**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 an inverse dynamics formulation where deterministic state trajectories are prescribed and uncertain actuator inputs are quantified. The resulting design determines a feasible time optimal motion plan—subject to actuator and obstacle avoidance constraints—for all possible systems within the probability space. The forward dynamics formulation (using deterministic actuator inputs and uncertain state trajectories) 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 downstream 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 an inverse dynamics formulation where deterministic state trajectories are prescribed and uncertain actuator inputs are quantified. The benefits of the ability to quantify the resulting actuator uncertainty are illustrated in a time optimal motion planning case-study of a serial manipulator pick-and-place application. The resulting design determines a time optimal motion plan—subject to actuator and obstacle avoidance constraints—for all possible systems within the probability space.

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

It’s important to point out that the new framework is not dependent on the specific formulation of the dynamical 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 inverse dynamics; Section 5 introduces the new framework for motion planning of uncertain fully-actuated dynamical systems based on an uncertain inverse 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],

$$
\mathbf{M}(q(t), \dot{q}(t))\ddot{q}(t) + \mathbf{C}(q(t), \dot{q}(t), \theta(t))\dot{q}(t) + \mathbf{N}(q(t), \dot{q}(t), \theta(t)) = \tau(t)
$$

where $q(t) \in \mathbb{R}^{n_d}$ 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); $\dot{q}(t) \in \mathbb{R}^{n_d}$ are the rates of the generalized coordinates and $\ddot{q}(t)$ are the associated accelerations—using Newton’s dot notation for a time derivative; $\theta(t) \in \mathbb{R}^{n_l}$ includes system parameters of interest (specifically, those with uncertainty as described in Section 4); $\mathbf{M}(\dot{q}(t), \ddot{q}(t)) \in \mathbb{R}^{n_d \times n_d}$ is the square positive definite inertia matrix; $\mathbf{C}(q(t), \dot{q}(t), \theta(t)) \in \mathbb{R}^{n_d \times n_d}$ includes centrifugal, gyroscopic and Coriolis effects; $\mathbf{N}(q(t), \dot{q}(t), \theta(t)) \in \mathbb{R}^{n_d}$ the generalized gravitational and joint forces; and $\tau(t) \in \mathbb{R}^{n_l}$ are the $n_l$ 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 $q(0) = q_0$ and $\dot{q}(0) = \ddot{q}_0$. Also, the system measured outputs are defined by,

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

where $y \in \mathbb{R}^{n_l}$ with $n_l$ equal to the number of outputs. A close inspection of (1) shows that the wrench inputs of fully-actuated systems can be calculated through direct algebraic evaluations if the state trajectories are “known”, or prescribed,

$$
\tau = \mathcal{F}(q, \dot{q}, \ddot{q}, \theta)
$$

This is the inverse dynamics (ID) formulation and does not require numerical integration and can result in significant computational savings.

3 DETERMINISTIC INVERSE 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 [7–12].

Use of an inverse dynamics formulation requires a practitioner to define the state trajectory, $(q, \dot{q}, \ddot{q})$, 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 system’s configuration, $q$, can be represented with B-Splines,

$$
q(P, t) = \sum_{i=0}^{n_p} \beta_i \psi_i(t) p^i
$$

where $p \in \mathbb{R}^{n_d}$ are $n_p$ control points; $\beta$ and $p$ are the B-Splines’ basis functions and degree, respectively. The corresponding parameterizations for $(q, \dot{q}, \ddot{q})$ are also B-Splines derived from $q(P, t)$ [13]; $P \in \mathbb{R}^{n_p \times n_d}$ is an vector of control points, $p$.

Once the state trajectories have been parameterized the NLP-based deterministic motion planning problem may be formulated as,

$$
\begin{align*}
\min_{x \in [P]} & \quad J \\
\text{s.t.} & \quad \tau = \mathcal{F}(q, \dot{q}, \ddot{q}, \theta) \\
& \quad y = \mathcal{O}(q, \dot{q}, \theta) \\
& \quad \mathcal{C}(y, \tau, \theta) \leq 0
\end{align*}
$$

where the state’s explicit dependence on their associated control points has been dropped for notational brevity. Equation (5) seeks to find the control points $P$ that minimize some prescribed objective function, $J$, while being subject to the inverse dynamic constraints defined in (3). 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, $\mathcal{C}(y, \tau, \theta) \leq 0$.

The literature contains a variety of objective function definitions for $J$ when used in a motion planning setting. Some commonly defined objective functions are,

$$
\begin{align*}
J_{D1} &= t_f \\
J_{D2} &= \sum_{i=1}^{n_l} t_i^2, \quad \forall t \\
J_{D3} &= \sum_{i=1}^{n_l} |rf_i|, \quad \forall t \\
J_{D4} &= \sum_{i=1}^{n_l} t_i^2, \quad \forall t
\end{align*}
$$

where (6) represents a time optimal design; (7) minimizes effort, (8) power, and (9) jerk.

The solution to (5) produces an optimal motion plan under the assumption that all system properties are known (i.e., (3) is completely deterministic). The primary contribution of this work is the
presentation of a variant of (5) that allows (3) 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 input wrenches.

4 GENERALIZED POLYNOMIAL CHAOS

Generalized Polynomial Chaos (gPC), first introduced by Wiener [14], is an efficient method for analyzing the effects of uncertainties in second order random processes [15]. This is accomplished by approximating a source of uncertainty, \( \theta \), 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,

\[
\theta(\xi) = \sum_{j=0}^{p_o} \theta^j \psi^j(\xi(\omega)) \tag{10}
\]

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 \( \theta \) that maps the random event \( \omega \in \Omega \), from the sample space, \( \Omega \), to the domain of the orthogonal polynomial basis (e.g., \( \xi; \Omega \rightarrow [-1,1] \)).

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

\[
\langle \psi^i(\xi), \psi^j(\xi) \rangle = \int_{-1}^{1} \psi^i(\xi) \psi^j(\xi) w(\xi) d\xi = 0, \quad \text{for } i \neq j \tag{11}
\]

where \( w(\xi) \) is the weighting function that is equal to the joint probability density function of the random variable \( \xi \). 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 (11).

Generalized Polynomial Chaos can be applied to multibody dynamical systems described by differential equations [16, 17]; 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) [1], or uncertain inputs, as in an inverse dynamics formulation (3). Therefore, proceeding with the inverse dynamics formulation, the uncertain input wrenches can be approximated in a similar fashion as (10),

\[
\tau_i(t; \xi) = \sum_{j=0}^{n_b} \tau^j_i(t) \psi^j \left( \xi \right), \quad i = 1 ... n_i \tag{12}
\]

where \( \tau^j_i(t) \in \mathbb{R}^{n_b} \) again represents the stochastic coefficients—for the \( i \)th input wrench—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 inputs 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, \( \xi \) becomes a vector of random variables, \( \xi = [\xi_1, ..., \xi_{n_p}] \in \mathbb{R}^{n_p} \) and maps the sample space, \( \Omega \), to an \( n_p \) dimensional cuboid, \( \xi; \Omega \rightarrow [-1,1]^{n_p} \) (as in the example of Jacobi chaoses).

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

\[
\psi^j = \psi^j_1 \psi^j_2 ... \psi^j_{n_p}, \quad i_k = 0 ... p_o, k = 1 ... n_p \tag{13}
\]

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 \( n_b = \frac{(p_o + 1)^{n_p}}{n_p^{p_o}} \) basis terms. The minimal basis set can be determined by the products resulting from these index ranges,

\[
i_1 = 0 ... p_o, \quad i_2 = 0 ... (p_o - i_1), ..., \quad i_{n_p} = 0 ... (p_o - i_1 - i_2 - ... - i_{n_p-1}) \tag{14}
\]

The number of multidimensional terms, \( n_b \), grows quickly with the number of uncertain parameters, \( n_p \), and polynomial order, \( p_o \).

Sanda et. al. showed that gPC is most appropriate for modeling systems with a relatively low number of uncertainties [16, 17] but can handle large nonlinear uncertainty magnitudes.

Substituting (10) and (12) into (3) produces the following uncertain inverse dynamics (UID),

\[
\sum_{j=0}^{n_b} \tau^j_i(t) \psi^j \left( \xi_k \right) = F \left( q, \dot{q}, \ddot{q}, \Theta_r(t; \mu) \right), \quad i = 1 ... n_b, k = 1 ... n_p \tag{15}
\]

where the unknowns are now the \( n_b n_i \) stochastic input coefficients, \( \tau^j_i(t) \).

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 \( n_b \) random variables, \( \xi \).

The Galerkin Projection Method (GPM) is a commonly used method for solving (14), however, this is a very intrusive technique and requires a custom formulation of the inverse dynamic EOMs. As an alternative, sample-based collocation techniques can be used without the need to modify the base EOMs.

Sanda et. al. [16, 18] showed that the collocation method solves (14) by solving (3) at a set of points, \( q, \dot{q}, \ddot{q}, \Theta_r(t; \mu) \), selected from the \( n_p \) dimensional domain of the random variables \( \xi \in \mathbb{R}^{n_p} \). Meaning, at any given instance in time, the random variables’ domain is sampled and solved \( n_p \) times with \( \xi = \mu \) (updating the approximations of all sources of uncertainty for each solve), then the known stochastic input coefficients \( \tau^j_i(t) \) can be determined at that given time instance. This can be accomplished by defining the intermediate variables,

\[
k^j \Theta_r(t; \mu) = \sum_{j=0}^{n_b} \tau^j_i(t) \psi^j \left( \mu \right), \quad i = 1 ... n_b, k = 0 ... n_p \tag{16}
\]

and substitute them into (14). This yields,

\[
k^j \Theta_r(t; \mu) = F \left( q, \dot{q}, \ddot{q}, k^j \Theta_r(t; \mu) \right), \quad i = 1 ... n_i, k = 0 ... n_p, r = 1 ... n_p \tag{17}
\]

where,

\[
k^j \Theta_r(t; \mu) = \sum_{j=0}^{p_o} \psi^j \left( \mu \right), \quad k = 0 ... n_p, r = 1 ... n_p \tag{18}
\]

Equation (16) provides a set of \( n_p \) independent equations whose solutions determine the stochastic coefficients, \( \tau^j_i(t) \). This is accomplished by recalling the relationship of the stochastic
coefficients to the solutions, $kT_i$, shown in (15). In matrix notation (15) can be expressed for all inputs,

$$T_i = (\tau(t))^T \Psi(\mu), \quad i = 1 \ldots n_t$$

(18)

where the matrix,

$$A_{k,j} = \Psi_j(\mu), \quad j = 0 \ldots n_b, k = 0 \ldots n_{cp}$$

(19)

is defined as the collocation matrix. It’s important to note that $n_b \leq n_{cp}$. The stochastic coefficients can now be solved for using (18),

$$\tau_i^j(t) = A_i^T T_i, \quad i = 1 \ldots n_t, j = 0 \ldots n_b$$

(20)

where $A_i^T$ is the pseudo inverse of $A$ if $n_b < n_{cp}$. If $n_b = n_{cp}$, then (20) is simply a linear solve. However, [18] presented the least-squares collocation method (LSCM) where the stochastic state coefficients are solved for, in a least squares sense, using (20) when $n_b < n_{cp}$. [18] also showed that as $n_{cp} \to \infty$ the LSCM approaches the GPM solution; where 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 $p_0 \to \infty$.

The unintrusive nature of the LSCM sampling approach is arguably its greatest benefit; (3) 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 [16, 18-21] for more information.

5 UNCERTAIN INVERSE 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, 22] and frequently only treats the uncertainty as a pseudo-random process. The literature thus far has primarily addressed sensor and/or actuator noise [4, 22] and frequently only treats the uncertainty as a pseudo-random process.

In [25], 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, in that the author has adapted the gPC framework to model uncertain fully-actuated multibody dynamical systems, formulated with uncertain inverse dynamics, is,

$$\min_{x \in \mathcal{P}} \quad J(t; \xi)$$

s.t. $\tau(t; \xi) = F(q, q, \dot{q}, \Theta(t; \xi))$

$$y = 0(q, \dot{q}, \ddot{q})$$

$$C(y, \tau, \theta(t; \xi)) \leq 0$$

(21)

where (21) is a reformulation of (5) with uncertain actuator inputs. The most interesting part of (21) 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. Recalling the definitions of an expected value and variance, (7)–(9) may be redefined statistically:

$$J_{S1} = E \left[ \sum_{i=1}^{n_t} z_i(\tau(t_i; \xi))^2 \right]$$

(22)

$$= \sum_{i=1}^{n_t} \sum_{j=0}^{n_b} z_i^j(\tau_i^j)^2 \langle \psi^j, \psi^j \rangle \}, \forall t$$

$$J_{S2} = E \left[ \sum_{i=1}^{n_t} z_i(\tau_i; \xi^j)^2 \right]$$

$$= \sum_{i=1}^{n_t} \sum_{j=0}^{n_b} z_i^j(\tau_i^j)^2 \langle \psi^j, \psi^j \rangle \}, \forall t$$

(23)

$$J_{S3} = E \left[ \sum_{i=1}^{n_t} z_i(\tau_i; \xi^j)^2 \right]$$

$$= \sum_{i=1}^{n_t} \sum_{j=0}^{n_b} z_i^j(\tau_i^j)^2 \langle \psi^j, \psi^j \rangle \}, \forall t$$

(24)

where $z$ is a vector of optional scalarization weights; (22) encapsulates expected effort; (23) expected power; and (24) expected jerk. 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. When using inverse dynamics, the input wrenches are uncertain. This uncertain quantity may also be bound by physical limits of the actuator; as an example, inequality constraints may be formulated as,

$$C = \{ \mu_{\tau_i} + \sigma_{\tau_i} \leq \bar{\tau}, \quad \bar{\tau} \leq \mu_{\tau_i} - \sigma_{\tau_i}, \quad i = 1,2 \}$$

(25)

where the mean $\mu_{\tau_i} = \bar{\tau}$ is defined as in (22)–(24), the standard deviation $\sigma_{\tau_i} = \sqrt{\sum_{i=1}^{n_b} \tau_i^2}$ is the root of the variance, and $\{\bar{\tau}, \bar{\tau}\}$ are the minimum/maximum input bounds respectively.

Deterministic terms, such as (6)–(9), may be combined with appropriately selected statistically based terms, such as (22)–(25), to form a final motion planning problem. This will be illustrated in the serial manipulator case-study in the following section.

![Figure 1](http://proceedings.asmedigitalcollection.asme.org/)

**Figure 1**—A simple illustration of the fully-actuated uncertain inverse dynamics motion planning formulation; this problem aims to determine a time 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 (21), the serial manipulator “pick-and-place” problem will be used (see Figure 1). The design objective is to minimize the time it takes to move the manipulator from its initial configuration, $q_0$, to the target configuration, $q_{Tt}$. This results in a
The time optimal wrench distribution for the deterministic serial manipulator 'pick-and-place' problem. This optimal solution resulted in a $t_f = 1.12$ (s).

Figure 3—The time optimal uncertain input wrench time histories for the uncertain serial manipulator ‘pick-and-place’ problem. Each input wrench is displaying its mean value and bounding $\mu_i + \sigma_i$ time histories. This optimal solution resulted in a $t_f = 1.2$ (s).

Therefore, the time optimal solution from the uncertain problem resulted in a more conservative answer (1.2 seconds as compared to 1.12 seconds). This is a sensible solution; close inspection of Figure 2 shows the deterministic solution drove the input wrenches to their extreme bounds of $\pm 10$ (Nm) at certain points during the motion profile. Clearly, introducing the uncertain mass to the system affected the amount of input torque required for the system to reliably follow the specified state trajectory. In fact, Figure 3 shows the distribution of input wrenches induced by the uncertain mass. The uncertain optimal motion plan from (21) effectively pushed the input wrench distribution inside the actuation limits, $\{T, \tau\}$; this results in a slower time 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 time optimal solution to (21) produces the minimum
time for the entire family of systems. Relying only on the contemporary deterministic problem formulation in (5) results in an unrealizable trajectory for a subset of the realizable systems.

Additionally, the author’s companion paper [1] presents data showing that 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.

A final observation is that the uncertain inverse dynamics motion planning framework embodied in (21) is most applicable to configuration/position controlled systems, where states are prescribed as they are in (21). However, force controlled systems may be better designed through application of the companion framework based on uncertain forward dynamics presented by the authors in [1].

**Figure 4**—The final optimal configuration time history of the uncertain serial manipulator ‘pick-and-place’ application involving collision avoidance and actuator constraints.

7 CONCLUSIONS

This work presents 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 inverse dynamics using a least-squares collocation method. The uncertainty aware design is obtained by including statistical information of the uncertain inverse dynamics 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