Computer Science Technical Report
TR-11-15
August 16, 2011

Joe Hays, Adrian Sandu[*], Corina Sandu[^], Dennis Hong[~]

"Simultaneous Optimal Uncertainty Apportionment and Robust Design Optimization of Systems Governed by Ordinary Differential Equations"

Computational Science Laboratory[*]
Advanced Vehicle Dynamics Laboratory[^]
Robotics & Mechanisms Laboratory[~]
Computer Science & Mechanical Engineering Departments
Virginia Polytechnic Institute and State University
Blacksburg, VA 24061
Phone: (540)-231-2193
Fax: (540)-231-9218
Email: sandu@cs.vt.edu
Web: http://www.eprints.cs.vt.edu

Simultaneous Optimal Uncertainty Apportionment and Robust Design Optimization of Systems Governed by Ordinary Differential Equations

Abstract

The inclusion of uncertainty in design is of paramount practical importance because all real-life systems are affected by it. Designs that ignore uncertainty often lead to poor robustness, suboptimal performance, and higher build costs. Treatment of small geometric uncertainty in the context of manufacturing tolerances is a well studied topic. Traditional sequential design methodologies have recently been replaced by concurrent optimal design methodologies where optimal system parameters are simultaneously determined along with optimally allocated tolerances; this allows to reduce manufacturing costs while increasing performance. However, the state of the art approaches remain limited in that they can only treat geometric related uncertainties restricted to be small in magnitude.

This work proposes a novel framework to perform robust design optimization concurrently with optimal uncertainty apportionment for dynamical systems governed by ordinary differential equations. The proposed framework considerably expands the capabilities of contemporary methods by enabling the treatment of both geometric and non-geometric uncertainties in a unified manner. Additionally, uncertainties are allowed to be large in magnitude and the governing constitutive relations may be highly nonlinear.

In the proposed framework, uncertainties are modeled using Generalized Polynomial Chaos and are solved quantitatively using a least-square collocation method. The computational efficiency of this approach allows statistical moments of the uncertain system to be explicitly included in the optimization-based design process. The framework formulates design problems as constrained multi-objective optimization problems, thus enabling the characterization of a Pareto optimal trade-off curve that is off-set from the traditional deterministic optimal trade-off curve. The Pareto off-set is shown to be a result of the additional statistical moment information formulated in the objective and constraint relations that account for the system uncertainties. Therefore, the Pareto trade-off curve from the new framework characterizes the entire family of systems within the probability space; consequently, designers are able to produce robust and optimally performing systems at an optimal manufacturing cost.

A kinematic tolerance analysis case-study is presented first to illustrate how the proposed methodology can be applied to treat geometric tolerances. A nonlinear vehicle suspension design problem, subject to parametric uncertainty, illustrates the capability of the new framework to produce an optimal design at an optimal manufacturing cost, accounting for the entire family of systems within the associated probability space. This case-study highlights the general nature of the new framework which is capable of optimally allocating uncertainties of multiple types and with large magnitudes in a single calculation.

Keywords: Uncertainty Apportionment, Tolerance Allocation, Robust Design Optimization, Dynamic Optimization, Nonlinear Programming, Multi-Objective Optimization, Multibody Dynamics, Uncertainty Quantification, Generalized Polynomial Chaos

List of Variables (Nomenclature)

Independent variables	
t	Time
ω	Random event
General	
x, X	Non-bolded variables generally indicate a scalar quantity
$\boldsymbol{x}, \boldsymbol{X}$	Bolded lower case variables are vectors, upper case variables are matrices
$\vec{\boldsymbol{x}}$	Alternative vector notation
$\angle \boldsymbol{x}$	Angle of the vector $\boldsymbol{x}$
ξ	Random variable
x_i	Bottom right index *generally* indicates a state (with occasional exceptions).
x^j	Top right index generally indicates a stochastic coefficient, or mode.
${}_k x$	Bottom left index generally associates x to a specific collocation point.
${}_k x_i^j$	The major variable annotations
$(\ )^{\boldsymbol{T}}$	Transpose
$(\ )_{\mathbf{q}}$, $\frac{\partial}{\partial q}$	Partial derivative notations
$(\ )^{-\mathbf{1}}$, $(\ )^{\#}$	Matrix inverse and pseudo inverse
$(\ )^{\circ}$	The quantity represents rotations in degrees
$\underline{x}$, $\overline{x}$	Lower and upper bounds on x
$E[x]$, μ_x	Expected value, or mean, of x
$Var[x]$, σ_x^2	Variance of x
$std[x]$, σ_x	Standard Deviation of x
Indexes & dimensions	
$n_d \in \mathbb{N}$	Number of degrees-of-freedom (DOF)
$n_{gc} \in \mathbb{N}$	Number of generalized coordinates, where $n_{gc} \geq n_d$, dependent on kinematic representation of rotation.
$n_s \in \mathbb{N}$	Number of states, $n_s = (n_{gc} + n_d)$
$n_o \in \mathbb{N}$	Number of outputs, $\boldsymbol{y} \in \mathbb{R}^{n_o}$
$n_i \in \mathbb{N}$	Number of input wrenches, $\boldsymbol{\tau} \in \mathbb{R}^{n_i}$
$n_p \in \mathbb{N}$	Number of uncertain parameters
$p_o \in \mathbb{N}$	Polynomial order
$n_b \in \mathbb{N}$	Number of multidimensional basis terms

$n_{cp} \in \mathbb{N}$	Number of collocation points
$n_x \in \mathbb{N}$	Number of optimization variables ('manipulated variables')
$n_a \in \mathbb{N}$	Number of apportionment variables ('allocated variables'), where $n_a \leq n_p$
$n_c \in \mathbb{N}$	Number of constraint equations
$n_{iv} \in \mathbb{N}$	Number of independent variables
$n_{dv} \in \mathbb{N}$	Number of dependent variables
Tolerance Analysis	
$\boldsymbol{\Phi}^{\mathrm{ol}}$	Open-loop algebraic kinematic constraints defining **P**
$\boldsymbol{\Phi}^{\mathrm{cl}}$	Closed-loop algebraic kinematic constraints
P	Algebraic relations describing kinematic assembly features (e.g., gaps, clearances, positions, etc…)
Dynamics	
$\boldsymbol{q} \in \mathbb{R}^{n_{gc}}$	Independent generalized coordinates
$\dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}}$	Rates and accelerations of generalized coordinates
$\boldsymbol{v} \in \mathbb{R}^{n_d}$	Generalized velocities
$\dot{\boldsymbol{v}}$	Generalized accelerations
$\boldsymbol{q}(\mathbf{0}) = \boldsymbol{q}_0$, $\boldsymbol{v}(\mathbf{0}) = \boldsymbol{v}_0$	Initial conditions
$\boldsymbol{H} \in \mathbb{R}^{n_s \times n_s}$	Kinematic mapping matrix relating rates of generalized coordinates to generalized velocities
$\boldsymbol{\tau} \in \mathbb{R}^{n_i}$	Input wrenches (e.g., forces and/or torques)
$\boldsymbol{M} \in \mathbb{R}^{n_s \times n_s}$	Square inertia matrix
$\boldsymbol{C} \in \mathbb{R}^{n_s}$	Centrifugal, gyroscopic and Coriolis terms
$\boldsymbol{N} \in \mathbb{R}^{n_s}$	Generalized gravitational and joint forces
$\boldsymbol{\mathcal{F}}$	Differential operator
$\boldsymbol{y} \in \mathbb{R}^{n_o}$	System outputs
$\boldsymbol{\mathcal{O}} \in \mathbb{R}^{n_o}$	Output operator
A	Road irregularity amplitude
ϖ	Road irregularity frequency
l	Road irregularity length
v	Longitudinal vehicle speed
Uncertainty Quantification	
Ω	Random event sample space
$f(\xi)$	Joint probability density function
$\boldsymbol{\theta} \in \mathbb{R}^{n_{iv}}$	Vector of uncertain independent variables
$\boldsymbol{\lambda} \in \mathbb{R}^{n_{dv}}$	Vector of uncertain dependent variables
$\psi \in \mathbb{R}^{p_o+1}$	Single dimensional basis terms
$\Psi \in \mathbb{R}^{n_b}$	Multidimensional basis terms
$H_n(x)$	Hermite polynomial basis
$L_n(x)$	Legendre polynomial basis
$\mathcal{N}(\mu, \sigma)$	Normal, or Gaussian, distribution with mean and standard deviation of $[\mu, \sigma]$
$\mathcal{U}(a, b)$	Uniform distribution with range of $[a, b]$
$\mathcal{A}$	Algebraic operator
$\mathfrak{D}$	Differential operator
${}_k\boldsymbol{\mu}$, $\boldsymbol{\mu} \in \mathbb{R}^{n_{cp}}$	K$^{\mathrm{th}}$ collocation point (confusing notation with mean, consider changing…)
${}_k\Lambda_i$, $\Lambda_i \in \mathbb{R}^{n_{cp}}$	K$^{\mathrm{th}}$ intermediate variable of the i$^{\mathrm{th}}$ uncertain dependent variable, λ_i, representing expanded quantity
$A \in \mathbb{R}^{n_b \times n_{cp}}$	Collocation matrix
Dynamic Optimization	
$\min_{\mathbf{x}}$	Optimization objective through manipulation of **x**
x	List of manipulated variables
J	Scalar objective function
w_i	Scalarization weights for the individual objective function terms
$\boldsymbol{\mathcal{C}} \in \mathbb{R}^{n_c}$	Inequality constraints (typically bounding constraints)
a_i	Standard deviation scaling parameters
ς	*Soft* constraint penalty weight

1 INTRODUCTION

The design for manufacturability community has long studied the adverse effects of uncertainty in kinematically assembled mechanical systems; these studies are generally referred to as *tolerance analysis* and *tolerance allocation*. *Tolerance analysis* approaches the problem from the perspective of analyzing uncertainty accumulation in assembled systems. Conversely, *tolerance allocation* determines the best distribution and maximized magnitudes of uncertainties such that the final assembled system satisfies specified assembly constraints. Initially, these studies only treated rigid kinematic relations of the dimensional uncertainties [1-4], but have been extended to include flexible [5] and dynamical systems [6-8].

Early works by Chase and co-workers investigated various techniques for the allocation of geometric tolerances subject to cost-tolerance tradeoff curves. Their approaches included both nonlinear programming (NLP) and analytic Lagrange multiplier solutions for two-dimension (2D) and three-dimension (3D) problems where a number of cost-tolerance models were analyzed and compared [1, 2]. Extensions of their initial work included the

costs of different manufacturing processes for a given feature; thus, an optimal cost-to-build included the optimal set of processes to complete the production of the various components [3].

The conventional design methodology prescribed a sequential approach where optimal dimensions were designed first followed by a *tolerance analysis* or *tolerance allocation* [6]. However, researchers quickly learned that superior designs could be achieved by designing the optimal system parameters concurrently with the allocation of the geometric tolerances [7, 9].

It is the understanding of the authors of this work that these studies have focused solely on the affects of geometric related uncertainties of relatively small magnitude. Conversely, this work presents a novel framework that extends the ability to apportion uncertainties including, but not limited to, geometric related tolerances. Specifically, the new framework simultaneously performs robust design optimization (RDO) and optimal uncertainty apportionment (OUA) of dynamical systems described by linear or nonlinear ordinary differential equations (ODEs). Examples of non-geometric uncertainties include: initial conditions (ICs), sensor and actuator noise, external forcing, and non-geometric related system parameters. Uncertainties from geometric and non-geometric sources may be relatively large in magnitude and are addressed in a unified manner; namely, they are modeled using Generalized Polynomial Chaos (gPC) and are solved quantitatively using a non-intrusive least-square collocation method (LSCM). The computational efficiencies gained by gPC and LSCM enable the inclusion of uncertainty statistics in the optimization process; thus enabling the concurrent OUA and RDO.

The new framework is initially presented in a constrained NLP-based formulation; this formulation is general and independent to the constrained NLP solver approach selected (i.e. gradient versus non-gradient based solvers); however, in the event that an unconstrained solver technique (such as Genetic Algorithms, Differential Evolution, or some other Evolutionary Algorithm) is selected, an unconstrained penalty-based formulation is also provided.

A simple kinematic *tolerance analysis* case study is presented to first illustrate how geometric tolerances may be accounted for through gPC. The benefits of the new framework are then illustrated in an optimal vehicle suspension design case-study where the optimal apportionment of system parameters minimizes the cost-to-build while maintaining optimal and robust performance. The suspension case-study was specifically selected to illustrate the apportionment of non-geometric related uncertainties.

The author's prior work related to RDO [10] and motion planning [11-14] of uncertain dynamical systems may be found in these respective references. These works contain an additional review of literature and information related to uncertainty quantification, gPC, RDO, and motion planning of uncertain systems.

2 UNCERTAINTY QUANTIFICATION

2.1 GENERALIZED POLYNOMIAL CHAOS

Generalized Polynomial Chaos (gPC), first introduced by Wiener [15], is an efficient method for analyzing the effects of uncertainties in second order random processes [16]. This is accomplished by approximating a source of uncertainty, θ, with an infinite series of weighted orthogonal polynomial bases called Polynomial Chaoses. Clearly, an infinite series is impractical; therefore, a truncated set of $p_o + 1$ terms is used with $p_o \in \mathbb{N}$ representing the *order* of the approximation. Or,

$$\theta(\xi) = \sum_{j=0}^{p_o} \theta^j \psi^j(\xi) \tag{1}$$

where $\theta^j \in \mathbb{R}$ are known coefficients; and $\psi^j \in \mathbb{R}$ are individual single dimensional orthogonal basis terms (or modes). The bases are orthogonal with respect to the ensemble average inner product,

$$\langle \psi^i(\xi), \psi^j(\xi) \rangle = \int_\Omega \psi^i(\xi)\psi^j(\xi) f(\xi) d\xi = 0, \quad \text{for i}\neq\text{j} \tag{2}$$

where $\xi(\omega) \in \mathbb{R}$ is a random variable that maps the random event $\omega \in \Omega$, from the sample space, Ω, to the domain of the orthogonal polynomial basis (e.g., $\xi: \Omega \to [-1,1]$); $f(\xi)$ is the weighting function that is equal to the joint probability density function of the random variable. Also, $\langle \Psi^j, \Psi^j \rangle = 1, \forall j$ when using *normalized basis*; *standardized basis* are constant and may be computed off-line for efficiency using (2).

The choice of basis to be used is dependent on the type of statistical distribution that best models a given uncertain parameter. In [16], a family of orthogonal polynomials and statistical distribution pairs was presented. Therefore, gPC allows a designer to pick an appropriate distribution and polynomial pair to model the uncertainty. For example, the *tolerance analysis* community generally models geometric uncertainties with either a Normal, or Gaussian, distribution, denoted by $\mathcal{N}(\mu, \sigma)$, or with a Uniform distribution, denoted by $\mathcal{U}(a, b)$; where μ is the mean, σ is the standard deviation, and a and b are the lower and upper bounds of the distribution range, respectively. When modeling an uncertainty with $\mathcal{N}(\mu, \sigma)$ the corresponding expansion basis is the probabilist's Hermite polynomials, $H_n(\xi)$, and the expansion with known coefficients is,

$$\theta(\xi) = \mu\, H_0(\xi) + \sigma\, H_1(\xi) + 0\, H_2(\xi) + \cdots + 0\, H_{p_o}(\xi) \tag{3}$$

where the domain is $\xi: \Omega \to [-\infty, \infty]$. Similarly, when the uncertainty is better modeled with $\mathcal{U}(a, b)$, then Legendre polynomials are used, $L_n(\xi)$, and the expansion with known coefficients is,

$$\theta(\xi) = \left(a + \frac{b-a}{2}\right) L_0(\xi) + \left(\frac{b-a}{2}\right) L_1(\xi) + 0\, L_2(\xi) + \cdots + 0\, L_{p_o}(\xi) \tag{4}$$

where the domain is $\xi: \Omega \to [-1,1]$. (For more information regarding possible distribution/polynomial pairs the interested reader should refer to [16].)

Any quantity dependent on a source of uncertainty becomes uncertain and can be approximated in a similar fashion as (1),

$$\lambda(\theta(\xi)) = \lambda(\xi) = \sum_{j=0}^{n_b} \lambda^j \Psi^j(\xi) \tag{5}$$

where $\lambda(\xi)$ is an approximated dependent quantity; λ^j are the unknown gPC expansion coefficients; and $n_b \in \mathbb{N}$ is the number of basis terms in the approximation.

The orthogonal basis may be multidimensional in the event that there are multiple sources of uncertainty. The multidimensional basis functions are represented by $\Psi^{j} \in \mathbb{R}^{n_b}$. Additionally, $\boldsymbol{\xi}$ becomes a vector of random variables, $\boldsymbol{\xi} = \{\xi_1, \dots, \xi_{n_p}\} \in \mathbb{R}^{n_p}$, and maps the sample space, Ω, to an n_p dimensional cuboid, $\boldsymbol{\xi}: \Omega \to [-1,1]^{n_p}$ (as in the example of Legendre chaoses).

The multidimensional basis is constructed from a product of the single dimensional basis in the following manner,

$$\Psi^{j} = \psi_1^{i_1} \psi_2^{i_2} \dots \psi_{n_p}^{i_{n_p}}, \quad i_k = 0 \dots p_o, k = 1 \dots n_p \tag{6}$$

where subscripts represent the uncertainty source and superscripts represent the associated basis term (or mode). A complete set of basis may be determined from a full tensor product of the single dimensional bases. This results in an excessive set of $(p_o + 1)^{n_p}$ basis terms. Fortunately, the multidimensional sample space can be spanned with a minimal set of $n_b = (n_p + p_o)!/(n_p!\ p_o!)$ basis terms. The minimal basis set can be determined by the products resulting from these index ranges,

$$\begin{gathered} i_1 = 0 \dots p_o, \\ i_2 = 0 \dots (p_o - i_1), \\ \vdots \\ i_{n_p} = 0 \dots \left(p_o - i_1 - i_2 - \cdots - i_{(n_p - 1)}\right) \end{gathered}$$

The number of multidimensional terms, n_b, grows quickly with the number of uncertain parameters, n_p, and polynomial order, p_o. Sandu et. al. showed that gPC is most appropriate for modeling systems with a relatively low number of uncertainties [17, 18] but can handle large nonlinear uncertainty magnitudes.

Once all sources of uncertainty, $\theta(\xi)$, and dependent quantities, $\lambda(\xi)$, have been expanded then the constitutive relations defining a given problem may be updated. For example, constitutive relations may be algebraic or differential equations,

$$\mathcal{A}(\lambda(\xi), \theta(\xi)) = 0 \tag{7}$$

$$\dot{\lambda}(t; \xi) = \mathfrak{D}(\lambda(t; \xi), \theta(\xi)) \tag{8}$$

Equation (7) is an implicit algebraic constitutive relation and (8) is a differential constitutive relation. It is important to note that the dependent quantities are functions of time in (8), therefore, (5) is modified to,

$$\lambda(t, \theta(\xi)) = \lambda(t; \xi) = \sum_{j=0}^{n_b} \lambda^{j}(t)\ \Psi^{j}(\xi) \tag{9}$$

It is instructive to notice how time and randomness are decoupled within a single term of the gPC expansion. Only the expansion coefficients are dependent on time, and only the basis terms are dependent on the n_b random variables, $\boldsymbol{\xi}$. If any sources of uncertainty are also functions of time then (1) needs to be updated in a similar fashion as (9) and then all dependent quantities will have to be expanded using (9). Without loss of generality, the proceeding presentation will assume that all sources of uncertainty are time-independent to simplify the notation.

Substituting the appropriate expansions from (1), (5), or (9) into the constitutive relations results in uncertain constitutive equations,

$$\mathcal{A}\left(\sum_{j=0}^{n_b} \lambda^{j}(t)\ \Psi^{j}(\xi), \sum_{j=0}^{p_o} \theta^{j} \psi^{j}(\xi)\right) = 0 \tag{10}$$

$$\sum_{j=0}^{n_b} \dot{\lambda}^{j}(t; \xi)\ \Psi^{j}(\xi) = \mathfrak{D}\left(\sum_{j=0}^{n_b} \lambda^{j}(t)\ \Psi^{j}(\xi), \sum_{j=0}^{p_o} \theta^{j} \psi^{j}(\xi)\right) \tag{11}$$

where the n_b expansion coefficients, λ^{j} or $\lambda^{j}(t)$, from the dependent quantities are the unknowns to be solved for.

There are a number of methods in the literature for solving equations such as (10)–(11). The Galerkin Projection Method (GPM) is a commonly used method; however, this is a very intrusive technique and requires a custom reformulation of (10)–(11). As an alternative, sample-based collocation techniques can be used without the need to modify the base equations.

Sandu et. al. [18, 19] showed that the collocation method solves equations such as (10)–(11) by solving (7)–(8) at a set of points, ${}_k\boldsymbol{\mu} \in \mathbb{R}^{n_p}$, $k = 1 \dots n_{cp}$, selected from the n_p dimensional domain of the random variables $\boldsymbol{\xi} \in \mathbb{R}^{n_p}$. Meaning, at any given instance in time, the random variables' domain is sampled and solved n_{cp} times with $\boldsymbol{\xi} = {}_k\boldsymbol{\mu}$ (updating the approximations of all sources of uncertainty for each solve), then the uncertain coefficients of the dependent quantities can be determined. This can be accomplished by defining intermediate variables such as,

$$\Lambda({}_k\boldsymbol{\mu}) = \sum_{j=0}^{n_b} \lambda^{j} \Psi^{j}(\xi), \quad k = 0 \dots n_{cp} \tag{12}$$

Substituting the appropriate intermediate variables into (10) and (11) respectively yields,

$$\mathcal{A}\left(\Lambda({}_k\boldsymbol{\mu}), {}_k\Theta_r({}_k\boldsymbol{\mu})\right) = 0 \tag{13}$$

$$\dot{\Lambda}({}_k\boldsymbol{\mu}) = \mathfrak{D}\left(\Lambda({}_k\boldsymbol{\mu}), {}_k\Theta_r({}_k\boldsymbol{\mu})\right) \tag{14}$$

where $k = 0 \dots n_{cp}, r = 1 \dots n_p$, and each uncertainty's intermediate variable is,

$${}_k\Theta_r({}_k\boldsymbol{\mu}) = \sum_{j=0}^{p_o} \theta_r^{j}\ \psi^{j}({}_k\boldsymbol{\mu}) \tag{15}$$

Equations (13)–(14) provide a set of n_{cp} independent equations whose solutions determine the uncertain expansion coefficients of the dependent quantities. Since (13) is implicitly defined, there are two options in determining $\Lambda({}_k\boldsymbol{\mu})$: use a numerical nonlinear system solver such as Newton-

Raphson, or, solve (7) for a new relation that defines $\Lambda\left({}_k\boldsymbol{\mu}\right)$ explicitly. The uncertain expansion coefficients of the dependent quantities are determined by recalling the relationship of the expansion coefficients to the solutions as in (12). In matrix notation, (12) can be expressed as,

$$ {}_k\Lambda = \boldsymbol{\Psi}\left({}_k\boldsymbol{\mu}\right)^{\mathrm{T}}\boldsymbol{\lambda} \tag{16}$$

where the matrix,

$$ A_{k,j} = \Psi^j\left({}_k\mu\right),\ \ j = 0 \ldots n_b, k = 0 \ldots n_{cp} \tag{17}$$

is defined as the *collocation matrix*. It's important to note that $n_b \leq n_{cp}$. The expansion coefficients can now be solved for using (16),

$$ \boldsymbol{\lambda} = \boldsymbol{A}^{\#}\boldsymbol{\Lambda} \tag{18}$$

where $A^{\#}$ is the pseudo inverse of A if $n_b < n_{cp}$. If $n_b = n_{cp}$, then (18) is simply a linear solve. However, [19-23] presented the least-squares collocation method (LSCM) where the stochastic dependent variable coefficients are solved for, in a least squares sense, using (18) when $n_b < n_{cp}$. Reference [19] also showed that as $n_{cp} \to \infty$ the LSCM approaches the GPM solution; by selecting $3n_b \leq n_{cp} \leq 4n_b$ the greatest convergence benefit is achieved with minimal computational cost. LSCM also enjoys the same exponential convergence rate as $p_o \to \infty$.

The nonintrusive nature of the LSCM sampling approach is arguably its greatest benefit; (7) or (8) may be repeatedly solved without modification. Also, there are a number of methods for selecting the collocation points and the interested reader is recommended to consult [18, 19, 24-26] for more information.

Once the expansion coefficients of the dependent quantities are determined then statistical moments, such as the expected value and variance, can be efficiently calculated. Arguably the greatest benefit of modeling uncertainties with gPC is the computational efficiencies gained when calculating the various statistical moments of the dependent quantities. For example, [27] defines the statistical expected value as,

$$ \mu_\lambda = E[\lambda(\boldsymbol{\xi})] = \int_{\Omega} \lambda(\boldsymbol{\xi})\, w(\boldsymbol{\xi}) d\boldsymbol{\xi} \tag{19}$$

and the variance,

$$ \sigma_\lambda^2 = Var[\lambda(\boldsymbol{\xi})] = \int_{\Omega} (\lambda(\boldsymbol{\xi}) - \mu_\lambda)^2\, w(\boldsymbol{\xi}) d\boldsymbol{\xi} = E[(\lambda(\boldsymbol{\xi}) - \mu_\lambda)^2] \tag{20}$$

with the standard deviation, $\sigma_\lambda = \sqrt{Var[\lambda(\boldsymbol{\xi})]}$. Given these definitions and leveraging the orthogonality of the gPC basis, these moments may be efficiently computed by a reduced set of arithmetic operations of the expansion coefficients,

$$ \mu_\lambda = E[\lambda(\boldsymbol{\xi})] = \lambda^0 \langle \Psi^0, \Psi^0 \rangle \tag{21}$$

$$ \sigma_\lambda = \sqrt{E[(\lambda(\boldsymbol{\xi}) - \mu_\lambda)^2]} = \sqrt{\sum_{j=1}^{n_b} (\lambda^j)^2 \langle \Psi^j, \Psi^j \rangle} \tag{22}$$

Also, recall that $\langle \Psi^j, \Psi^j \rangle = 1, \forall j$ when using *normalized basis*; *standardized basis* are constant and may be computed off-line for efficiency using (2). A number of efficient statistical moments may be determined from the expansion coefficients. The authors presented a number of gPC based measures using efficient moments such as (21)–(22) in [10-14].

To summarize, the following basic steps are taken in order to model uncertainty and solve for statistical moments of quantities dependent on uncertainties:

1. Model all sources of uncertainty by associating an appropriate probability density function (PDF).
2. Expand all sources of uncertainty with an appropriate single dimensional orthogonal polynomial basis. The known expansion coefficients are determined from the PDF modeling the uncertainty.
3. Expand all dependent quantities with an appropriately constructed multi-dimensional basis.
4. Update constitutive relations with the expansions from steps 2–3. The new unknowns are now the expansion coefficients from the dependent quantities.
5. Solve the uncertain constitutive relations for the unknown expansion coefficients; this work uses the LSCM technique.
6. Calculate appropriate statistical moments from the expansion coefficients.

The following two sections summarize this material in the context of: uncertain kinematic assemblies, and uncertain dynamical systems.

2.2 UNCERTAIN KINEMATIC ASSEMBLIES

Generalized Polynomial Chaos may be employed for *tolerance analysis*, where the effects of geometric uncertainties in kinematically assembled systems are quantified. The assembly features to be analyzed may be defined through an explicit algebraic constitutive relation such as,

$$ \mathbf{P} = \boldsymbol{\Phi}^{\mathrm{ol}}(\boldsymbol{\lambda}, \boldsymbol{\theta}) \tag{23}$$

where **P** represents assembly relations such as gaps, positions, and orientations of subcomponent features [28]. The dependent assembly variables, $\boldsymbol{\lambda}$, must satisfy closed kinematic constraints, if any; these may be implicitly defined as was shown in (7),

$$ \boldsymbol{\Phi}^{\mathrm{cl}}(\boldsymbol{\lambda}, \boldsymbol{\theta}) = \mathbf{0} \tag{24}$$

After solving (24) then any assembly feature defined by (23) may be evaluated.

Once the independent kinematic features, or contributing dimensions, have manufacturing tolerances prescribed, $\theta_i(\boldsymbol{\xi})$, then (23)–(24) become uncertain constitutive relations. Solving the uncertain versions of (24) yields the uncertain dependent assembly variables, $\boldsymbol{\lambda}(\boldsymbol{\xi})$, and (23) may then be solved for the uncertain assembly features, $\mathbf{P}(\boldsymbol{\xi})$.

The procedure defined in Section 2.1 for algebraic constitutive relations allows designers to calculate statistical moments of the dependent assembly features and quantify the effects of the prescribed dimension tolerances. A simple *tolerance analysis* example based on gPC is presented in Section 4.

2.3 UNCERTAIN DYNAMICAL SYSTEMS

Generalized Polynomial Chaos has previously been shown to be an effective tool in quantifying uncertainty within RDO [10] and robust motion control [11-14] settings. This work presents a new framework that enables RDO concurrently with OUA of dynamical systems described by ODEs. The new framework is not dependent on a specific formulation of the equations-of-motion (EOMs); formulations such as Newtonian, Lagrangian, Hamiltonian, and Geometric methodologies are all applicable. An Euler-Lagrange ODE formulation may describe a dynamical system [29, 30] by,

$$\begin{aligned}\boldsymbol{M}(\boldsymbol{q}(t),\boldsymbol{\theta})\dot{\boldsymbol{v}}(t)+\boldsymbol{C}(\boldsymbol{q}(t),\boldsymbol{v}(t),\boldsymbol{\theta})\boldsymbol{v}(t)+\boldsymbol{N}(\boldsymbol{q}(t),\boldsymbol{v}(t),\boldsymbol{\theta})\\ =\boldsymbol{\mathcal{F}}(\boldsymbol{q}(t),\boldsymbol{v}(t),\dot{\boldsymbol{v}}(t),\boldsymbol{\theta})=\boldsymbol{\tau}(t)\end{aligned} \tag{25}$$

where $\boldsymbol{q}(t)\in\mathbb{R}^{n_{gc}}$ are the generalized coordinates with $n_{gc}\geq n_d$; $\boldsymbol{v}(t)\in\mathbb{R}^{n_d}$ are the generalized velocities and—using Newton's *dot* notation—$\dot{\boldsymbol{v}}(t)$ contains their time derivatives; $\boldsymbol{\theta}\in\mathbb{R}^{n_p}$ includes system parameters of interest; $M(\boldsymbol{q}(t),\boldsymbol{\theta})\in\mathbb{R}^{n_d\times n_d}$ is the square inertia matrix; $\boldsymbol{C}(\boldsymbol{q}(t),\boldsymbol{v}(t),\boldsymbol{\theta})\in\mathbb{R}^{n_d\times n_d}$ includes centrifugal, gyroscopic, and Coriolis effects; $\boldsymbol{N}(\boldsymbol{q}(t),\boldsymbol{v}(t),\boldsymbol{\theta})\in\mathbb{R}^{n_d}$ are the generalized gravitational and joint forces; and $\boldsymbol{\tau}(t)\in\mathbb{R}^{n_i}$ are the n_i applied wrenches (e.g., forces or torques). (For notational brevity, all future equations will drop the explicit time dependence.)

The relationship between the time derivatives of the independent generalized coordinates and the generalized velocities is,

$$\dot{\boldsymbol{q}}=\boldsymbol{H}(\boldsymbol{q},\boldsymbol{\theta})\boldsymbol{v} \tag{26}$$

where $\boldsymbol{H}(\boldsymbol{q},\boldsymbol{\theta})$ is a skew-symmetric matrix that is a function of the selected kinematic representation (e.g., Euler Angles, Tait-Bryan Angles, Axis-Angles, Euler Parameters, etc.) [13, 31, 32]. However, if (25) is formulated with independent generalized coordinates and the system has a fixed base then (26) becomes $\dot{\boldsymbol{q}}=\boldsymbol{v}$.

The trajectory of the system is determined by solving (25)–(26) as an initial value problem, where $\boldsymbol{q}(0)=\boldsymbol{q}_0$ and $\boldsymbol{v}(0)=\boldsymbol{v}_0$. Also, the system measured outputs are defined by,

$$\boldsymbol{y}=\boldsymbol{\mathcal{O}}(\boldsymbol{q},\dot{\boldsymbol{q}},\boldsymbol{\theta}) \tag{27}$$

where $\boldsymbol{y}\in\mathbb{R}^{n_o}$ with n_o equal to the number of outputs. One helpful observation is that the dynamic outputs defined in (27) are analogous to the kinematic assembly features in (23); they are both functions of the dependent quantities of their respective systems defined in (24) and (25)–(26), respectively.

The EOMs defined in (25)–(26) have the form of the differential constitutive relations defined in (8). Any uncertainties in ICs, actuator inputs, sensor outputs, or system parameters are accounted for by $\boldsymbol{\theta}(\boldsymbol{\xi})$. All system states, $\{\boldsymbol{q},\boldsymbol{v}\}$, and associated outputs, $\boldsymbol{y}$, are dependent quantities represented by $\boldsymbol{\lambda}(\boldsymbol{\xi})$.

The procedure defined in Section 2.1 for differential constitutive relations allows designers to calculate statistical moments of the dynamic states and outputs, thus quantifying the effects of the system uncertainties over the trajectory of the system. The new framework presented in Section 3 will build upon this formulation to enable RDO concurrently with OUA of dynamical systems described by ODEs.

3 OPTIMAL DESIGN AND UNCERTAINTY APPORTIONMENT

The new framework for simultaneous RDO and OUA of dynamical systems described by ODEs is now presented. This formulation builds upon the gPC-based uncertainty quantification techniques presented in Sections 2.1 and 2.3. Sources of uncertainty may come from ICs, actuator inputs, sensor outputs, and system parameters; where parametric uncertainties may include both geometric and non-geometric sources. The NLP-base formulation of the new framework is,

$$\begin{aligned}\min_{\boldsymbol{x}}\quad & \mathrm{J}=\boldsymbol{\mathcal{W}}\big(\boldsymbol{\sigma}(\boldsymbol{\xi})\big)\\ \text{s.t.}\quad & \boldsymbol{\mathcal{F}}\big(\boldsymbol{q}(t;\boldsymbol{\xi}),\boldsymbol{v}(t;\boldsymbol{\xi}),\dot{\boldsymbol{v}}(t;\boldsymbol{\xi}),\boldsymbol{\theta}(t;\boldsymbol{\xi})\big)=\boldsymbol{\tau}(t;\boldsymbol{\xi})\\ & \dot{\boldsymbol{q}}(t;\boldsymbol{\xi})=\boldsymbol{H}\big(\boldsymbol{q}(t;\boldsymbol{\xi}),\boldsymbol{\theta}(t;\boldsymbol{\xi})\big)\boldsymbol{v}(t;\boldsymbol{\xi})\\ & \boldsymbol{y}(t;\boldsymbol{\xi})=\boldsymbol{\mathcal{O}}\big(\boldsymbol{q}(t;\boldsymbol{\xi}),\dot{\boldsymbol{q}}(t;\boldsymbol{\xi}),\boldsymbol{\theta}(t;\boldsymbol{\xi})\big)\\ & \boldsymbol{C}(\boldsymbol{y}(t;\boldsymbol{\xi}),\boldsymbol{\theta}(t;\boldsymbol{\xi}),\boldsymbol{\tau}(t;\boldsymbol{\xi}),t)\leq\boldsymbol{0}\\ & \boldsymbol{q}(0;\boldsymbol{\xi})=\boldsymbol{q}_0,\boldsymbol{q}\big(t_f;\boldsymbol{\xi}\big)=\boldsymbol{q}_{t_f}\\ & \dot{\boldsymbol{q}}(0;\boldsymbol{\xi})=\dot{\boldsymbol{q}}_0,\dot{\boldsymbol{q}}\big(t_f;\boldsymbol{\xi}\big)=\dot{\boldsymbol{q}}_{t_f}\end{aligned} \tag{28}$$

where the problem objective, J, is a weighted vector function, $\boldsymbol{\mathcal{W}}\in\mathbb{R}^{n_a}$, defining n_a cost-uncertainty trade-off curves for the manufacturing cost associated with each uncertainty being apportioned. Equation (28) is subject to the dynamic constitutive relations defined in (25)–(27), and their associated ICs and optional terminal conditions (TCs). When performing simultaneous RDO and OUA, the list of optimization variables, $\boldsymbol{x}\in\mathbb{R}^{n_x}$, includes select nominal design parameters as well as variances, or standard deviations, of the uncertainties to be apportioned. Concurrent RDO and OUA is possible by converting the robust performance objectives of RDO to constraints and adding them to the list of problem constraints itemized in $\boldsymbol{C}(\boldsymbol{y},\boldsymbol{\theta},\boldsymbol{\tau},t)\leq\boldsymbol{0}$. The authors' work in [10] illustrates how robust performance objectives may be defined within a gPC setting. Therefore, the solution of (28) yields a system design that minimizes the manufacturing cost-to-build subject to specified robust performance criterion defined through constraints. Equation (28) is formulated as a *constrained multi-objective optimization* (cMOO) problem, meaning, as long as at least two performance constraints have opposing influences on the optimum, then a Pareto optimal set may be determined as the constraint boundaries are adjusted. For example, if each robust performance constraint, β, is bounded in the following manner $\underline{\beta}\leq\beta\leq\overline{\beta}$. A Pareto set will be obtained for unique values of $\overline{\beta}$ and/or $\underline{\beta}$ as long as the associated constraint is active. Once a given constraint becomes inactive, that constraint has no influence on the optimal value.

The NLP defined in (28) may be approached from either a *sequential nonlinear programming* (SeqNLP), or from a *simultaneous nonlinear programming* (SimNLP) perspective [33, 34]. (The literatures occasionally refers to the SeqNLP approach as *partial discretization* and to the SimNLP as *full discretization* [35].) In the SeqNLP approach, the dynamical equations (25)–(27) remain as continuous functions that may be integrated with standard off-the-shelf ODE solvers (such as Runge-Kutta). This directly leverages the LSCM-based gPC techniques described in Section 2.1 and yields a smaller optimization problem as only the optimization variables, $\boldsymbol{x}$, are discretized. On the contrary, the SimNLP approach discretizes (25)–(27) over the trajectory of the system and treats the complete set of equations as equality constraints for the NLP. The discretized

state variables are added to $\boldsymbol{x}$ to complete the full discretization. As such, the SimNLP approach requires a slight modification in the formulation to account for the full discretization of (25)–(27) in light of the LSCM technique:

$$\begin{aligned}
\min_{\boldsymbol{x}} \quad & \mathrm{J} = \boldsymbol{\mathcal{W}}\big(\boldsymbol{\sigma}(\boldsymbol{\xi})\big) \\
\text{s.t.} \quad & \boldsymbol{\mathcal{F}}\Big(\boldsymbol{q}\big({}_1\boldsymbol{\mu}\big), \boldsymbol{v}\big({}_1\boldsymbol{\mu}\big), \dot{\boldsymbol{v}}\big({}_1\boldsymbol{\mu}\big), \boldsymbol{\theta}\big({}_1\boldsymbol{\mu}\big)\Big) = \boldsymbol{\tau}\big({}_1\boldsymbol{\mu}\big) \\
& \dot{\boldsymbol{q}}\big({}_1\boldsymbol{\mu}\big) = \boldsymbol{H}\Big(\boldsymbol{q}\big({}_1\boldsymbol{\mu}\big), \boldsymbol{\theta}\big({}_1\boldsymbol{\mu}\big)\Big)\boldsymbol{v}\big({}_1\boldsymbol{\mu}\big) \\
& \boldsymbol{y}\big({}_1\boldsymbol{\mu}\big) = \boldsymbol{\mathcal{O}}\Big(\boldsymbol{q}\big({}_1\boldsymbol{\mu}\big), \dot{\boldsymbol{q}}\big({}_1\boldsymbol{\mu}\big), \boldsymbol{\theta}\big({}_1\boldsymbol{\mu}\big)\Big) \\
& \boldsymbol{q}\big(0;\, {}_1\boldsymbol{\mu}\big) = \boldsymbol{q}_{0,1}, \dot{\boldsymbol{q}}\big(0;\, {}_1\boldsymbol{\mu}\big) = \dot{\boldsymbol{q}}_{0,1} \\
& \qquad\qquad \vdots \\
& \boldsymbol{\mathcal{F}}\Big(\boldsymbol{q}\big({}_{n_{cp}}\boldsymbol{\mu}\big), \boldsymbol{v}\big({}_{n_{cp}}\boldsymbol{\mu}\big), \dot{\boldsymbol{v}}\big({}_{n_{cp}}\boldsymbol{\mu}\big), \boldsymbol{\theta}\big({}_{n_{cp}}\boldsymbol{\mu}\big)\Big) = \boldsymbol{\tau}\big({}_{n_{cp}}\boldsymbol{\mu}\big) \\
& \dot{\boldsymbol{q}}\big({}_{n_{cp}}\boldsymbol{\mu}\big) = \boldsymbol{H}\Big(\boldsymbol{q}\big({}_{n_{cp}}\boldsymbol{\mu}\big), \boldsymbol{\theta}\big({}_{n_{cp}}\boldsymbol{\mu}\big)\Big)\boldsymbol{v}\big({}_{n_{cp}}\boldsymbol{\mu}\big) \\
& \boldsymbol{y}\big({}_{n_{cp}}\boldsymbol{\mu}\big) = \boldsymbol{\mathcal{O}}\Big(\boldsymbol{q}\big({}_{n_{cp}}\boldsymbol{\mu}\big), \dot{\boldsymbol{q}}\big({}_{n_{cp}}\boldsymbol{\mu}\big), \boldsymbol{\theta}\big({}_{n_{cp}}\boldsymbol{\mu}\big)\Big) \\
& \boldsymbol{q}\big(0;\, {}_{n_{cp}}\boldsymbol{\mu}\big) = \boldsymbol{q}_{0,n_{cp}}, \dot{\boldsymbol{q}}\big(0;\, {}_{n_{cp}}\boldsymbol{\mu}\big) = \dot{\boldsymbol{q}}_{0,n_{cp}} \\
& \boldsymbol{\mathcal{C}}(\boldsymbol{y}(\boldsymbol{\xi}), \boldsymbol{\theta}(\boldsymbol{\xi}), \boldsymbol{\tau}(\boldsymbol{\xi}), t) \leq \mathbf{0}
\end{aligned} \tag{29}$$

Equation (29) duplicates the deterministic dynamical equations (25)–(27) n_{cp} times where each set has a unique collocation point, ${}_k\boldsymbol{\mu}$. Each unique set of dynamical equations is then fully discretized and $\boldsymbol{x}$ is updated appropriately. However, the system constraints, $\boldsymbol{\mathcal{C}}(\boldsymbol{y}(\boldsymbol{\xi}), \boldsymbol{\theta}(\boldsymbol{\xi}), \boldsymbol{\tau}(\boldsymbol{\xi}), t) \leq \mathbf{0}$, are calculated using the statistical properties determined by the LSCM and the n_{cp} sets of dynamical equations. Thus, the SimNLP approach has a much larger set of constraints and optimization variables than the SeqNLP approach, but, enjoys a more structured NLP that typically experiences faster convergence.

The Direct Search (DS) class of optimization solvers—techniques such as Genetic Algorithms, Differential Evolution, and Particle Swarm—typically only treat unconstrained optimization problems. To use this kind of solver, all the inequality constraints in (28) need to be converted from *hard* constraints to *soft* constraints; where *hard* constraints are explicitly defined as shown in (28), and constraints that are added to the definition of the objective function, J, are referred to as *soft* constraints. This is accomplished by additional objective penalty terms of the form,

$$\mathrm{J}_{\mathrm{const}}(\mathrm{t}; \boldsymbol{\xi}) = \sum_{i=1}^{n_c} \varsigma \; max\; (0, \mathcal{C}_i(\mathrm{t}; \boldsymbol{\xi}))^2 \tag{30}$$

where n_c represents the number of system constraints, and ς is a large constant. With a large ς, this relationship is analogous to an inequality like penalizing term, meaning, if the constraint $\mathcal{C}_i$ is outside of its bounds—or outside the feasible region—then it's heavily penalized. When it's within the feasible region there is no penalty. Also, by squaring the *max* function its discontinuity is smoothed out; however, this is an optional feature and only necessary for a solver that uses gradient information.

Once the inequality constraints have been converted to objective penalty terms, equation (28) can be reformulated as,

$$\begin{aligned}
\min_{\boldsymbol{x}} \quad & \mathrm{J} = \boldsymbol{\mathcal{W}}\big(\boldsymbol{\sigma}(\boldsymbol{\xi})\big) \; + \mathrm{J}_{\mathrm{const}}(\mathrm{t}; \boldsymbol{\xi}) \\
\text{s.t.} \quad & \boldsymbol{\mathcal{F}}\big(\boldsymbol{q}(t; \boldsymbol{\xi}), \boldsymbol{v}(t; \boldsymbol{\xi}), \dot{\boldsymbol{v}}(t; \boldsymbol{\xi}), \boldsymbol{\theta}(t; \boldsymbol{\xi})\big) = \boldsymbol{\tau}(t; \boldsymbol{\xi}) \\
& \dot{\boldsymbol{q}}(t; \boldsymbol{\xi}) = \boldsymbol{H}\big(\boldsymbol{q}(t; \boldsymbol{\xi}), \boldsymbol{\theta}(t; \boldsymbol{\xi})\big)\boldsymbol{v}(t; \boldsymbol{\xi}) \\
& \boldsymbol{y}(t; \boldsymbol{\xi}) = \boldsymbol{\mathcal{O}}\big(\boldsymbol{q}(t; \boldsymbol{\xi}), \dot{\boldsymbol{q}}(t; \boldsymbol{\xi}), \boldsymbol{\theta}(t; \boldsymbol{\xi})\big) \\
& \boldsymbol{q}(0; \boldsymbol{\xi}) = \boldsymbol{q}_0, \dot{\boldsymbol{q}}(0; \boldsymbol{\xi}) = \dot{\boldsymbol{q}}_0
\end{aligned} \tag{31}$$

where the equality constraints from the continuous dynamics are implicit in the calculation of the objective function. This SeqNLP approach enables (31) to be solved by the DS class of unconstrained solvers.

The new framework presented in (28), (29) or (31) allows designers to directly treat the effects of modeled uncertainties during a concurrent RDO and OUA design process. The formulations are independent of the optimization solver selected; meaning, if a constrained NLP solver—such as *sequential quadratic programming* (SQP) or *interior point* (IP)—is selected, then any of the three formulations presented is appropriate, depending upon the designer's preferences regarding *hard/soft* constraint definition and partial/full discretization. On the contrary, if an unconstrained solver is selected, then (31) is the formulation of choice.

The computational efficiencies of gPC enable the inclusion of statistical moments in the OUA objective function definition as well as in the RDO constraint specifications; these statistical measures are available at a reduced computational cost as compared to contemporary techniques. However, the framework does introduce an additional layer of modeling and computation [12]. In [10], the authors presented general guidelines of when to apply the framework for RDO problems. From an OUA perspective, the following general guidelines can help determine if a given design will benefit from the new framework:

1. **Non-Geometric Uncertainties:** Traditional *tolerance allocation* techniques have been developed for the apportionment of geometric related uncertainties. The new framework provides a unified framework that enables the simultaneous apportionment of both geometric and non-geometric related uncertainties simultaneously in dynamical systems.
2. **Large Magnitude Uncertainties:** Again, traditional *tolerance allocation* techniques have been developed under the assumption that the uncertainty magnitudes are sufficiently small. This assumption is generally valid for geometric manufacturing related uncertainties, however, it may not be valid for non-geometric related uncertainties. This point is illustrated in the case study presented in Section 5.
3. **Simultaneous RDO/OUA Design:** As mentioned in Section 1, the research community has already found that concurrent optimal design and tolerance allocation yields a superior design than the traditional sequential optimization approach. However, the concurrent design studies to date have only treated geometric uncertainties; the new framework in (28), (29) or (31) enables RDO concurrently with OUA and treats non-geometric uncertainties in addition to the geometric.

In what follows, Section 4 presents a *tolerance analysis* example for a simple assembly described by kinematic algebraic constraints. The purpose of Section 4 is to show that the gPC framework is immediately applicable to *tolerance analysis* problems. Section 5 show-cases the new framework for simultaneous RDO and OUA when applied to a vehicle suspension subject to non-geometric uncertainties.

4 TOLERANCE ANALYSIS BASED ON GPC

This section illustrates gPC-based *tolerance analysis* using a one-way mechanical clutch found in a lawn mower or some other small machinery; the problem definition was borrowed from [36] and will be used for comparison purposes. The clutch assembly, shown in Figure 1, is comprised of an outer ring, a hub, and a roller bearing in a close-loop kinematic relation. Leveraging the system symmetry, the problem considers explicitly only a quarter of the mechanism; its independent assembly variables are $\boldsymbol{\theta} = \{a, c, d, e\}$ and dependent variables are $\boldsymbol{\lambda} = \{b, \chi, \gamma\}$. The basic goal of the *tolerance analysis* is to ensure that the pressure angle remains within the specified range of $6^\circ \le \chi \le 8^\circ$.

The assembly feature being analyzed, the pressure angle χ, is a dependent variable, therefore, there is no need for an explicit assembly feature relation as defined in (23), only the closed-loop kinematic constraint equation (24) is necessary. This two dimensional vector relation is:

$$\boldsymbol{\Phi}(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}, \boldsymbol{d}, \boldsymbol{e}, \chi, \gamma) = \begin{cases} \vec{\boldsymbol{a}} + \vec{\boldsymbol{b}} + \vec{\boldsymbol{c}} + \vec{\boldsymbol{d}} + \vec{\boldsymbol{e}} = \boldsymbol{0} \\ \angle\boldsymbol{a} + \angle\boldsymbol{b} + \angle\boldsymbol{c} + \angle\boldsymbol{d} + \angle\boldsymbol{e} + \chi + \gamma = 0 \end{cases} \tag{32}$$

resulting in three equations for the three dependent unknowns.

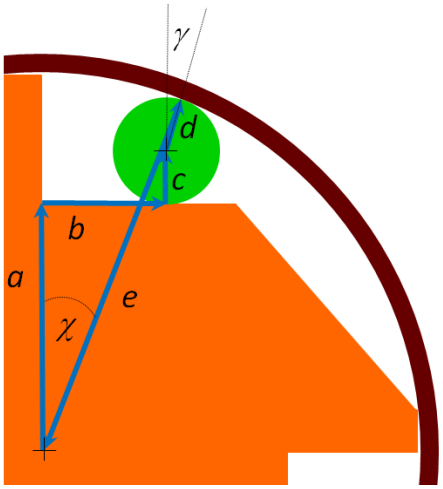

Figure 1—One quarter of a clutch assembly. The independent variables are dimensions $\{a, c, d, e\}$ and the dependent variables are $\{b, \chi, \gamma\}$. Proper operation of the clutch requires $6^\circ \le \chi \le 8^\circ$.

The independent variables, or dimensions, are assigned tolerances where each is assumed to have a normal distribution; their respective mean and standard deviation values are presented in Table 1.

Table 1—Independent variable mean and standard deviations

Parameter	Mean (μ)	Std (σ)	Units (SI)
a	27.645	0.0167	mm
c, d	11.430	0.0033	mm
e	50.800	0.0042	mm

The analysis proceeds by applying the procedure outlined in Section 2.1, where each independent variable $\theta_i(\boldsymbol{\xi})$ is approximated as shown in (1); uncertain dependent variables $\boldsymbol{\lambda}(\boldsymbol{\xi})$ are expanded using (5); and all approximations are substituted into (32) resulting in uncertain algebraic constitutive relations. The LSCM method samples the probability space n_{cp} times and solves for the dependent variable expansion coefficients through least-squares using (18). The respective mean and standard deviation statistical moments are then efficiently determined by (21)–(22). The results of this analysis for various gPC approximation orders are shown in Table 2.

Table 2—Various results for the $3\sigma_\chi$ one-way clutch pressure angle variation

Parameter	Variation (3σ)	$\lvert\Delta_{MC}\rvert$	# of samples, or collocation points	Computation time (s)
DLM	0.65788	0.00135	n/a	0.06
gPC, $p_o = 2$	0.65822	0.00101	30	0.34
gPC, $p_o = 3$	0.65900	0.00023	60	0.51
gPC, $p_o = 5$	0.65901	0.00022	168	2.73
MC	0.65923	0	2,500,000	217.42

Table 2 also contains the results of the analysis when applying the direct linearization method (DLM), as used in [1, 36], as well as a Monte Carlo (MC) based analysis using 2.5 million samples. Additionally, Table 2 reports the absolute value of the errors between the solution of the different methods when compared to the MC results, $|\Delta_{MC}|$, in the third column; the number of samples used are shown in the fourth column; and the associated computation times are shown the fifth column. All methods validate that the pressure angle remains within the specified range; for example, the DLM method had the largest reported variation and its $3\sigma_\chi$ solution had a range of $6.3605^\circ \le \chi \le 7.6763^\circ$, which is within specification.

Using the high sample MC solution as a baseline for comparison shows that the 2nd order gPC solution results in comparable results to the DLM. The 3rd order gPC solution seems to be the best approximation point when considering both computational cost and accuracy. The reported computation times were taken from an unoptimized Mathematica code running on an HP Pavilion with the Intel i7 processor and 6 GB of RAM.

Clearly the gPC approach is more computationally burdensome than the DLM, however, when considering the more general nature of the gPC approach—as discussed in Section 2.1 and Section 3—and the relatively cheap cost to increase the accuracy of the solution beyond what the DLM can provide, a designer may find that this trade-off is worth the expense. In Section 5, the benefits of the gPC approach become more apparent when large magnitude variations and non-geometric uncertainties are included within a problem's scope.

5 AN ILLUSTRATIVE CASE-STUDY OF CONCURRENT RDO & OUA

This section illustrates the benefits of the new framework presented in Section 3 through a vehicle suspension design case-study where RDO and OUA are carried out concurrently. The case-study showcases OUA for a nonlinear system subject to large magnitude non-geometric uncertainties.

A brief overview of the system dynamics follows to help clarify the concurrent RDO/OUA design presented thereafter; however, the interested reader may consult [10] where more detailed information is presented to explain the cMOO formulation that is used to carry out both a deterministic and RDO of the suspension. This work focuses on illustrating OUA when performed simultaneously with RDO.

5.1 VEHICLE SUSPENSION MODEL

The case-study uses an idealized two degree-of-freedom (DOF) nonlinear quarter-car suspension model which is shown in Figure 2. When all the system parameters are known exactly, the system is described by the following deterministic nonlinear dynamical EOMs,

$$\begin{aligned} \ddot{z}_s &= -\frac{k_s}{m_s}(z_s - z_u) - \frac{1}{m_s}\mathcal{D}(\dot{z}_s, \dot{z}_u) \\ \ddot{z}_u &= \frac{k_s}{m_u}(z_s - z_u) + \frac{1}{m_u}\mathcal{D}(\dot{z}_s, \dot{z}_u) - \frac{k_u}{m_u}(z_u - z_g) \end{aligned} \tag{33}$$

The model has sprung and unsprung masses, m_s and m_u; vertical mass positions about the equilibrium, $\{z_s, z_u\}$, and velocities $\{\dot{z}_s, \dot{z}_u\}$; suspension spring and damping coefficients, k_s and b_s; tire spring coefficient, k_u; and ground input position, z_g. The system is nonlinear due to the asymmetric damping force that is dependent on the velocity direction.

$$\mathcal{D}(\dot{z}_s, \dot{z}_u) = \begin{cases} b_s(\dot{z}_s - \dot{z}_u), & (\dot{z}_s - \dot{z}_u) \geq 0 \\ \eta\, b_s(\dot{z}_s - \dot{z}_u), & (\dot{z}_s - \dot{z}_u) \geq 0 \end{cases} \tag{34}$$

The ratio of damping forces is determined by the scalar η.

The ground input position, z_g, is modeled by a series of isolated road bumps defined by,

$$z_g = A\, sin(\varpi\, t) \tag{35}$$

where A represents the amplitude of the bump; $\varpi = \pi\, v/l$ is the frequency of the irregularity determined by the vehicle velocity v and base length of the irregularity l; and t represents time. Each bump was uniquely spaced with no overlap with one another and their amplitudes were $A = 0.15$ meters. The frequencies of the speed bumps were selected to be $\boldsymbol{\varpi} = [1,\ 5, 10, 15]$ Hertz. Filtered Gaussian noise with a maximum amplitude of $A = 0.03$ meters was super-imposed over the series of speed bumps. The cut-off frequency of the filtered Gaussian noise was 35 Hertz.

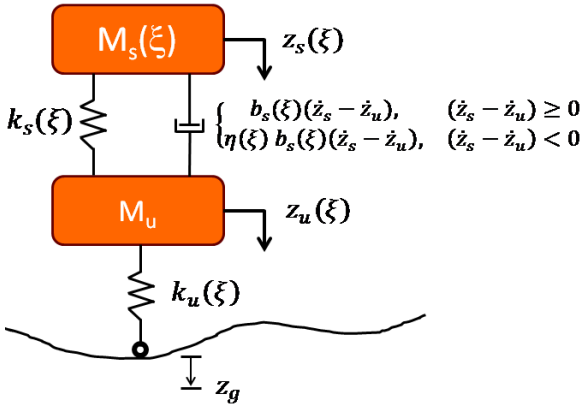

Figure 2—An uncertain 2-DOF quarter-car suspension model with a nonlinear asymmetric damper. The five uncertain parameters are, $\boldsymbol{\theta(\xi) = \{m_s(\xi), k_s(\xi), b_s(\xi), \eta(\xi), k_u(\xi)\}}$.

Varying passenger and cargo loads, fatiguing/deteriorating suspension components, and variations in tire air pressure are all very practical sources of uncertainty in a vehicle. Therefore, five uncertain system parameters were selected for this study, $\boldsymbol{\theta(\xi)} = \{m_s(\boldsymbol{\xi}), k_s(\boldsymbol{\xi}), b_s(\boldsymbol{\xi}), \eta(\boldsymbol{\xi}), k_u(\boldsymbol{\xi})\}$. Each uncertain parameter is assumed to have a uniform distribution and is therefore modeled with a Legendre polynomial expansion. This takes the form of,

$$\theta_r(\xi_r) = \theta_r^0 + \theta_r^1 \xi_r, \qquad r = 1 \ldots n_p \tag{36}$$

This system results in the following set of uncertain nonlinear EOMs,

$$\begin{aligned} \ddot{z}_s(\boldsymbol{\xi}) &= -\frac{k_s(\boldsymbol{\xi})}{m_s(\boldsymbol{\xi})}\big(z_s(\boldsymbol{\xi}) - z_u(\boldsymbol{\xi})\big) - \frac{1}{m_s(\boldsymbol{\xi})}\mathcal{D}\big(\dot{z}_s(\boldsymbol{\xi}), \dot{z}_u(\boldsymbol{\xi})\big) \\ \ddot{z}_u(\boldsymbol{\xi}) &= \frac{k_s(\boldsymbol{\xi})}{m_u}\big(z_s(\boldsymbol{\xi}) - z_u(\boldsymbol{\xi})\big) + \frac{1}{m_u}\mathcal{D}\big(\dot{z}_s(\boldsymbol{\xi}), \dot{z}_u(\boldsymbol{\xi})\big) - \frac{k_u(\boldsymbol{\xi})}{m_u}\Big(z_u(\boldsymbol{\xi}) - z_g(\boldsymbol{\xi})\Big) \end{aligned} \tag{37}$$

The ultimate goal for an optimal vehicle suspension design is to characterize the trade-off effects between three conflicting objectives: the passenger comfort (*ride*), which is modeled as the acceleration of the sprung mass; the suspension displacement (*rattle*); and the tire road holding forces (*holding*). Once the optimal trade-off relationship of these opposing objectives is determined, the engineers are more to select a parameter set that optimally satisfies the design's requirements.

5.2 CONCURRENT OUA/RDO

It is assumed that the standard deviations of the uncertain sprung mass, $\sigma_{m_s} = m_s^1$, and tire spring constant, $\sigma_{k_s} = k_u^1$, cannot be manipulated by the design; therefore, they are treated as fixed uncertainties. It is also assumed that the mean nonlinear damping coefficient, $\mu_{\eta_s} = \eta_s^0$, is fixed. Therefore, the search variables to carry out the OUA are $\boldsymbol{x}_{OUA} = \{ k_s^1, b_s^1, \eta_s^1\}$. The same search variables used in [10] for RDO are reused here; they are $\boldsymbol{x}_{RDO} = \{ k_s^0, b_s^0\}$. The final list of problem search variables is the union of the two sets, or $\boldsymbol{x}_{OUA} \cup \boldsymbol{x}_{RDO}$.

There are a number of methods presented in the literature for defining a cost-uncertainty trade-off curve [3]. This case-study assumes that the *reciprocal power* cost-uncertainty trade-off curve used by the manufacturing community is a reasonable definition for the selected non-geometric uncertainties in this case study. The *reciprocal power* curve is defined as,

$$\mathcal{W}_i = A_i + \frac{B_i}{\delta_i^k}, \qquad i = 1 \dots n_a \tag{38}$$

where A_i is a bias cost associated with the i^{th} source of uncertainty being apportioned; B_i is the cost that is scaled by the reciprocal power of the selected variation magnitude, δ_i^k, with $k \in \mathbb{R}$ defining the exponential power. Table 3 shows the A, B, and k values used for the uncertainties associated with $\boldsymbol{\delta}^k = \boldsymbol{x}OUA$.

Table 3—Independent variable means and standard deviations

Parameter	Bias (A_i)	Scaled Cost (B_i)	Power (k)
k_s^1	0.5	1.40 e+6	1.5
b_s^1	0.75	3.31 e+5	2
η_s^1	0.85	7.33 e-1	1

With this selected set of search variables, defined cost-uncertainty trade-off curves, and uncertain vehicle suspension dynamics described in (37), the corresponding concurrent OUA/RDO problem may be defined as,

$$\begin{aligned}
\min_{\boldsymbol{x}=\{ k_s^0,k_s^1,b_s^0,b_s^1,\eta_s^1\}} \quad & \sum_{i=1}^{n_a} w_i \left(A_i + \frac{B_i}{\delta_i^k} \right) \\
\text{s.t.} \quad & \ddot{z}_s(\boldsymbol{\xi}) = -\frac{k_s(\boldsymbol{\xi})}{m_s(\boldsymbol{\xi})}\big(z_s(\boldsymbol{\xi}) - z_u(\boldsymbol{\xi})\big) - \frac{1}{m_s(\boldsymbol{\xi})}\mathcal{D}\big(\dot{z}_s(\boldsymbol{\xi}), \dot{z}_u(\boldsymbol{\xi})\big) \\
& \ddot{z}_u(\boldsymbol{\xi}) = \frac{k_s(\boldsymbol{\xi})}{m_u}\big(z_s(\boldsymbol{\xi}) - z_u(\boldsymbol{\xi})\big) + \frac{1}{m_u}\mathcal{D}\big(\dot{z}_s(\boldsymbol{\xi}), \dot{z}_u(\boldsymbol{\xi})\big) - \frac{k_u(\boldsymbol{\xi})}{m_u}\big(z_u(\boldsymbol{\xi}) - z_g\big) \\
& \boldsymbol{y} = \begin{bmatrix} z_s(\boldsymbol{\xi}) \\ \dot{z}_s(\boldsymbol{\xi}) \\ z_u(\boldsymbol{\xi}) \\ \dot{z}_u(\boldsymbol{\xi}) \\ \int_0^{t_f} \big(\ddot{z}_s(\boldsymbol{\xi})\big)^2 \, dt \\ rms\big(z_u(\boldsymbol{\xi}) - z_g\big) \end{bmatrix} \\
& J_{ride}(\boldsymbol{\xi}) - \overline{J}_{ride} \le 0 \\
& J_{rattle}(\boldsymbol{\xi}) - \overline{J}_{rattle} \le 0, \ \ (extension) \\
& \underline{J}_{rattle} - J_{rattle}(\boldsymbol{\xi}) \le 0, \ \ (compression) \\
& J_{holding}(\boldsymbol{\xi}) - \overline{J}_{holding} \le 0 \\
& \underline{k}_s \le k_s^0(\boldsymbol{\xi}) \le \overline{k}_s \\
& \underline{b}_s \le b_s^0(\boldsymbol{\xi}) \le \overline{b}_s \\
& \boldsymbol{z}(0) = \boldsymbol{z}_0, \ \dot{\boldsymbol{z}}(0) = \dot{\boldsymbol{z}}_0
\end{aligned} \tag{39}$$

where w_i is a scalarization weighting factor for the i^{th} apportionment cost and the uncertain asymmetric damping force is,

$$\mathcal{D}\big(\dot{z}_s(\boldsymbol{\xi}), \dot{z}_u(\boldsymbol{\xi})\big) = \begin{cases} b_s(\boldsymbol{\xi})\big(\dot{z}_s(\boldsymbol{\xi}) - \dot{z}_u(\boldsymbol{\xi})\big), & \big(\dot{z}_s(\boldsymbol{\xi}) - \dot{z}_u(\boldsymbol{\xi})\big) \ge 0 \\ \eta(\boldsymbol{\xi})\, b_s(\boldsymbol{\xi})\big(\dot{z}_s(\boldsymbol{\xi}) - \dot{z}_u(\boldsymbol{\xi})\big), & \big(\dot{z}_s(\boldsymbol{\xi}) - \dot{z}_u(\boldsymbol{\xi})\big) \ge 0 \end{cases} \tag{40}$$

Therefore, (39) simultaneously performs RDO and OUA subject to the uncertain system dynamics defined in (37) and opposing performance constraints for vehicle *ride*, *rattle*, and *holding*. In other words, (39) determines the optimal apportionment of the uncertainties in $\boldsymbol{x}_{OUA}$ that satisfy the performance constraints; this is accomplished by simultaneously determining optimal nominal suspension values in $\boldsymbol{x}_{RDO}$. Equation (39) is a robust cMOO formulation in that a Pareto optimal set accounting for the system's uncertainties may be constructed by sweeping through a range of the performance constraint bounds, $\left\{\overline{J}_{ride}, \underline{J}_{rattle}, \overline{J}_{rattle}, \overline{J}_{holding}\right\}$. Recalling the mean and standard deviation definitions provided in (21)–(22), the performance constraint definitions are,

$$\begin{gathered}
\big(\mu_{y_5} + a_1\sigma_{y_5}\big) - \overline{J}_{ride} \le 0 \\
\Big(\big(\mu_{z_s} + a_2\sigma_{z_s}\big) - \big(\mu_{z_u} - a_3\sigma_{z_u}\big)\Big) - \overline{J}_{rattle} \le 0 \\
\underline{J}_{rattle} - \Big(\big(\mu_{z_s} - a_4\sigma_{z_s}\big) - \big(\mu_{z_u} + a_5\sigma_{z_u}\big)\Big) \le 0 \\
\big(\mu_{y_6} + a_6\sigma_{y_6}\big) - \overline{J}_{holding} \le 0
\end{gathered} \tag{41}$$

Therefore, the performance constraints bound the mean values plus or minus a weighted standard deviation. The constants a_i are weighting factors of the standard deviations; setting $a_i > 1.0$ yields a more conservative design. The interested reader is referred to [10] for more details regarding the definition of the uncertain performance constraints and this approach to robust cMOO.

5.3 RESULTS

The following results show the progression of the suspension design from a deterministic optimal performance design, to an RDO design, and finally the concurrent OUA/RDO design based on the new framework. Figure 3 best helps visualize the differences between the solutions obtained from the three designs; in the following figures the deterministic design optimization is referenced as *dOpt*; RDO is referenced as *uOpt*; and the simultaneous OUA/RDA is referenced as *aOpt*.

Figure 3 presents a specific 3D optimal solution in the constraint space as projections onto the three orthogonal 2D planes, $holding - rattle, holding - ride, rattle - ride$; this solution is obtained when the bounding constraints are set to $\bar{J}_{ride} = 220\ [m^2/s^3]$, $\underline{J}_{rattle} = \bar{J}_{rattle} = 0.203\ [m]$, and $\bar{J}_{holding} = 0.034\ [m]$. The *dOpt* solution is represented by a solid dot and results in an active *holding* constraint. The mean of the *uOpt* solution is represented by an asterisk. When all weighting factors are set to $a_i = 1$, the mean solution is enclosed in a 3D cuboid whose dimensions are determined by the original non-optimal uncertainty standard deviations. The projections of the *uOpt* cuboid are denoted by the dotted lines. Upon close inspection of Figure 3 it is apparent that the *uOpt* solution has an active *rattle* constraint. Since the σ_{rattle}^{uOpt} is so large, the *uOpt* solution was pushed to a significantly lower $\mu_{holding}^{uOpt}$ value when compared to the deterministic $x_{holding}^{dOpt}$ value.

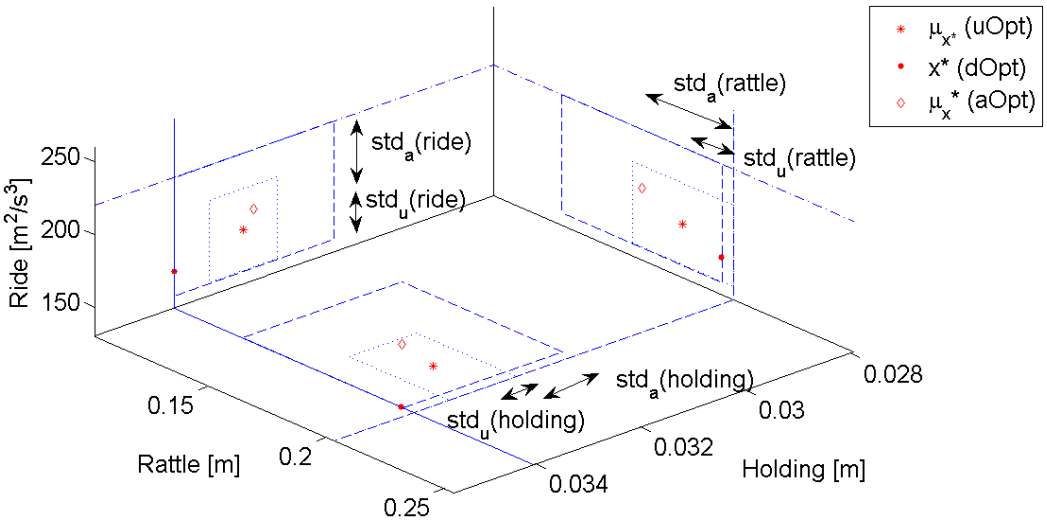

Figure 3— Projection of the 3D *dOpt*, *uOpt* , and *aOpt* optimal solutions onto the three orthogonal 2D planes. These optimal solutions were determined when: $\bar{J}_{ride} = 220\ [m^2/s^3]$, $\underline{J}_{rattle} = \bar{J}_{rattle} = 0.203\ [m]$, and $\bar{J}_{holding} = 0.034\ [m]$. Notice how each solution has a different set of active constraints.

Finally, the optimal mean solution obtained from the new framework for a simultaneous OUA/RDO design is represented by a diamond. Again, when all weighting factors are set to $a_i = 1$, the mean solution is enclosed in a 3D cuboid whose dimensions are determined by the optimally apportioned uncertainty standard deviations, $\boldsymbol{x}_{OUA}$. The projections of the *aOpt* cuboid are denoted by the dashed lines. Notice how the apportioned uncertainties for the *aOpt* solution are significantly larger than the *uOpt* solution. Also, the result of the uncertainty apportionment yields active *ride* and *holding* constraints. Figure 3 confirms that $\boldsymbol{x}_{OUA}$ and $\boldsymbol{x}_{RDO}$ are coupled in that the *aOpt* solution is shifted when compared to that found by *uOpt*; therefore, the simultaneous search finds the true optimal solution.

A Pareto optimal trade-off curve may be obtained by sweeping through a range of values of the bounding constraints. Since there are four performance constraints the actual Pareto trade-off is a 5D surface; however, to illustrate the *ride* and *holding* constraints are fixed and the OUA/RDO optimal solution is determined for a range of values of the *rattle* bound; where $\underline{J}_{rattle} = \bar{J}_{rattle}$. The resulting Pareto curve may be illustrated as a 2D projection into the objective/*rattle* plane; this is illustrated in Figure 4.

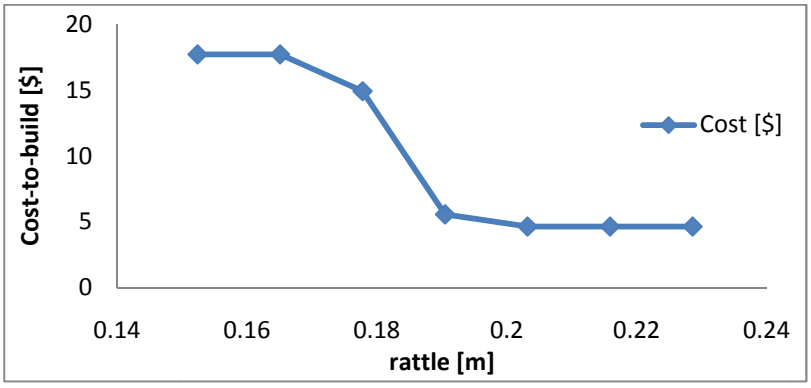

Figure 4—A single 2D plane from the 5D Pareto optimal set showing the trade-off between the cost-to-build objective and *rattle* constraint bound; the other constraints are held constant at $\bar{J}_{ride} = 220\ [m^2/s^3]$, and $\bar{J}_{holding} = 0.034\ [m]$. The first two points on the curve are set to zero as these resulted in infeasible *aOpt* designs.

One interesting observation is that the OUA/RDO framework determined the first two design points shown in Figure 4 to be infeasible points; therefore, their values were set to the objective value that results if no optimization was performed. The reason for these infeasible constraints is best illustrated by Figure 5, which shows the 2D projection of the *uOpt ride* objective trade-off with the *rattle* constraint resulting from an RDO, or *uOpt*, design [10]; the *holding* bound is fixed at $\bar{J}_{holding} = 0.034\ [m]$. The diamond curve is obtained from a *dOpt* optimal design and the square curve is obtained from the RDO design, however, the triangle line shows the bounding constraint value assigned for the OUA/RDO, or *aOpt*, design ($\bar{J}_{ride} = 220\ [m^2/s^3]$). When the *ride* bound (triangle line) is below the *uOpt* curve (square curve) the OUA/RDO design requires the uncertainties to be reduced, or tightened, to satisfy the constraints. However, since the variations of the uncertain mass, m_s^1, and tire spring constant, k_u^1, were assumed to be uncontrollable, the required uncertainty reductions for the first two points ended up being infeasible.

The resulting optimally allocated uncertainties for the various design points illustrated in Figure 4 are shown in Figure 6. The optimal distribution of the allocated uncertainties is a function of the selected cost-uncertainty trade-off curves defined by (38) and each associated weighting factor w_i; Table 4 documents the apportionment results obtained from the specific case when $\bar{J}_{ride} = 220\ [m^2/s^3]$, $\underline{J}_{rattle} = \bar{J}_{rattle} = 0.203\ [m]$, and $\bar{J}_{holding} = 0.034\ [m]$. The final apportionment of optimal standard deviations resulted in changes as large as 300% relative to their respective initial values and as large as 64% relative to their respective mean values; this clearly show-cases the new concurrent OUA/RDO framework's ability to treat uncertainties with large magnitudes.

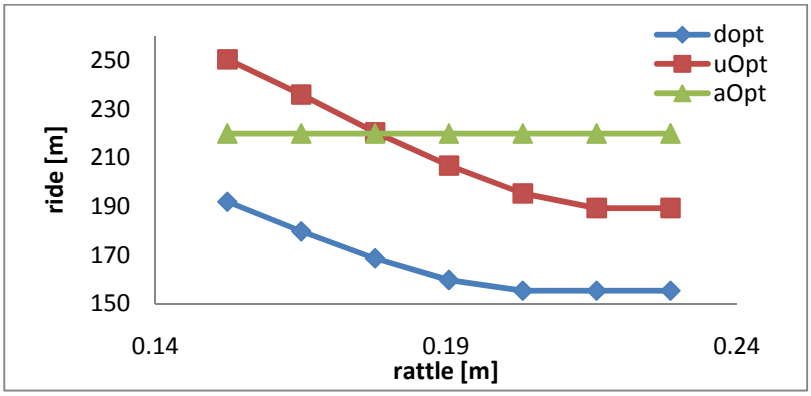

Figure 5—A single 2D plane showing the trade-off between the *ride* objective and *rattle* constraint bound when performing an RDO design; where $\bar{J}_{holding} = 0.034\ [m]$ is held fixed. The triangle curve represents the bounding value for $\bar{J}_{ride}$ when performing the OUA/RDO designs associated with Figure 3 and Figure 4. The first two points on the triangle curve require too large of a reduction in uncertainties and therefore result in infeasible *aOpt* designs.

The final resulting cost-to-manufacture for the *dOpt* and *uOpt* designs was $17.70; this is based on the initial non-optimal uncertainty levels which apply to both designs. However, the *aOpt* solutions result in significant cost-to-manufacture reductions; Figure 4 shows the cost savings for the various design points, where savings as high as 74% were experienced. Clearly, the actual cost-savings achievable by applying the new simultaneous OUA/RDO framework defined in (28), (29) or (31) is very dependent upon the definition of the respective cost-uncertainty trade-off curves, however, this case-study illustrates the power of treating uncertainty up-front during the design process; robust optimally performing systems are designed at an optimal cost-to-manufacture.

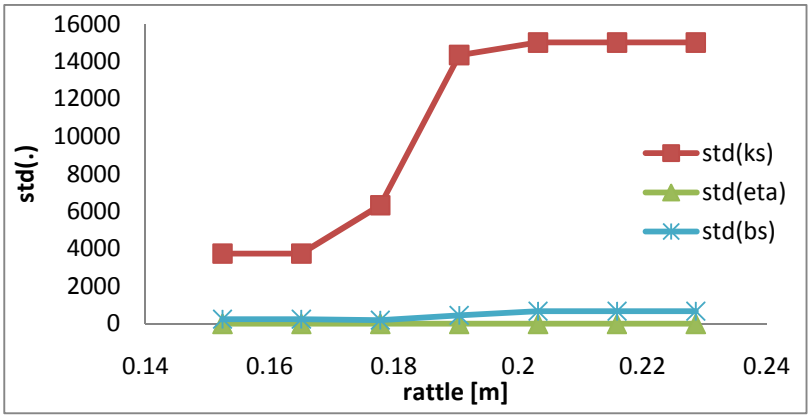

Figure 6—Optimally apportioned uncertainties determined from the new simultaneous OUA/RDO framework. The first two points were infeasible designs and therefore the values were set to their initial values.

Table 4—Apportionment results obtained from the new concurrent OUA/RDO framework corresponding to the case when $\bar{J}_{ride} = 220\ [m^2/s^3]$, $\underline{J}_{rattle} = \bar{J}_{rattle} = 0.203\ [m]$, and $\bar{J}_{holding} = 0.034\ [m]$.

	Δ from non-optimal	% Δ from non-optimal	% from μ
k_s^1	11250	300.00%	64.27%

b_s^1	423.0763	20.19%	29.70%
η_s^1	0.5213	300.12%	50.00%

6 CONCLUSIONS

This work presents a novel framework that enables RDO concurrently with OUA for dynamical systems described by ODEs. Current state of the art methodologies are limited to only treating geometric uncertainties of small magnitude. The new framework removes these limitations by treating both geometric and non-geometric uncertainties in a unified manner within a concurrent RDO/OUA setting. Additionally, uncertainties may be relatively large in magnitude and the system constitutive relations may be highly nonlinear. The vehicle suspension design case-study supports this message; uncertainty allocations on the order of 300% of the initial value were obtained.

The computational efficiency of the selected gPC approach allows statistical moments of the uncertain system to be explicitly included in the optimization-based design process. The framework, presented in a cMOO formulation, enables a Pareto optimal trade-off surface characterization for the entire family of systems within the probability space. The Pareto trade-off surface from the new framework is shown to be off-set from the traditional deterministic optimal trade-off surface as a result of the additional statistical moment information formulated into the objective and constraint relations. As such, the vehicle suspension case-study Pareto trade-off surface from the new framework enables a more robust and optimally performing design at an optimal manufacturing cost.

In future work, the authors will expand the new framework to treat constrained dynamical systems described by differential algebraic equations (DAEs).

ACKNOWLEDGEMENTS

This work was partially supported by the Automotive Research Center (ARC), Thrust Area 1, the National Science Foundation (NSF), the Computational Science Laboratory (CSL), Advanced Vehicle Dynamics Laboratory (AVDL) and the Robotics and Mechanisms laboratory (RoMeLa) at Virginia Tech.

REFERENCES

[1] Chase, K. W., 1999, Dimensioning and Tolerancing Handbook, McGraw-Hill Professional, Minimum-Cost Tolerance Allocation.
[2] Chase, K. W., 1999, "Tolerance Allocation Methods for Designers," ADCATS Report, 99(6), pp. 1-28.
[3] Chase, K. W., Greenwood, W. H., Loosli, B. G., and Hauglund, L. F., 1990, "Least Cost Tolerance Allocation for Mechanical Assemblies with Automated Process Selection," Manufacturing Review, 3(1), pp. 49-59.
[4] Barraja, M., and Vallance, R. R., 2002, "Tolerance Allocation for Kinematic Couplings," eds., pp.
[5] Merkley, K. G., 1998, "Tolerance Analysis of Compliant Assemblies," Ph.D. thesis, Citeseer,
[6] Krishnaswami, P., and Kelkar, A. G., 2003, "Optimal Design of Controlled Multibody Dynamic Systems for Performance, Robustness and Tolerancing," Engineering with Computers, 19(1), pp. 26-34.
[7] Arenbeck, H., Missoum, S., Basudhar, A., and Nikravesh, P., 2010, "Reliability-Based Optimal Design and Tolerancing for Multibody Systems Using Explicit Design Space Decomposition," Journal of Mechanical Design, 132(2), pp. 021010-11.
[8] Lee, S. J., Gilmore, B. J., and Ogot, M. M., 1993, "Dimensional Tolerance Allocation of Stochastic Dynamic Mechanical Systems through Performance and Sensitivity Analysis," Journal of Mechanical Design, 115(3), pp. 392-402.
[9] Rout, B., and Mittal, R., 2010, "Simultaneous Selection of Optimal Parameters and Tolerance of Manipulator Using Evolutionary Optimization Technique," Structural and Multidisciplinary Optimization, 40(1), pp. 513-528.
[10] Hays, J., Sandu, A., Sandu, C., and Hong, D., 2011, "Parametric Design Optimization of Uncertain Ordinary Differential Equation Systems," Technical Report No. TR-11-06, Virginia Tech, Blacksburg, VA, USA.
[11] Hays, J., Sandu, A., Sandu, C., and Hong, D., 2011, "Motion Planning of Uncertain Fully-Actuated Dynamical Systems—an Inverse Dynamics Formulation," eds., Washington, DC, USA, pp. (accepted).
[12] Hays, J., Sandu, A., Sandu, C., and Hong, D., 2011, "Motion Planning of Uncertain Fully-Actuated Dynamical Systems—a Forward Dynamics Formulation," eds., Washington, DC, USA, pp. (accepted).
[13] Hays, J., Sandu, A., Sandu, C., and Hong, D., 2011, "Motion Planning of Uncertain under-Actuated Dynamical Systems—a Hybrid Dynamics Formulation," eds., Denver, CO, USA, pp. (submitted).
[14] Hays, J., Sandu, A., Sandu, C., and Hong, D., 2011, "Motion Planning of Uncertain Ordinary Differential Equation Systems," Technical Report No. TR-11-04, Virginia Tech, Blacksburg, VA, USA.
[15] Wiener, N., 1938, "The Homogeneous Chaos," American Journal of Mathematics, 60(4), pp. 897-936.
[16] Xiu, D., and Karniadakis, G., 2003, "The Wiener-Askey Polynomial Chaos for Stochastic Differential Equations," pp.
[17] Sandu, C., Sandu, A., and Ahmadian, M., 2006, "Modeling Multibody Systems with Uncertainties. Part Ii: Numerical Applications," Multibody System Dynamics, 15(3), pp. 241-262.
[18] Sandu, A., Sandu, C., and Ahmadian, M., 2006, "Modeling Multibody Systems with Uncertainties. Part I: Theoretical and Computational Aspects," Multibody System Dynamics, 15(4), pp. 369-391.
[19] Cheng, H., and Sandu, A., 2009, "Efficient Uncertainty Quantification with the Polynomial Chaos Method for Stiff Systems," Mathematics and Computers in Simulation, 79(11), pp. 3278-3295.
[20] Cheng, H., and Sandu, A., 2007, "Numerical Study of Uncertainty Quantification Techniques for Implicit Stiff Systems," eds., Winston-Salem, NC, USA, pp. 367-372.
[21] Cheng, H., and Sandu, A., 2009, "Uncertainty Quantification in 3d Air Quality Models Using Polynomial Chaoses," Environmental Modeling and Software, 24(8), pp. 917-925.
[22] Cheng, H., and Sandu, A., 2009, "Uncertainty Apportionment for Air Quality Forecast Models," eds., Honolulu, HI, USA, pp. 956-960.
[23] Cheng, H., and Sandu, A., 2010, "Collocation Least-Squares Polynomial Chaos Method," eds., Orlando, FL, USA, pp. 80.
[24] Xiu, D., and Hesthaven, J. S., 2005, "High-Order Collocation Methods for Differential Equations with Random Inputs," SIAM Journal on Scientific Computing, 27(3), pp. 1118-1139.
[25] Xiu, D., 2007, "Efficient Collocational Approach for Parametric Uncertainty Analysis," Communications in Computational Physics, 2(2), pp. 293-309.

[26] Xiu, D., 2009, "Fast Numerical Methods for Stochastic Computations: A Review," Communications in Computational Physics, 5(2-4), pp. 242-272.
[27] Papoulis, A., Pillai, S., and Unnikrishna, S., 2002, *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill New York,
[28] Law, M., 1996, "Multivariate Statistical Analysis of Assembly Tolerance Specifications," Ph.D. thesis, Brigham Young University, Provo, UT, USA.
[29] Greenwood, D., 2003, *Advanced Dynamics*, Cambridge Univ Pr,
[30] Murray, R., Li, Z., Sastry, S., and Sastry, S., 1994, *A Mathematical Introduction to Robotic Manipulation*, CRC Press, Inc, Boca Raton, FL, USA.
[31] Nikravesh, P. E., 2004, Product Engineering, Springer, An Overview of Several Formulations for Multibody Dynamics.
[32] Haug, E. J., 1989, *Computer Aided Kinematics and Dynamics of Mechanical Systems. Vol. 1: Basic Methods*, Allyn & Bacon, Inc.,
[33] Diehl, M., Ferreau, H., and Haverbeke, N., 2009, "Efficient Numerical Methods for Nonlinear Mpc and Moving Horizon Estimation," Nonlinear Model Predictive Control, pp. 391-417.
[34] Biegler, L. T., 2003, "Optimization of Ode/Dae Constrained Models," Technical Report No.
[35] Biegler, L. T., and Grossmann, I. E., 2004, "Retrospective on Optimization," Computers & Chemical Engineering, 28(8), pp. 1169-1192.
[36] Adcats, 2006, One-Way Clutch Verification, http://adcats.et.byu.edu/WWW/ADCATS/Example_Problems/ProE_Verify/2clutch/2clutch.html