Proceedings of the ASME 2011 International Design Engineering Technical Conferences &
Computers and Information in Engineering Conference
IDETC/CIE 2011
August 28-31, 2011, Washington, DC, USA

DETC2011-48233

MOTION PLANNING OF UNCERTAIN FULLY-ACTUATED DYNAMICAL SYSTEMS—A FORWARD DYNAMICS FORMULATION

Joe Hays
Mechanical Engineering
Virginia Tech
Blacksburg, VA 24061

Adrian Sandu
Computational Science Laboratory
Computer Science Department
Virginia Tech
Blacksburg, VA 24061

Corina Sandu
Advanced Vehicle Dynamics Laboratory
Mechanical Engineering
Virginia Tech
Blacksburg, VA 24061

Dennis Hong
Robotics and Mechanisms Laboratory
Mechanical Engineering
Virginia Tech
Blacksburg, VA 24061

Keywords: Motion Planning, Trajectory Planning, Optimization, Nonlinear Programming, Multibody Dynamics, Uncertainty Quantification

Abstract

This work presents a novel nonlinear programming based motion planning framework that treats uncertain fully-actuated dynamical systems described by ordinary differential equations. Uncertainty in multibody dynamical systems comes from various sources, such as: system parameters, initial conditions, sensor and actuator noise, and external forcing. Treatment of uncertainty in design is of paramount practical importance because all real-life systems are affected by it; ignoring uncertainty during design may lead to poor robustness and suboptimal performance. System uncertainties are modeled using Generalized Polynomial Chaos and are solved quantitatively using a least-square collocation method. The computational efficiency of this approach enables the inclusion of uncertainty statistics in the nonlinear programming optimization process. As such, new design questions related to uncertain dynamical systems can now be answered through the new framework.

Specifically, this work presents the new framework through a forward dynamics formulation where deterministic actuator inputs are prescribed and uncertain state trajectories are quantified. The benefits of the ability to quantify the resulting state uncertainty are illustrated in an effort optimal motion planning case-study of a serial manipulator pick-and-place application. The resulting design determines a feasible effort optimal motion plan—subject to actuator and obstacle avoidance constraints—for all possible systems within the probability space. Variance of the system's terminal conditions are also minimized in a Pareto optimal sense. The inverse dynamics formulation (using deterministic state trajectories and uncertain actuator inputs) is presented in a companion paper.

1 INTRODUCTION

Design engineers cannot quantify exactly every aspect of a given system. These uncertainties frequently create difficulties in accomplishing design goals and can lead to poor robustness and suboptimal performance. Tools that facilitate the analysis and characterization of the effects of uncertainties enable designers to develop more robustly performing systems. The need to analyze the effects of uncertainty is particularly acute when designing dynamical systems. Frequently, engineers do not account for various uncertainties in their design in order to save time and to reduce costs. However, this simply delays, or hides, the cost which is inevitably incurred down-stream in the design flow; or worse, after the system has been deployed and fails to meet the design goals. Ultimately, if a robust system design is to be achieved, uncertainties must be accounted for up-front during the design process.

This work presents a novel nonlinear programming (NLP) based motion planning framework that treats uncertain fully-actuated dynamical systems described by ordinary differential equations (ODEs). System uncertainties, such as parameters, initial conditions, sensor/actuator noise, or forcing functions, are modeled using Generalized Polynomial Chaos (gPC) and are solved quantitatively using a least-square collocation method (LSCM). The computational efficiencies gained by gPC and LSCM enable the inclusion of uncertainty statistics in the NLP optimization process.

Specifically, this work presents the new framework through a *forward dynamics* formulation where deterministic input wrenches are prescribed and uncertain system states are quantified. The benefits of the ability to quantify the resulting state uncertainty are illustrated in a *effort optimal* motion planning case-study of a serial manipulator pick-and-place application. The resulting design determines an *effort optimal* motion plan—subject to actuator and obstacle avoidance constraints—for all possible systems within the probability space.

The companion formulation based on *uncertain inverse dynamics* is presented by the authors in [1]. Application of the *uncertain inverse dynamics* has particular advantages for configuration/position controlled systems, while the *uncertain forward dynamics* formulation presented in this work is more suitable for force controlled systems.

It's important to point out that the new framework is not dependent on the specific formulation of the equations of motion (EOMs); formulations such as, Newtonian, Lagrangian, Hamiltonian, and Geometric methodologies are all applicable. This work applies the analytical Lagrangian EOM formulation which is briefly introduced in Section 2; Section 3 briefly discusses the well studied deterministic motion planning problem; Section 4 reviews the Generalized Polynomial Chaos methodology for uncertainty quantification when using *forward dynamics*; Section 5 introduces the new framework for motion planning of uncertain fully-actuated dynamical systems based on an *uncertain forward dynamics* formulation; finally, Section 6 illustrates the strengths of the new framework through a serial manipulator pick-and-place application which is followed by concluding remarks in Section 7.

2 MULTIBODY INVERSE DYNAMICS

As a very brief overview, the Euler-Lagrange ODE formulation for a multibody dynamical system can be described by [2, 3],

$$\begin{aligned}\boldsymbol{M}\big(\boldsymbol{q}(t),\boldsymbol{\theta}(t)\big)\ddot{\boldsymbol{q}}(t)+\boldsymbol{C}\big(\boldsymbol{q}(t),\dot{\boldsymbol{q}}(t),\boldsymbol{\theta}(t)\big)\dot{\boldsymbol{q}}(t)\\ +\boldsymbol{N}\big(\boldsymbol{q}(t),\dot{\boldsymbol{q}}(t),\boldsymbol{\theta}(t)\big)\\ =\boldsymbol{\mathcal{F}}\big(\boldsymbol{q}(t),\dot{\boldsymbol{q}}(t),\ddot{\boldsymbol{q}}(t),\boldsymbol{\theta}(t)\big)=\boldsymbol{\tau}(t)\end{aligned} \tag{1}$$

where $\boldsymbol{q}(t)\in\mathbb{R}^{n_d}$ are independent generalized coordinates equal in number to the number of degrees of freedom, n_d (the illustrating case study uses relative joint angles but the formulation is not limited to such a choice); $\dot{\boldsymbol{q}}(t)\in\mathbb{R}^{n_d}$ the rates of the generalized coordinates and $\ddot{\boldsymbol{q}}(t)$ are the associated accelerations—using Newton's *dot* notation for a time derivative; $\boldsymbol{\theta}(t)\in\mathbb{R}^{n_p}$ includes system parameters of interest (specifically, those with uncertainty as described in Section 4); $M\big(\boldsymbol{q}(t),\boldsymbol{\theta}(t)\big)\in\mathbb{R}^{n_d\times n_d}$ is the square positive definite inertia matrix; $\boldsymbol{C}\big(\boldsymbol{q}(t),\dot{\boldsymbol{q}}(t),\boldsymbol{\theta}(t)\big)\in\mathbb{R}^{n_d\times n_d}$ includes centrifugal, gyroscopic and Coriolis effects; $\boldsymbol{N}\big(\boldsymbol{q}(t),\dot{\boldsymbol{q}}(t),\boldsymbol{\theta}(t)\big)\in\mathbb{R}^{n_d}$ the generalized gravitational and joint forces; and $\boldsymbol{\tau}(t)\in\mathbb{R}^{n_i}$ are the n_i applied input wrenches. (For notational brevity, all future equations will drop the explicit time dependence.)

The trajectory of the system is determined by solving (1) as an initial value problem, where $\boldsymbol{q}(0)=\boldsymbol{q}_0$ and $\dot{\boldsymbol{q}}(0)=\dot{\boldsymbol{q}}_0$. Also, the system measured outputs are defined by,

$$\boldsymbol{y}=\boldsymbol{\mathcal{O}}(\boldsymbol{q},\dot{\boldsymbol{q}},\boldsymbol{\theta}) \tag{2}$$

where $\boldsymbol{y}\in\mathbb{R}^{n_o}$ with n_o equal to the number of outputs.

3 DETERMINISTIC FORWARD DYNAMICS MOTION PLANNING

The task of dynamic system motion planning is a well studied topic; it aims to determine a state, or input, trajectory to realize some prescribed objective. Sampled-based motion planning formulations, such as Rapid-exploring Random Trees (RRTs), primarily focus on finding a feasible solution [4-6]; where nonlinear programming formulations seek to determine at least a local optimal solution [7-13].

Use of a *forward dynamics* formulation requires a practitioner to define the input wrench profile, $\tau(t)$, over the entire motion plan. This is an infinite dimensional problem. Parameterized trajectories are commonly used to reduce the problem to a finite dimensional search. For example, the input wrench profile can be represented with B-Splines,

$$\boldsymbol{\tau}(\boldsymbol{P},t)=\sum_{i=0}^{n_{sp}}\beta^{i,\wp-1}(t)\boldsymbol{p}^i \tag{3}$$

where $\boldsymbol{p}\in\mathbb{R}^{n_d}$ are n_{sp} control points; β and $\wp$ are the B-Splines' basis functions and degree, respectively [14]; $\boldsymbol{P}\in\mathbb{R}^{n_{sp}\times n_d}$ is an vector of control points, $\boldsymbol{p}$.

Once the input wrench profiles have been parameterized the NLP-based deterministic motion planning problem may be formulated as,

$$\begin{aligned}\min_{x=\{\boldsymbol{P}\}}\quad & J\\ \text{s.t.}\quad & \boldsymbol{\mathcal{F}}(\boldsymbol{q},\dot{\boldsymbol{q}},\ddot{\boldsymbol{q}},\boldsymbol{\theta})=\boldsymbol{\tau}\\ & \mathbf{y}=\boldsymbol{\mathcal{O}}(\boldsymbol{q},\dot{\boldsymbol{q}},\boldsymbol{\theta})\\ & \boldsymbol{\mathcal{C}}(\boldsymbol{y},\boldsymbol{\tau},\boldsymbol{\theta})\le\mathbf{0}\\ & \boldsymbol{q}(0)=\boldsymbol{q}_0,\boldsymbol{q}(t_f)=\boldsymbol{q}_{t_f}\\ & \dot{\boldsymbol{q}}(0)=\dot{\boldsymbol{q}}_0,\dot{\boldsymbol{q}}(t_f)=\dot{\boldsymbol{q}}_{t_f}\end{aligned} \tag{4}$$

where the input wrenches' explicit dependence on their associated control points has been dropped for notational brevity. Equation (4) seeks to find the control points $\boldsymbol{P}$ that minimize some prescribed objective function, J, while being subject to the *forward dynamic* constraints defined in (1). Additional constraints may also be defined; for example, maximum/minimum actuator and system parameter limits or physical system geometric limits can be represented as inequality relations, $\boldsymbol{\mathcal{C}}(\boldsymbol{y},\boldsymbol{\tau},\boldsymbol{\theta})\le\mathbf{0}$.

In [1] the authors itemized common motion planning objective function definitions from the literature; the *effort optimal* objective function, used in the accompanying case study, is repeat for convenience ,

$$\mathrm{J}_{\mathrm{D1}}=\sum_{i=1}^{n_i}\tau_i{}^2,\quad\forall t \tag{5}$$

The solution to (4) produces an optimal motion plan under the assumption that all system properties are known (i.e., (1) is completely deterministic). The primary contribution of this work is the presentation of a variant of (4) that allows (1) to contain uncertainties of diverse types (e.g., parameters, initial conditions, sensor/actuator noise, or forcing functions). The following section will briefly introduce Generalized Polynomial Chaos (gPC) which is used to model the uncertainties and to quantify the resulting uncertain system states.

4 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,

$$\theta(\xi) = \sum_{j=0}^{p_o} \theta^j \psi^j(\xi(\omega)) \quad (6)$$

where $\theta^j \in \mathbb{R}$ represent known stochastic coefficients; $\psi^j \in \mathbb{R}$ represent individual single dimensional orthogonal basis terms (or modes); $\xi(\omega) \in \mathbb{R}$ is the associated random variable for θ that maps the random event $\omega \in \Omega$, from the sample space, Ω, to the domain of the orthogonal polynomial basis (e.g., $\xi: \Omega \to [-1,1]$).

Polynomial chaoses are orthogonal with respect to the ensemble inner product,

$$\langle \psi^i(\xi), \psi^j(\xi) \rangle = \int_{-1}^{1} \psi^i(\xi)\psi^j(\xi)w(\xi)d\xi = 0, \quad \text{for } i \neq j \quad (7)$$

where $w(\xi)$ is the weighting function that is equal to the joint probability density function of the random variable ξ. 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 (7).

Generalized Polynomial Chaos can be applied to multibody dynamical systems described by differential equations [17, 18]; where sources of uncertainty, such as parameters, initial conditions, sensor/actuator noise, or forcing functions, are all treated in a unified manner. The presence of uncertainty in the system results in either uncertain states, as in a *forward dynamics* formulation (1), or uncertain inputs, as in an *inverse dynamics* formulation [1]. Therefore, proceeding with the *forward dynamics* formulation, the uncertain states can be approximated in a similar fashion as (6),

$$q_i(\xi; t) = \sum_{j=0}^{n_b} q_i^j(t)\Psi^j(\xi), \qquad i = 1 \dots n_s \quad (8)$$

where $q_i^j(t) \in \mathbb{R}^{n_b}$ again represents the stochastic coefficients—for the i^{th} state—but are now unknown functions of time, with $n_b \in \mathbb{N}$ representing the number of basis terms in the approximation.

The stochastic basis of the states may be multidimensional in the event 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 Jacobi chaoses).

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

$$\Psi^j = \psi_1^{i_1}\psi_2^{i_2} \dots \psi_{n_p}^{i_{n_p}}, \quad i_k = 0 \dots p_o, k = 1 \dots n_p \quad (9)$$

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 $n_b = \frac{(n_p+p_o)!}{n_p!\,p_o!}$ basis terms. The minimal basis set can be determined by the products resulting from these index ranges,

$$i_1 = 0 \dots p_o,$$
$$i_2 = 0 \dots (p_o - i_1), \dots,$$
$$i_{n_p} = 0 \dots \left(p_o - i_1 - i_2 - \dots - i_{(n_p-1)}\right)$$

The number of multidimensional terms, n_b, grows quickly with the number of uncertain parameters, n_p, and polynomial order, p_o. Sandu, Sandu, and Ahmadian 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.

Substituting (6) and (8) into (1) produces the following *uncertain forward dynamics* (UFD),

$$\begin{aligned} &\mathcal{F}(\textstyle\sum_{j=0}^{n_b} q_i^j(t)\Psi^j(\boldsymbol{\xi}), \sum_{j=0}^{n_b} \dot{q}_i^j(t)\Psi^j(\boldsymbol{\xi}), \\ &\quad \textstyle\sum_{j=0}^{n_b} \ddot{q}_i^j(t)\Psi^j(\boldsymbol{\xi}), \sum_{j=0}^{p_o} \theta_k^j(t)\,\psi_k^j(\xi_k)), \\ &\quad i = 1 \dots n_s, k = 1 \dots n_p \end{aligned} \quad (10)$$

where the unknowns are now the $n_b n_s$ stochastic state coefficients, $q_i^j(t)$.

It is instructive to notice how time and randomness are decoupled within a single term after the gPC expansion. Only the stochastic coefficients are dependent on time, and only the basis terms are dependent on the n_p random variables, $\boldsymbol{\xi}$.

The Galerkin Projection Method (GPM) is a commonly used method for solving (10), however, this is a very intrusive technique and requires a custom formulation of the *forward dynamic* EOMs. As an alternative, sample-based collocation techniques can be used without the need to modify the base EOMs.

Reference [17] showed that the collocation method solves (10) by solving (1)–(2) 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 unknown stochastic state coefficients q_i^j can be determined at that given time instance. This can be accomplished by defining the intermediate variables,

$${}_kQ_i(t;\, {}_k\boldsymbol{\mu}) = \sum_{j=0}^{n_b} q_i^j(t)\Psi^j({}_k\boldsymbol{\mu}), \qquad i = 1 \dots n_s, k = 0 \dots n_{cp} \quad (11)$$

and substituting them into (10). This yields,

$$\begin{aligned} {}_k\dot{Q}_i(t;\, {}_k\mu) &= \boldsymbol{\mathcal{F}}\left({}_kQ(t;\, {}_k\boldsymbol{\mu}),\, {}_k\Theta_r(t;\, {}_k\boldsymbol{\mu})\right), \\ &\quad i = 1 \dots n_s, k = 0 \dots n_{cp}, r = 1 \dots n_p \end{aligned} \quad (12)$$

where,

$${}_k\Theta_r(t;\, {}_k\boldsymbol{\mu}) = \sum_{j=0}^{p_o} \theta_r^j(t)\,\psi^j({}_k\boldsymbol{\mu}_i), \;\; k = 0 \dots n_{cp}, r = 1 \dots n_p \quad (13)$$

Equation (12) provides a set of n_{cp} independent equations whose solutions determine the stochastic coefficients, q_i^j. This is accomplished by recalling the relationship of the stochastic coefficients to the solutions, ${}_kQ_i$, shown in (11). In matrix notation (11) can be expressed for all states,

$$\boldsymbol{Q}_i = \left(q_i(t)\right)^T \boldsymbol{\Psi}(\boldsymbol{\mu}), \qquad i = 1 \dots n_s \quad (14)$$

where the matrix,

$$A_{k,j} = \Psi^j({}_k\boldsymbol{\mu}), \;\; j = 0 \dots n_b, k = 0 \dots n_{cp} \quad (15)$$

is defined as the *collocation matrix*. It's important to note that $n_b \le n_{cp}$. The stochastic coefficients can now be solved for using (14),

$$q_i^j(t) = \boldsymbol{A}^{\#}\boldsymbol{Q}_i, \qquad i = 1 \dots n_s, j = 0 \dots n_b \quad (16)$$

where $A^{\#}$ is the pseudo inverse of A if $n_b < n_{cp}$. If $n_b = n_{cp}$, then (16) is simply a linear solve. However, [19] presented the least-squares collocation method (LSCM) where the stochastic state coefficients are solved for, in a least squares sense, using (16) when $n_b < n_{cp}$. Reference [19] also showed that as $n_{cp} \to \infty$ the LSCM approaches the GPM solution; where by selecting $3n_b \le n_{cp} \le 4n_b$ the greatest convergence benefit is achieved with minimal computational cost. LSCM also enjoys the same exponential convergence rate as $p_o \to \infty$.

The unintrusive nature of the LSCM sampling approach is arguably its greatest benefit; (1) 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 [17, 19-22] for more information.

5 UNCERTAIN FORWARD DYNAMICS MOTION PLANNING

Little work is found in the literature addressing motion planning for uncertain systems. The literature thus far has primarily addressed sensor and/or actuator noise [4, 23] and frequently only treats the system's kinematics [24, 25].

In [26], Kewlani presents an RRT planner for mobility of robotic systems based on gPC but refers to it as a stochastic response surface method (SRSM). Kewlani's work is similar in spirit to this work, however, the main difference is Kewlani's solution is developed only for determining a feasible motion plan. The motion planning framework of this paper is formulated within a NLP setting and thus benefits from the more efficient gradient-based searching techniques providing at least a locally optimal design.

As such, the new NLP-based framework for motion planning of uncertain fully-actuated multibody dynamical systems, formulated with *uncertain forward dynamics*, is,

$$\begin{aligned}
&\min_{\boldsymbol{x}=\{\boldsymbol{P}\}} && \mathrm{J}\\
&\text{s.t.} && \boldsymbol{\mathcal{F}}\big(\boldsymbol{q}(\xi),\dot{\boldsymbol{q}}(\xi),\ddot{\boldsymbol{q}}(\xi),\boldsymbol{\theta}(\xi)\big)=\boldsymbol{\tau}\\
& && \boldsymbol{y}(\xi)=\boldsymbol{\mathcal{O}}\big(\boldsymbol{q}(\xi),\dot{\boldsymbol{q}}(\xi),\boldsymbol{\theta}(\xi)\big)\\
& && \boldsymbol{\mathcal{C}}\big(\boldsymbol{y}(\xi),\boldsymbol{\theta}(\xi),\boldsymbol{\tau}\big)\le\boldsymbol{0}\\
& && \boldsymbol{q}(0;\xi)=\boldsymbol{q}_0,\ \boldsymbol{q}\big(t_f;\xi\big)=\boldsymbol{q}_{t_f}\\
& && \dot{\boldsymbol{q}}(0;\xi)=\dot{\boldsymbol{q}}_0,\ \dot{\boldsymbol{q}}\big(t_f;\xi\big)=\dot{\boldsymbol{q}}_{t_f}
\end{aligned} \tag{17}$$

where (17) is a reformulation of (4) with uncertain system states. The most interesting part of (17) comes in the definition of the objective function terms and constraints. These terms now have the ability to approach the design accounting for uncertainties by way of expected values, variances, and standard deviations. In the *uncertain forward dynamics* formulation, the input wrench profiles are deterministic. Therefore, the objective function for *time optimal*, *effort optimal*, and *jerk optimal* (as presented in [1]) are deterministically defined. However, a *power optimal* design would necessitate statistical objective functions of the form,

$$\begin{aligned}
\mathrm{J_{S1}} &= E\left[\sum_{i=1}^{n_i}\left|\mathrm{z}_i\tau_i\mathrm{y}_i(\boldsymbol{\xi})\right|\right]\\
&= \textstyle\sum_{i=1}^{n_i}\left|\mathrm{z}_i\tau_i\mathrm{y}_i^0\langle\Psi^0,\Psi^0\rangle\right|,\ \forall t
\end{aligned} \tag{18}$$

where $\mathbf{z}$ is a vector of optional scalarization weights; (18) is the expected power profile.

Designs may necessitate statistically penalizing terminal conditions (TC) of the state trajectory in the objective function (occasionally referred to as *soft* constraints). Two candidates are,

$$\begin{aligned}
\mathrm{J_{S3}} &= \left\|\mu_{e(t_f)}\right\| = \left\|E\left[\boldsymbol{e}\left(t_f;\boldsymbol{\xi}\right)\right]\right\|\\
&= \left\|\boldsymbol{y}_{ref}\left(t_f\right)-\boldsymbol{y}^0\left(t_f\right)\langle\Psi^0,\Psi^0\rangle\right\|
\end{aligned} \tag{19}$$

$$\begin{aligned}
\mathrm{J_{S4}} &= \left\|\sigma^2_{e(t_f)}\right\|\\
&= \left\|E\left[\left(\boldsymbol{e}(\mathrm{t_f};\boldsymbol{\xi})-\mu_{e(t_f)}\right)^2\right]\right\|\\
&= \left\|\textstyle\sum_{j=0}^{n_b}\left(\boldsymbol{y}^j(\mathrm{t_f})\right)^2\langle\Psi^j,\Psi^j\rangle\right\|
\end{aligned} \tag{20}$$

where $\boldsymbol{e}(\mathrm{t_f};\boldsymbol{\xi}) = \boldsymbol{y}_{ref}(\mathrm{t_f}) - \boldsymbol{y}(\mathrm{t_f};\boldsymbol{\xi})$; (19) is the expected value of the TC's error; (20) is the corresponding variance of the TC's error. Notice that due to the orthogonality of the polynomial basis these computations result in a reduced set of efficient operations on the respective stochastic coefficients.

The inequality constraints may also benefit from added statistical information; for example, bounding the expected values can be expressed as,

$$\boldsymbol{\mathcal{C}}(t;\xi) = \underline{\boldsymbol{y}} \le E[\boldsymbol{y}(\boldsymbol{\xi})] \le \overline{\boldsymbol{y}} \tag{21}$$

where $E[\boldsymbol{y}(\boldsymbol{\xi})] = \mu_{\boldsymbol{y}} = \boldsymbol{y}^0\langle\Psi^j,\Psi^j\rangle$, and $\{\underline{\boldsymbol{y}},\overline{\boldsymbol{y}}\}$ are the minimum/maximum output bounds respectively.

Collision avoidance constraints would ideally involve *supremum* and *infimum* bounds,

$$\underline{\boldsymbol{y}} \le \inf\big(\boldsymbol{y}(t;\boldsymbol{\xi})\big),\ \ \sup(\boldsymbol{y}(t;\boldsymbol{\xi})) \le \overline{\boldsymbol{y}} \tag{22}$$

However, one major difficulty with *supremum* and *infimum* bounds is that they are expensive to calculate. An alternative can be to constrain the uncertain configuration in a standard deviation sense; collision constraints would then take the form,

$$\begin{aligned}
\mu_{\boldsymbol{y}} + \sigma_{\boldsymbol{y}} &\le \overline{\boldsymbol{y}}\\
\underline{\boldsymbol{y}} &\le \mu_{\boldsymbol{y}} - \sigma_{\boldsymbol{y}}
\end{aligned} \tag{23}$$

where $stddev[\boldsymbol{y}(\boldsymbol{\xi})] = \sigma_{\boldsymbol{y}} = \sqrt{\sum_{j=1}^{n_b}\mathbf{y}^{\mathrm{j}}\langle\Psi^j,\Psi^j\rangle}$.

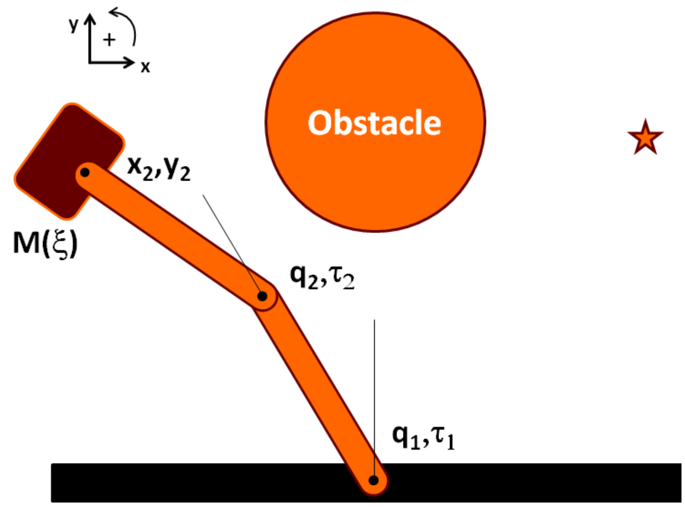

Figure 1— A simple illustration of the fully-actuated *uncertain forward dynamics* motion planning formulation; this problem aims to determine an *effort optimal* motion plan subject to input wrench and geometric collision constraints. This is an uncertain system due to the uncertain mass of the payload.

6 A SERIAL MANIPULATOR PICK-AND-PLACE CASE-STUDY

As an illustration of (17), the serial manipulator "pick-and-place" problem will be used (see Figure 1). The design objective is to minimize the effort it takes to move the manipulator from its initial configuration, $\boldsymbol{q}_0$, to the target configuration, $\boldsymbol{q}_{t_f}$ in a prescribed amount of time, t_f. This results in a deterministic objective function of, $J = \sum_{i=1}^{n_i}\mathrm{z}_i\tau^2$, which is frequently referred to as an *effort optimal* design. However, the payload mass, $\mathrm{M}(\xi)$, is defined to be uncertain rendering the system dynamics uncertain. Since the uncertain serial manipulator is a fully actuated system, where the joints $\boldsymbol{q} = \{q_1, q_2\}$

are actuated with the input wrenches $\boldsymbol{\tau} = \{\tau_1, \tau_2\}$, the motion planning problem may be appropriately defined by (17).

By parameterizing the input wrench profiles with B-Splines, as in (3), (17) results in a finite search problem seeking for spline control points, $\mathbf{P}$, that minimize the actuation effort defined in J. Therefore, the problem's optimization variables are $\boldsymbol{x} = \{\mathbf{P}\}$.

The actuators are bounded in their torque supply and the manipulator should neither hit the wall it's mounted to nor the obstacle. The constraints may therefore be defined as,

$$\boldsymbol{C}: \begin{cases} \underline{\boldsymbol{\tau}} \leq \boldsymbol{\tau} \leq \bar{\boldsymbol{\tau}} \\ \mu_y \pm \sigma_y \leq \mathbf{0} \\ -\mathfrak{D}_{i,j}(\mu_y \pm \sigma_y) \leq 0 \end{cases} \tag{24}$$

where $i = 1{,}2$ and $j = obstacle$ for the signed distance, $\mathfrak{D}_{i,j}(\mu_y \pm \sigma_y)$, measured from each link of the serial manipulator to the obstacle calculated using the statistical mean and standard deviations of the configuration/outputs; and $\{\underline{\tau}, \bar{\tau}\}$ are the minimum/maximum input bounds respectively.

This formulation allows a design engineer to answer the question,

> *Given actuator and obstacle constraints, what's the "effort optimal" motion plan that accounts for all possible systems within the probability space?*

Without accounting for the uncertainty directly in the dynamics and motion planning formulations, design engineers would have a difficult time answering this question.

The solution to this problem with the deterministic formulation, as defined in (4), results in an *effort optimal* solution of $J = 2770\ (\mathrm{Nm})^2$; where $\mathrm{t_f} = 1.5$ seconds; all system parameters are set equal to one, $\theta_i = 1$ (with SI units); initial conditions $\boldsymbol{q}(0) = \{\frac{\pi}{6}, \frac{\pi}{6}\}$ and $\dot{\boldsymbol{q}}(0) = \{0, 0\}$ radians; terminal conditions $\boldsymbol{q}(\mathrm{t_f}) = \{-\frac{\pi}{6}, -\frac{\pi}{6}\}$ and $\dot{\boldsymbol{q}}(\mathrm{t_f}) = \{0, 0\}$ radians; and $\underline{\tau} = -10, \bar{\tau} = 10$ (Nm). The resulting optimal configuration time history is shown in Figure 2.

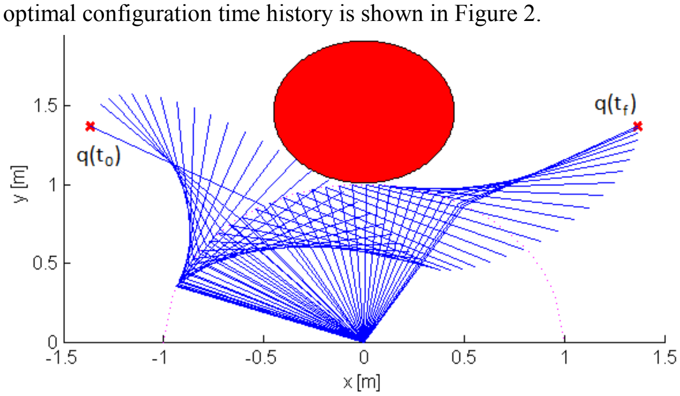

Figure 2—The *effort optimal* configuration time histories for the deterministic serial manipulator 'pick-and-place' problem. This optimal solution resulted in a $J = 2770\ (\mathrm{Nm})^2$ design.

The solution from the new formulation, as defined in (17) with constraints defined by (24), results in an *effort optimal* solution of $J = 3530\ (\mathrm{Nm})^2$; where all system parameters and initial/terminal conditions are defined the same as in the deterministic problem. The only difference in this problem definition, as compared to the deterministic problem, is the uncertain pay-load mass modeled with a uniform distribution having a unity mean and 0.5 variance. The resulting optimal uncertain end-effector Cartesian position time history is illustrated in Figure 3; where the mean and bounding $\mu_y \pm \sigma_y$ time histories are displayed.

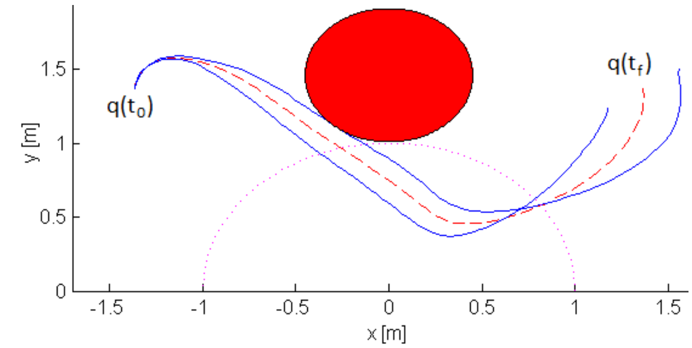

Figure 3—The *effort optimal* uncertain end-effector Cartesian position time history for the uncertain serial manipulator 'pick-and-place' problem. The mean and bounding $\mu_y \pm \sigma_y$ time histories are displayed. This optimal solution resulted in a $J = 3530\ (\mathrm{Nm})^2$ design.

Therefore, the *effort optimal* solution from the uncertain problem resulted in a more conservative answer—$3530\ (\mathrm{Nm})^2$ as compared to $2770\ (\mathrm{Nm})^2$. This is a sensible solution; close inspection of Figure 2 shows the deterministic solution drove the configuration as close to the obstacle as possible. The introduction of uncertainty in the pay-load mass affected the amount of input torque required for the system to reliably avoid the obstacle for all systems within the probability space. In fact, Figure 3 shows the distribution of end-effector Cartesian position trajectory induced by the uncertain pay-load. The uncertain optimal motion plan from (17) effectively pushed the end-effector configuration distribution away from the obstacle; this results in a larger *effort optimal* solution, however, all realizable systems within the probability space of the uncertain mass are now guaranteed to satisfy the constraints. In other words, the *effort optimal* solution to (17) produces the minimum effort design for the entire family of systems. Relying only on the contemporary deterministic problem formulation in (4) results in an unrealizable trajectory for a subset of the realizable systems.

A third study provides some additional insight to what the new framework can provide. By redefining the objective function for (17) as (20) the uncertain design is no longer an *effort optimal* but *terminal variance optimal* design. In other words, the new design question is,

> *Given actuator and obstacle constraints, what motion plan will minimize the variance of the terminal condition's (TC) error when accounting for all possible systems within the probability space?*

The *effort optimal* design resulted in a TC error standard deviation of $\sigma_{e(t_f)} = [0.191, 0.133]\ (m)$; where the standard deviation is the square root of the variance. Redesigning the motion plan using an objective function defined by (20) results in a TC error standard deviation of $\sigma_{e(t_f)} = [0.144, 0.114]\ (m)$, as shown in Figure 4. Therefore, a modest reduction in the TC error standard deviation was realized, however, the effort of the new design increased from $3530\ (\mathrm{Nm})^2$ to $5910\ (\mathrm{Nm})^2$. These results indicate a Pareto optimal trade-off between the effort and TC's variance. Therefore, designers may define a hybrid objective function with a scalarization between the *effort optimal* and *terminal variance optimal* terms.

One additional insight gained from the *terminal variance optimal* design is related to the controllability of an uncertain system's TC variance. If the TC variance was fully controllable then the *terminal* *variance optimal* design would be able to reduce it to zero. This initial investigation indicates that the variance is not fully controllable. A rigorous uncertain system controllability investigation is out of the scope of this work but will be considered for future research.

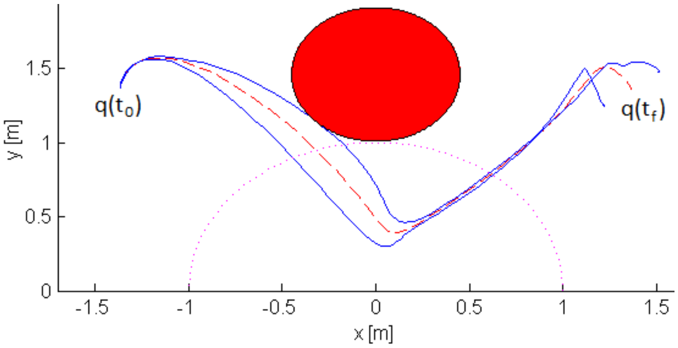

Figure 4—The *terminal variance optimal* uncertain end-effector Cartesian position time history for the uncertain serial manipulator 'pick-and-place' problem. The mean and bounding $\mu_y \pm \sigma_y$ time histories are displayed. This optimal solution resulted in a $J = 5910\ (\mathrm{Nm})^2$ design.

This case study provides insight into the computational complexity of the new framework. Given that the problem only had a single source of uncertainty and used a 5th order gPC approximation, the minimum number of basis terms was, $n_b = 6$. However, since the LSCM method described in Section 4 was used, the total number of collocation points, or samples of the probability space, used was $n_{ncp} = 3 * n_b = 18$. If a Monte Carlo technique was used, the number of probability space samples required for a similar level of approximation accuracy would have been on the order of $O(10^3)$~$O(10^4)$ [16].

Tables 6.1 and 6.2 show the timing results from a Matlab implementation for a single function evaluation and full optimization of both the deterministic and uncertain systems, respectively, where Matlab's parallel For-Loop, PARFOR, was first enabled and then disabled. The starting point for the uncertain system optimization was the solution from the deterministic system solve, therefore, the total calculation time of the uncertain solve is the deterministic time plus the uncertain time (as shown in Table 6.2). These results were obtained on a workstation with 8 Intel Xeon 2.33 GHz CPU cores.

Table 6.1—Function evaluation times (in seconds) for the deterministic and uncertain systems executed with Matlab's parallel For-Loop enabled/disabled. Ratios between the deterministic and uncertain function evaluation times and parallel and non-parallel times are provided.

	Parallel	Non-Parallel	Non/Parallel
dOpt	0.18	0.18	0.99
uOpt	0.64	1.28	0.50
dOpt/uOpt	0.28	0.14	

As can be seen in Table 6.1, the natural parallelism of the LSCM method reduces the computational cost of the uncertain system evaluation to roughly 50% of the non-parallel implementation, and Table 6.2 shows that the parallel uncertain system optimization only increases the computational cost by roughly three times. With an increase in the number of available processors the additional costs of the new framework, presented in Section 5, will reduce. Clearly a production grade implementation in a more efficient language, such as C or Fortran, would yield higher performance, however, the relative timings from the Matlab implementation illustrate the computational complexities of the framework. Therefore, use of a parallelized LCSM based gPC in the new framework allows for efficient optimal motion planning of uncertain dynamical systems; where the additional cost reduces as the number of available parallel processors increases.

Table 6.2—Full optimization solve times (in seconds) for the deterministic and uncertain systems executed with Matlab's parallel For-Loop enabled/disabled. Ratios between the deterministic and uncertain system solve times and parallel and non-parallel times are provided.

	Parallel	Non-Parallel	Parallel/Non
dOpt	1709	5344	0.32
uOpt	5071	33063	0.15
uOpt (total)	6780	38407	0.18
uOpt/dOpt	2.97	6.19	

A final observation is that the *uncertain forward dynamics* motion planning framework embodied in (17) is most applicable to force controlled systems where input wrenches are prescribed. However, configuration/position controlled systems may be better designed through application of the companion framework based on *uncertain inverse dynamics* presented by the authors in [1].

7 CONCLUSIONS

This work has presented a new nonlinear programming based motion planning framework that treats uncertain fully-actuated dynamical systems. The framework allows practitioners to model sources of uncertainty using the Generalized Polynomial Chaos methodology and to solve the *uncertain forward dynamics* using a least-squares collocation method. Subsequently, statistical information of the *uncertain forward dynamics* may be included in the NLP's objective function and constraints. The serial manipulator case study illustrated how the new framework produces an optimal design that accounts for the entire family of systems enabling a practitioner to design an optimally performing system that is also robust.

ACKNOWLEDGEMENTS

This work was partially supported by the Automotive Research Center (ARC), Thrust Area 1

REFERENCES

[1] Hays, J., Sandu, A., and Sandu, C., 2011, "Motion Planning of Uncertain Fully-Actuated Dynamical Systems—an Inverse Dynamics Formulation," ASME IDETC/CIE Conference, Washington, DC, USA, pp. (submitted).

[2] Greenwood, D., 2003, *Advanced Dynamics*, Cambridge Univ Pr,

[3] Murray, R., Li, Z., Sastry, S., and Sastry, S., 1994, *A Mathematical Introduction to Robotic Manipulation*, CRC Press, Inc, Boca Raton, FL, USA.

[4] Lavalle, S., 2006, *Planning Algorithms*, Cambridge Univ Press, New York, NY, USA.

[5] Choset, H., 2005, *Principles of Robot Motion: Theory, Algorithms, and Implementation*, The MIT Press, Cambridge, MA, USA.

[6] Karaman, S., and Frazzoli, E., 2010 (submitted), "Incremental Sampling-Based Algorithms for Optimal Motion Planning," International Journal of Robotics Research, pp.

[7] Park, J., 2007, Industrial Robotics, Programming, Simulation and Applications, Verlag, Croatia, Optimal Motion Planning for Manipulator Arms Using Nonlinear Programming.
[8] Sohl, G. A., and Bobrow, J. E., 2001, "A Recursive Multibody Dynamics and Sensitivity Algorithm for Branched Kinematic Chains," Transactions of the ASME. Journal of Dynamic Systems, Measurement and Control, 123 (Copyright 2002, IEE), pp. 391-9.
[9] Bobrow, J., Martin, B., Sohl, G., Wang, E., Park, F., and Kim, J., 2001, "Optimal Robot Motions for Physical Criteria," Journal of Robotic systems, 18 (12), pp. 785-795.
[10] Sohl, G., 2000, "Optimal Dynamic Motion Planning for Underactuated Robots," PhD thesis, University of California, Irvine.
[11] Xiang, Y., Arora, J., and Abdel-Malek, K., 2010, "Physics-Based Modeling and Simulation of Human Walking: A Review of Optimization-Based and Other Approaches," Structural and Multidisciplinary Optimization, 42 (1), pp. 1-23.
[12] Kim, H., Wang, Q., Rahmatalla, S., Swan, C., Arora, J., Abdel-Malek, K., and Assouline, J., 2008, "Dynamic Motion Planning of 3d Human Locomotion Using Gradient-Based Optimization," Journal of biomechanical engineering, 130, pp. 031002.
[13] Lee, S. H., Kim, J., Park, F. C., Kim, M., and Bobrow, J. E., 2005, "Newton-Type Algorithms for Dynamics-Based Robot Movement Optimization," Robotics, IEEE Transactions on, 21 (4), pp. 657-667.
[14] Piegl, L. A., and Tiller, W., 1997, *The Nurbs Book*, Springer Verlag, Berlin, Germany.
[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, 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.
[18] 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.
[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] 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.
[21] Xiu, D., 2007, "Efficient Collocational Approach for Parametric Uncertainty Analysis," Communications in Computational Physics, 2 (2), pp. 293-309.
[22] Xiu, D., 2009, "Fast Numerical Methods for Stochastic Computations: A Review," Communications in Computational Physics, 5 (2-4), pp. 242-272.
[23] Erdmann, M., 1984, "On Motion Planning with Uncertainty," Masters thesis, Massachusetts Institute of Technology, Boston.
[24] Barraquand, J., and Ferbach, P., 1995, "Motion Planning with Uncertainty: The Information Space Approach," International Conference on Robotics and Automation, 2, pp. 1341-1348.
[25] Park, W., Liu, Y., Zhou, Y., Moses, M., and Chirikjian, G., 2008, "Kinematic State Estimation and Motion Planning for Stochastic Nonholonomic Systems Using the Exponential Map," Robotica, 26 (04), pp. 419-434.
[26] Kewlani, G., Ishigami, G., and Iagnemma, K., 2009, "Stochastic Mobility-Based Path Planning in Uncertain Environments," pp. 1183-1189.