Final Technical Report

DOE Award Number: DE-SC0001980

Recipient: Johns Hopkins University

Project Title: Analysis and Reduction of Complex Networks Under Uncertainty

Principal Investigator: Omar M. Knio
Department of Mechanical Engineering
Johns Hopkins University
3400 N. Charles Street
Baltimore, MD 21218

Date: April 9, 2014

Project Goals

This is a collaborative proposal that aims at developing new methods for the analysis and reduction of complex multiscale networks under uncertainty. The approach is based on combining methods of computational singular perturbation (CSP) and probabilistic uncertainty quantification. In deterministic settings, CSP yields asymptotic approximations of reduced-dimensionality "slow manifolds" on which a multiscale dynamical system evolves. Introducing uncertainty raises fundamentally new issues, particularly concerning its impact on the topology of slow manifolds, and means to represent and quantify associated variability. To address these challenges, this project uses polynomial chaos (PC) methods to reformulate uncertain network models, and to analyze them using CSP in probabilistic terms. Specific objectives include (1) developing effective algorithms that can be used to illuminate fundamental and unexplored connections among model reduction, multiscale behavior, and uncertainty, and (2) demonstrating the performance of these algorithms through applications to model problems.

Accomplishments

Random eigenvalue problem

A spectral decomposition methodology has been developed of the Jacobian of stochastic chemical systems. These algorithm establish the capability of expressing both the eigenvalues and eigenvectors of the Jacobian in the form of compact Polyomial Chaos (PC) representations, namely in the space of the physical eigenvectors.

Approach

The development is based on associating with each realization, $\boldsymbol{\xi}$, of the germ parameterizing the random inputs a system Jacobian matrix, $J(\boldsymbol{\xi})$, defined in terms of its entries,

$$J_{ij} = \frac{\partial g_i}{\partial y_j}, \quad i,j = 1,\ldots,n. \tag{1}$$

Provided that the stochastic Jacobian J is in $\left(L^2(\Omega^*)\right)^{n\times n}$ and is almost surely $\mathbb{R}$-diagonalizable with n distinct (stochastic) eigenvalues, we define a spectral representation of $J(\boldsymbol{\xi})$ according to:

$$J(\boldsymbol{\xi}) = \sum_{k=0}^{P} J^k \Psi_k(\boldsymbol{\xi}), \quad \boldsymbol{\xi} \in \Omega^*. \tag{2}$$

Such a spectral representation of J enables us to characterize the dependence of system eigenvectors on the uncertainty spawned by $\boldsymbol{\xi}$, and provides useful expressions of the Jacobian that preserves the physical interpretation of its eigenvectors; that is, each eigenvector is in $\mathbb{R}^n$.

Let $\boldsymbol{u}$ be a random n-vector in with spectral expansion $\boldsymbol{u} = \sum_{k=0}^{P} \boldsymbol{u}^k \Psi_k$. The modes $\boldsymbol{u}^k$ define the coordinates of $\boldsymbol{u}$ in $(\mathcal{V}^p)^n$ which we denote by $[\boldsymbol{u}]$. The matrix $\mathcal{J} \in \mathbb{R}^{N\times N}$ is indexed so that it acts on a random n-vector $\boldsymbol{u} \in (\mathcal{V}^p)^n$ through its coordinate vector $[\boldsymbol{u}]$ as follows:

$$\mathcal{J}[\boldsymbol{u}] = \begin{pmatrix} \mathcal{J}^{00} & \mathcal{J}^{01} & \dots & \mathcal{J}^{0P} \\ \mathcal{J}^{10} & \mathcal{J}^{11} & \dots & \mathcal{J}^{1P} \\ \dots & \dots & \dots & \dots \\ \mathcal{J}^{P0} & \mathcal{J}^{P1} & \dots & \mathcal{J}^{PP} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}^0 \\ \boldsymbol{u}^1 \\ \vdots \\ \boldsymbol{u}^P \end{pmatrix},$$

with $\mathcal{J}^{kl} \in \mathbb{R}^{n\times n}$ defined by

$$\left\langle \Psi_k^2 \right\rangle \mathcal{J}_{ij}^{kl} = \left\langle J_{ij}(\boldsymbol{y}) \Psi_k \Psi_l \right\rangle, \quad i,j = 1,\dots,n,\ k,l = 0,\dots,P. \tag{3}$$

A useful relationship was also established between the spectral representation for J in (2) and the matrix $\mathcal{J} = \left(\mathcal{J}_{ij}^{kl}\right)$ defined in (3). For $i,j = 1,\dots,n$ and $k = 0,\dots,P$ we have

$$\mathcal{J}_{ij}^{k0} = \frac{\langle J_{ij}\Psi_k\Psi_0\rangle}{\left\langle \Psi_k^2 \right\rangle} = \frac{\left\langle \left(\sum_{m=0}^{P} J_{ij}^m \Psi_m\right)\Psi_k \right\rangle}{\left\langle \Psi_k^2 \right\rangle} = \sum_{m=0}^{P} J_{ij}^m \frac{\langle \Psi_k \Psi_m\rangle}{\left\langle \Psi_k^2 \right\rangle} = \sum_{m=0}^{P} J_{ij}^m \frac{\delta_{km}\left\langle \Psi_m^2 \right\rangle}{\left\langle \Psi_k^2 \right\rangle} = J_{ij}^k.$$

That is,

$$\mathcal{J}^{k0} = J^k, \quad k = 0,\dots,P. \tag{4}$$

Solution algorithm

Solution algorithms were developed for the purpose of computing the chaos coefficients of the eigenvalues $\lambda(\boldsymbol{\xi})$, and right (left) eigenvectors $\boldsymbol{a}(\boldsymbol{\xi})$ ($\boldsymbol{b}(\boldsymbol{\xi})$) of the stochastic Jacobian by solving the following eigenvalue problem:

$$J(\boldsymbol{\xi})\boldsymbol{a}(\boldsymbol{\xi}) = \lambda(\boldsymbol{\xi})\boldsymbol{a}(\boldsymbol{\xi}), \qquad \boldsymbol{a}(\boldsymbol{\xi}) \cdot \boldsymbol{a}(\boldsymbol{\xi}) = 1, \tag{5}$$

where $\cdot$ denotes the Euclidean inner product in $\mathbb{R}^n$. The PC representations of $\lambda(\boldsymbol{\xi})$ and $\boldsymbol{a}(\boldsymbol{\xi})$ are

$$\lambda(\boldsymbol{\xi}) = \sum_{k=0}^{P} \lambda^k \Psi_k, \qquad \boldsymbol{a}(\boldsymbol{\xi}) = \sum_{k=0}^{P} \boldsymbol{a}^k \Psi_k. \tag{6}$$

Using (4) to recover the PC representation of $J(\boldsymbol{\xi})$,

$$J(\boldsymbol{\xi}) = \sum_{k=0}^{P} J^k \Psi_k(\boldsymbol{\xi}), \quad J^k = \mathcal{J}^{k0},$$

and using the PC representations of λ, $\boldsymbol{a}$, and J in conjunction with (5), we obtain

$$\sum_{i=0}^{P}\sum_{j=0}^{P}\left(J^i-\lambda^i I\right)\boldsymbol{a}^j\Psi_i\Psi_j=r_1,\qquad \sum_{i=0}^{P}\sum_{j=0}^{P}\boldsymbol{a}^i\cdot\boldsymbol{a}^j\Psi_i\Psi_j-1=r_2, \tag{7}$$

where I is the $n\times n$ identity matrix, and r_1 and r_2 are the residuals due to the finite truncation of the PC expansions. Following the method of Ghanem and Ghosh (*Int. J. Num. Meth. Engng.* 72:486-504, 2007) these residuals are minimized by forcing their orthogonality to the basis of $\mathcal{V}^p$. The stochastic eigenvalue problem becomes:

$$\begin{cases}\displaystyle\sum_{i=0}^{P}\sum_{j=0}^{P}\left(J^i-\lambda^i I\right)\boldsymbol{a}^j\langle\Psi_i\Psi_j\Psi_k\rangle=0,\quad k=0,\dots,P,\\ \displaystyle\sum_{i=0}^{P}\sum_{j=0}^{P}\boldsymbol{a}^i\cdot\boldsymbol{a}^j\langle\Psi_i\Psi_j\Psi_k\rangle-\delta_{k0}=0,\quad k=0,\dots,P.\end{cases} \tag{8}$$

The system (8) can be simply expressed as:

$$\begin{cases}(J-\lambda I)\otimes\boldsymbol{a}=\bar{\boldsymbol{\xi}},\\ \boldsymbol{a}\odot\boldsymbol{a}=1.\end{cases}$$

We compute λ_k and $\boldsymbol{a}^k$ for $k=0\dots P$ by solving the nonlinear system in (8) using Newton-Raphson iterations, and refer to this approach as the residual minimization method.

We also remark that the method of Ghanem and Ghosh (2007) may be extended to compute the right stochastic eigenvectors as well, namely by solving:

$$\boldsymbol{b}(\boldsymbol{\xi})^T J(\boldsymbol{\xi})=\lambda(\boldsymbol{\xi})\boldsymbol{b}(\boldsymbol{\xi})^T,\qquad \boldsymbol{b}(\boldsymbol{\xi})\cdot\boldsymbol{b}(\boldsymbol{\xi})=1.$$

This can be performed easily by replacin J^i by $(J^i)^T$ in (8).

Relation between nominal and stochastic eigenvectors

Consider the eigenpairs of the nominal system Jacobian $\bar{J}=J(\bar{\boldsymbol{\xi}})$,

$$(\bar{\lambda}_i,\bar{\boldsymbol{a}}_i),\quad \bar{\boldsymbol{a}}\cdot\bar{\boldsymbol{a}}=1,\quad i=1,\dots,n,$$

where we have used overbars to indicate that these eigenpairs correspond to the nominal system Jacobian. For a given $\boldsymbol{\xi}\in\Omega^*$, consider the eigenpair $(\lambda_i(\boldsymbol{\xi}),\boldsymbol{a}_i(\boldsymbol{\xi}))$ of $J(\boldsymbol{\xi})$, where $\boldsymbol{a}_i$ is a *unit* eigenvector. Under the assumption that $\lambda_i(\boldsymbol{\xi})$ is a simple real eigenvalue, the choice of $\boldsymbol{a}_i(\boldsymbol{\xi})$ is unique up to its sign. To make the choice of $\boldsymbol{a}_i$ unique, we may multiply $\boldsymbol{a}_i$ by $\operatorname{sgn}(\boldsymbol{a}_i\cdot\bar{\boldsymbol{a}}_i)$, where sgn denotes the signum function.

Assuming that J depends continuously on the parameter $\boldsymbol{\xi}$, it is well known that the eigenvalues of $J(\boldsymbol{\xi})$ depend continuously on $\boldsymbol{\xi}$; that is, for each eigenvalue $\lambda_i(\boldsymbol{\xi})$, we have $\lim_{\boldsymbol{\xi}\to\bar{\boldsymbol{\xi}}}\lambda_i(\boldsymbol{\xi})=\bar{\lambda}_i$. In the case J depends differentiably on $\boldsymbol{\xi}$, we can say that an eigenvector of $J(\boldsymbol{\xi})$ (chosen as above to make it unique for each $\boldsymbol{\xi}$) corresponding to a simple eigenvalue depends differentiably on $\boldsymbol{\xi}$ in a neighborhood of $\boldsymbol{\xi}=\bar{\boldsymbol{\xi}}$; this in particular implies that,

$$\lim_{\boldsymbol{\xi}\to\bar{\boldsymbol{\xi}}}\boldsymbol{a}_i(\boldsymbol{\xi})=\boldsymbol{a}_i(\bar{\boldsymbol{\xi}})=\bar{\boldsymbol{a}}_i. \tag{9}$$

A measure of deviation of $\boldsymbol{a}_i(\boldsymbol{\xi})$ from $\bar{\boldsymbol{a}}$ is the angle ζ_i bewteen them:

$$\cos(\zeta_i(\boldsymbol{\xi}))=\boldsymbol{a}_i(\boldsymbol{\xi})\cdot\bar{\boldsymbol{a}}_i. \tag{10}$$

Since the eigenvectors $\bar{\boldsymbol{a}}_i$ form a basis of $\mathbb{R}^n$, we may expand the vectors $\boldsymbol{a}_i$ in the basis $\{\bar{\boldsymbol{a}}_1, \ldots, \bar{\boldsymbol{a}}_n\}$:

$$\boldsymbol{a}_i(\boldsymbol{\xi}) = \sum_{j=1}^{n} h_{ij}(\boldsymbol{\xi})\bar{\boldsymbol{a}}_i. \tag{11}$$

The coefficients h_{ij} provide a further means to assess how close the eigenvectors are to those of the nominal system. Note that,

$$\bar{\boldsymbol{b}}_k \cdot \boldsymbol{a}_i(\boldsymbol{\xi}) = \bar{\boldsymbol{b}}_k \cdot \Big(\sum_{j=1}^{n} h_{ij}(\boldsymbol{\xi})\bar{\boldsymbol{a}}_i \Big) = h_{ik}(\boldsymbol{\xi}).$$

That is, $h_{ij}(\boldsymbol{\xi}) = \bar{\boldsymbol{b}}_j \cdot \boldsymbol{a}_i(\boldsymbol{\xi})$. We find it convenient to consider the matrix $H = H(\boldsymbol{\xi})$ defined by $H_{ij} = h_{ij}$, $i, j = 1, \ldots, n$.

In view of (9) we have

$$\lim_{\boldsymbol{\xi} \to \bar{\boldsymbol{\xi}}} \cos(\zeta_i(\boldsymbol{\xi})) = 1, \tag{12}$$

and

$$\lim_{\boldsymbol{\xi} \to \bar{\boldsymbol{\xi}}} h_{ij}(\boldsymbol{\xi}) = \lim_{\boldsymbol{\xi} \to \bar{\boldsymbol{\xi}}} \bar{\boldsymbol{b}}_j \cdot \boldsymbol{a}_i(\boldsymbol{\xi}) = \bar{\boldsymbol{b}}_j \cdot \bar{\boldsymbol{a}}_i = \delta_{ij}.$$

That is, $\lim_{\boldsymbol{\xi} \to \bar{\boldsymbol{\xi}}} h_{ij} = \delta_{ij}$; this can be written more concisely as

$$\lim_{\boldsymbol{\xi} \to \bar{\boldsymbol{\xi}}} \|H(\boldsymbol{\xi}) - I\|_F = 0, \tag{13}$$

where $\|\cdot\|_F$ denotes the Frobenius norm:

$$\|A\|_F = \Big(\sum_{i=1}^{m} \sum_{j=1}^{n} |A_{ij}|^2 \Big)^{1/2}, \quad A \in \mathbb{R}^{m \times n}.$$

The uncertainty measures $\cos(\zeta_i(\boldsymbol{\xi}))$ and

$$\eta(\boldsymbol{\xi}) = \|H(\boldsymbol{\xi}) - I\|_F$$

to provide useful means for studying the deviation of the stochastic eigenvectors from their nominal counterparts.

The Stochastic CSP framework

Based on the development above, we have generalized the deterministic CSP formalism to the stochastic case. Our approach is based on exploiting the spectral representation of the n-dimensional stochastic eigenvectors as CSP vectors. Accordingly, the resulting CSP formalism involves a projection onto a stochastic manifold. Two variants of the stochastic CSP methodology have been developed. The first, is based on using the stochastic eigenvectors as CSP vectors. The second variant is based on using the deterministic eigenvectors of the nominal system as CSP vectors. The latter is inspired by the observation that in many cases the stochastic eigenvalues exhibit a pronounced dependence on $\boldsymbol{\xi}$, whereas the stochastic eigenvectors are weakly sensitive to the uncertainty. Thus, in these situations the stochastic CSP algorithm may rely on projection onto a deterministic manifold, though the projection still involves stochastic mode amplitudes. One of the main advantages offered by both of these methods is that they provide simple (averaged) criteria for determination of (stochastic) fast subspaces of an uncertain ODE system.

Stochastic eigenvectors as CSP vectors

For the uncertain ODE system of the form $\boldsymbol{y} = \boldsymbol{g}(\boldsymbol{y})$, we seek a decomposition of the system right-hand-side in terms of the stochastic eigenvectors of $J(\boldsymbol{\xi})$. By our assumptions on the eigenstructure of $J(\boldsymbol{\xi})$, we know that its right eigenvectors $\{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n\}$ form a basis of $\mathbb{R}^n$ and hence $\boldsymbol{g}$ can be expanded in this basis; however, the expansion coefficients f_i will also be random. Considering the weak form of the governing equation, we arrive at:

$$\dot{\boldsymbol{y}} = \boldsymbol{g}(\boldsymbol{y}) = \sum_{i=1}^{n} \boldsymbol{a}_i \otimes f_i, \tag{14}$$

with $\boldsymbol{y} \in (\mathcal{V}^p)^n$, $\boldsymbol{a}_i \in (\mathcal{V}^p)^n$, and $f_i \in \mathcal{V}^p$. Thus, we have to compute the n stochastic coordinates $f_i(\boldsymbol{\xi})$ of the system right-hand side in the space spanned by the right stochastic eigenvectors. The coefficients f_i are evaluated using the Galerkin dot product:

$$f_i \doteq \boldsymbol{b}_i \odot \boldsymbol{g}, \tag{15}$$

where we have relied on the approximate orthogonality between the stochastic left and right eigenvectors.

With m fast modes, the system (14) is reduced according to:

$$\dot{\boldsymbol{y}} \approx \sum_{i=m+1}^{n} \boldsymbol{a}_i \otimes f_i \approx \underbrace{(I - \sum_{i=1}^{m} \boldsymbol{a}_i \otimes \boldsymbol{b}_i)}_{P} \otimes \boldsymbol{g} =: P \otimes \boldsymbol{g}. \tag{16}$$

The difficulty in the stochastic case is that in general m will depend on $\boldsymbol{\xi}$. Instead of seeking a criterion guaranteeing m fast modes for all (or almost all) $\boldsymbol{\xi} \in \Omega^*$, which is a difficult task, we formulate an approximate criterion in a mean-square sense. Specifically, we deem the first m modes fast if the following inequality holds:

$$\left\langle (\tau_{m+1} \otimes \sum_{i=1}^{m} \boldsymbol{a}_i \otimes f_i)^2 \right\rangle < \varepsilon_{rel} \left\langle \boldsymbol{y}^2 \right\rangle + \varepsilon_{abs} \mathbf{1}, \tag{17}$$

where it is understood that the square operator applies component-wise on the corresponding vectors; that is, $\boldsymbol{y}^2$ denotes a vector whose n components are the squares of the corresponding components of $\boldsymbol{y}$. Note that the criterion in (17) is convenient to implement, because in actual computations the stochastic quantities are represented by their coordinates in the orthogonal basis of $\mathcal{V}^p$. For instance,

$$\left\langle \boldsymbol{y}^2 \right\rangle = \sum_{k=0}^{P} \left\langle \Psi_k^2 \right\rangle \left(\boldsymbol{y}^k\right)^2.$$

Different criteria can be considered [J1].

Algorithm for CSP with stochastic eigenvectors: Assembling all the previous components, we end up with the following stochastic CSP algorithm 1, for the integration of the solution over a time step. Note that for a system with m fast modes, Algorithm 1 can be used for integration of the simplified stochastic system $\dot{\boldsymbol{y}} = P \otimes \boldsymbol{g}$ with time steps on the order of $\alpha \tau_{m+1}^0$ with $0 < \alpha \lesssim 1$.

We observe that Algorithm 1 involves solving the stochastic eigenproblem along with several Galerkin products. In addition, it involves the inversion of the stochastic matrix $\Lambda(t)$, which can be accomplished either through a Galerkin linear solve or adaptation of an iterative method for approximation of a matrix inverse to the stochastic case [J1]. In cases when the nominal system dynamics can be reliably used to approximate the stochastic system trajectory, we may use the nominal system eigenvectors as CSP vectors. As shown in [J1], This simplifies many of the computational steps in Algorithm 1.

Algorithm 1 One step of CSP integration (assuming modes $1, \ldots, m$ are fast) for the stochastic system using the stochastic right eigenvectors as CSP vectors.

$$\Lambda(t) = \big[\Lambda_{ij}(t)\big] = \Big(\dot{\boldsymbol{b}}_i^T + \boldsymbol{b}_i^T \otimes J\Big) \otimes \boldsymbol{a}_j\Big|_t$$

$$\Gamma(t) = \big(\Lambda(t)\big)^{-1}$$

$$\hat{\boldsymbol{y}}(t+\Delta t) = \boldsymbol{y}(t) + \int_t^{t+\Delta t} P \otimes \boldsymbol{g}(\boldsymbol{y}(s))\, ds \qquad \{\text{Integration of the slow dynamics}\}$$

$$f_j(t) = \boldsymbol{b}^j(t) \odot \boldsymbol{g}\big[\hat{\boldsymbol{y}}(t+\Delta t)\big], \quad j = 1, \ldots, m$$

$$\delta\boldsymbol{y}(t) = \sum_{i=1}^{m} \gamma_i(t) \otimes \boldsymbol{a}_i(t), \quad \text{where } \gamma_i(t) = \sum_{j=1}^{m} \Gamma_{ij}(t) \otimes f_j(t)$$

$$\boldsymbol{y}(t+\Delta t) = \hat{\boldsymbol{y}}(t+\Delta t) - \delta\boldsymbol{y}(t) \qquad \{\text{Radical correction}\}$$

Nominal system eigenvectors as CSP vectors:

We have seen that the extension of the deterministic CSP method to the stochastic case involves the calculation of the left and right stochastic eigenvectors and eigenvalues of the stochastic Jacobian $J(\boldsymbol{\xi})$. These are computed through the solution of nonlinear deterministic problems after stochastic discretization in $\mathcal{V}^p$. It can be anticipated that the determination of the stochastic eigenspaces constitutes the most computationally demanding part of the stochastic CSP method, justifying the question of the possibility to bypass this step in order to reduce the computational cost. When the uncertainty in the system induces a moderate variability of the eigenvectors, the stochastic eigenvectors can be well approximated by the deterministic eigenvectors of the nominal Jacobian $\bar{J} = J(\bar{\boldsymbol{\xi}})$, allowing for the expansion of the stochastic system right-hand-side in the basis of the deterministic nominal eigenvectors. Using the nominal system eigenvectors for CSP vectors effectively uses the dynamics of the nominal system as a guide for determining the fast and slow subspaces for the uncertain system. This naturally leads to significant computational savings highlighted by not having to compute the stochastic eigenvectors.

Assuming such approximation is valid, we reformulate the uncertain system by expanding its right-hand-side in the basis of the nominal eigenvectors $\{\bar{\boldsymbol{a}}_1, \ldots, \bar{\boldsymbol{a}}_n\}$:

$$\dot{\boldsymbol{y}} = \boldsymbol{g}(\boldsymbol{y}) = \sum_{i=1}^{n} \bar{\boldsymbol{a}}_i f_i. \tag{18}$$

The expansion coefficients (mode amplitudes) f_i are stochastic and are given by

$$f_i(\boldsymbol{\xi}) = \bar{\boldsymbol{b}}_i^T \boldsymbol{g}(\boldsymbol{y}, \boldsymbol{\xi}), \quad i = 1, \ldots, n, \tag{19}$$

where $\bar{\boldsymbol{b}}_{\boldsymbol{i}} \in \mathbb{R}^n$ are the nominal left eigenvectors.

The stochastic governing equation can be projected onto $\mathcal{V}^p$, resulting in

$$\dot{\boldsymbol{y}}^k = \sum_{i=1}^{n} \bar{\boldsymbol{a}}_i f_i^k, \quad k = 0, \ldots, P. \tag{20}$$

For m fast modes, we obtain the following simplified system

$$\dot{\boldsymbol{y}}^k \approx \sum_{i=m+1}^{n} \bar{\boldsymbol{a}}_i f_i^k = \underbrace{(I - \sum_{i=1}^{m} \bar{\boldsymbol{a}}_i \bar{\boldsymbol{b}}_i^T)}_{\bar{P}} \boldsymbol{g}^k =: \bar{P}\boldsymbol{g}^k. \tag{21}$$

Note that, presently, the projection operator $\bar{P}$ on the slow dynamics space is deterministic, and the same for *all* the stochastic modes of the solution. This has to be contrasted with the method developed above.

Also note that the present simplification calls enables us to adopt a simplified criterion for mode exhaustion. Specifically, an approximate criterion of the form:

$$\left\langle \left(\bar{\tau}_{m+1} \sum_{i=1}^{m} \bar{\boldsymbol{a}}_{\boldsymbol{i}} f_i\right)^2 \right\rangle < \varepsilon_{rel} \left\langle \boldsymbol{y}^2 \right\rangle + \varepsilon_{abs} \mathbf{1}.$$

can be implemented. Note that in this case $\bar{\tau}_{m+1} = 1/\bar{\lambda}_{m+1}$ and $\bar{\boldsymbol{a}}_i$ are deterministic.

Algorithm for the simplified CSP with nominal eigenvectors: For a system with m fast modes, Algorithm 2 can be used for integration of the simplified stochastic system $\dot{\boldsymbol{y}}^k = \bar{P}\boldsymbol{g}^k$, with time steps on the order of $\alpha\bar{\tau}_{m+1}$ with $0 < \alpha \lesssim 1$.

Algorithm 2 One step of CSP integration (assuming modes $1, \ldots, m$ are fast) for the stochastic system using the nominal right eigenvectors as CSP vectors.

$\bar{\Lambda}(t) = \left[\bar{\Lambda}_{ij}(t)\right] = \left(\dot{\bar{\boldsymbol{b}}}_i^T + \bar{\boldsymbol{b}}_i^T \bar{J}\right) \bar{\boldsymbol{a}}_j \Big|_t$

$\bar{\Gamma}(t) = \left(\bar{\Lambda}(t)\right)^{-1}$

for $k = 0$ to P **do**

$\hat{\boldsymbol{y}}^k(t + \Delta t) = \boldsymbol{y}^k(t) + \int_t^{t+\Delta t} \bar{P}\boldsymbol{g}^k(\boldsymbol{y}(s))\, ds$ {Integration of the slow dynamics}

end for

$f_j^k(t) = \bar{\boldsymbol{b}}^j(t) \cdot \boldsymbol{g}^k\left[\hat{\boldsymbol{y}}(t + \Delta t)\right], \quad j = 1, \ldots, m, \quad k = 0, \ldots, P$

for $k = 0$ to P **do**

$\delta\boldsymbol{y}^k(t) = \sum_{i=1}^{m} \gamma_i^k(t)\bar{\boldsymbol{a}}_i(t), \quad$ where $\gamma_i^k(t) = \sum_{j=1}^{m} \bar{\Gamma}_{ij}(t) f_j^k(t)$

$\boldsymbol{y}^k(t + \Delta t) = \hat{\boldsymbol{y}}^k(t + \Delta t) - \delta\boldsymbol{y}^k(t)$ {Radical correction}

end for

Implementations

The CSP algorithms outlined above were applied to a model chemical system, and computational experiments were conducted to illustrate the eigenvector representations and to test the performance of the CSP integration algorithms. The analysis showed that for system considered, the stochastic eigenvalues exhibit a rich PC spectrum and accordingly pronounced variability, whereas the eigenvectors exhibit an essentially degenerate spectrum with the dominant amplitude lying in the mean. Thus, the random eigenvectors were found to remain closely aligned with the corresponding nominal ones. These features were exploited to simplify CSP projection onto deterministic (nominal) manifolds, and consequently enhance the efficiency of the integration. The simulations were also used to validate the explicit Galerkin solution, namely by comparison with direct Monte Carlo integration. The validated Galerkin solution was then used to conduct performance tests. In all cases considered, a close agreement between the predictions was found, and substantial speedup in the integration of the uncertain dynamical system was achieved. In particular, when CSP using nominal eigenvectors was used, order-of-magnitude improvements in integration efficiency were observed.

A detailed account of algorithmic constructions and computational tests is provided in [J1].

Preconditioned Stochastic Projection

A well-known difficulty that occurs with time-dependent dynamical systems concerns the broadening of the spectrum of the PC representation over time. This in turns makes it difficult to approximate model variables via finite PC expansions, and to select a PC basis dimension that remains suitable over the entire integration horizon. The underlying difficulty is that different realizations of the random solution trajectory fail to stay in phase as time progresses. In the context of chemical systems, for instance, some realizations can reach equilibrium very quickly, while others can exhibit a transient behavior over substantially larger timescales. This decorrelation of trajectories entails the generation of multiple modes in the PC representation of the model variables over time. Therefore, the corresponding PC expansions require a large number of terms for a reasonably accurate approximation of the stochastic behavior.

In order to address this difficulty, we have developed a stochastic preconditioning approach, in the context of non-intrusive methods. The method addresses the issue of widening spectra by introducing a global, multiple time scale, linear or affine stretching of the solution variables so as to synchronize realizations, and consequently control their variance and PC spectrum. The basic idea behind the method is to work with transformed variables in scaled time which are in phase and have controlled spectra, *i.e.* the transformed variables have tight low order PC representations which can be computed with a limited number of realizations. Subsequently, the distribution of the original variables can be easily recovered, from that of transformed ones, through a sampling strategy.

Approach

The approach we have developed generally involves three distinct steps, namely the definition of the transformation, the projection of the transformed variables, and finally the recovery of the original variable, for the purpose of sampling and analysis.

Taking advantage of the non-intrusive character of the method, the preconditioning is performed separately for each component of the system. Therefore, for convenience, we on a generic scalar variable, X, and its transform, Y.

Variable transformations: Recall that the state variable X is a stochastic process, $X : [0, T_{fin}] \times \Omega^* \to \mathbb{R}$. We introduce the transformed variable $Y = Y(\tau, \boldsymbol{\xi})$ that depends on the scaled time, τ. The scaled time $\tau = \tau(t, \boldsymbol{\xi})$ is defined through,

$$\tau(t, \boldsymbol{\xi}) = \frac{t}{\hat{t}(\boldsymbol{\xi})}, \quad t \in [0, +\infty), \tag{22}$$

where $\hat{t} : \Omega^* \to (0, \infty)$ is a random variable which we call the time scaling factor. We assume that there exist positive constants $\kappa_1 < \kappa_2$ such that

$$\kappa_1 \leq \hat{t}(\boldsymbol{\xi}) \leq \kappa_2, \quad \text{for almost all } \boldsymbol{\xi} \in \Omega^*.$$

In general, we define the scaled variable Y by

$$Y(\tau(t, \boldsymbol{\xi}), \boldsymbol{\xi}) = \Phi[X(t, \boldsymbol{\xi})].$$

where Φ is an invertible mapping on $L^2(\Omega^*)$. We mainly focused on the case of *linear scaling* where,

$$\Phi[X(t, \boldsymbol{\xi}); \hat{c}] = \frac{1}{\hat{c}(\boldsymbol{\xi})} X(t, \boldsymbol{\xi}), \tag{23}$$

where $\hat{c} : \Omega^* \to (0, \infty)$ is an amplitude scaling factor; we assume that there exist positive constants ν_1 and ν_2 that are independent of $\boldsymbol{\xi}$ and,

$$\nu_1 \leq \hat{c}(\boldsymbol{\xi}) \leq \nu_2, \quad \text{for almost all } \boldsymbol{\xi} \in \Omega^*. \tag{24}$$

Note that in the case where $X(t,\boldsymbol{\xi})$ is positive valued, as is the case for chemical species concentrations, we may also consider *log-linear scaling*,

$$\Phi[X(t,\boldsymbol{\xi});\hat{c}] = \log\left(\frac{1}{\hat{c}(\boldsymbol{\xi})}X(t,\boldsymbol{\xi})\right). \tag{25}$$

An advantage of the log-linear scaling is in preserving positivity of the recovered state variables. Moreover, since the scaled variables are ultimately approximated using PC expansions, sometimes the log-linear scaling may result in a better behaved PC representations for the scaled variables.

Projection of the scaled variables: PC representations of the scaling factors, $\hat{c}$ and $\hat{t}$, are computed using a NISP sample set $\mathcal{S}$. We assume that for each $\boldsymbol{\xi}_j \in \mathcal{S}$ one is able to define a corresponding pair of factors $(\hat{t}(\boldsymbol{\xi}_j);\hat{c}(\boldsymbol{\xi}_j))$, the corresponding realization $X(\boldsymbol{\xi}_j,t)$ is used for this purpose. The expansions of the scaling factors are

$$\hat{c} \doteq \sum_{k=0}^{P} c_k\Psi_k, \qquad \hat{t} \doteq \sum_{k=0}^{P} t_k\Psi_k,$$

where the expansion coefficients are computed through a classical NISP procedure:

$$c_k = \sum_{j=1}^{N_q} \Pi_{k,j}\hat{c}(\boldsymbol{\xi}_j), \quad t_k = \sum_{j=1}^{N_q} \Pi_{k,j}\hat{t}(\boldsymbol{\xi}_j).$$

Since $\hat{c}$ and $\hat{t}$ are positive we may also consider the expansion of their logarithms. This again has the advantage of preserving their positivity and in some cases yielding nicer (tighter and/or sparser) PC spectra. In particular, we will project the logarithm of the normalized variables, $\hat{c}(\boldsymbol{\xi})/\hat{c}(\boldsymbol{\xi}=\mathbf{0})$ and $\hat{t}(\boldsymbol{\xi})/\hat{t}(\boldsymbol{\xi}=\mathbf{0})$:

$$\log\left(\frac{\hat{c}(\boldsymbol{\xi})}{\hat{c}(\mathbf{0})}\right) \doteq \sum_{k=0}^{P} \sigma_k\Psi_k, \qquad \log\left(\frac{\hat{t}(\boldsymbol{\xi})}{\hat{t}(\mathbf{0})}\right) \doteq \sum_{k=0}^{P} \theta_k\Psi_k,$$

where

$$\sigma_k = \sum_{j=1}^{N_q} \Pi_{k,j}\log\left(\frac{\hat{c}(\boldsymbol{\xi}_j)}{\hat{c}(\mathbf{0})}\right), \quad \theta_k = \sum_{j=1}^{N_q} \Pi_{k,j}\log\left(\frac{\hat{t}(\boldsymbol{\xi}_j)}{\hat{t}(\mathbf{0})}\right). \tag{26}$$

Accordingly $\hat{c}$ and $\hat{t}$ are approximated through,

$$\hat{c} \doteq \hat{c}(\mathbf{0})\exp\left(\sum_{k=0}^{P} \sigma_k\Psi_k\right), \qquad \hat{t} \doteq \hat{t}(\mathbf{0})\exp\left(\sum_{k=0}^{P} \theta_k\Psi_k\right). \tag{27}$$

Preconditioned projection algorithm: The major computational work in the method involves the non-intrusive projection of the transformed variable, $Y(\tau(t,\boldsymbol{\xi}),\boldsymbol{\xi}) = \Phi\big[X(t,\boldsymbol{\xi});\hat{c}\big]$, to get its PC representation. This calls for time discretization. Owing to the assumed properties of the time scaling in (22), for every $t \in [0,+\infty)$, $\tau(t,\cdot) \in [0,+\infty)$ almost surely. Moreover, for a given $(t',\boldsymbol{\xi}) \in [0,+\infty)\times\Omega^*$, there corresponds an (unscaled) time t given by $t = t'\times\hat{t}(\boldsymbol{\xi})$, and we have that

$$Y(t',\boldsymbol{\xi}) = \Phi\Big[X\big(t'\times\hat{t}(\boldsymbol{\xi}),\boldsymbol{\xi}\big);\hat{c}\Big].$$

This allows us to discretize the transformed variable Y on a fixed *deterministic* grid of times t' in the scaled space. With the slight abuse of notation where there is no confusion between τ and t', the PC expansion sought is then expressed as

$$Y(\tau, \boldsymbol{\xi}) \doteq \sum_{k=0}^{P} Y_k(\tau)\Psi_k(\boldsymbol{\xi}) \approx \Phi\Big[X(\tau \times \hat{t}(\boldsymbol{\xi}), \boldsymbol{\xi}); \hat{c}\Big], \tag{28}$$

where the PC coefficients $Y_k(\tau)$ are to be computed on a fixed grid of scaled times τ. Specifically, given $\Delta\tau > 0$ and defining $\tau_l \doteq l\Delta\tau$, $l = 0, 1, \ldots$ the NISP projection of the k-PC mode of Y would be

$$Y_k(\tau_l) = \sum_{j=1}^{N_q} \Pi_{k,j}\Phi\Big[X(\tau_l \times \hat{t}(\boldsymbol{\xi}_j), \boldsymbol{\xi}_j); \hat{c}\Big]. \tag{29}$$

The expression in (29) suggests an implementation where one first selects the time step $\Delta\tau$ and then performs N_q successive deterministic solves for $X(t, \boldsymbol{\xi}_j)$, $\boldsymbol{\xi}_j \in \mathcal{S}$, recording and projecting the current state when t reaches $t = (l\Delta\tau) \times \hat{t}(\boldsymbol{\xi}_j)$. However, since each component X^i of the state vector $\boldsymbol{X}$ has its own time scale factor $\hat{t}^i(\boldsymbol{\xi})$, we have preferred a different approach resulting in a deterministic time integration of $\boldsymbol{X}(t, \boldsymbol{\xi}_j)$ with a fixed time step, Δt, for all the elements in $\mathcal{S}$, and to rely on an interpolation procedure to retrieve the values $X^i(\tau_l^i \times \hat{t}^i(\boldsymbol{\xi}_j), \boldsymbol{\xi}_j)$ needed for the projection in (29).

Recovering the original variables: We now turn to the problem of reconstructing $\boldsymbol{X}(t, \boldsymbol{\xi})$ from the transformed variable $\boldsymbol{Y}(\tau, \boldsymbol{\xi})$. The objective is to derive an efficient procedure to resample $\boldsymbol{X}(t, \boldsymbol{\xi})$, particularly to perform various statistical analyses. Specifically, we want to reconstruct the corresponding value of $\boldsymbol{X}(t, \boldsymbol{\xi}_l)$.

The recovery amounts to the inversion of the transformation:

$$X(t, \boldsymbol{\xi}) = \Phi^{-1}\left[Y(\tau(t, \boldsymbol{\xi}), \boldsymbol{\xi}); \hat{c}(\boldsymbol{\xi})\right]. \tag{30}$$

For the case of linear scaling, we have

$$X(t, \boldsymbol{\xi}) = \Phi^{-1}\left[Y(\tau(t, \boldsymbol{\xi}), \boldsymbol{\xi}); \hat{c}(\boldsymbol{\xi})\right] = \hat{c}(\boldsymbol{\xi})Y(\tau(t, \boldsymbol{\xi}), \boldsymbol{\xi}), \tag{31}$$

while for the case of log-linear scaling, we have

$$X(t, \boldsymbol{\xi}) = \Phi^{-1}\left[Y(\tau(t, \boldsymbol{\xi}), \boldsymbol{\xi})\right] = \hat{c}(\boldsymbol{\xi}) \exp\big(Y(\tau(t, \boldsymbol{\xi}), \boldsymbol{\xi})\big). \tag{32}$$

Of course, in practice we will approximate X by inserting the PC expansions of the scaling factors and scaled variables in (30). Introducing the notation $\tilde{X}$ to denote the recovered state variable X from the PC expansions of the scaled variable through the above relations, we have,

$$\begin{cases} \tilde{X}(t, \boldsymbol{\xi}) = \hat{c}(\boldsymbol{\xi})\Big\{ \sum_{k=0}^{P} Y_k(\tau(t, \boldsymbol{\xi}))\Psi_k(\boldsymbol{\xi}) \Big\} & \text{linear scaling,} \\ \tilde{X}(t, \boldsymbol{\xi}) = \hat{c}(\boldsymbol{\xi})\Big\{ \exp\big(\sum_{k=0}^{P} Y_k(\tau(\boldsymbol{\xi}, t))\Psi_k(\boldsymbol{\xi}) \big) \Big\} & \text{log-linear scaling.} \end{cases} \tag{33}$$

We can then sample the expansion of $\tilde{X}(t^*, \cdot)$ in (33) to generate an approximate distribution of $X(t^*, \cdot)$ fairly efficiently. The only issue remaining is that the PC coefficients of the scaled variable Y are known on the fixed grid of scaled times τ_l which generally will not coincide with the time $\tau(t^*, \cdot)$ in (33). A piecewise linear interpolation may be used to compute the PC coefficients Y_k at the needed scaled time τ.

Finally, a complete sampling method is immediately constructed through a Monte Carlo procedure wrapped around the recovery procedure. For fixed t^*, one only has to sample Ω^* from the distribution $F_{\boldsymbol{\xi}}$ to obtain a sampling of the distribution of $\tilde{X}(t^*, \cdot)$.

Implementations

The approach above was illustrated for the case of a stiff oxidation problem with large uncertainties in rate parameters. Computations relied on first constructing a coarse computational database of individual realizations. The database was then used to define appropriate characteristic time and concentration scales for each species and each of the realizations. A NISP approach is then used to express these *global* characteristic scales in terms of appropriate, low-order PC expansions, and the latter are used to scale individual realizations. NISP is also used to project the resulting stretched realizations onto the PC basis. This leads to a reprensentation of the response of the stochastic system in terms of an appropriate combination of the PC expansions for the preconditioners and for the stretched variables. Computational experiments conducted using a coarse computed database showed that the preconditioning approach can naturally lead to tight low-order PC representations, and that the latter may be efficiently sampled to recover the distribution of the original variables in the physical timescale.

A detailed account of algorithmic constructions and computational tests is provided in [J2].

Bayesian Preconditioning Approach

When chemical system dynamics are modeled using stochastic simulation algorithms (SSAs), several challenges arise due to the compound impact of high fluctuation levels (intrinsic stochasticity due to finite system size), and of the steep dependence of the system response on uncertain rates. In these situations, a straightforward NISP approach can be problematic especially when sparse SSA sampling must be utilized to estimate the quantity of interest (QoI) at the quadrature nodes. Specifically, accuracy may severely deteriorate due to large sampling errors.

To address these hurdles, an alternative to the projection based approach described above was also developed. The method extends the stochastic preconditioning approach discussed for deterministic systems. It utilizes appropriate transformations (preconditioning) of the state variables and infers a PC model for the transformed (preconditioned) variables via Bayesian regression. Two possibilities were considered for computing the preconditioner, which is characterized by time and amplitude scaling factors. The first approach uses Bayesian regression, whereas PC expansions of the scaling factors are determined in the second using deterministic realizations of the reaction rate expressions (RREs) for the chemical system. One then proceeds to discretize the resulting preconditioned variables on a scaled time mesh, and Bayesian regression is used to infer a PC response surface for them. The process results in response surfaces for the expected trajectory and variance of the stochastic system, involving combinations of deterministic PC models for the preconditioners and stochastic PC models of the preconditioned variables.

The approach was illustrated for the case of a Michaelis-Menten dynamics and a genetic positive feedback loop. Both of the examples show large noise amplitudes, the latter also exhibiting steep response to uncertainty in the reaction rates. Computational experiments show that Bayesian preconditioning algorithms can simultaneously accommodate large variation with uncertain inputs and high fluctuation levels, and that robust estimates can be obtained with a moderate number of SSA samples. This includes the dependence of both the mean and variance of the state variables on the uncertain inputs. This constitutes a significant advantage, since performing a large number of SSA replicas in high dimensions can be prohibitively expensive.

Details of the algorithmic constructions, numerical constructions and performance metics are provided in [J3]; for brevity, they are omitted from this report.

Publications

Journal Publications:

J1 M. Salloum, A. Alexanderian, O.P. Le Maître, H.N. Najm, O.M. Knio (2012) "Simplified CSP Analysis of a Stiff Stochastic ODE System," *Comput. Methods Appl. Mech. Engrg.* **217**, 121-138.

J2 A. Alenxanderian, O.P. Le Maître, H.N. Najm, M. Iskandarani, O.M. Knio (2012) "Multiscale stochastic preconditioners in non-intrusive spectral projection," *J. Sci. Comput.* **50**, 306-340.

J3 A. Alexanderian, F. Rizzi, M. Rathinam, O.P. Le Maître, O.M. Knio (2014) "Preconditioned Bayesian Regression for Stochastic Chemical Kinetics," *Journal of Scientific Computing* **58**, 592-626.

Other Publications Acknowledging DOE Support:

O1 O.P. Le Maître, L. Mathelin, O.M. Knio & M.Y. Hussaini (2010) "Asynchronous Time Integration for Polynomial Chaos Expansion of Uncertain Periodic Dynamics," *Discrete and Continuous Dynamical Systems* **28**, 199-226.

O2 O.P. Le Maître & O.M. Knio (2010) *Spectral Methods for Uncertainty Quantification*, Springer-Verlag Series on Scientific Computing.

Lectures and Presentations:

L1 O. Knio, "Spectral methods for uncertainty quantification," Fall Meeting of the Washington-Baltimore Section of SIAM, December 9, 2009.

L2 O. Knio, "Spectral Methods for uncertainty quantification," University of Miami, April 6, 2010.

L3 O. Knio, "Spectral Methods for uncertainty quantification," MIT, April 23, 2010.

L4 O. Knio, "Spectral Methods for uncertainty quantification," Duke University, April 27, 2010.

L5 O. Le Maître, L. Mathelin, O. Knio, M.Y. Hussaini (2010) "Asynchronous Time Integration for Polynomial Chaos Expansion of Uncertain Periodic Dynamics," presented at IV European Conference on Computational Mechanics, Paris, France, May 16-21, 2010.

L6 O. Knio, "Spectral Methods for uncertainty quantification," Free University of Berlin, May 31, 2010.

L7 O. Le Maître, L. Mathelin, O. Knio, M.Y. Hussaini (2010) "Asynchronous Time Integration for Polynomial Chaos Expansion of Uncertain Periodic Dynamics," presented at SIAM Annual Meeting, Pittsburgh, PA, July 12-16, 2010.

L8 O. Knio, "Spectral Approach to Uncertainty Quantification in Complex Dynamical Systems," 2010 MetStroem Program Meeting, FU Berlin, October 28, 2010. (**plenary**)

L9 O. Knio, "Spectral Approach to Uncertainty Quantification in Complex Dynamical Systems," UMBC, October 15, 2010.

L10 O. Knio, "Spectral Approaches of Uncertainty Quantification in Complex Dynamical Systems," UC Berkeley, February 16, 2011.

L11 A. Alexanderian, M. Salloum, O. Le Maitre, H. Najm, O. Knio (2011) "Simplified CSP Analysis of a Stochastic Chemical System," presented at SIAM Conference on Computational Science and Engineering, February 28 - March 4, 2011, Reno, NV.

L12 A. Alexanderian, M. Salloum, O. Le Maitre, H. Najm, M. Iskandarani, O.M. Knio (2011) "Recent experiences with stochastic preconditioners and random eigenvalues," presented at Workshop on Uncertainty Quantification for Multiphysics and Multiscale Systems, USC, March 7-8, 2011.

L13 M. Salloum, O.M. Knio, H.N. Najm (2011) "Analysis and Reduction of a Simplified Stochastic Chemical System via CSP,"presented at 3rd International Workshop on Model Reduction in Reacting Flows April 27-29, 2011, Corfu, Greece.

L14 A. Alexanderian, O. Le Maitre, H.N. Najm, M. Iskandarani, O.M. Knio (2011) "Multiscale Stochastic Preconditioners in Non-Instrusive Spectral Projection," presented at ICIAM 2011, July 18-22, 2011, Vancouver BC.

L15 O. Knio, "Probabilistic Methods for Uncertainty Quantification: Recent Developments and Implications to Extreme Scale Computing," DOE SCGF Conference, ORNL, July 18, 2011.

L17 O. Knio, "Uncertainty Quantification Challenges in Modeling Complex Systems," Winter Enrichment Program, KAUST, January 15, 2012.

L18 O. Knio, "Spectral Uncertainty Quantification in Stiff Dynamical Systems," CIMPA School in Applied Mathematics on Uncertainty Quantification, January 5, 2012.

L19 O.M. Knio (2012) "Polynomial Chaos Approaches to Multiscale and Data Intensive Computations," presented at SIAM Conference on Uncertainty Quantification, Raleigh, NC, April 2-5, 2012. (**plenary**)