Random Features Approximation for Fast Data-Driven Control

Kimia Kazemian
Cornell University
kk983@cornell.edu

Sarah Dean
Cornell University
sdean@cornell.edu

Abstract

The goal of data-driven nonlinear control problems is to guarantee stability or safety of an unknown system. We consider a method based on Control Certificate Functions (CCFs) that uses Gaussian Process (GP) regression to learn unknown quantities for control affine dynamics. Computing the GP estimator can become prohibitively expensive for large datasets, which is an issue since speed is critical in real time control systems. We introduce a random feature approximation of the affine compound kernel to speed up training and prediction time. To ensure that the controller can be robust to these approximations, we provide an error analysis on the approximate mean and variance estimates. Finally, we propose a fast and robust convex optimization based min-norm controller using the error bounds and present preliminary experiments comparing the random features approximation to kernel methods.

1 Introduction

It is important to guarantee properties of safety and stability in modern control applications, even when the model of the system to be controlled is unknown and must be learned from data. *Control Certificate Functions* (CCF) are utilized to build convex optimization based controllers for achieving properties such as safety and stability [18]. CCFs have been used for attaining such properties in a range of applications from robotics to multi-agent systems [2, 1, 14, 12, 9, 13].

Gaussian Processes (GP) are used in various settings in reinforcement learning and control when dynamics of models are not perfectly known [10, 5, 4, 8, 11]. In particular, [6, 7] use this method to learn models for a CCF convex optimization controller. However controller feasibility requires dense sampling of the state space [7], and kernel methods scale poorly in number of training data. This poses an issue in real time feedback systems. To remedy the general problem, [15] introduced Random Fourier Features (RF) to approximate symmetric kernels with lower dimensional feature vectors. This method can significantly improve time and memory complexity of computations. This approach has since been further improved and analyzed in the context of kernel ridge regression [17, 3].

We propose an RF method suitable for safety-critical control of an unknown model from data. We adapt the RF approach to an affine dot product kernel to approximate GP regression. This allows for efficient estimation of model dependent quantities necessary for control. We further provide a probabilistic error analysis to propose a robust second order cone program based controller. Finally, we validate our approach with preliminary numerical experiments comparing the accuracy and speed of our approximation. We hope to further develop this work to compare the behavior of approximate closed-loop control with the full GP based control.

2022 NeurIPS Workshop on Gaussian Processes, Spatiotemporal Modeling, and Decision-making Systems.

2 Background and Problem Setting

In this work, we consider the design of control laws for nonlinear control-affine systems of the form $\dot{x} = f(x) + g(x)u$ where $x \in \mathcal{X} \subset \mathbb{R}^n$ is the system state and $u \in \mathbb{R}^m$ is the control input. We assume that the functions $f : \mathbb{R}^n \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^{n\times m}$ are locally Lipschitz continuous and, without loss of generality, that the origin is the equilibrium point, so $f(0) = 0$. Asuming that we observe the state directly, our goal is to design a state feedback controller for such a system without direct knowledge of f or g. In the following subsections, we detail elements of this approach.

2.1 Control Certificate Functions

To design controllers that guarantee properties like safety and stability, we use an approach based on *control certificate functions* proposed by [18].

Definition 1. *(Control Certificate Function-CCF) A continuously differentiable function $C : \mathbb{R}^n \to \mathbb{R}$ is a* Control Certificate Function (CCF) *with* comparison function $\alpha : \mathbb{R} \to \mathbb{R}_+$ *for control affine dynamics defined by f, g if:*

$$\inf_{u\in\mathbb{R}^m} \underbrace{\nabla C(x)^\top f(x) + \nabla C(x)^\top g(x)u}_{:=\dot{C}(x,u)} + \alpha(C(x)) \leq 0 \quad \forall\ x \in \mathcal{X} \setminus \{0\}\,. \tag{1}$$

When the inequality $\dot{C}(x, u) + \alpha(C(x)) \leq 0$ is satisfied, desired system properties are certified. For example, control Lyapunov functions (CLFs) and control barrier functions (CBFs) are types of CCFs that respectively guarantee stability and safety. To synthesize controllers that ensure the inequality is satisfied, we use it as a constraint in a min-norm quadratic program (QP):

$$u^*(x) = \arg\min_{u\in\mathbb{R}^m} \|u\|_2^2 \text{ s.t. } \nabla C(x)^\top f(x) + \nabla C(x)^\top g(x)u + \alpha(C(x)) \leq 0. \tag{CCF-QP}$$

This controller is feasible at every x by definition, and it guarantees that the system satisfies desired properties. However, to implement this QP, is it necessary to know the dynamics f and g. We consider a setting where the model is unknown[1], but that a valid CCF C and α for the true plant is known. This is a structural assumption and it is met for feedback linearizable systems if the degree of actuation of the true dynamics model is known [19]. For example, many robotic systems satisfy this assumption.

2.2 Data-driven Controller with Gaussian Process Regression

We develop a data-driven control synthesis method which uses data of the form $\{(x_i, \dot{C}(x_i, u_i))\}_{i=1}^n$. We can approximately measure $\dot{C}(x, u)$ using finite differencing methods on sampled trajectories from the true system. By learning a model of $\dot{C}(x, u)$ from this data, the CCF constraint can be implemented in a data-driven manner. However, to ensure the the true system certifies the desired property, we must account for prediction errors. Therefore, we use GP regression.

Under GP assumptions, given a set of finite measurements of features and labels of the form $\{(s_j, h(s_j) + \epsilon_j)\}_{j=1}^n$, where $\epsilon_j \sim \mathcal{N}(0, \lambda_n^2)$ is white measurement noise, and a query point s, a posterior distribution for $h(s)$ can be derived. This can be used as a prediction of $h(s)$ at a query point s, with mean and variance

$$\mu_x(u) = z^\top (K + \lambda_n^2 I)^{-1} k_s, \quad \sigma_x(u)^2 = k(s, s) - k_s^\top (K + \lambda_n^2 I)^{-1} k_s \tag{2}$$

where $K \in \mathbb{R}^{n\times n} = [k(s_i, s_j)]_{i,j\in[n]}$, $k_s = [k(s, s_1);\ \cdots\ ; k(s, s_n)] \in \mathbb{R}^n$, and $z \in \mathbb{R}^n$ is the vector containing the output measurements $z_j = h(s_j) + \epsilon_j$.

Due to the affine structure of the dynamics function with respect to the input signal u, we make use of a specialized kernel first introduced by [7]. Notice that

$$\dot{C}(x, u) = \nabla C(x)^\top f(x) + (\nabla C(x)^\top g(x))u = \Phi_C \begin{bmatrix} 1 \\ u \end{bmatrix} \tag{3}$$

[1] Incorporating a known nominal model to this framework is straightforward [18, 7]

where $\Phi_C \in \mathbb{R}^{1\times(m+1)}$. For convenience of notation, we define $y := [1 \quad u^\top]^\top$, denote $s = [x^\top \quad y^\top]^\top$, and define $\chi = \mathcal{X} \times \{1\} \times \mathbb{R}^m$. This is used to construct the following kernel:

Definition 2. *Define $k_c : \chi \times \chi \to \mathbb{R}$ given by $k_c(s, s') := y^\top \mathrm{diag}([k_1(x, x'), \cdots, k_{m+1}(x, x')])y'$ as the Affine Dot Product (ADP) compound kernel of $(m+1)$ individual kernels $k_i : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$.*

By Theorem 6 in [16], with a probability of at least $1 - \delta_1$, the approximation error of a GP is bounded $|\mu(s) - h(s)| \leq \beta\sigma(s)$ for a some $\beta \geq 0$ depending on δ_1. Applying this result to the control setting where $\mu(s) := \mu_x(u)$, $\sigma(s) := \sigma_x(u)$ are the mean and standard deviation of the GP prediction of $\dot{C}$ at the query point (x, u), we have the following inequality, which holds with a probability of at least $1 - \delta_1$ for $\forall x \in \mathcal{X}$, $[1\ u^\top]^\top \in \mathcal{Y}$, a bounded subset of $\mathbb{R}^{m+1}$

$$\dot{C}_x(u) \leq \mu_x(u) + \beta\sigma_x(u)\,. \tag{4}$$

As first proposed by [7], we construct a Second-Order Cone Program (SOCP) which defines a data-driven min-norm stabilizing feedback control law $u^* \colon \mathbb{R}^n \to \mathbb{R}^m$:

$$u^*(x) = \underset{u\in\mathbb{R}^m}{\arg\min} \|u\|_2^2 \text{ s.t. } \mu_x(u)+\beta\sigma_x(u)+\alpha(C(x)) \leq 0 \qquad \textbf{(GP-CCF-SOCP)}$$

2.3 Random Features Approximation

Random Fourier Features is a technique that can scale up methods for shift invariant kernels, i.e. kernels satisfying $k(x, x') = k(x - x')$, where k is a positive definite function. We consider kernels with $k(0) = 1$. These hold for e.g. the popular RBF kernel. We define

$$\psi(x) := \sqrt{\frac{2}{D}}\left[sin(\omega_1^\top x) \quad cos(\omega_1^\top x) \quad \ldots \quad sin(\omega_{D/2}^\top x) \quad cos(\omega_{D/2}^\top x)\right]^\top \tag{5}$$

where $\omega_i \overset{iid}{\sim} p(\omega)$ and $p(\omega)$ is the Fourier transform of the kernel (which by Böchner's theorem, is a valid density function). Thus we have that $k(x - x') = \mathbb{E}_\omega[\psi_\omega(x)^\top \psi_\omega(x')]$. Sutherland and Schneider [17] show that with probability $1 - \delta_2$,

$$\sup_{x\in\mathcal{X}} |\psi(x)^\top \psi(x) - k(x)| \leq \epsilon \quad \text{for} \quad D \geq \frac{8(d + 2\alpha_\epsilon)}{\epsilon^2}\left[\frac{2}{1 + \frac{2}{d}} \log\frac{\sigma_p l}{\epsilon} + log\frac{\beta_d}{\delta}\right],$$

where l is the diameter of $\mathcal{X}$, $\sigma_p^2 = \mathbb{E}_p\|\omega\|^2$ and β_d, α_e defined in Proposition 1 of [17] is bounded by a constant. For kernel regression, we can compute a random feature approximation in $O(nD^2)$ time and $O(nD)$ memory, which is computationally attractive if $D < n$.

3 Results on Random Features GP Control

To make practical use of the data-driven GP controller introduced above, it is necessary to compute posterior estimates in real-time feedback systems. The time and memory complexity of the kernel computations are thus of immediate concern. As a result, previous work has used only subsets of available training data in an ad hoc manner [7].

Instead, we use a random features approximation to estimate the ADP compound kernel. We approximate only the state-dependent portion, defining the random feaures as:

$$\varphi(s_i)^\top = \left[y_i^1\psi_1(x_i)^\top \quad \ldots \quad y_i^{m+1}\psi_{m+1}(x_i)^\top\right] = (\phi(x_i)y_{x_i})^\top \tag{6}$$

where $\phi(x) := \mathrm{blkdiag}[\psi_1(x); ...; \psi_{m+1}(x)]$. Let $\Phi \in \mathbb{C}^{n\times D(m+1)}$ be the matrix whose i^{th} row is $\varphi(s_i)^\top$. Then using random features, the posterior mean and covariance can be approximated by

$$\hat{\mu}_x(u) = \varphi(s)^\top(\Phi^\top\Phi + \lambda_n I)^{-1}\Phi^\top z \quad \text{and} \quad \hat{\sigma}_x(u) = \lambda_n\varphi(s)^\top(\Phi^\top\Phi + \lambda_n I)^{-1}\varphi(s).$$

Note that the time complexity of computing μ is $O(n(m + 1)^2D^2)$ as opposed to $O(n^3)$.

For the purposes of robustly guaranteeing the CCF condition (1), it is necessary to track how the approximation error accumulates in our computation of the posterior. We provide an error analysis which allows for the design of a quadratic SOCP (RF-CCF-SOCP) which is both computationally efficient and robust.

3.1 Error Bounds

Recall that k_s is a vector containing the kernel $k(s_i, s)$ for $i = 1, ..., n$. Define $U = [u_1; ...; u_n]^\top$.

Proposition 1. *Assume each i-th element of Φ_c (3) is a member of $\mathcal{H}_{k_i}$ with bounded RKHS norm, for $i = 1, ..., m+1$. Assume the ADP compound kernel with bounded kernels k_i and that we have access to measurements z. Assume $\|k_s\| \leq \sqrt{n}\kappa$ and that $\lambda_n = n\lambda$. Let σ_{max} be the max singular value of U. Then with a probability of $1-(\delta_1+\delta_2)$ we have:*

$$|\dot{C}(s) - \hat{\mu}_x(u)| \leq \beta\hat{\sigma}_x(u) + \epsilon(\nu\|u_x\| + \iota\|u_x\|^2 + \Delta) \tag{7}$$

where $\nu := \frac{\sigma_{max}}{\sqrt{n\lambda}}(\sigma_n + \frac{2\beta\kappa}{\sqrt{n}} + 2\beta\epsilon), \iota = \frac{\beta\epsilon\sigma_{max}^2}{n\lambda}, \Delta = \beta\Delta_\sigma + (\beta\kappa + \sqrt{n}\sigma_n)\Delta_\mu,$
$\Delta_\mu = \frac{1}{\lambda\sqrt{n}}[1 + \frac{\kappa\sigma_{max}}{n\sqrt{n\lambda}} + \frac{\kappa}{\sqrt{n\lambda}}], \Delta_\sigma = 1 + \epsilon + \frac{\kappa}{\sqrt{n\lambda}}$

We present the proof of this result in the appendix. This bound allow us to derive a robust and fast data-driven controller:

$$u^*(x) = \underset{u\in\mathbb{R}^m}{\arg\min}\|u\|_2^2 \quad \text{s.t.} \quad \hat{\mu}_x(u) + \beta\hat{\sigma}_x(u) + \epsilon(\nu\|u_x\| + \iota\|u_x\|^2 + \Delta) + \alpha(C(x)) \leq 0$$

(RF-CCF-SOCP)

3.2 Experiments

We present preliminary experiments of the random features approach for an adaptive cruise control system with underlying dynamics:

$$\begin{bmatrix}\dot{v}\\ \dot{w}\end{bmatrix} = \begin{bmatrix}-F_r(v)/m\\ v_0 - v\end{bmatrix} + \begin{bmatrix}1/m\\ 0\end{bmatrix} u,$$

where $x \in \mathbb{R}^2$ is the system state, with v being the forward velocity of the ego car and w the distance between the ego car and the front car, and $u \in \mathbb{R}$ is the ego car's wheel force as control input. The velocity of the front car v_0 is assumed to be a constant 14 m/s, the mass of the ego car m is 3300kg, and $F_r(v) = 0.2 + 10v + 0.5v^2$ is the rolling resistance acting on the ego car. We use these dynamics to generate data, but do not assume they are available for control design.

We use data $\{(x_i, u_i)\}_{i=1}^n$ with $n = 1160$ collected episodically from this system at a sampling frequency of 0.02 under a combined CBF and CLF controller, further detailed in [7]. We set up a supervised learning problem for $C(x) = w - T_h v$. We compute $C(x_i)$ and use forward finite differencing on this sequence to estimate $z_i \approx \dot{C}(x_i, u_i)$.

Using this dataset, we investigate the performance of the random features approximation for a GP with RBF kernel and parameters $\sigma_n = 1, D = 700, \beta = 1$. Both GP and RF regression are implemented in numpy. The training time takes 0.420 seconds for the ADP kernel and 0.065 seconds for its RF approximation. This is a reduction of 85%. Figure 1 plots a subset of this data (from the final episode) and compares the mean and and variance of the predictions. We see that the RF predictions are nearly identical, although sometimes it underestimates the variance compared with the full ADP kernel. This observation highlights the importance of incorporating the error analysis presented in the previous section.

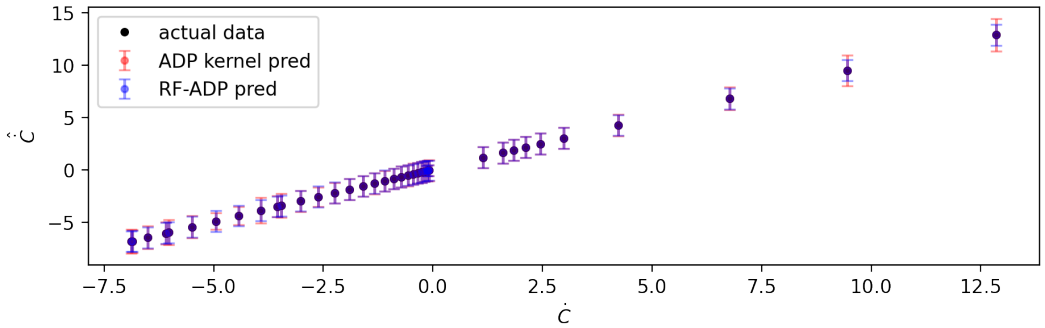

Figure 1: Predicted mean (dot) and variance (bars) for $\dot{C}$ in the final episode of training data.

References

[1] Aaron D Ames, Jessy W Grizzle, and Paulo Tabuada. Control barrier function based quadratic programs with application to adaptive cruise control. In *53rd IEEE Conference on Decision and Control*, pages 6271–6278. IEEE, 2014.

[2] Zvi Artstein. Stabilization with relaxed controls. *Nonlinear Analysis: Theory, Methods & Applications*, 7(11):1163–1173, 1983.

[3] Haim Avron, Michael Kapralov, Cameron Musco, Christopher Musco, Ameya Velingker, and Amir Zandieh. Random Fourier features for kernel ridge regression: Approximation bounds and statistical guarantees. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 253–262. PMLR, 06–11 Aug 2017.

[4] Eric Bradford, Lars Imsland, and Ehecatl Antonio del Rio-Chanona. Nonlinear model predictive control with explicit back-offs for gaussian process state space models. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 4747–4754, 2019.

[5] Jack Caldwell and Joshua A. Marshall. Towards efficient learning-based model predictive control via feedback linearization and gaussian process regression. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4306–4311, 2021.

[6] Fernando Castañeda, Jason J Choi, Bike Zhang, Claire J Tomlin, and Koushil Sreenath. Gaussian process-based min-norm stabilizing controller for control-affine systems with uncertain input effects and dynamics. In *2021 American Control Conference (ACC)*, pages 3683–3690. IEEE, 2021.

[7] Fernando Castañeda, Jason J. Choi, Bike Zhang, Claire J. Tomlin, and Koushil Sreenath. Pointwise feasibility of gaussian process-based safety-critical control under model uncertainty. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 6762–6769, 2021.

[8] Lukas Hewing, Juraj Kabzan, and Melanie N. Zeilinger. Cautious model predictive control using gaussian process regression. *IEEE Transactions on Control Systems Technology*, 28(6):2736–2743, 2020.

[9] S Mohammad Khansari-Zadeh and Aude Billard. Learning control lyapunov function to ensure stability of dynamical system-based robot reaching motions. *Robotics and Autonomous Systems*, 62(6):752–765, 2014.

[10] Torsten Koller, Felix Berkenkamp, Matteo Turchetta, and Andreas Krause. Learning-based model predictive control for safe exploration. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 6059–6066, 2018.

[11] Fei Li, Huiping Li, and Yuyao He. Adaptive stochastic model predictive control of linear systems using gaussian process regression. *IET Control Theory & Applications*, 15(5):683–693, 2021.

[12] Quan Nguyen, Ayonga Hereid, Jessy W Grizzle, Aaron D Ames, and Koushil Sreenath. 3d dynamic walking on stepping stones with control barrier functions. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 827–834. IEEE, 2016.

[13] Daniel Pickem, Paul Glotfelter, Li Wang, Mark Mote, Aaron Ames, Eric Feron, and Magnus Egerstedt. The robotarium: A remotely accessible swarm robotics research testbed. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1699–1706. IEEE, 2017.

[14] Stephen Prajna and Ali Jadbabaie. Safety verification of hybrid systems using barrier certificates. In *International Workshop on Hybrid Systems: Computation and Control*, pages 477–492. Springer, 2004.

[15] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Proceedings of the 20th International Conference on Neural Information Processing Systems*, NIPS'07, page 1177–1184, Red Hook, NY, USA, 2007. Curran Associates Inc.

[16] Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*, 2009.

[17] Dougal Sutherland and Jeff Schneider. On the error of random fourier features. *Uncertainty in Artificial Intelligence - Proceedings of the 31st Conference, UAI 2015*, 06 2015.

[18] Andrew J. Taylor, Victor D. Dorobantu, Sarah Dean, Benjamin Recht, Yisong Yue, and Aaron D. Ames. Towards robust data-driven control synthesis for nonlinear systems with actuation uncertainty. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 6469–6476, 2021.

[19] Andrew J Taylor, Victor D Dorobantu, Hoang M Le, Yisong Yue, and Aaron D Ames. Episodic learning with control lyapunov functions for uncertain robotic systems. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6878–6884. IEEE, 2019.

A Proof of Approximation Error Bounds

Define the random features approximation as $\hat{k}_s := \Phi\varphi(s) = \Phi\phi(x)y$.

Proof. we split the proof to several steps:

1. Approximating the compound kernel:

$$\begin{aligned}|k(s,s') - \hat{k}(s,s')| &= y^\top \mathrm{diag}[k_1(s,s')...k_{m+1}(s,s')]y' - y^\top \mathrm{diag}[\hat{k}_1(s,s')...\hat{k}_{m+1}(s,s')]y' \\ &\leq \epsilon(u^\top u' + 1)\end{aligned}$$

which implies:

$$\begin{aligned}\|K - \hat{K}\|_2 &\leq \epsilon\|[u_i^\top u_j + 1]_{i,j}\|_2 \leq \epsilon\sigma_{max}^2 + \epsilon n \\ \|k_s - \hat{k}_s\| &\leq \epsilon\|[u_x^\top u_i + 1]_i\| \leq \epsilon\|u_x\|.\|[u_1,...,u_N]\| + \epsilon\sqrt{n} \leq \epsilon\sigma_{max}\|u_x\| + \epsilon\sqrt{n}\end{aligned}$$

2. Approximating the mean:

$$\begin{aligned}|\mu_x(u) - \hat{\mu}_x(u)| &= \|z^\top\|\|(\hat{K} + \lambda_n I)^{-1}(k_s - \hat{k}_s) + ((K + \lambda_n I)^{-1} - (\hat{K} + \lambda_n I)^{-1})k_s\| \\ &\leq \frac{\|z\|}{\lambda_n}\|\hat{k}_s - k_s\| + \frac{\|z\|.\|\hat{K} - K\|}{\lambda_n^2}\|k_s\| \\ &\leq \frac{\sqrt{n}\sigma_n}{n\lambda}\|\hat{k}_s - k_s\| + \frac{\sqrt{n}\sigma_n\kappa}{n^2\lambda^2}\|\hat{K} - K\| \\ &\leq \frac{\sqrt{n}\sigma_n}{n\lambda}(\epsilon\sigma_{max}\|u_x\| + \epsilon\sqrt{n}) + \frac{\sqrt{n}\sigma_n\kappa}{n^2\lambda^2}(\epsilon\sigma_{max}^2 + \epsilon n) \\ &\leq \epsilon\|u_x\|\frac{\sigma_{max}\sigma_n}{\sqrt{n}\lambda} + \sqrt{n}\sigma_n\Delta_\mu\end{aligned}$$

Where we used $(K+\lambda I)^{-1} - (\hat{K}+\lambda I)^{-1} = (\hat{K}+\lambda I)^{-1}(\hat{K}-K)(K+\lambda I)^{-1}$, and that the smallest eigenvalue of $\hat{K} + \lambda I$ and $K + \lambda I$ is at least λ

3. Approximating the variance: Assume $\sigma_x(u) + \hat{\sigma}_x \leq 1$

$$\begin{aligned}|\sigma_x(u) - \hat{\sigma}_x(u)| &\leq |\sigma_x(u)^2 - \hat{\sigma}_x(u)^2| \\ &\leq \epsilon + |k_s[(K + \lambda)^{-1}k_s^\top - (\hat{K} + \lambda)^{-1}\hat{k}_s^\top] + [k_s - \hat{k}_s](\hat{K} + \lambda)^{-1}\hat{k}_s^\top| \\ &\leq \epsilon + \|k_s\|\frac{|\mu_x(u) - \hat{\mu}_x|}{\|z\|} + \frac{\epsilon\sigma_{max}\|u_x\| + \epsilon\sqrt{n}}{n\lambda}\|\hat{k}_s\| \\ &\leq \epsilon + \kappa(\epsilon\|u_x\|\frac{\sigma_{max}^2}{n\lambda} + \Delta_\mu) + \frac{\epsilon\sigma_{max}\|u_x\| + \epsilon\sqrt{n}}{n\lambda}(\kappa + \epsilon\sigma_{max}^2\|u_x\| + \epsilon\sqrt{n}) \\ &\leq \|u_x\|\frac{2\epsilon\sigma_{max}^2}{\sqrt{n}\lambda}(\frac{\kappa}{\sqrt{n}} + \epsilon) + \|u_x\|^2\frac{\epsilon^2\sigma_{max}^2}{n\lambda} + \Delta_\sigma + \kappa\Delta_\mu\end{aligned}$$

4. Bounds on total error: with a probability of $1-(\delta_1+\delta_2)$:

$$\begin{aligned}|\dot{C}_x(u) - \hat{\mu}_x(u)| &\leq |\mu_x(u) - \dot{C}_x(u)| + |\mu_x(u) - \hat{\mu}_x(u)| \\ &\leq \beta\hat{\sigma}_x(u) + \beta|\sigma_x(u) - \hat{\sigma}_x(u)| + |\mu_x(u) - \hat{\mu}_x(u)| \\ &\leq \beta\hat{\sigma}_x(u) + \|u_x\|\nu + \|u_x\|^2\iota + \Delta_s\end{aligned}$$

□