808003725.

[3] P.D. Couchman, M. Cannon, and B. Kouvaritakis. "Stochastic MPC with inequality stability constraints". In: *Automatica* 42.12 (2006), pp. 2169–2174. ISSN: 0005-1098. DOI: https://doi.org/10.1016/j.automatica.2006.07.006. URL: https://www.sciencedirect.com/science/article/pii/S0005109806002846.

[4] Aharon Ben-Tal, Stephen Boyd, and Arkadi Nemirovski. "Extending Scope of Robust Optimization: Comprehensive Robust Counterparts of Uncertain Problems". In: *Mathematical Programming* 107.1 (2006), pp. 63–89. ISSN: 1436-4646. DOI: 10.1007/s10107-005-0679-z. URL: https://doi.org/10.1007/s10107-005-0679-z.

[5] A. Ben-Tal et al. "Adjustable Robust Solutions of Uncertain Linear Programs". In: *Mathematical Programming* 99.2 (2004), pp. 351–376. ISSN: 1436-4646. DOI: 10.1007/s10107-003-0454-y. URL: https://doi.org/10.1007/s10107-003-0454-y.

[6] Paul Goulart, Eric Kerrigan, and Jan Maciejowski. "Optimization Over State Feedback Policies for Robust Control with Constraints". In: *Automatica* 42 (Apr. 2006), pp. 523–533. DOI: 10.1016/j.automatica.2005.08.023.

[7] P. R. Kumar and P. Varaiya. *Stochastic Systems: Estimation, Identification, and Adaptive Control*. Prentice Hall, 1986.

[8] Peter Hokayem et al. "Stochastic Receding Horizon Control with Output Feedback and Bounded Controls". In: *Automatica* 48.1 (2012), pp. 77–88. DOI: 10.1016/j.automatica.2011.09.048. URL: https://doi.org/10.1016/j.automatica.2011.09.048.

[9] K. C. Toh, M. J. Todd, and R. H. Tütüncü. "SDPT3 — a Matlab software package for semidefinite programming". In: *Optimization Methods and Software* 11.1-4 (1999), pp. 545–581. DOI: 10.1080/10556789908805762.

[10] R. H. Tütüncü, K. C. Toh, and M. J. Todd. "Solving semidefinite-quadratic-linear programs using SDPT3". In: *Mathematical Programming*. Series B 95 (2003), pp. 189–217. DOI: 10.1007/s10107-002-0347-5.

[11] J. Löfberg. "YALMIP : A Toolbox for Modeling and Optimization in MATLAB". In: *In Proceedings of the CACSD Conference*. Taipei, Taiwan, 2004.