State Constrained Controller Design for Uncertain Linear Systems using Polynomial Chaos

Souransu Nandi[1], Victor Migeon[2], Tarunraj Singh[3] and Puneet Singla[4]

Abstract— **The focus of this paper is on the design of state constrained controllers which are robust to time invariant uncertain variables. Polynomial Chaos spectral expansion is used to parameterize the uncertain variables, which permits evaluation of the evolution of the uncertain states. The coefficients of the truncated polynomial chaos expansion are determined using the Galerkin projection resulting in a set of deterministic equations. A mapping into Bernstein polynomial space permits determination of bounds on the evolving states. Linear programming is used on the deterministic set of equation with constraints as the predetermined bounds to determine controllers which are robust to the epistemic uncertainties. Numerical examples are used to illustrate the benefit of the proposed technique for the design of rest-to-rest controllers subject to deformation constraints; which are robust to uncertainties in the stiffness coefficient for the benchmark spring-mass system.**

I. Introduction

The problem of precision motion control of structures in the presence of model parameter uncertainties has been addressed by numerous researchers [1], [2], [3], [4]. The technique can be broadly classified into two categories: (1) sensitivity based robust control design which considers the nominal model in the design and (2) an uncertain domain based design which often is a worst case design.

Most of the controllers have considered minimizing the residual energy, i.e., the undesired energy in the system at the end of the maneuver. This is important in rest-to-rest class of problems. This paper considers applications where constraints have to be imposed on states over the duration of the maneuver, in the presence of model parameter uncertainties. Building on the rich literature of the use of polynomial chaos in control [5], this paper presents an approach which rewrites the polynomials associated with the specified distributions of the uncertain variables using Bernstein polynomials. The mapping permits determination of time evolutions of the bounds on the uncertain states. An optimization problem can now be posed to ensure that the state constraints are not violated for all possible realizations of the uncertain variables. The proposed approach is attractive since it does not require Monte Carlo simulations to estimate the bounds on the states. A linear programming problem can be posed to determine a controller which is robust to the uncertain variables.

The structure of the paper is as follows: first, a review of polynomial chaos is provided to characterize uncertainties and propagate them through the dynamical model. This is followed in Section III by the polynomial chaos mapping using Bernstein polynomial basis function which illustrate how the bounds on the states are determined. Section IV elaborates the improvement on these bounds by splitting the Bernstein coefficients. The final section illustrates the robust controller design using linear programming and the final results.

II. Polynomial Chaos (PC) Expansion

Polynomial Chaos, introduced by Norbert Wiener in [6], first approximated a stochastic state following a Gaussian process by an infinite series expansion of orthogonal Hermite polynomials. Xiu and Karniadakis in [7] proved that the convergence of this series was exponential only for Gaussian processes. Furthermore, they showed that stochastic processes with other distributions could also be expressed as series expansions with exponential convergence if appropriate orthogonal polynomials were chosen. This development was termed as the generalised Polynomial Chaos (gPC) theory.

A. Methodology

Let a stochastic dynamical system be expressed in the form

$$\dot{\boldsymbol{x}}(t,\boldsymbol{\xi}) = f(\boldsymbol{x}(t),\boldsymbol{p}(\boldsymbol{\xi}),\boldsymbol{u}(t)) \text{ and } \boldsymbol{x}(t_0,\boldsymbol{\xi}) = \boldsymbol{x_0} \quad (1)$$

where, $\boldsymbol{x} \in \mathbb{R}^n$ is the state vector, $\boldsymbol{\xi} \in \mathbb{R}^m$, the vector of random variables, and $\boldsymbol{u}(t)$ the control input.

From gPC, the states can be expressed as

$$\boldsymbol{x}(t,\boldsymbol{\xi}) = \sum_{i=0}^{\infty} \boldsymbol{x}_{\bar{\boldsymbol{i}}}(t)\Psi_i(\boldsymbol{\xi}) \quad (2)$$

where, $\Psi_i(\boldsymbol{\xi})$ is a complete set of multivariate orthogonal (w.r.t the pdf of $\boldsymbol{\xi}$) polynomials and $\boldsymbol{x}_{\bar{\boldsymbol{i}}} \in \mathbb{R}^n$ is the time varying coefficient vector of $\Psi_i(\boldsymbol{\xi})$. The selection of the set of orthogonal polynomials for popular distributions is given by the Wiener-Askey scheme [7].

The expansion is typically truncated to a finite number of terms (depending on the desired accuracy) as an approximation [7]. Hence, equation (2) is rewritten as

$$\boldsymbol{x}(t,\boldsymbol{\xi}) \approx \sum_{i=0}^{N} \boldsymbol{x}_{\bar{\boldsymbol{i}}}(t)\Psi_i(\boldsymbol{\xi}) \quad (3)$$

The objective is to evaluate the unknown vectors $\boldsymbol{x}_{\bar{\boldsymbol{i}}}(t)$ over time. Equation (3) is substituted in equation (1) to get

$$\sum_{i=0}^{N} \dot{\boldsymbol{x}}_{\bar{\boldsymbol{i}}}(t)\Psi_i(\boldsymbol{\xi}) = f\Big(\sum_{i=0}^{N} \boldsymbol{x}_{\bar{\boldsymbol{i}}}(t)\Psi_i(\boldsymbol{\xi}),\boldsymbol{p}(\boldsymbol{\xi}),\boldsymbol{u}(t)\Big) \quad (4)$$

The essence of PC expansion is to form a set of deterministic differential equations from the stochastic equation (4); whose solution allows us to approximate the states over time. This can be done by performing the Galerkin Projection on it over each of the orthogonal basis functions (i.e. Ψ_k, where $k = 0, 1, \ldots, N$). The solution to these equations yield the desired elements of $\boldsymbol{x_{\bar{i}}}(t)$.

B. Numerical Example

An implementation of PC on a simple 2 mass spring damper system (Fig. 1) has been shown and compared with Monte Carlo (MC) simulations. It is assumed that the spring constant (k) is uncertain with a uniform distribution: $k \in \mathcal{U}(0.7, 1.3)$. The orthogonal basis functions $\Psi_i(\boldsymbol{\xi})$ for a uniform distribution are Legendre polynomials. Since the domain of orthogonality is [-1,1], k is rewritten in terms of another random variable ξ ($\in \mathcal{U}(-1, 1)$) as: $k = 1 + 0.3\xi$. The non-uncertain parameters were chosen as: $c = 1$; $m_1 = 5$; $m_2 = 5$; $\boldsymbol{y}(0) = (1, 0, 0, 0)^T$ and $u(t) = 0$. Using the Galerkin approach and truncating the PC expansion to $N = 5$, ($(N+1)n = 24$) deterministic equations are formed which are used to evaluate the PC coefficients. Fig. 2(a) shows the mean of the position of m_1 evaluated using PC. It is seen to overlap with the mean derived from 10000 Monte Carlo (MC) simulations proving convergence in the given interval of time.

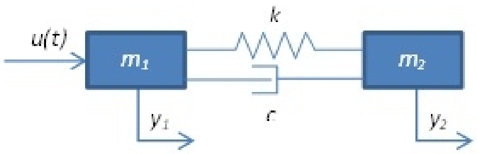


Fig. 1. 2 mass spring damper system

III. Determination of Bernstein Bounds

In numerous fields of engineering, it is often desired to determine the bounds on the range of a particular state or function. If the particular function of interest is, or can be well approximated by a multivariate polynomial, Bernstein polynomials can be exploited to determine these bounds [8]. In fact, algorithms have also been proposed to determine the exact range of multivariate polynomials [9]. The work presented here, however, makes use of a basis transformation from the existing one (of PC expansion) to the Bernstein one to use the bounding properties. As bounds only exist for compact support, the procedure only works when variables have a finite distribution.

A. Methodology

A polynomial function expressed in the othogonal bases (like Legendre, Jacobi, etc.) can also be expressed in power (i.e. $P_i\,:\,1, \xi, \xi^2, \xi^3, \ldots$) or Bernstein bases. Since, such polynomials with finite orthogonality domain are used as bases in PC expansion of stochastic states with compact support, the first objective is a basis transformation from the orthogonal bases to the Bernstein one.

A stochastic state expressed as a PC Expansion (equation (3)) now needs to be expressed as

$$x_j(t, \boldsymbol{\xi}) = \sum_{i=0}^{N} b_{ji}(t)\mathcal{B}_i(\boldsymbol{\xi}) \tag{5}$$

where x_j is the j^{th} state of the model, $\mathcal{B}_i$ are the multivariate Bernstein polynomials with domain $\boldsymbol{\xi} = [0, 1]^m$ and $b_{ji}(t)$ are their coefficients.

The linear transformation of the coefficients involves 3 stages. The first stage requires the bases Ψ_i (which has domain of orthogonality $[a, b]$) to be transformed to orthogonal bases Ψ_i' with orthogonality domain $[0, 1]$ (as the property of Bernstein polynomials to be utilized is satisfied only within that domain). The second stage requires the transformation from Ψ_i' to the power bases P_i followed by the final transformation from P_i to $\mathcal{B}_i$.

Without loss of generality, the subscript j is dropped (to indicate any state member) for convenience from equation (5) and the equation is rewritten as

$$x(t, \boldsymbol{\xi}) = \sum_{\xi_1, \xi_2, \ldots \xi_m = 0}^{d, d, \ldots d} b_{\xi_1, \xi_2, \ldots \xi_m}(t) B^d_{\xi_1, \xi_2, \ldots \xi_m}(\boldsymbol{\xi}) \tag{6}$$

where $B^d_{\xi_1, \xi_2, \ldots \xi_m}(\boldsymbol{\xi}) = \prod_{i=0}^{m} B^d_{\xi_i}(\xi_i)$ are the m-dimensional Bernstein polynomials, $b_{\xi_1, \xi_2, \ldots \xi_m}$ are the Bernstein coefficients, and d is the degree of the univariate polynomials. Now, the range enclosing property of Bernstein polynomials over the box $\boldsymbol{\xi} = [0, 1]^m$

$$x(t, \boldsymbol{\xi}) \subseteq \left[\min_{\xi_1, \xi_2, \ldots \xi_m = 0}^{d, d, \ldots d} b_{\xi_1, \xi_2, \ldots \xi_m}, \max_{\xi_1, \xi_2, \ldots \xi_m = 0}^{d, d, \ldots d} b_{\xi_1, \xi_2, \ldots \xi_m} \right] \tag{7}$$

can be used to immediately obtain bounds on the states. Equation (7) can also be expressed as a convex hull property by defining control points.

$$\left\{ \begin{pmatrix} \boldsymbol{\xi} \\ x(t, \boldsymbol{\xi}) \end{pmatrix} : \boldsymbol{\xi} \in [0, 1]^m \right\} \subseteq$$

$$conv\left\{ \begin{pmatrix} (b-a) * (\frac{\xi_1}{d}, \frac{\xi_2}{d}, \ldots \frac{\xi_m}{d}) \\ b_{\xi_1, \xi_2, \ldots \xi_m}(t) \end{pmatrix} : \right.$$

$$\left. \xi_1, \xi_2, \ldots \xi_m = 0, 0, \ldots 0 \;\; to \;\; d, d, \ldots d \right\} \tag{8}$$

where $conv\{M\}$ denotes the convex hull of set M.

B. Illustrative Example (contd.)

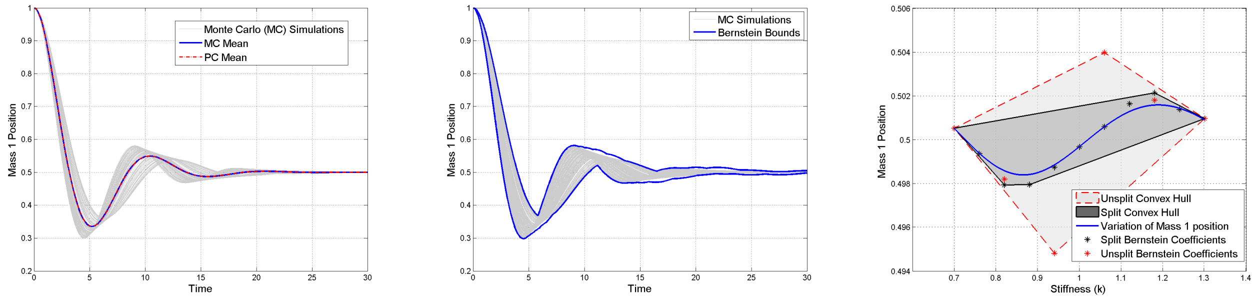
The numerical example from section II-B is continued to elaborate the method. Bounds determination is, however, shown only for one of the states (Mass 1 position). Bounds on the other states can be determined in the same way. In Legendre bases, we have

$$y_1(t) = y_{10}(t)\Psi_0 + y_{11}(t)\Psi_1 + \ldots + y_{15}(t)\Psi_5. \tag{9}$$

After a basis transformation described in [10], the state is expressed as

$$y_1(t) = b_{10}(t)B^5_0(\xi) + \ldots + b_{15}(t)B^5_5(\xi) \tag{10}$$

Using the property mentioned in equation (7), at every instant in time, the maximum and minimum values of the state are obtained directly by observing the coefficients $[b_{10}, \ldots, b_{15}]^T$. The maximum value of the coefficients provide the upper



(a) (b) (c)

Fig. 2. (a) MC Simulations (10000), MC Mean and PC Mean for Mass 1 Position - One uncertain variable, (b) Definite Bounds obtained from Bernstein coefficients, (c) Unsplit and Split coefficients at time = 29

bound while the minimum values provide the lower bound. Fig. 2(b) shows that the state trajectories from MC simulations of the model all lie within the envelope determined by the bounds.

A slice taken from Fig. 2(b) at time t = 29 is shown in Fig. 2(c). The red stars denote the control points derived from equation (8).

IV. Splitting Bernstein Coefficients

Fig. 2(c) makes a visual point towards stating that the bounds are conservative and have room for improvement. Garloff in [11] shows that these bounds can be improved if the domain of the Bernstein polynomials is subdivided. This subdivision can be done using De Casteljau's algorithm [12].

A. Methodology

Splitting of Bezier curves was first presented by French engineer Paul de Casteljau [13]. His algorithm is used to split the convex hull into segments. In our case, that reduces the conservative nature of the bounds.

The lower and upper bounds for the curve in Fig. 2(c) after splitting are given by: $\min(b_{1i}) = 0.4979$ and $\max(b_{1i}) = 0.5021$ respectively. The lower bound increases and the upper bound decreases, thus improving the definitive bound. In this particular case, the range on the bounds see a substantial improvement of 54.13 % when the coefficients are split. Throughout the procedure, the splitting has been done once and for all cases at $\xi = 0.5$. Splitting the coefficients more than once is possible and does improve bounds. However, splitting the coefficients more than once greatly increases computational requirements considering that the computations have to be performed at all times. Splitting can also be done at values other than $\xi = 0.5$ and would yield different results. It would be ideal if the splitting could be done at an optimal point, however determining the optimal point at every stage is once again expensive and a computational compromise is made.

V. Controller Design

Determination of definite bounds on the states of a stochastic process leads to an obvious application in controller design. The information from the Bernstein coefficients can now be used to implement a robust controller for the states, in spite of the uncertainties in the model.

The controller is designed in a discrete setting of the system. A Linear Programming (LP) formulation is used (as the constraints can be reduced to linear equalities and inequalities) to solve for the desired control input. The LP problem formulation for linear dynamical systems as elaborated by Singh in [14] is first discussed before elaborating the incorporation of the Bernstein bounds.

A. LP Problem for Linear Dynamical Systems

Consider the linear discrete system

$$\boldsymbol{x}(k+1) = A\boldsymbol{x}(k) + B\boldsymbol{u}(k) \tag{11}$$

The recursive relation of (11) can be expressed as a linear expression of the initial conditions and the control input $\boldsymbol{u}$ as

$$\boldsymbol{x}(k+1) = A^k\boldsymbol{x}(1) + \sum_{i=1}^{k} A^{k-i}B\boldsymbol{u}(i) \tag{12}$$

where $\boldsymbol{x}(1)$ is the initial condition of the states.

The objective of the problem is to determine a control input $\boldsymbol{u}$ which drives the dynamical system from an initial state $\boldsymbol{x}(1)$ to a final state $\boldsymbol{x}(N_t+1)$, where $k = N_t$ is the final iterative step of equation (11) in a given finite interval of time. Thus, the known quantities of the problem are initial time (T_0), final time (T_f), initial conditions ($\boldsymbol{x}(1)$) and final conditions ($\boldsymbol{x}(N_t+1)$).

The LP problem is posed as a feasibility problem in this case with linear constraints. From equation (12), the final condition constraint is given by

$$\begin{bmatrix} A^{N_t-1}B & A^{N_t-2}B & \dots & AB & B \end{bmatrix} \begin{bmatrix} u(1) \\ u(2) \\ \vdots \\ u(N_t) \end{bmatrix} = \boldsymbol{x}(N_t+1) - A^{N_t}\boldsymbol{x}(1) \tag{13}$$

The constraints on the control can be easily implemented as

$$u_{lb} \leqslant u(k) \leqslant u_{ub} \quad \forall\, k \in [1:N_t] \tag{14}$$

where u_{lb} and u_{ub} are lower and upper bounds on the control input respectively. The constraints on the states can be implemented using equation (12) and an output matrix. The solution to the problem stated above yields the desired control input over all time instants ($k = [1, ..., N_t]$).

B. LP Problem for Stochastic Processes

The LP formulation for a stochastic process has a similar framework as that of the deterministic problem. However, the primary difference lies in the fact that even though the desired final states are known, the final state values for a stochastic process remains uncertain.

To formulate the problem considering uncertainty, the residual energy of the final states is chosen to be the cost to be minimized. The residual potential energy is a function of the position states and the residual kinetic energy is only a function of the velocity states. Hence, constraining the residual energy constrains the final state residues.

However, the residual energy is a quadratic term and does not fit the requirements of a LP problem. Hence, the residual energy constraint is equivalently rewritten to conform to LP rules. This framework to realize a control input is also known as the $Minimax\ Control$ as the scheme seeks to minimize the maximum magnitude of residual energy of the final states over the domain of uncertainty [4].

The final problem can be posed as

$$\begin{aligned}
&\text{minimize}_{\boldsymbol{u},f} && f \\
&\text{subject to} && -f \leqslant y_1^i(N_t+1) \leqslant f \\
& && -f \leqslant y_2^i(N_t+1) \leqslant f \\
& && \vdots \\
& && -f \leqslant y_r^i(N_t+1) \leqslant f \qquad \forall i = 0, 1, \dots p \\
& && -f \leqslant \dot{y}_1^i(N_t+1) \leqslant f \\
& && \vdots \\
& && -f \leqslant \dot{y}_r^i(N_t+1) \leqslant f \\
& && u_{lb}(k) \leqslant u(k) \leqslant u_{ub}(k) \quad \forall k = 1, 2, \dots N_t \\
& && \text{State Constraints} \qquad \forall k = 1, 2, \dots N_t
\end{aligned}$$

where p is the number of uncertain models selected over the space of uncertainty on which the residue is to be minimized and u_{lb} & u_{ub} are the iterative bounds on the control. y_i represents the states transformed using the mass and stiffness matrices of the system and the state constraints are the desired state restrictions during the control operation. A similar approach can be used to formulate a LP problem with a L_1 cost function [14].

C. LP problem with Bernstein coefficients

In the previous section it was stated that the number of models used to form the constraints was p. Accurate representations of multivariate stochastic systems with sampling require a large number of such samples. The magnitude of this number exponentially increases with the dimensionality of the uncertainty [15]. Hence, the robustness of the controller is limited by the number of samples of the model chosen.

However, when the stochastic process has a Bernstein formulation, the control points of the Bernstein expansion provide absolute deterministic bounds on the truncated PC expansion of the states. Hence, a smart sampling method which requires the model samples to be the control points (which make the convex hull) always makes the control design valid.

The case for using the unsplit control points is first presented before the split control points case. Once again, the same example is used to show the theory discussed.

D. Illustrative Example

In this section, the method to formulate the LP problem and solve it has been shown. Considering the model from Fig. 1, the dynamics are given by the chosen matrices

$$M = \begin{bmatrix} 4 & 0 \\ 0 & 2 \end{bmatrix},\ C = \begin{bmatrix} 0.3 & -0.3 \\ -0.3 & 0.3 \end{bmatrix},\ K = \begin{bmatrix} k & -k \\ -k & k \end{bmatrix} \tag{15}$$

For this example, it is once again assumed that the uncertainty in the model is one dimensional and the uncertainty lies in the spring constant value

$$k \in \mathcal{U}(0.8, 1.2) \tag{16}$$

Hence, To calculate the L_1 or L_∞ form of the residual energy, the square root of K matrix is desired. Thus, a pseudo spring (with spring constant k_p) is attached to the first mass to make the stiffness matrix positive definite. Therefore,

$$K_h = \begin{bmatrix} k + k_p & -k \\ -k & k \end{bmatrix} \tag{17}$$

For all simulations, K_h is used as the effective stiffness matrix with the value of k_p being 0.05.

As seen previously, in order to formulate the LP problem, a discrete setting is necessary. The whole procedure for PC expansion is followed to obtain the Legendre states in discrete format.

Using the same notation as in section II, the state space model equation can be written as

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \\ \dot{z}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{-a}{m_1} & \frac{a}{m_1} & \frac{-c}{m_1} & \frac{c}{m_1} \\ \frac{a}{m_2} & \frac{-a}{m_2} & \frac{c}{m_2} & \frac{-c}{m_2} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{m_1} \\ 0 \end{bmatrix} u \tag{18}$$

where $a = (1 + 0.2\xi)$. Following a PC expansion and a Galerkin projection, a set of deterministic equations are obtained.

$$\dot{z}_{10}\langle\Psi_0, \Psi_0\rangle = f_{10}(z_{10}, \dots, z_{45}, u) \tag{19}$$

$$\vdots$$

$$\dot{z}_{45}\langle\Psi_5, \Psi_5\rangle = f_{45}(z_{10}, \dots, z_{45}, u) \tag{20}$$

Equations (19) through (20) are discretized to yield

$$\boldsymbol{Z}(k+1) = A\boldsymbol{Z}(k) + Bu(k) = A^k\boldsymbol{Z}(1) + \sum_{i=1}^{k} A^{k-i}Bu(i) \tag{21}$$

where $\boldsymbol{Z} = [z_{10}, \dots, z_{45}]^T$ are the Legendre coefficients and k is the iteration number. The Legendre states at final time is given by

$$\boldsymbol{Z}(T_f) = \boldsymbol{Z}(N_t+1) = A^{N_t}\boldsymbol{Z}(1) + \sum_{i=1}^{N_t} A^{N_t-i}Bu(i) \quad (22)$$

The transformation from Legendre states to Bernstein states is linear and the transformation matrix is said to be represented by M_N. Therefore, the terminal states in Bernstein form are

$$\begin{aligned}\boldsymbol{Z_B}(T_f) &= M_N\boldsymbol{Z}(N_t+1) = \\ &M_N A^{N_t}\boldsymbol{Z}(1) + \sum_{i=1}^{N_t} M_N A^{N_t-i}Bu(i)\end{aligned} \quad (23)$$

where $\boldsymbol{Z_B}$ is a vector of dimension $(n(N+1) = 24)$ containing all the Bernstein coefficients of the Bernstein basis functions obtained under the influence of the input $u(k)$. The objective of the example is to find a control input which can drive the system from its initial states ($[0,0,0,0]^T$) to its final states ($[1,1,0,0]^T$) while constraining the input to remain within the domain $[-1 \;\; 1]$ and constraining the relative displacement of the masses to remain within 0.1. From the desired final states, the desired Bernstein coefficients at the final time can be easily determined as

$$\boldsymbol{Z_{Bd}}(T_f) = \begin{bmatrix}1,\dots & (12) & \dots,1,0,\dots & (12)\dots, & 0\end{bmatrix}^T \quad (24)$$

Thus, the residual states are

$$\boldsymbol{X}(T_f) = \begin{bmatrix}\boldsymbol{x}_1 & \boldsymbol{x}_2 & \dot{\boldsymbol{x}}_1 & \dot{\boldsymbol{x}}_2\end{bmatrix}^T |_{T_f} = \boldsymbol{Z_B}(T_f) - \boldsymbol{Z_{Bd}}(Tf) \quad (25)$$

where $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are vectors of dimension 6 having the Bernstein coefficients of the final residual positions of mass 1 and 2 respectively. Similarly, $\dot{\boldsymbol{x}}_1$ and $\dot{\boldsymbol{x}}_2$ hold the coefficients for their residual velocities.

Now depending on the way in which the residual energy constraint is implemented (L_1 or L_∞), separate formulations of the LP problem can be made.

Defining new states

$$y_1^{(i)} = \sqrt{K_h}^{(i)}_{(1,1)}\boldsymbol{x}_1^{(i)} + \sqrt{K_h}^{(i)}_{(1,2)}\boldsymbol{x}_2^{(i)} \quad (26)$$

$$y_2^{(i)} = \sqrt{K_h}^{(i)}_{(2,1)}\boldsymbol{x}_1^{(i)} + \sqrt{K_h}^{(i)}_{(2,2)}\boldsymbol{x}_2^{(i)} \quad (27)$$

$$y_3^{(i)} = \sqrt{M}_{(1,1)}\dot{\boldsymbol{x}}_{\boldsymbol{1}}{}^{(i)} + \sqrt{M}_{(1,2)}\dot{\boldsymbol{x}}_{\boldsymbol{2}}{}^{(i)} \quad (28)$$

$$y_4^{(i)} = \sqrt{M}_{(2,1)}\dot{\boldsymbol{x}}_{\boldsymbol{1}}{}^{(i)} + \sqrt{M}_{(2,2)}\dot{\boldsymbol{x}}_{\boldsymbol{2}}{}^{(i)} \quad (29)$$

where $i = 0, 1, \dots, p$ are the sample models and $\boldsymbol{x}_a^{(i)}$ is the i^{th} member of $\boldsymbol{x}_a$. In this framework, as stated earlier, the samples are selected to be the control points of the convex hull enveloping the uncertain region. Therefore, $p = N = 5$. $\sqrt{K_h}^{i}_{(r,c)}$ refers to the r^{th} row and c^{th} column of the matrix $\sqrt{K_h}$ defined for $k^{(i)}$ where $k^{(i)}$ is defined as

$$k^{(i)} = 0.8 + (1.2 - 0.8)\frac{i}{N} \quad (30)$$

For a L_∞ formulation, the nature of constraints is provided in the optimization problem in section (V-B). Since there are 4 states in the model, the constraint equations are

$$-f \leqslant y_j^{(i)} \leqslant f \quad for \;\; j = 1, 2, 3, 4 \quad (31)$$

These constraints can be exercised with the help of an output matrix $C_{l\infty}$ The constraints can therefore be written in a matrix format as

$$\underbrace{\begin{bmatrix}\pm C_{l\infty}M_N A^{N_t-1}B \dots \pm C_{l\infty}M_N B & -1_{24,1}\end{bmatrix}}_{H_{1\pm}} \underbrace{\begin{bmatrix}u(1) \\ \vdots \\ u(N_t) \\ f\end{bmatrix}}_{\tilde{\boldsymbol{z}}} \leqslant \pm\boldsymbol{h_1} \quad (32)$$

where $\boldsymbol{h_1}$ is the vector of constant terms from equation (31). Hence, the final problem can be stated as

$$\begin{aligned} &\text{minimize} && f = \tilde{\boldsymbol{c}}^T\tilde{\boldsymbol{z}} \\ &\text{subject to} && H\tilde{\boldsymbol{z}} \leqslant \boldsymbol{h} \\ &&& \tilde{\boldsymbol{z}}_{min} \leqslant \tilde{\boldsymbol{z}} \leqslant \tilde{\boldsymbol{z}}_{max} \\ &&& |\text{Relative Mass Position}| \leqslant 0.1 \end{aligned} \quad (33)$$

where

$$\tilde{\boldsymbol{c}}^T = [0 \dots 0 \; 1], \quad \tilde{\boldsymbol{z}} = [u(1) \dots u(N_t) \; f]^T \quad (34)$$

$$H = \begin{bmatrix}H_{1+} \\ H_{1-}\end{bmatrix}, \quad \boldsymbol{h} = \begin{bmatrix}\boldsymbol{h_1} \\ -\boldsymbol{h_1}\end{bmatrix} \quad (35)$$

$$\begin{bmatrix}u_{min}(1) \\ \vdots \\ u_{min}(N_t) \\ 0\end{bmatrix} \leqslant \tilde{\boldsymbol{z}} \leqslant \begin{bmatrix}u_{max}(1) \\ \vdots \\ u_{max}(N_t) \\ \infty\end{bmatrix} \quad (36)$$

On solving the LP problems the desired control inputs are obtained. Fig. 3(a) shows the variation of residual energy with stiffness for both norms.

E. LP Problem with split Bernstein coefficients

Using the previously illustrated example, the methodology to apply the split Bernstein coefficients to the LP problem is shown in this section.

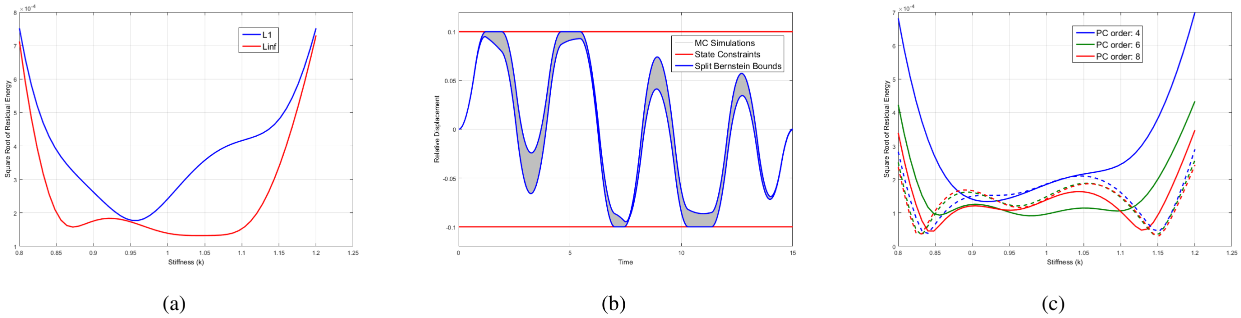
Initially, the order of PC expansion chosen was ($N = 5$). Hence, each state was represented by ($N + 1 = 6$) coefficients. Since there are 4 states in our chosen problem, the total number of coefficients were ($(N+1) \times 4 = 24$). As the Bernstein coefficients are split, the Bernstein coefficients representing each state increases to ($2N + 1 = 11$) and the total number of coefficients become ($11 \times 4 = 44$).

Following the procedure described previously, the desired set of Bernstein coefficients at final time (T_f) is now given by

$$\boldsymbol{Z_{Bd}}(T_f) = \begin{bmatrix}1,\dots & (22) & \dots,1,0,\dots & (22)\dots, & 0\end{bmatrix}^T \quad (37)$$

where $\boldsymbol{Z_{Bd}}(T_f) \in \mathbb{R}^{44}$.

New states are once again defined by equations (26) through (29). As the number of control points have increased on



(a) (b) (c)

Fig. 3. (a) Residual Energy Sensitivity, (b) Relative Displacement with Bernstein envelope, (c) Residual Energy sensitivity comparison

splitting the Bernstein coefficients, the number of samples increase to $p = 2N + 1 = 11$. All the Bezier curves (i.e. the residual state curves with respect to stiffness) were split at their midpoints (i.e. at $k = 1$ or $\xi = 0.5$). Hence, $k^{(i)}$ for this scheme of implementation is defined as

$$k^{(i)} = 0.8 + (1.2 - 0.8)\frac{i}{2N+1} \quad \forall i = 0, 1, \dots, p = 11 \tag{38}$$

The rest of the formulation remains identical. Considering C_l as the output matrix (depending on the L_1 or the L_∞ formulation), the inequality constraints are given by

$$\underbrace{\left[\pm C_l M_B M_N A^{N_t - 1} B \dots \pm C_l M_B M_N B, \boldsymbol{-1}\right]}_{H_{1\pm}} \underbrace{\begin{bmatrix} u(1) \\ \vdots \\ u(N_t) \\ f \end{bmatrix}}_{\tilde{\boldsymbol{z}}} \leqslant \pm \boldsymbol{h_1} \tag{39}$$

where $\boldsymbol{h_1}$ is the vector of constant terms derived from the constraint equations. The final problem can be summarized by equations (33) through (36).

On solving the split LP problem, the control is used to make MC simulations of the relative displacement of the masses (Fig. 3(b)) and is shown to lie within the constraints ($\leqslant 0.1$) as well as within the envelope of Bernstein Bounds. A comparison between the unsplit and split methods based on residual energy sensitivity is shown in Fig. 3(c). The dotted curves correspond to the split method while the solid lines correspond to the unsplit method. The difference in colour corresponds to results obtained from different orders of PC expansion.

It is evident that in case of the split method, the residual energy has lower values at the edges. Also, as expected, with an increase in the order of the PC expansion, the residual energy improves.

VI. Conclusion

This paper proposes to exploit the Bernstein Polynomials to determine the bounds of states of uncertain dynamical systems. A two mass spring system is used to illustrate the development of a set of deterministic differential equations after parameterizing the time-invariant model parameter uncertainties using polynomial chaos series. The coefficient of the Bernstein polynomials define a convex hull which permits determination of the upper and lower bounds of the uncertain evolution of the system states. The splitting of the Bernstein coefficients allow the development of a tighter convex hull. Finally, a rest-to-rest maneuver for a system with uncertain stiffness is used to illustrate the design of a state constrained controller which is robust to model parameter uncertainties.

References

[1] T. Singh and S. R. Vadali, "Robust time-delay control," *ASME J. Dyn. Syst. Meas. and Control*, vol. 115, no. 2, pp. 303–306, 1993.

[2] N. C. Singer and W. P. Seering, "Preshaping command inputs to reduce system vibrations," *ASME J. Dyn. Syst. Meas. and Control*, vol. 112, pp. 76–82, 1990.

[3] T. Singh and S. R. Vadali, "Robust time-optimal control: Frequency domain approach," *J. Guid. Control Dyn.*, vol. 17, no. 2, pp. 346–353, 1994.

[4] T. Singh, "Minimax design of robust controllers for flexible structures," *J. Guid. Control Dyn.*, vol. 25, no. 5, pp. 868–875, 2002.

[5] R. Bhattacharya, "A polynomial chaos framework for designing linear parameter varying control systems," in *American Control Conference (ACC), 2015*. IEEE, 2015, pp. 409–414.

[6] N. Wiener, "The homogeneous chaos," *American Journal of Mathematics*, vol. 60, no. 4, pp. pp. 897–936, 1938. [Online]. Available: http://www.jstor.org/stable/2371268

[7] D. Xiu and G. E. Karniadakis, "The Wiener–Askey Polynomial Chaos for Stochastic Differential Equations," *SIAM journal on scientific computing*, vol. 24, no. 2, pp. 619–644, 2002.

[8] J. Garloff, "The Bernstein Expansion and its Applications," *J. Am. Roumanian Acad*, vol. 25, no. 27, pp. 80–85, 2003.

[9] S. Ray and P. Nataraj, "An Efficient Algorithm for Range Computation of Polynomials using the Bernstein form," *Journal of Global Optimization*, vol. 45, no. 3, pp. 403–426, 2009.

[10] R. T. Farouki, "Legendre–Bernstein Basis Transformations," *Journal of Computational and Applied Mathematics*, vol. 119, no. 1, pp. 145–160, 2000.

[11] J. Garloff, "The Bernstein Algorithm," *Interval Computations*, vol. 2, no. 6, pp. 154–168, 1993.

[12] M. d. Berg, O. Cheong, M. v. Kreveld, and M. Overmars, *Computational Geometry: Algorithms and Applications*, 3rd ed. Santa Clara, CA, USA: Springer-Verlag TELOS, c2008.

[13] P. De Casteljau, "Outillages méthodes calcul," *Andr e Citro en Automobiles SA, Paris*, h1959.

[14] T. Singh, *Optimal Reference Shaping for Dynamical Systems: Theory and Applications*. CRC Press, Taylor and Francis Group, o2009.

[15] A. H. Stroud and D. Secrest, *Gaussian quadrature formulas*. Prentice-Hall Englewood Cliffs, NJ, i1966, vol. 39.