

Biased Kernel Density Estimators for Chance Constrained Optimal Control Problems

Rachel E. Keil
Alexander Miller
Mrinal Kumar
Anil V. Rao

Abstract—A method is developed for transforming chance constrained optimization problems to a form numerically solvable. The transformation is accomplished by reformulating the chance constraints as nonlinear constraints using a method that combines the previously developed Split-Bernstein approximation and kernel density estimator (KDE) methods. The Split-Bernstein approximation in a particular form is a biased kernel density estimator. The bias of this kernel leads to a nonlinear approximation that does not violate the bounds of the original chance constraint. The method of applying biased KDEs to reformulate chance constraints as nonlinear constraints transforms the chance constrained optimization problem to a deterministic optimization problems that retains key properties of the chance constrained optimization problem and can be solved numerically. This method can be applied to chance constrained optimal control problems. As a result, the Split-Bernstein and Gaussian kernels are applied to a chance constrained optimal control problem and the results are compared.

I. INTRODUCTION

Chance constrained optimization problems arise in various engineering and non-engineering applications as such problems account for uncertainty in the constraints of optimization problems. One application of interest is chance constrained optimal control. For chance constrained optimal control, possible applications include trajectory optimization under uncertain conditions including GPS blackout during atmospheric entry and fuzzy boundaries on geographic no fly zones, as well as optimal natural disaster tracking such as forest fire growth and propagation. Due to the probabilistic formulation of chance constraints, numerically solving these chance constrained optimization problems is challenging. As a result, methods are being developed that look independently at chance constraints to transform them to a more tractable form in terms of computation. The methods can

then be applied to chance constrained optimization problems and by extension chance constrained optimal control problems.

Chance constraints are defined in probability space as being dependent on random variables and bounded by some risk violation parameter. This probability is equal to the expectation (mean) of the indicator function and so the constraint can be redefined in terms of this expectation. Solving optimization problems with chance constraints defined relative to the indicator function also suffers from intractability issues. It has been shown in recent works [1]–[4] that functions can be designed with expectations that are conservative approximations of the expectation of the indicator function. In the method of [1]–[3], Markov Chain Monte Carlo (MCMC) sampling is used to obtain samples of the random variable under the assumption that determining the expectation of the functions analytically is difficult or impossible. In the method of [4], the random variables are assumed to have certain properties by which samples can be readily obtained. Thus by applying some form of sampling, the described methods approximate the chance constraints as expectations of functions designed such that the expectation does not violate the bounds of the original chance constraint. The disadvantage of this class of methods is that the resulting approximate nonlinear constraint can be too conservative as compared to the original chance constraint.

Another class of methods for transforming the chance constraints to nonlinear constraints focuses on obtaining approximations that are not too conservative, at the expense of possibly violating the bounds of the original chance constraint. These methods define approximate probability density functions (PDFs) of the random variables. In the methods of [5] and [6], non-parametric approximate PDFs are obtained from samples of the random variables. In particular, the method of [5] obtains these PDF approximations using kernel density estimators (KDEs). The expectation of the integrated kernels of the KDEs is an approximate cumulative distribution function (CDF). This approximate CDF is a nonlinear constraint approximation of the original chance constraint that may violate the bounds of the original chance constraint. Despite the disadvantage of this possible violation, the use of KDEs to obtain nonlinear constraints is an attractive option as there are several well-known kernels including the Epanechnikov, Gaussian and triangular

Rachel Keil is a Ph.D. Candidate in the Department of Mechanical and Aerospace Engineering at University of Florida, Gainesville, FL 32611-6250. E-mail: rekeil@ufl.edu

Alexander Miller is a Ph.D. Candidate in the Department of Mechanical and Aerospace Engineering at University of Florida, Gainesville, FL 32611-6250. E-mail: alexandertmiller@ufl.edu

Mrinal Kumar is an Associate Professor in the Department of Mechanical and Aerospace Engineering at The Ohio State University, Columbus, OH 43210. AIAA Senior Member. E-mail: kumar.672@osu.edu

Anil V. Rao is an Professor, Erich Farber Faculty Fellow and University Term Professor in the Department of Mechanical and Aerospace Engineering at University of Florida, Gainesville, FL 32611-6250. E-mail: anilvrao@ufl.edu.

kernels that can be applied to obtain a nonlinear constraint approximation that is not overly-conservative relative to the chance constraint.

Looking again at the method from [2] and [3], where a Split-Bernstein constraint approximation is defined that does not violate the bounds of the chance constraint, it can be shown that with the correct choice of parameters this approximation is a *biased* KDE. This bias is what leads to an approximation of the chance constraint that does not violate the bounds of the chance constraint. Additionally, the Split-Bernstein approximation in KDE form is a less conservative approximation of the chance constraint. As a result, the objective of this paper is to show the connection of biasing kernels to obtain a KDE constraint approximation that does not violate the bounds of the chance constraint. Two of these biased kernels will then be applied to a chance constrained optimal control problem to transform the problem to a deterministic optimal control problem that is solved using a Legendre-Gauss-Radau (LGR) collocation method [7]–[9]. The solutions will be compared to study the differences between the two kernels.

The paper is organized as follows. Section II briefly describes the Split-Bernstein approximation and its KDE form. Section III derives an expression for the bias of the Split-Bernstein kernel as well as providing a comparison to other kernels. Section IV describes a chance constrained optimal control problem. Section V describes the LGR collocation method. Section VI describes the chance constrained optimal control problem and solutions obtained using both kernels. Section VII provides some brief conclusions.

II. SPLIT-BERNSTEIN: KERNEL DENSITY ESTIMATOR

Consider the chance constrained optimization problem

$$P(\mathbf{F}(\mathbf{z}, \xi) \geq \mathbf{q}) \geq 1 - \varepsilon, \quad (1)$$

where $\mathbf{z}$ is a decision variable defined on the feasible set $\mathbf{Z} \subset \mathbb{R}^n$ and ξ is a random variable with a probability density function (PDF) $\mathbf{f}_\xi(\xi)$ supported on set $\Omega \subseteq \mathbb{R}^d$. To account for the possibility of the PDF being difficult or impossible to integrate analytically, either Markov Chain Monte Carlo (MCMC) sampling will be applied to obtain samples or it is assumed that samples are available. The function $\mathbf{F}(\mathbf{z}, \xi) > \mathbf{q}$ is an event in the probability space $P(\cdot)$, where $\mathbf{F}(\cdot)$ maps $\mathbb{R}^n \times \mathbb{R}^d \rightarrow \mathbb{R}^{n_g}$ and ε is the risk violation parameter. The function $(\mathbf{F}(\mathbf{z}, \xi))$ is itself a random variable whose probabilistic properties are assumed unknown. As a result, the complement of the constraint of (1) can be defined as

$$P(\psi < \mathbf{q}) \leq \varepsilon. \quad (2)$$

The event of the chance constraint is in vector form, and so the constraint is a joint chance constraint. Using Boole's inequality and the approach of [2] and [10], the original chance constraint can be redefined as the following set of conservative scalar constraints

$$P(\psi < q_m) \leq \varepsilon_m, \quad \sum_{m=1}^{n_g} \varepsilon_m \leq \varepsilon. \quad (3)$$

where $m \in 1, \dots, n_g$ is the index corresponding to the m th component of the event.

The following is a known property of probability:

$$1 - \varepsilon_m \leq P(\psi \geq q_m) = \mathbb{E}_\xi[1_{[q_m, +\infty)}(\alpha(\psi - q_m))], \quad (4)$$

where $\mathbb{E}_\xi[\cdot]$ is the expectation. Applying the expectation of the Split-Bernstein function $\Xi_\alpha(\cdot)$ from [2] to (1) gives:

$$\mathbb{E}_\xi[1_{[q_m, +\infty)}(\alpha(\psi - q_m))] \leq \mathbb{E}_\xi[\Xi_\alpha(\alpha(\psi - q_m))], \quad (5)$$

$$\Xi_\alpha(\alpha(\psi - q_m)) = \begin{cases} \exp(\alpha_+(\psi - q_m)), & \text{if } \psi - q_m \geq 0, \\ \exp(\alpha_-(\psi - q_m)), & \text{if } \psi - q_m < 0. \end{cases} \quad (6)$$

When enough MCMC samples of the random variable ξ are available, the inequality from (5) can be approximated as the following inequality (as proven in [2]):

$$\frac{1}{N} \sum_{j \in J_+} \exp(\alpha_+(q_m - \psi_j)) + \frac{1}{N} \sum_{j \in J_-} \exp(\alpha_-(q_m - \psi_j)) \leq \varepsilon_m, \quad (7)$$

where $j = 1, \dots, N$ is the number of MCMC samples, α_+ and α_- are user defined parameters, and the event is sorted per MCMC sample as $J_+ \in (q_m - \psi) > 0$ and $J_- \in (q_m - \psi) \leq 0$. The left hand side of (7) can exceed unity and so is not an approximate cumulative distribution function (CDF), leading to an overly-conservative chance constraint approximation. It is possible, however, to set α_+ equal to zero, and α_- to a sufficiently large number. As a result, substituting $\alpha_+ = 0$ and $\alpha_- = \alpha$ in (7) results in the following reformulation of (7):

$$\frac{1}{N} \sum_{j \in J_+} 1 + \frac{1}{N} \sum_{j \in J_-} \exp(\alpha(q_m - \psi_j)) \leq \varepsilon_m. \quad (8)$$

The left hand side of (8) is now an approximate CDF that is a less conservative chance constraint approximation than (7). Using (8), the approximate Split-Bernstein PDF is then defined as:

$$\hat{f}_\psi(q_m) = \frac{1}{N} \sum_{j \in J_+} 0 + \frac{1}{N} \sum_{j \in J_-} \alpha \exp(\alpha(q_m - \psi_j)). \quad (9)$$

To show that the Split-Bernstein approximate PDF in the form of (9) is a KDE, first the general form of a KDE is defined as

$$\hat{f}_\psi(q_m) = \frac{1}{N} \sum_{j=1}^N \frac{1}{h} k(\eta_j), \quad (10)$$

where $k(\cdot)$ is the kernel, h is the bandwidth, and η is defined as follows:

$$\eta_j = \frac{q_m - \psi_j}{h}. \quad (11)$$

Setting α from (9) as:

$$\alpha = \frac{1}{h}, \quad (12)$$

(9) is in the form of a KDE from (10) with the Split-Bernstein kernel k_{SB} defined as

$$k_{SB}(\eta_j) = \begin{cases} 0, & \eta_j > 0 \\ \exp(\eta_j), & \eta_j \leq 0. \end{cases} \quad (13)$$

With the approximate PDF of the Split-Bernstein method now in the form of a KDE, it is left to prove that the derived Split-Bernstein “kernel” satisfies the criteria for an actual kernel. This criteria is not universally defined [11], [12], so the authors here apply the following criteria that encompasses generally accepted requirements as

- 1) $k(\eta_j) \geq 0$,
- 2) $\int_{-\infty}^{+\infty} k(x)dx = 1$,
- 3) $0 < \int_{-\infty}^{+\infty} x^2 k(x)dx < +\infty$.

For case (1), the Split-Bernstein kernel is zero for $\eta > 0$ and always greater than zero for $\eta \leq 0$. Showing that case (2) is satisfied is straightforward:

$$\int_{-\infty}^{+\infty} k(x)dx \Rightarrow \int_{-\infty}^0 \exp(x)dx + \int_0^{+\infty} 0 \Rightarrow \exp(x) \Big|_{-\infty}^0 + 0 = 1. \quad (14)$$

Case(3) is proven as follows:

$$\int_{-\infty}^{+\infty} x^2 k(x)dx \Rightarrow \int_{-\infty}^0 x^2 \exp(x)dx + \int_0^{+\infty} 0dx = 2. \quad (15)$$

It has been shown that the Split-Bernstein approximation method yields a kernel that satisfies the definition of a kernel for $\alpha_+ = 0$. When this kernel is applied, the Split-Bernstein approximation is a KDE.

III. BIASED KERNEL DENSITY ESTIMATORS

KDEs place a kernel at each sample point and approximate the true PDF of the samples as the average weighted distance evaluated at each kernel. Integrating the kernel results in a function that converges to the indicator function with decreasing bandwidth. The bandwidth cannot be set equal to zero, however, as the variance of the kernel increases while the bandwidth decreases. Care must be taken in choosing a bandwidth that best accounts for this trade off. The expectation of the integrated kernels at each sample is an approximate CDF, and so provides an approximation of the chance constraint. The relation between the probability of the chance constraint and the approximate CDF is influenced by shifting the expectation of the kernel, which will be referred to as *biasing* the kernel.

In order to better characterize the bias, first the form of the KDE approximate CDF is derived as follows. Consider the KDE approximate probability of the chance constraint from (3) in terms of the approximate PDF from (10):

$$\hat{p}(\psi < q_m) = \int_{-\infty}^{q_m} \hat{f}_{\psi}(x)dx = \frac{1}{Nh} \sum_{j=1}^N \int_{-\infty}^{q_m} k\left(\frac{x - \psi_j}{h}\right) dx. \quad (16)$$

Using the definition of η_j from (11), (16) can be reformulated as

$$\frac{1}{Nh} \sum_{j=1}^N \int_{-\infty}^{q_m} k\left(\frac{x - \psi_j}{h}\right) dx = \frac{1}{N} \sum_{j=1}^N \int_{-\infty}^{\frac{q_m - \psi_j}{h}} k(\eta_j) d\eta_j. \quad (17)$$

Substituting the result of (17) into (16), the KDE approximate probability in terms of the chance constraint is defined

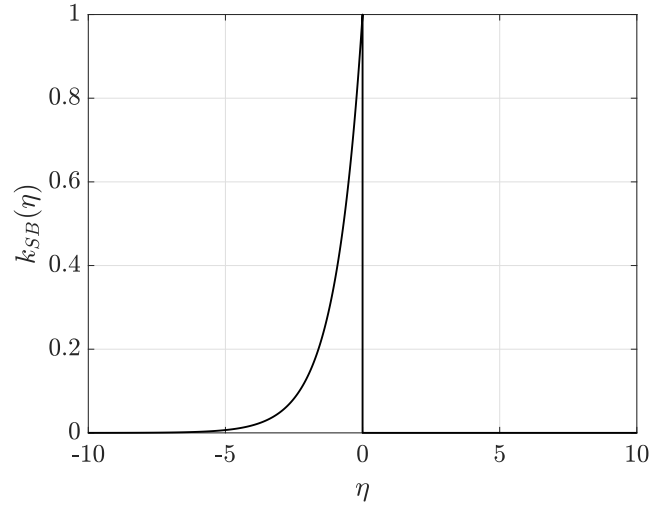


Fig. 1. Split-Bernstein kernel.

as follows:

$$P(\psi < q_m) \approx \frac{1}{N} \sum_{j=1}^N \int_{-\infty}^{\frac{q_m - \psi_j}{h}} k(\eta_j) d\eta_j, \quad (18)$$

where the right hand side is an approximate CDF that can possibly violate the bounds of the chance constraint. By contrast, applying the Split-Bernstein kernel to define an approximate CDF results in an approximate CDF that remains within the bounds of the chance constraint as seen in (8), due to the bias of Split-Bernstein kernel.

The Split-Bernstein kernel from (2) has a sail like shape from $-\infty$ to zero and is zero otherwise as can be seen in Fig. 1. This shape implies that the kernel is biased. To prove this bias, first a point w is defined at which half of the total area under of the curve of the kernel lies. Next the integration of the Split-Bernstein kernel from $-\infty$ to w can be defined as:

$$\begin{aligned} \int_{-\infty}^w \exp(x)dx &= 0.5, \\ \Rightarrow \exp(x) \Big|_{-\infty}^w &= 0.5, \\ \Rightarrow 0 + \exp(w) &= 0.5, \\ \Rightarrow w &= \ln(0.5). \end{aligned} \quad (19)$$

Then applying the definition of η from (11), the mean of the Split-Bernstein kernel μ can be determined as:

$$w = \frac{\mu - \psi_j}{h} = \ln\left(\frac{1}{2}\right) < 0, \Rightarrow \mu = \psi_j + h \ln\left(\frac{1}{2}\right), \quad (20)$$

proving that the Split-Bernstein kernel has a bias of $h \ln\left(\frac{1}{2}\right)$ that is dependent on the bandwidth.

The bias of the Split-Bernstein kernel ensures that the expectation of the integrated kernel (which is the Split-Bernstein function Ξ_{α}) results in an upper bound on the expectation of the indicator function as defined in (5). To demonstrate the relation of bias to the indicator function,

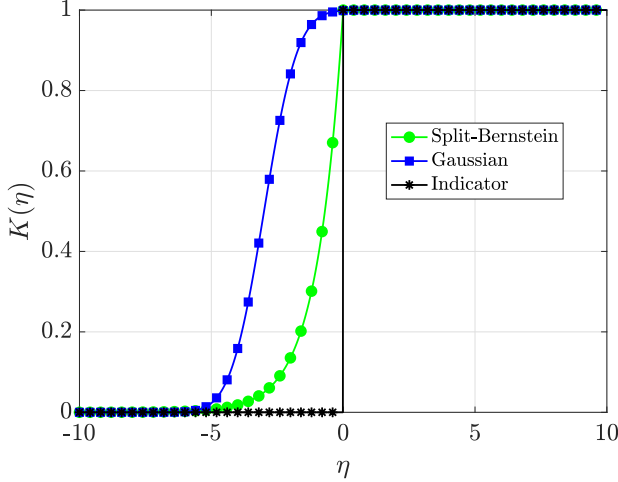


Fig. 2. Biased integrated kernel functions and the indicator function.

the function $K(\cdot)$ for the Split-Bernstein kernel and biased Gaussian kernel is plotted in Fig. 2 along with the indicator function. For Fig. 2, the function $K(\cdot)$ is the integration of the Split-Bernstein and Gaussian kernels. The Gaussian kernel in Fig. 2 has been biased by $3 * h_G$ to ensure that at least 99% of the Gaussian function $K(\cdot)$ is greater than the indicator function, where h_G is the Gaussian bandwidth. In summary, biasing a kernel such that its integrated function $K(\cdot)$ is greater than the indicator function and applying MCMC sampling leads to the following inequality:

$$\frac{1}{N} \sum_{j=1}^N \int_{-\infty}^{\frac{q_m - \psi_j}{h}} k(\eta_j) d\eta_j \leq P(\psi < q_m) \leq \epsilon_m, \quad (21)$$

where the left most expression is the biased KDE nonlinear constraint approximation. This method of biasing kernels and applying MCMC sampling to reformulate chance constraints as nonlinear constraints transforms chance constrained optimization problems to deterministic optimization problems. Extending this discussion to chance constrained optimal control problems (CCOCs), the method of biased KDEs can be applied to transform CCOCs to deterministic optimal control problems solvable using numerical methods.

IV. CHANCE CONSTRAINED OPTIMAL CONTROL PROBLEM

Without loss of generality, consider the following CCOC. Determine the state $\mathbf{y}(\tau) \in \mathbb{R}^{n_y}$ and the control $\mathbf{u}(\tau) \in \mathbb{R}^{n_u}$ on the domain $\tau \in [-1, +1]$, the initial time, t_0 , and the terminal time t_f that minimize the cost functional

$$\mathcal{J} = \mathcal{M}(\mathbf{y}(-1), t_0, \mathbf{y}(+1), t_f) + \frac{t_f - t_0}{2} \int_{-1}^{+1} \mathcal{L}(\mathbf{y}(\tau), \mathbf{u}(\tau), t(\tau, t_0, t_f)) d\tau, \quad (22a)$$

subject to the dynamic constraints

$$\frac{d\mathbf{y}}{d\tau} - \frac{t_f - t_0}{2} \mathbf{a}(\mathbf{y}(\tau), \mathbf{u}(\tau), t(\tau, t_0, t_f)) = \mathbf{0}, \quad (22b)$$

the inequality path constraints

$$\mathbf{c}_{\min} \leq \mathbf{c}(\mathbf{y}(\tau), \mathbf{u}(\tau), t(\tau, t_0, t_f)) \leq \mathbf{c}_{\max}, \quad (22c)$$

the boundary conditions

$$\mathbf{b}_{\min} \leq \mathbf{b}(\mathbf{y}(-1), t_0, \mathbf{y}(+1), t_f) \leq \mathbf{b}_{\max}, \quad (22d)$$

and the chance-constraints

$$P(\mathbf{F}(\mathbf{y}(\tau), \mathbf{u}(\tau), t(\tau, t_0, t_f); \xi) \leq \mathbf{q}) \geq 1 - \epsilon. \quad (22e)$$

It is noted that the time interval $\tau \in [-1, +1]$ can be transformed to the time interval $t \in [t_0, t_f]$ via the affine transformation

$$t \equiv t(\tau, t_0, t_f) = \frac{t_f - t_0}{2} \tau + \frac{t_f + t_0}{2}. \quad (23)$$

where $\mathbf{y}$ is the state defined on the feasible set $\mathbf{Y} \subset \mathbb{R}^n$. $\mathbf{u}$ is the control defined on the feasible set $\mathbf{U} \subset \mathbb{R}^m$. $J \in \mathbb{R}$ is the cost functional. The variables of $P(\cdot)$ are the same as those defined in Sect. II. It should be noted that dynamic, path, and event constraints could be formulated as chance constraints.

In order to apply numerical methods to the CCOC, the continuous time CCOC is discretized on the domain $\tau \in [-1, +1]$ which is partitioned into a *mesh* consisting of K mesh intervals $\mathcal{S}_k = [T_{k-1}, T_k]$, $k = 1, \dots, K$, where $-1 = T_0 < T_1 < \dots < T_K = +1$. The mesh intervals have the property that $\cup_{k=1}^K \mathcal{S}_k = [-1, +1]$. Let $\mathbf{y}^{(k)}(\tau)$ and $\mathbf{u}^{(k)}(\tau)$ be the state and control in $\mathcal{S}_k$. Using the transformation given in Eq. (23), the CCOC of Eqs. (22a)-(22e) can then be rewritten as follows. Minimize the cost functional

$$\mathcal{J} = \mathcal{M}(\mathbf{y}^{(1)}(-1), t_0, \mathbf{y}^{(K)}(+1), t_f) + \frac{t_f - t_0}{2} \sum_{k=1}^K \int_{T_{k-1}}^{T_k} \mathcal{L}(\mathbf{y}^{(k)}(\tau), \mathbf{u}^{(k)}(\tau), t) d\tau, \quad (24a)$$

subject to the dynamic constraints

$$\frac{d\mathbf{y}^{(k)}(\tau)}{d\tau} - \frac{t_f - t_0}{2} \mathbf{a}(\mathbf{y}^{(k)}(\tau), \mathbf{u}^{(k)}(\tau), t) = \mathbf{0}, \quad (k = 1, \dots, K), \quad (24b)$$

the path constraints

$$\mathbf{c}_{\min} \leq \mathbf{c}(\mathbf{y}^{(k)}(\tau), \mathbf{u}^{(k)}(\tau), t) \leq \mathbf{c}_{\max}, \quad (k = 1, \dots, K), \quad (24c)$$

the boundary conditions

$$\mathbf{b}_{\min} \leq \mathbf{b}(\mathbf{y}^{(1)}(-1), t_0, \mathbf{y}^{(K)}(+1), t_f) \leq \mathbf{b}_{\max}, \quad (24d)$$

and the chance-constraints

$$P(\mathbf{F}(\mathbf{y}^{(k)}(\tau), \mathbf{u}^{(k)}(\tau), t; \xi) \leq \mathbf{q}) \geq 1 - \epsilon. \quad (24e)$$

Because the state must be continuous at each interior mesh point, it is required that the condition $\mathbf{y}(T_k^-) = \mathbf{y}(T_k^+)$, $(k = 1, \dots, K-1)$ be satisfied at the interior mesh points $(T_1, \dots, T_{K-1})$.

V. LEGENDRE-GAUSS-RADAU COLLOCATION

The multiple-interval form of the continuous-time CCOCP in Section IV is discretized using collocation at Legendre-Gauss-Radau (LGR) points [7]–[9]. In the LGR collocation method, the state of the continuous-time CCOCP is approximated in $\mathcal{S}_k$, $k \in [1, \dots, K]$, as

$$\mathbf{y}^{(k)}(\tau) \approx \mathbf{Y}^{(k)}(\tau) = \sum_{j=1}^{N_k+1} \mathbf{Y}_j^{(k)} \ell_j^{(k)}(\tau), \quad (25)$$

$$\ell_j^{(k)}(\tau) = \prod_{\substack{l=1 \\ l \neq j}}^{N_k+1} \frac{\tau - \tau_l^{(k)}}{\tau_j^{(k)} - \tau_l^{(k)}},$$

where $\tau \in [-1, +1]$, $\ell_j^{(k)}(\tau)$, $j = 1, \dots, N_k + 1$, is a basis of Lagrange polynomials, $(\tau_1^{(k)}, \dots, \tau_{N_k}^{(k)})$ are the Legendre-Gauss-Radau (LGR) [7] collocation points in $\mathcal{S}_k = [T_{k-1}, T_k)$, and $\tau_{N_k+1}^{(k)} = T_k$ is a noncollocated point. Differentiating $\mathbf{Y}^{(k)}(\tau)$ in Eq. (25) with respect to τ gives

$$\frac{d\mathbf{Y}^{(k)}(\tau)}{d\tau} = \sum_{j=1}^{N_k+1} \mathbf{Y}_j^{(k)} \frac{d\ell_j^{(k)}(\tau)}{d\tau}. \quad (26)$$

Defining $t_i^{(k)} = t(\tau_i^{(k)}, t_0, t_f)$ using Eq. (23), the dynamics are then approximated at the N_k LGR points in mesh interval $k \in [1, \dots, K]$ as

$$\sum_{j=1}^{N_k+1} D_{ij}^{(k)} \mathbf{Y}_j^{(k)} - \frac{t_f - t_0}{2} \mathbf{a}(\mathbf{Y}_i^{(k)}, \mathbf{U}_i^{(k)}, t_i^{(k)}) = \mathbf{0}, \quad (i = 1, \dots, N_k), \quad (27)$$

where $D_{ij}^{(k)} = d\ell_j^{(k)}(\tau_i^{(k)})/d\tau$, $(i = 1, \dots, N_k)$, $(j = 1, \dots, N_k + 1)$ are the elements of the $N_k \times (N_k + 1)$ Legendre-Gauss-Radau differentiation matrix [7] in mesh interval $\mathcal{S}_k$, $k \in [1, \dots, K]$. The LGR discretization then leads to the following nonlinear programming problem (NLP). Minimize

$$\mathcal{J} \approx \mathcal{M}(\mathbf{Y}_1^{(1)}, t_0, \mathbf{Y}_{N_K+1}^{(K)}, t_f) + \sum_{k=1}^K \sum_{j=1}^{N_k} \frac{t_f - t_0}{2} w_j^{(k)} \mathcal{L}(\mathbf{Y}_j^{(k)}, \mathbf{U}_j^{(k)}, t_j^{(k)}), \quad (28)$$

subject to the collocation constraints of Eq. (27) and the constraints

$$\mathbf{c}_{\min} \leq \mathbf{c}(\mathbf{Y}_i^{(k)}, \mathbf{U}_i^{(k)}, t_i^{(k)}) \leq \mathbf{c}_{\max}, \quad (i = 1, \dots, N_k), \quad (29)$$

$$\mathbf{b}_{\min} \leq \mathbf{b}(\mathbf{Y}_1^{(1)}, t_0, \mathbf{Y}_{N_K+1}^{(K)}, t_f) \leq \mathbf{b}_{\max}, \quad (30)$$

$$P(\mathbf{F}(\mathbf{Y}_i^{(k)}, \mathbf{U}_i^{(k)}, t_i^{(k)}; \xi) \leq \mathbf{q}) \geq 1 - \varepsilon, \quad (i = 1, \dots, N_k), \quad (31)$$

$$\mathbf{Y}_{N_k+1}^{(k)} = \mathbf{Y}_1^{(k+1)}, \quad (k = 1, \dots, K-1), \quad (32)$$

where $N = \sum_{k=1}^K N_k$ is the total number of LGR points and Eq. (32) is the continuity condition on the state and is enforced at the interior mesh points $(T_1, \dots, T_{K-1})$ by treating $\mathbf{Y}_{N_k+1}^{(k)}$ and $\mathbf{Y}_1^{(k+1)}$ as the same variable in the NLP.

VI. APPLICATION OF METHOD

A chance constrained variation of a UAV optimal control problem from [13] was solved using the software package GPOPS-III [14] that employs the Gaussian quadrature orthogonal collocation described in Sect. V implemented in MATLAB® version R2018a (build 9.4.0.813654). The NLP solver used in conjunction with GPOPS-III [14] was SNOPT [15], [16]. All runs were performed using a 3.50GHz Intel® Xeon(R) CPU E5-2637 v4 16 cores Dell Precision Tower 7910 running GNOME version 3.20.2 with 188.8 GiB and base system openSUSE Leap 42.3 64-bit.

The chance constrained UAV problem is defined as the following:

$$\min t_f, \quad (33)$$

subject to the dynamic constraints

$$\begin{aligned} \dot{x}(t) &= V \cos \theta(t), \\ \dot{y}(t) &= V \sin \theta(t), \\ \dot{\theta}(t) &= \frac{\tan(\sigma_{\max})}{V} u, \\ \dot{V} &= a, \end{aligned} \quad (34)$$

the boundary conditions

$$\begin{aligned} x(0) &= -1.3852, & x(t_f) &= 0.5160, \\ y(0) &= 0.5875, & y(t_f) &= 0.5021, \\ \theta(0) &= 0, & V(t_0) &= 0.2934, \end{aligned} \quad (35)$$

the control bounds

$$-1 \leq u \leq 1, \quad (36)$$

and the chance path constraint

$$P\left(\frac{1}{2}(R^2 - ((x + \xi_1) - x_c)^2 - ((y + \xi_2) - y_c)^2) > 0\right) \leq \varepsilon_d, \quad (37)$$

where R is the radius of the keep out zone and the center of the keep out zone is delineated by x_c and y_c . The random variables ξ_1 and ξ_2 represent vectors of 50,000 Markov Chain Monte Carlo (MCMC) samples generated with normal distributions $N(0, 0.0007^2)$ and $N(0, 0.0005^2)$, respectively, using a Hamiltonian Monte Carlo (HMC) method from Neal [17], [18]. The values for all parameters are provided in Table I.

TABLE I
PARAMETERS FOR EXAMPLE PROBLEM.

R	x_c	y_c	ε_d	a	$\sigma_{\max}$
0.17	0	0.52	0.01	-0.01	0.35

The kernels used for comparison in thirty runs of the optimal control problem for each kernel were the Split-Bernstein kernel with a bandwidth of 1×10^{-4} and the biased Gaussian kernel with bandwidth of 5×10^{-4} , where the bias of the Gaussian kernel was set to three times its bandwidth. The bandwidths were chosen to be as small as possible while still ensuring convergence of the NLP solver on all runs. The same initial guess of a line approximation between the initial and terminal conditions with a control of zero was provided

for both kernels. Table II contains the average optimal cost J^* and run time T results for all thirty runs of both kernels, where the run time T does not include setup time.

TABLE II
RUN RESULTS FOR KERNELS

	Biased Gaussian	Split-Bernstein
μ_{J^*}	7.6807	7.6549
σ_{J^*}	0.0024	1.6415×10^{-5}
μ_T	97.0609 sec	70.4740 sec
σ_T	22.2886 sec	37.4294 sec
$T_{\max}$	138.2312 sec	254.3716 sec
$T_{\min}$	61.8254 sec	48.8354 sec

The Gaussian kernel produced a more conservative average optimal cost than the Split-Bernstein kernel, due to a larger bandwidth and greater overshoot of the indicator function by its integrated kernel function as compared to the integrated kernel function of the Split-Bernstein kernel. The variation in the average optimal cost for the Gaussian kernel is higher than for the Split-Bernstein kernel. The Gaussian kernel additionally had higher average run times (T), though there was much less variation in run time for the Gaussian kernel than for the Split-Bernstein kernel. The maximum run time for the Gaussian kernel was a little over half that of the Split-Bernstein kernel as a result of the smoothness and continuous derivative form of the Gaussian kernel. Thus, a less conservative average optimal cost is obtained using the Split-Bernstein kernel, with the incurred penalty of possibly more expensive computations.

VII. CONCLUSIONS

The method of biased KDEs has been developed to reformulate chance constraints as nonlinear constraints that provide an upper bound on the chance constraint. The method transforms chance constrained optimization problems to deterministic optimization problems solvable numerically. Two kernels were then applied with the method of biased KDEs to transform a chance constrained optimal control problem to two different deterministic optimal control problems. The kernels compared were the Split-Bernstein kernel for its less conservative design and the biased Gaussian kernel for its smoothness. Thirty runs using optimal control software paired with an NLP solver were performed using each kernel. The Split-Bernstein kernel produced a less conservative average optimal cost. The average run time using the Gaussian kernel was higher, although the run times were more consistent than those obtained using the Split-Bernstein kernel. Thus the method of biased KDEs can be applied to efficiently solve CCOCs using numerical methods.

ACKNOWLEDGMENTS

The authors gratefully acknowledge support for this research from the U.S. Office of Naval Research under grant N00014-15-1-2048 and from the U.S. National Science

Foundation under grants DMS-1522629, DMS-1924762, and CMMI-1563225.

REFERENCES

- [1] R. Chai, A. Savvaris, A. Tsuordos, S. Chai, Y. Xia, and S. Wang, "Solving trajectory optimization problems in the presence of probabilistic constraints," *IEEE Transactions on Cybernetics (Early Access)*, pp. pp:1–14. DOI: 10.1109/TYCB.2019.2895305, February 2019.
- [2] Z. Zhao and M. Kumar, "Split-bernstein approach to chance constrained optimal control," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 11, pp. 2782–2795. DOI: 10.2514/1.G002551, November 2017.
- [3] Z. Zhao, F. Liu, M. Kumar, and A. V. Rao, "A novel approach to chance constrained optimal control problems," *2015 American Control Conference, IEEE, Chicago*, pp. 5611–5616, July 2015.
- [4] S. Ahmed, "Convex relaxations of chance constrained optimization problems," *Optimization Letters*, vol. 8, no. 1, pp. 1–12. DOI: 10.1007/s11590-013-0624-7, January 2014.
- [5] J.-B. Caillaud, M. Cerf, A. Sassi, E. Trelat, and H. Zidani, "Solving chance constrained optimal control problems in aerospace via kernel density estimation," *Optimal Control Applications and Methods*, vol. 39, no. 5, pp. 1833–1858. DOI: 10.1002/oca.2445, September/October 2018.
- [6] B. Gopalakrishnan, A. K. Singh, K. M. Krishna, and D. Manocha, "Solving chance constrained optimization under non-parametric uncertainty through hilbert space embedding," *arXiv*, p. 1811.09311v1 [cs.LG], November 2018.
- [7] D. Garg, M. A. Patterson, C. L. Darby, C. Francolin, G. T. Huntington, W. W. Hager, and A. V. Rao, "Direct Trajectory Optimization and Costate Estimation of Finite-Horizon and Infinite-Horizon Optimal Control Problems via a Radau Pseudospectral Method," *Computational Optimization and Applications*, vol. 49, no. 2, pp. 335–358. DOI: 10.1007/s10589-00-09291-0, June 2011.
- [8] D. Garg, M. A. Patterson, W. W. Hager, A. V. Rao, D. A. Benson, and G. T. Huntington, "A Unified Framework for the Numerical Solution of Optimal Control Problems Using Pseudospectral Methods," *Automatica*, vol. 46, no. 11, pp. 1843–1851. DOI: 10.1016/j.automatica.2010.06.048, November 2010.
- [9] M. A. Patterson, W. W. Hager, and A. V. Rao, "A ph mesh refinement method for optimal control," *Optimal Control Applications and Methods*, vol. 36, no. 4, pp. 398–421. <https://doi.org/10.1002/oca.2114>, July–August 2015.
- [10] L. Blackmore, M. Ono, and B. C. Williams, "Chance-constrained optimal path planning with obstacles," *IEEE Transactions on Robotics*, vol. 27, no. 6, pp. 1080–1094, December 2011.
- [11] J.-N. Hwang, S.-R. Lay, and A. Lippman, "Nonparametric multivariate density estimation: A comparative study," *IEEE Transactions on Signal Processing*, vol. 42, no. 10, pp. 2795 – 2810, October 1994.
- [12] Anurag and N. Paragios, "Motion-based background subtraction using adaptive kernel density estimation," *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE*, p. DOI: 10.1109/CVPR.2004.1315179, June/July 2004.
- [13] T. Jorris, "Common aero vehicle autonomous reentry trajectory optimization satisfying waypoint and no-fly zone constraints," Ph.D. dissertation, Air Force Institute of Technology, 2004.
- [14] M. A. Patterson and A. V. Rao, "GPOPS – II, A MATLAB Software for Solving Multiple-Phase Optimal Control Problems Using hp -Adaptive Gaussian Quadrature Collocation Methods and Sparse Nonlinear Programming," *ACM Transactions on Mathematical Software*, vol. 41, no. 1, pp. 1:1–1:37. <https://doi.org/10.1145/2558904>, October 2014.
- [15] P. E. Gill, W. Murray, and M. A. Saunders, "Snopt: An sqp algorithm for large-scale constrained optimization," *SIAM Review*, vol. 47, no. 1, pp. 99–131, January 2002.
- [16] —, *User's Guide for SNOPT Version 7: Software for Large Scale Nonlinear Programming*, February 2006.
- [17] R. M. Neal, *Handbook of Markov Chain Monte Carlo*. Boca Raton, Florida: CRC Press, 2011, ch. 5, pp. 113–162.
- [18] —, *Probabilistic Inference Using Markov Chain Monte Carlo Methods*, Department of Computer Science, University of Toronto, September 1993, technical Report, CRG-TR-93-1.