Computers and Chemical Engineering 34 (2010) 1953–1961

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/compchemeng

Stochastic dynamic predictions using Gaussian process models for nanoparticle synthesis

Andres F. Hernandez, Martha A. Grover*

311 Ferst Dr. NW, School of Chemical & Biomolecular Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0100, USA

ARTICLE INFO

Article history:
Received 9 December 2009
Received in revised form 15 July 2010
Accepted 16 July 2010
Available online 23 July 2010

Keywords:
Gaussian process model
Dynamic systems modeling
Spatial statistics
Reduced-order model
Nanoparticle synthesis

ABSTRACT

Gaussian process modeling (also known as kriging) is an empirical modeling approach that has been widely applied in engineering for the approximation of deterministic functions, due to its flexibility and ability to interpolate observed data. Despite its statistical properties, Gaussian process models (GPM) have not been employed to describe the dynamics of stochastic systems with multiple outputs. Our paper presents a methodology to construct approximate models for multivariate stochastic dynamic simulations using GPM, by combining ideas from design of experiments, spatial statistics and dynamic systems modeling. The methodology is the first application in dynamic systems modeling that combines parameter and state uncertainty propagation in Gaussian process models. We apply the methodology in the prediction of a dynamic size distribution during the synthesis of nanoparticles. The method is robust to the simulation noise, and is able to learn the dynamics using a small number of sequentially designed samples of the nanoparticle simulation.

1. Introduction

The complexity of dynamic models in engineering has been increasing in the last decades. Dynamic models span from simple linear approximations to detailed simulations with a large number of parameters to be specified. Some examples include fluid and atmospheric dynamics, combustion chemistry and nanoscale phenomena. Current research in nanoscale phenomena includes both experimental studies to understand the principles that govern these systems and also modeling efforts to mimic and predict the molecular behavior at the nanoscale level. Two common computational tools for nanoscale modeling are molecular dynamics and kinetic Monte Carlo simulations (Chatterjee & Vlachos, 2007; Ghoniem, Busso, Kioussis, & Huang, 2003). Unfortunately, these detailed and complex simulations are not well suited for more practical engineering tasks like process control and optimization. Simulation for these purposes requires a reduction in the computational effort without losing significant accuracy in the prediction. The development of accurate representations of the complex simulations by combining model-reduction methodologies with novel numerical methods could be a valuable tool to enable systems engineering of nanoscale systems.

The seminal paper by Pope (1997) presented a novel mathematical approach to create approximations for complex dynamic simulations based on pre-computed function evaluations. Pope proposed *in situ* adaptive tabulation (ISAT) for the creation of a database that stores dynamic simulations. By inspecting the table, a prediction for new entries is made using nearby elements in the database. A different approach called cell mapping (Hsu, 1980) divides the state-space into regions or "cells", and uses the database to associate constant values to each cell as an approximation of the complex dynamic simulation. These two approaches are based on local descriptions of the information contained in the database, suggesting that a large number of simulations are needed to completely characterize the system over the region of interest. From a different point of view, pre-computed function evaluations can be used to fit a dynamic mapping function over the state-space. This dynamic mapping function represents the evolution of the state from one time step to the next one. Polynomial regression functions (Brad, Tomlin, Fairweather, & Griffiths, 2007) and artificial neural networks (Blasco, Fueyo, Dopazo, & Ballester, 1998) have been used to model this mapping process in the context of combustion chemistry.

Gaussian process modeling (GPM) has been applied in machine learning as a nonparametric model for classification and regression problems. GPM combines global trend functions in the data with spatially correlated functions to improve the prediction, making GPM attractive for global and local interpolation. GPM has been applied in the field of dynamic systems modeling (Azman & Kocijan, 2007; Deisenroth, Rasmussen, & Peters, 2009; Gregorcic & Lightbody, 2009; Vinet & Vazquez, 2008) because of its flexibility

* Corresponding author. Tel.: +1 404 894 2878; fax: +1 404 894 2866.
E-mail address: martha.grover@chbe.gatech.edu (M.A. Grover).

0098-1354/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compchemeng.2010.07.023

and because it predicts the level of confidence in the model prediction.

This work presents an application of GPM as a surrogate of complex dynamic simulations for nanoscale phenomena. The structure of this paper is as follows. Section 2 explains the synthesis of platinum nanoparticles as a case study for GPM. Section 3 presents a brief background of GPM and its application in dynamic systems modeling. Section 4 describes our methodology contribution, first presenting a GPM reduced-order model with input and parameter uncertainty propagations and then, a sequential design of computer experiments approach that improves the reduced-order model. Section 5 shows the results of using GPM to approximate the dynamics of a nanoparticle size distribution and Section 6 summarizes the article.

2. Nanoscale system: synthesis of platinum nanoparticles

Metal nanoparticles have a wide range of industrial applications in areas including catalysis, microelectronics, magnetics, electrochemistry and optics (Saquing, Kang, Aindow, & Erkey, 2005). In particular, platinum nanoparticles are used as the electrocatalyst in polymer electrolyte membrane (PEM) and phosphoric acid fuel cells. One of the most challenging problems in nanoparticle synthesis is the controlled generation of a monodisperse size distribution while sustaining a high yield. Because of the small size scale, atomic scale discreteness must be included in modeling the manufacturing process.

To analyze this manufacturing problem, we describe our model and simulations for platinum deposition on a porous alumina support under supercritical CO_2 conditions (Gummalla, Tsapatsis, Watkins, & Vlachos, 2004), including single adatom processes (Ratsch & Venables, 2003). The first step in the synthesis is the adsorption of an organometallic platinum precursor P in supercritical CO_2 in the presence of hydrogen. The adsorption step describes the deposition of P on the surface of the alumina support. The second step is the nucleation of platinum nanoparticles. The formation of stable platinum nuclei occurs when two adsorbed precursor molecules on the alumina surface (from now on, called intermediate I) react, releasing the organic ligands in the precursor from the elemental platinum atoms to form the nuclei. Finally, the growth step is a chemical reduction reaction where hydrogen reduces the intermediate I to release elemental platinum atoms that incorporate into the platinum nanoparticle in an autocatalytic reaction. Table 1 contains a formal description of the three chemical reactions with their reaction rate expressions employed in this work.

In Table 1, M (mol/cm^3) is the total concentration of platinum nanoparticles on the alumina surface and OL represents the remaining organic ligand in the organometallic precursor. Using the reaction rate expressions in Table 1, a discrete population balance model is used to describe the nanoparticle size distribution. The complete system of differential equations is presented as follows

$$\begin{aligned}
\frac{d[H_2]}{dt} &= -r_{ads} - r_{gro} \\
\frac{d[I]}{dt} &= r_{ads} - 2r_{nuc} - r_{gro} \\
\frac{d[P]}{dt} &= -r_{ads} \\
G &= \frac{r_{gro}}{\sum_{i=2}^{T}[Pt_i]} = \frac{r_{gro}}{M} \\
\frac{d[Pt_2]}{dt} &= r_{nuc} - G[Pt_2] \\
\frac{d[Pt_i]}{dt} &= G[Pt_{i-1}] - G[Pt_i] \quad i = 3, \ldots, T-1 \\
\frac{d[Pt_T]}{dt} &= G[Pt_{T-1}]
\end{aligned} \tag{1}$$

Table 1
Reaction rate expressions to model the synthesis of platinum nanoparticles.

Reaction steps	Reaction rate expression	Rate constant
$H_2 + P \xrightarrow{k_{ads}} I + \text{Byproducts}$	$r_{ads} = k_{ads}[H_2][P]$	k_{ads}=1 × 10^6 cm^3/(mol s)
$2I \xrightarrow{k_{nuc}} M + 2OL$	$r_{nuc} = k_{nuc}[I]^2$	k_{nuc}=1 × 10^3 cm^3/(mol s)
$I + H_2 \xrightarrow{M,\ k_{gro}} M + OL$	$r_{gro} = k_{gro}[I][H_2]^{0.5}$	k_{gro}=2 × 10^1 cm$^{1.5}$/(mol$^{0.5}$ s)

where $[Pt_i]$ (mol/cm^3) is the concentration of nanoparticles with i platinum atoms. The growth rate of nanoparticles, G (s^{-1}), is assumed to be independent of nanoparticle size. The number of bins used to describe the distribution was T = 100 and the system of ODEs was solved up to t = 60 s, when the system reaches a steady-state condition, with a sampling time of Δt = 1 s.

The system described by Eq. (1) models the macroscopic dynamics of the hydrogen and platinum precursor concentrations as well as the nanoscale dynamics of the nanoparticle distribution. The observed dynamic trajectory from this detailed simulation is the solution of Eq. (1) plus a white noise term in the evolution of the moments to generate the stochastic behavior. The set of potential initial concentrations of H_2 and P is

$$\begin{aligned}
H_2(t=0) &= [1 \times 10^{-4} – 5 \times 10^{-5}] \quad (\text{mol/cm}^3) \\
P(t=0) &= [1 \times 10^{-5} – 5 \times 10^{-6}] \quad (\text{mol/cm}^3)
\end{aligned} \tag{2}$$

This range of initial concentrations is our desired set of operating conditions for the nanoparticle system. Therefore, if an approximated model is required as a surrogate of the detailed simulation, this approximated model should describe the dynamic behavior of the system under any set of initial concentrations in this range.

3. Background

3.1. Gaussian process model (GPM)

Gaussian process regression, also known as kriging (Cressie, 1993), is a modeling approach for generalized linear regression models which accounts for the correlation in the residuals between the regression model and the observations (Martin & Simpson, 2005). To explain this correlation, assume a set S of input/output pairs of observations $\{(\mathbf{x}_i, y_i)\}$, where $\mathbf{x}_i \in \mathbb{R}^d$, $y_i \in \mathbb{R}$, $i = 1, \ldots, n$. In a GPM, all output observations y_i can be represented as

$$y_i = \mathbf{h}^T(\mathbf{x}_i)\boldsymbol{\beta} + z[0, V(S)] \tag{3}$$

where $\mathbf{h}(\mathbf{x}_i) \in \mathbb{R}^p$ and $\boldsymbol{\beta} \in \mathbb{R}^p$ explain the mean behavior of the observations as a linear model. Usually $\mathbf{h}^T(\mathbf{x})$ is modeled as a constant regression function or as a set of polynomial regression functions evaluated at the input $\mathbf{x}$. The second residual term $z \in \mathbb{R}$ is defined as zero mean with stationary regression covariance matrix $V \in \mathbb{R}^{n \times n}$.

The mathematical structure of the regression covariance matrix V describes the correlation between the residuals z in the set S. The usual stationary assumptions in the GPM regression covariance matrix V model the correlation as a function of the *distance* between input points $\mathbf{x}_i$ and $\mathbf{x}_j$. With this assumption, residuals at nearby input points are highly correlated. The most commonly used covariance function is (Rasmussen & Williams, 2006)

$$V_{ij}(\mathbf{x}_i, \mathbf{x}_j) = \sigma_c^2 \exp\left[-\frac{1}{2}\sum_{k=1}^{d}\frac{(\mathbf{x}_{i,k} - \mathbf{x}_{j,k})^2}{\omega_k^2}\right] + \sigma_u^2\delta_{ij} \tag{4}$$

where δ_{ij} is the Kronecker delta and $\boldsymbol{\theta} = [\sigma_c^2, \sigma_u^2, \omega_1, \ldots, \omega_d]$ are the unknown parameters that control the features of the correlation between inputs $\mathbf{x}_i$ in the set S. In particular, σ_u^2 is included to model a spatially uncorrelated residual that could be observed in the output

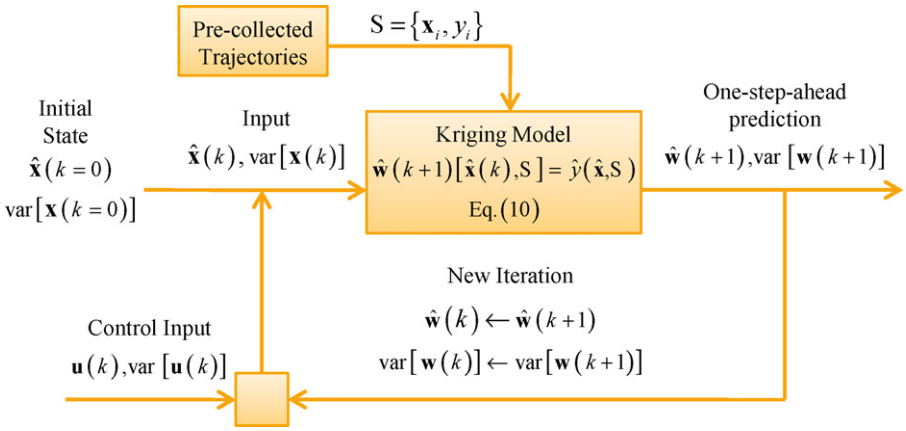

Fig. 1. Graphical representation of a dynamic Gaussian process model.

due to measurement noise. The stationary property is not the only way to represent the spatial dependence of the information. Other correlation functions using dot products between input points or periodic functions can be used, as long as the regression covariance matrix V is positive definite (Lophaven, Nielsen, & Sondergaard, 2002; Rasmussen & Williams, 2006).

Using the mathematical formulation from Eqs. (3) and (4), a linear predictor for an output y at a new input $\mathbf{x}$ can be constructed using the output observations in the set S. The linear predictor is estimated by the minimization of the prediction variance, constrained by the unbiased condition (Goldberger, 1962). This generalized linear regression model describes a Gaussian distribution of the output y given the input $\mathbf{x}$ and the set S:

$$(y|\mathbf{x}, S) \sim N[E(y|\mathbf{x}, S), var_y(\mathbf{x}, S)] \tag{5}$$

where

$$E(y|\mathbf{x}, S) = \hat{y}(\mathbf{x}, S) = \mathbf{h}(\mathbf{x})^T \hat{\boldsymbol{\beta}} + \mathbf{v}^T(\mathbf{x}, S) \cdot V^{-1}(S, S) \cdot (\mathbf{y} - H(S)\hat{\boldsymbol{\beta}}) \tag{6}$$

$$var_y(\mathbf{x}, S) = V(\mathbf{x}, \mathbf{x}) - [\mathbf{h}(\mathbf{x})^T \mathbf{v}^T(\mathbf{x}, S)] \begin{bmatrix} 0 & H^T(S) \\ H(S) & V(S, S) \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{h}(\mathbf{x}) \\ \mathbf{v}(\mathbf{x}, S) \end{bmatrix} \tag{7}$$

and $\mathbf{v}(\mathbf{x}, S) \in \mathbb{R}^n$ is the correlation vector between the new input $\mathbf{x}$ and each input $\mathbf{x}_i$ in the set S, using the correlation in Eq. (4). The terms $\mathbf{h}(\mathbf{x}) \in \mathbb{R}^p$, $H(S) \in \mathbb{R}^{n \times p}$ represent the set of p regression functions evaluated at the unknown input $\mathbf{x}$ and the inputs $\mathbf{x}_i$ in S, respectively. These regression functions are used to model the overall trends on the mean behavior of y over the input region. The vector $\mathbf{y} \in \mathbb{R}^n$ is simply the vector of observations such that $\mathbf{y} = [y_1, y_2, \ldots, y_n]^T$. The parameter estimate $\hat{\boldsymbol{\beta}} \in \mathbb{R}^p$ is the generalized least-squares estimator of the regression coefficients which corresponds to

$$\hat{\boldsymbol{\beta}}(\theta, S) = [H^T(S)V^{-1}(S, S)H(S)]^{-1}[H^T(S)V^{-1}(S, S)\mathbf{y}] \tag{8}$$

GPM can also be understood in the Bayesian framework as a distribution of functions with a characteristic covariance matrix or kernel matrix (Koehler & Owen, 1996), resulting in similar or equivalent expressions. Finally, the estimation of the parameter vector $\boldsymbol{\theta}$ in the regression covariance matrix is usually made by maximizing the log-likelihood function $\ln L$, over the multivariate distribution of the outputs in the set S as follows

$$\ln L(\boldsymbol{\theta}, S) = -\frac{n}{2}\ln(2\pi) - \frac{n}{2}\ln(|V(S, S)|) - \frac{1}{2}(\mathbf{y} - H(S)\hat{\boldsymbol{\beta}})^T \cdot V^{-1}(S, S) \cdot (\mathbf{y} - H(S)\hat{\boldsymbol{\beta}}) \tag{9}$$

3.2. Application of GPM in dynamic systems modeling

An approximated model for dynamic systems can be formulated based on the GPM structure presented in Section 3.1. A GPM can be used as an autoregressive model where the outputs are fed back and employed as inputs to predict a new output. Using the autoregressive formulation, GPM has been implemented to represent discrete-time dynamic systems (Azman & Kocijan, 2007). A recursive state-space formulation for the state variables $\mathbf{w} \in \mathbb{R}^m$ and control inputs $\mathbf{u} \in \mathbb{R}^l$ can be written as (Hernandez & Grover Gallivan, 2008):

$$\begin{aligned} &\hat{\mathbf{w}}(k+1) = \hat{f}[\hat{\mathbf{x}}(k), S] \quad k = 0, 1, 2, \ldots \\ &\hat{\mathbf{x}}(k) = [\hat{\mathbf{w}}(k), \mathbf{u}(k)] \\ &t = k\Delta t \\ &S = \{(\mathbf{x}_i,\ y_i)\},\ \ y_i = \mathbf{x}_i(k+1) = f[\mathbf{x}_i(k)] \end{aligned} \tag{10}$$

where $\hat{f}$ is the GPM in Eq. (6). Fig. 1 is a graphical representation of the expressions in Eq. (10). The GPM is used as a function to map all the information (state variables and control inputs) at discrete time k to the next $k+1$ discrete time step, in a one-step-ahead prediction. In this recursive formulation, each prediction is continually reused in the GPM to move forward in time. In the scheme presented by Fig. 1, the expected mean prediction $\hat{\mathbf{w}}(k)$ and the prediction variance var$[\mathbf{w}(k)]$ are fed back into the GPM, combined with the control input $\mathbf{u}(k)$ to create a new input $\mathbf{x}(k)$ for the GPM. For a system with multiple states, a GPM is constructed for each of the m predicted states that need to be fed back.

Notice that the GPM input $\mathbf{x}$ belongs to a higher dimension than the state variables $\mathbf{w}$ ($\mathbf{x} \in \mathbb{R}^d$, $\mathbf{w} \in \mathbb{R}^m$ and $d \geq m$) because the variables in the GPM could also depend on any parameter or external control input.

This approximated dynamic representation using GPM employs the concept of storage and retrieval of information (Pope, 1997), in which pre-computed function evaluations of the function to be approximated $f[\mathbf{x}_i(k)]$ at different input $\mathbf{x}_i(k)$ are stored in S as a database of the system dynamics. The database is used as reference information for state prediction at untried locations in the state-space by interpolating between the sampled points using GPM. In practice, a subset of information from the database may be extracted from the set S to approximate the dynamics at a new value of $\mathbf{w}$ with reduced computational demands.

The GPM framework has some resemblance to the equation-free prediction of system dynamics (Kevrekidis, Gear, & Hummer, 2004), in which evaluations of the original function $f[\mathbf{x}_i(k)]$ are made in the vicinity of $\mathbf{x}_i(k)$ and with small Δt, to approximate the dynamic evolution of the system. The difference between the two approaches is in the function evaluations from the detailed simulations. In the

equation-free approach, the function evaluations are made online while making the predictions, compared to the presented methodology where the evaluations are made offline and later used to map the states in time.

4. Methodology

4.1. Dynamic approximated model using GPM with input and parameter uncertainty propagation

Because of the recursive nature of Eq. (10), any prediction error that occurs in one step due to the GPM approximation will propagate along the dynamic trajectory. However, it is possible to formulate a correction in the prediction using a truncated Taylor-Series expansion, which combines error estimation in the GPM parameters $\boldsymbol{\theta}$ using the Fisher information matrix of the likelihood function in Eq. (9), along with error estimates from the previous GPM prediction via Eq. (7).

The Fisher information matrix can be used as a measurement of the uncertainty in the GPM parameters, given in the set S (Pardo-Iguzquiza & Dowd, 1998). The Fisher information matrix, $I \in \mathbb{R}^{(d+2)\times(d+2)}$, is defined as

$$I_{i,j}(\boldsymbol{\theta}) = -E\left[\frac{\partial^2[\ln L(\boldsymbol{\theta}, S|\boldsymbol{\theta})]}{\partial\theta_i\partial\theta_j}\right] \approx \frac{\partial^2[-\ln L(\hat{\boldsymbol{\theta}}, S)]}{\partial\theta_i\partial\theta_j} \tag{11}$$

where $L(\boldsymbol{\theta}, S)$ is the likelihood function of the GPM parameters as in Eq. (9). Kitanidis and Lane (1985) derived the analytical solution for each element of the Fisher information matrix using the multivariable Gaussian distribution as a likelihood function. By definition, the Fisher information matrix is evaluated at the true GPM parameter vector $\boldsymbol{\theta}$, which is unknown. Instead, the Fisher information matrix is typically evaluated at the estimated GPM parameter vector, $\hat{\boldsymbol{\theta}} \in \mathbb{R}^{d+2}$ (Todini & Ferraresi, 1996). This estimated value of $\hat{\boldsymbol{\theta}}$ can be calculated by maximizing the likelihood function, $\hat{\boldsymbol{\theta}} = arg\min_{\theta}[-\ln L(\boldsymbol{\theta}, S)]$. Once I is calculated, its inverse can be used as a parameter covariance matrix of the GPM parameters as:

$$var(\boldsymbol{\theta}) = I^{-1}(\boldsymbol{\theta}) \approx \left[\frac{\partial^2[-\ln L(\hat{\boldsymbol{\theta}}, S)]}{\partial\theta_i\partial\theta_j}\right]^{-1} \tag{12}$$

Because there are as many GPM models as predicted outputs in the approximated model, each GPM will have its own estimated vector parameter $\hat{\boldsymbol{\theta}}$ and therefore its own parameter covariance matrix var($\boldsymbol{\theta}$).

Combining Eqs. (6), (7) and (10), we can describe the predicted distribution of an uncertain state $\mathbf{w}(k)$ as a multinormal distribution with an expected mean vector $[\hat{\mathbf{w}}(k)|\hat{\mathbf{x}}(k-1), S]$ with corresponding state covariance matrix $var[\mathbf{w}(k)|\hat{\mathbf{x}}(k-1), S]$:

$$[\mathbf{w}(k)|\hat{\mathbf{x}}(k-1), S] \sim MN([\hat{\mathbf{w}}(k)|\hat{\mathbf{x}}(k-1), S], var[\mathbf{w}(k)|\hat{\mathbf{x}}(k-1), S]) \tag{13}$$

The expected mean vector $[\hat{\mathbf{w}}(k)|\hat{\mathbf{x}}(k-1), S] \in \mathbb{R}^m$ is constructed with the expected mean predictions of each of the m estimated states by their corresponding GPM from Eq. (6):

$$[\hat{\mathbf{w}}(k)|\hat{\mathbf{x}}(k-1), S] = [\hat{w}_a(k)|\hat{\mathbf{x}}(k-1), S] \quad a = 1, \ldots, m \tag{14}$$

Similarly, the state covariance matrix $var[\mathbf{w}(k)|\hat{\mathbf{x}}(k-1), S] \in \mathbb{R}^{m\times m}$ is constructed as a diagonal matrix, where the diagonal entries are the prediction variances from the GPM for each of the m estimated states, calculated in Eq. (7)

$$var[\mathbf{w}(k)|\hat{\mathbf{x}}(k-1), S] = \text{diag}\{var[w_a(k)|\hat{\mathbf{x}}(k-1), S]\} \quad a = 1, \ldots, m \tag{15}$$

Eq. (15) defines the update for the state covariance matrix as the dynamic prediction progresses in time. Notice that Eq. (15) only refers to the uncertainty in the state $\mathbf{w}(k)$, but not to the uncertainty in the control input $\mathbf{u}(k)$.

Using the expressions in Eqs. (12) and (15), we can correct the expected mean prediction $\hat{w}_a(k+1)$ of the GPM in Eq. (6) for each state a, given the uncertainty in the parameters $\hat{\boldsymbol{\theta}}$ and the previous state $\hat{\mathbf{w}}(k)$

$$\begin{aligned}
&[\hat{w}_{a,c}(k+1)|\hat{\mathbf{x}}(k), S] \\
&\quad = [\hat{w}_a(k+1)|\hat{\mathbf{x}}(k), S] \\
&\qquad + \frac{1}{2}\sum_{i=1}^{d+2}\sum_{j=1}^{d+2} cov(\hat{\theta}_i, \hat{\theta}_j)\frac{\partial^2\{[\hat{w}_a(k+1)|\hat{\mathbf{x}}(k), S]\}}{\partial\theta_i\partial\theta_j}\bigg|_{\hat{\theta},\hat{x}(k),S} \\
&\qquad + \frac{1}{2}\sum_{i=1}^{m} var[w_i(k)|\hat{\mathbf{x}}(k-1), S]\frac{\partial^2\{[\hat{w}_a(k+1)|\hat{\mathbf{x}}(k), S]\}}{\partial[w_i(k)]^2}\bigg|_{\hat{\theta},\hat{x}(k),S} \quad a \\
&\quad = 1, \ldots, m
\end{aligned} \tag{16}$$

where the subscript c stands for the corrected estimation. The first term of the right hand side of Eq. (16) is the expected mean prediction from Eq. (6) and the second term considers the GPM parameter uncertainty in the expected mean prediction. Notice that the third term in the right hand side of Eq. (16) only considers the state uncertainty, var[$\mathbf{w}(k)$], and not the overall input uncertainty var[$\mathbf{x}(k)$], because we have not presented control input uncertainties (that explains why the partial derivatives are with respect to $w_a(k)$ and the summation is up to m and not up to d), but it is not difficult to derive the remaining terms when control input uncertainties are also present. Eq. (16) is replicated for each of the m state dimensions, to obtain a complete prediction for the entire state vector.

The uncertainty propagation in GPMs is not a new topic. Parameter uncertainty has been an important topic in geostatistics (Kitanidis, 1987; Marchant & Larl, 2007; Todini & Ferraresi, 1996), where the impact of different covariance functions on the uncertainty propagation has been studied (Marchant & Lark, 2004). With the application of GPM in dynamic systems, the state uncertainty and its propagation on a one-step-ahead dynamic prediction has been also investigated. A correction was presented by (Girard & Murray-Smith, 2005) for the GPM predictive distribution for a one-step-ahead dynamic prediction given a noisy input $\mathbf{x}_i(k)$, assuming that the observations y_i comes from a zero-mean Gaussian distribution.[1] However, they did not consider the effect of uncertainty in the GPM parameters $\hat{\boldsymbol{\theta}}$. To the best of our knowledge, our contribution is the first application in dynamic systems modeling that combine parameter and state uncertainty propagation in a GPM. Additionally our contribution considers the uncertainty propagation associated with the regression functions $\mathbf{h}(\mathbf{x})$ and the regression coefficients $\boldsymbol{\beta}$.

4.2. Improving the approximated model: sequential design and analysis of computer experiments

A Gaussian process model is a local interpolator that establishes a spatial dependence of the residuals z as a function of the inputs in the multidimensional space, to improve the prediction of a global regression model. In the dynamic context presented in this paper, the input space for the set S contains the region of the state-space where the system dynamics evolve. As a local inter-

[1] Which it automatically implies that $\hat{\boldsymbol{\beta}} = 0$, eliminating many of the terms in Eqs. (6) and (7).

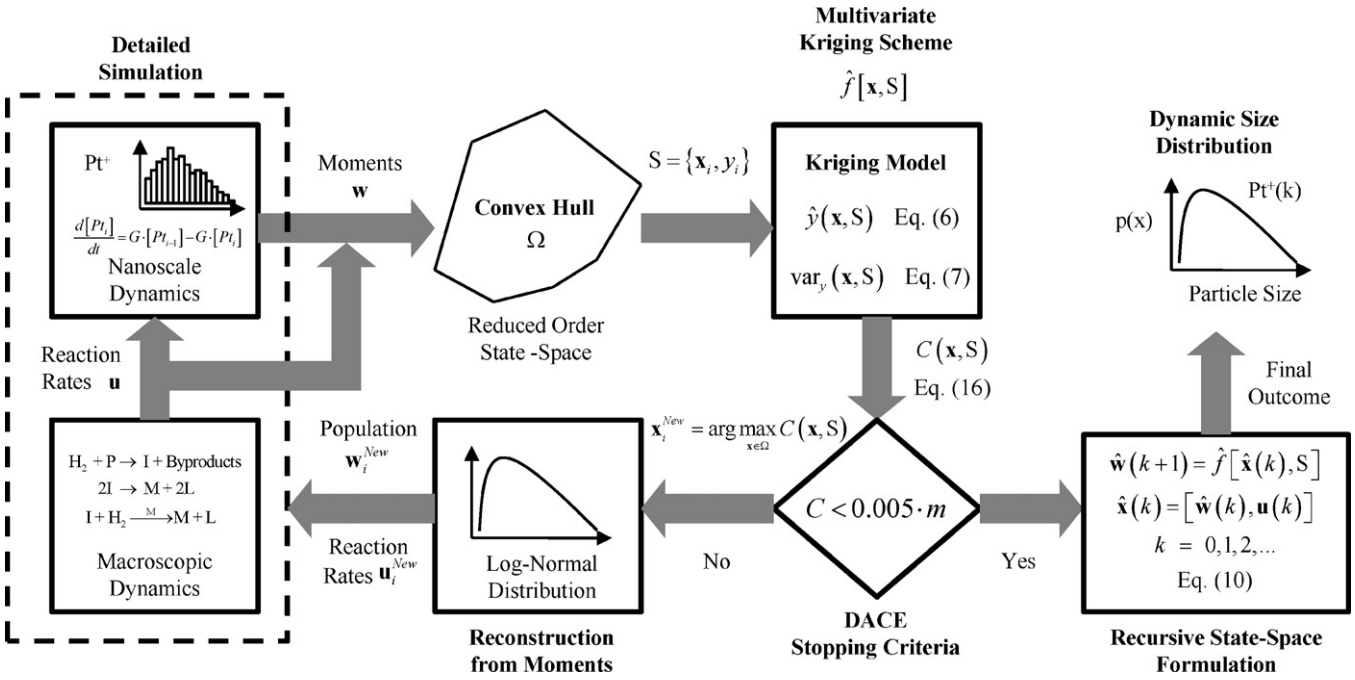

Fig. 2. Sequential DACE design to approximate nanoparticle size distribution using a multivariate GPM scheme.

polator, the position of the inputs in S is the key to accurately describing the dynamics of the system, over different conditions and regions across the state-space. To improve the accuracy of the approximated model, new samples of the full simulation should be obtained, and then added to the set S.

A sequential design of computer experiments strategy (from now on, called sequential DACE) (Sacks, Welch, Mitchell, & Wynn, 1989) is proposed to improve the construction of the input/output set S used in the GPM. This sequential DACE was implemented using the prediction variance of the GPM, Eq. (7), as an indicator of regions where the approximated model is more uncertain (Kleijnen & Beers, 2004). Similar improvement approaches have been developed for GPM in the machine learning community under the name of reinforcement learning (Engel, Mannor, & Meir, 2005). Since the multivariate approximated model is built with several GPMs – one for each of the predicted outputs – the uncertainty in all m models should be considered when designing new simulations. Here we use the sum of all m prediction variances C, as our measure of uncertainty.

$$C[\mathbf{x}(k), S] = \sum_{i=1}^{m} var[\hat{x}_i(k+1) \,|\, \mathbf{x}(k), S] \tag{17}$$

An optimization procedure was formulated to find the new input point $\mathbf{x}_i^{New}$ to be added to the set S, at the point where C is maximized. Fig. 2 shows a schematic of our algorithm for the sequential design of input points, applied to the detailed simulation for platinum nanoparticles in Eq. (1).

To limit the number m of GPMs required and the dimension of $\boldsymbol{\theta}$, we employ a reduced state dimension, using the first three moments of the nanoparticle distribution (g_0, g_1, g_2), along with the nucleation and growth rates from the macroscopic model. Detailed simulations from the nanoparticle model and simulation in Section 2 are collected for five representative settings of the initial concentration of hydrogen and precursor using a Latin Hypercube design, within the region described by Eq. (2). No nanoparticles are initially present.

All of the collected states from the dynamic simulation are used to construct a convex hull (Ω) to create a mathematical description of the dynamic region. The convex hull defines boundaries of the known input region from the pre-collected dynamic trajectories. From the convex hull, a set S of sampled information is selected to build the multivariate GPM. The set could be a subset of the pre-collected dynamic inputs, for which the observations are already collected, or it could be a completely different set of states across the convex hull, in which case additional one-step-ahead simulations will be needed.

After the construction and identification of the GPMs, the optimization problem to define a new input point $\mathbf{x}_i^{New}$ is solved. The optimization problem uses the uncertainty measure C from Eq. (17) as the objective function which is constrained to search within the convex hull Ω. A one-step-ahead prediction is then collected from the detailed simulation, beginning in the state $\mathbf{x}_i^{New}$. The reconstruction from the reduced-order state based on the moment approximation, back to the full description of the nanoparticle distribution, is made by assuming that the size distribution follows a log-normal distribution.

Finally, the multivariate approximated model is rebuilt, now incorporating the new sample point $\mathbf{x}_i^{New}$ in the set S. This new GPM is then used to design another sample point, iterating as shown in Fig. 2, until a desired level of confidence is achieved in the prediction. After the set S has been built, dynamic size distributions can be obtained using the final GPM, for different macroscopic process settings.

5. Results and discussion

Figs. 3, 4 and 6 present the predictions for a single dynamic trajectory, with initial concentrations of $H_2 = 5 \times 10^{-5}$ mol/cm^3 and $P = 1 \times 10^{-5}$ mol/cm^3. Fig. 3 presents two reconstructed nanoparticle size distributions. The predicted distribution is constructed from the first three moments assuming that the nanoparticle distribution is log-normal. The reconstruction is made by calculating the mean, variance and normalizing constant of the log-normal distribution from the predicted moments. In Fig. 3, the histogram is obtained directly from the full solution of the population balance in Eq. (1). The solid line is the log-normal reconstruction using the moments from the histogram in the population balance showing that the

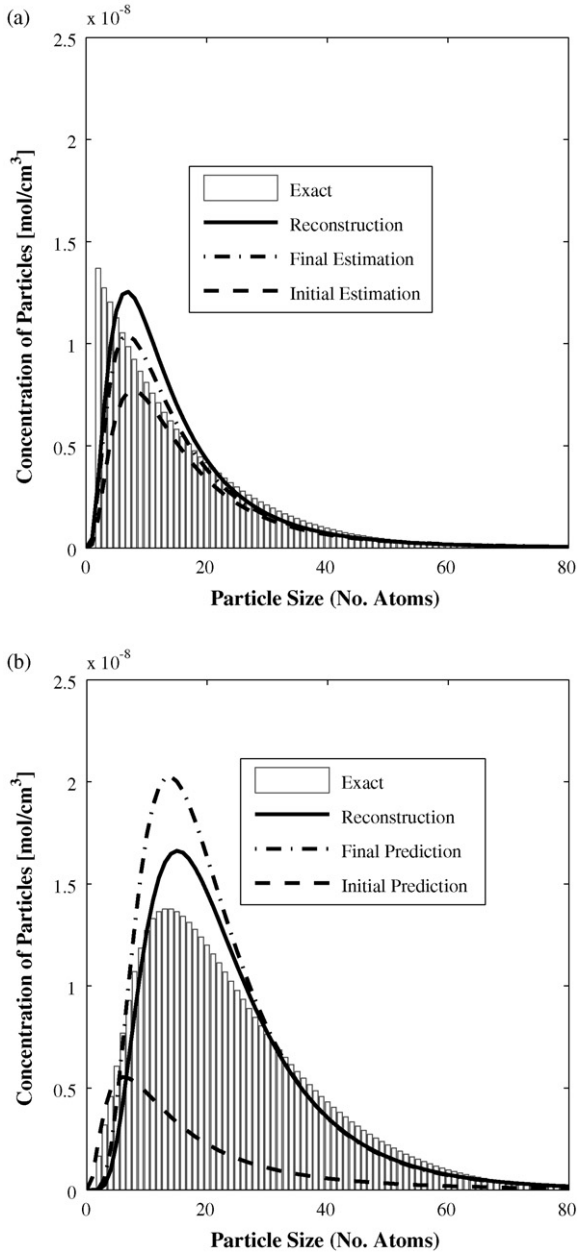

Fig. 3. Reconstructed nanoparticle size distributions using the first three moments predicted by the GPM. (a) Distributions at $t = 3$ s. (b) Distributions at $t = 60$ s. The initial concentrations are $H_2 = 5 \times 10^{-5}$ mol/cm^3 and $P = 1 \times 10^{-5}$ mol/cm^3.

full distribution is not in fact log-normal. The dashed and dotted lines represent the reconstructions of the distributions using our approximated model, before and after the sequential DACE strategy presented in Section 4.

The initial approximated model has $n = 10$ input points, collected from the initial set of dynamic trajectories, while the final approximated model uses a total of 23 input points, after adding 13 additional input points selected by the sequential DACE strategy. The final approximated model creates a good reconstruction of the location of the peak and the width of the distribution. Comparing the nanoparticle distribution prediction made at $t = 3$ s (Fig. 3a) and $t = 60$ s (Fig. 3b), the final GPM exhibits a more accurate prediction over time than the initial GPM.

Fig. 3 shows the comparison between the solution from the population balance and the reconstruction using the moment approximation. The approximated model will have an error because the reduction in the dimension to represent the nanoparticle size distribution. The methodology presented in this work could have a limitation in its application when the dimension of the state-space increases. The application of efficient dimensional reduction techniques will be required to make the application scalable to higher dimensions.

The effect of the sequential design in the uncertainty of the approximated model can be seen in Fig. 4, where the dynamics of the first three moments are shown. The addition of new input points improves the prediction of all moments in the entire time frame of interest, showing the robustness of the approximated model.

The implemented sequential DACE design uses the prediction variance from Eq. (7) as measurement to select the new points to be sampled. One characteristic of a GPM is that the prediction variance is minimized at the input points in S. This situation creates many local maxima in between the sampled points across the input space with most of them located at the boundaries of the convex hull. The optimal trade-off between the exploration of the dynamic region and the significance of the dynamic information obtained from unexplored regions remains an open problem in the methodology for further study.

We improve the probability of finding the global maximum by randomly selecting 25 initial values for the optimization. The uncertainty measure C from Eq. (17) is evaluated for all 25 points prior to conducting the optimization. The initial guess for the optimization is selected as the point with the maximum value of C. The results of the optimization procedure for the sequential DACE design are shown in Figs. 5 and 6.

The addition of the new 13 input points decreases the maximum value of the uncertainty measure C as well as improves the dynamic prediction of the first moment. The significance of this fact is that the approximated model can select automatically where new information is needed from the full simulation in order to improve the approximation.

A global dynamic evaluation was performed using 40 different initial process settings from Eq. (2). The overall evaluation was repeated 30 times for each initial setting, and the results were averaged. The Euclidean distance between the full detailed simulation and the GPM predicted trajectories at each time were used to quantify the prediction error. After our sequential DACE design was implemented, the overall dynamic error in the trajectories decreased by 37%.

The dynamic GPM is based on pre-collected dynamic simulations which guide the GPM, globally and locally, to estimate values at unrecorded locations. The creation of this database is key in the performance of the approximated model. The pre-collected set of dynamic trajectories also defines the dynamic region where the system dynamics evolve. Because these initial dynamic trajectories are obtained from expensive detailed simulations, it is desirable to minimize the number of simulations without sacrificing the accuracy of the prediction over the region of interest. Finally, even after the database is constructed, there is no guarantee that the collected samples will reflect completely the system dynamics because of the stochastic nature of the dynamic simulation. All these factors should be considered in determining the best implementation of the methodology, which remains as an open problem.

The sampling rate at which these trajectories are recorded is also important. The sampling time, Δt, in Eq. (10) represents the time resolution at which the one-step-ahead prediction will be made. A small sampling time maybe needed to capture the system dynamics but it also means that more input points will be needed to represent the overall state. We also need to consider whether different regions in the input space will require a finer local resolution (i.e. more input points over the same area) to capture the dynamics of those states. In that case, certain regions of the input space could be recorded with smaller sampling times than other areas. By com-

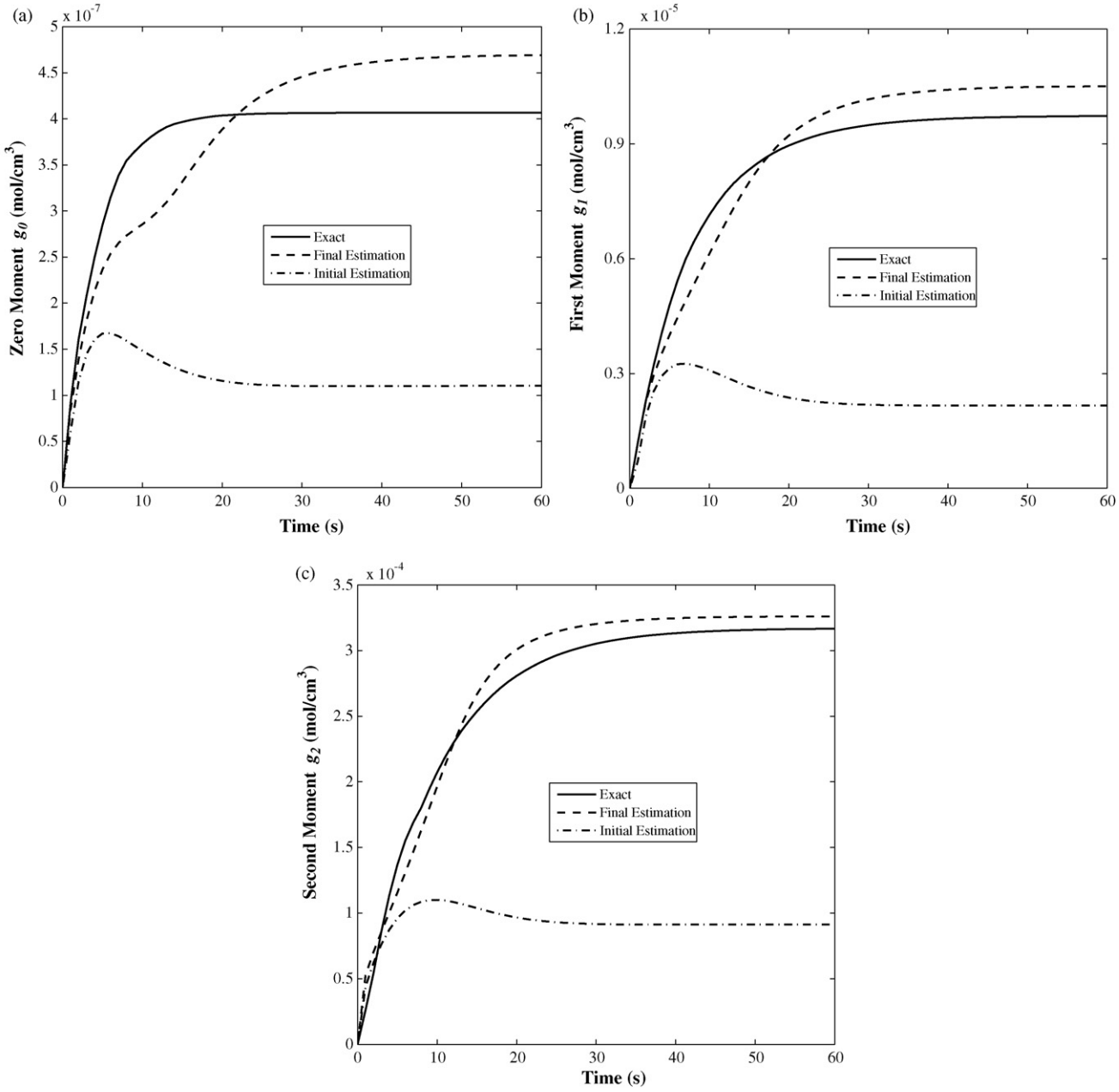

Fig. 4. Dynamic approximation of the first three moments of the nanoparticle size distribution before and after the use sequential DACE design. Initial concentrations: $H_2 = 5 \times 10^{-5}$ mol/cm^3 and $P = 1 \times 10^{-5}$ mol/cm^3.

bining different sampling times, the dynamic GPM methodology could handle different input/output sets S for each Δt, a problem that has not yet been formulated for GPM in dynamic systems applications.

Once the database of initial simulations has been collected, one should select a subset of points, avoiding redundant information to reduce the size of the set S for computational efficiency. This problem can be interpreted as a data mining problem to find the most appropriate subset of input points for the GPM. The goal is not only to obtain a good prediction over the state-space by spreading out the points, but also because it is important to identify the noise level in each of the output variables. The signal-to-noise problem has been recently studied (Kleijnen, Beers, & Nieuwenhuyse, 2010) by evaluating the number of repetitions for each input point in S, but it has not been evaluated in the dynamic context presented in this paper.

After the initial set S of input/output information has been selected, the functional form of a GPM requires the selection of the covariance function to model the regression covariance matrix V and the mean distribution of the observations $\mathbf{h}^T(\mathbf{x}_i)\hat{\boldsymbol{\beta}}(\hat{\theta}, S)$. The number of possible covariance or kernel functions that have been used in the GPM is large, with different studies describing how to select the best covariance function (Huang, Martinez, Mateu, & Montes, 2007). However, in the dynamic framework presented in this work, it is likely that the Gaussian covariance function will continue to be used because of its simplicity and because it is an infinitely differentiable function, which allows for the uncertainty propagation studies and corrections that have been presented in this work. A different situation occurs with the mean distribution of the observations, where the majority of the papers use a constant regression function with a known value of the regression coefficients $\boldsymbol{\beta}$. A few papers explore other mathematical formulations to

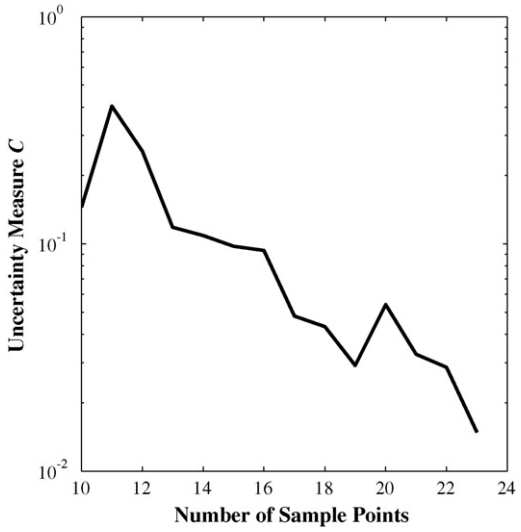

Fig. 5. Maximum value of the uncertainty measure C using sequential DACE design.

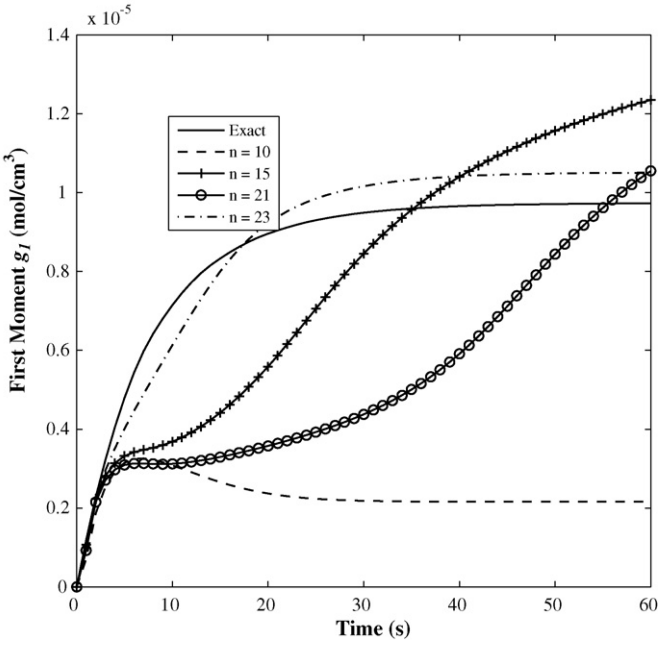

Fig. 6. Improvement in the dynamic approximation of the first moment using the sequential DACE design. Initial concentrations: $H_2 = 5 \times 10^{-5}$ mol/cm^3 and $P = 1 \times 10^{-5}$ mol/cm^3.

model the mean behavior, like a set of polynomial regression functions (Joseph, Hung, & Sudjianto, 2008). In fact, the linear model formulation of the mean behavior allows the user to implement other nonlinear functions instead of polynomial regression functions. The nonlinear functions could bring in a physical description of the system dynamics, which is then combined with the local correction via the residuals z to further improve the prediction of the system dynamics.

6. Conclusions

This paper presents the framework for building an approximate model for complex stochastic simulations using Gaussian process models. Using a detailed simulation for nanoparticle synthesis, our model approximates the nanoparticle size distribution obtained from the full simulations. The sequential DACE design based on the prediction uncertainty of the model shows a good performance in the selection of additional sample points to improve its accuracy. The methodology shows promising results as a surrogate for computationally expensive simulations, to make fast and accurate dynamic predictions.

The future application of the presented GPM methodology depends on the ability to reduce the computational effort to obtain dynamic predictions. The most computationally demanding step is the calculation of the inverse of the covariance matrix V, not only because the matrix could be ill-conditioned depending on GPM parameters, but also because the number of operations in the inverse calculation scales cubically with the number of input points $O(n^3)$ (Gregorcic & Lightbody, 2009). Most of the GPM applications in engineering use a Cholesky decomposition to mitigate both problems. Another computational aspect in GPM is the calculation of the summation in the covariance function of Eq. (4), which increases when the input dimension d and the number of input points n increases. Research has been made in order to reduce the computational cost associated with larger sets of input points in GPM (Gray, 2004). These results could be incorporated to make the methodology more practical for engineering problems.

A final aspect to analyze in the framework is the uncertainty propagation. Eq. (16) describes the correction to the GPM estimate due to uncertainty in the parameters and the state. In a more detailed inspection of Eq. (16), the parameter uncertainty is fixed once the GPM parameter vector has been identified. The Fisher information matrix will not change in time, but the state covariance matrix will evolve. A future improvement in the estimation of the state covariance matrix should include off-diagonal terms to represent the cross-correlation between predicted outputs from the approximated model.

Future research directions in Gaussian process modeling for dynamic systems include the evaluation of different sampling strategies to collect the input points from the pre-collected dynamic trajectories and the identification of the uncorrelated noise component from stochastic simulations.

Acknowledgement

The authors gratefully acknowledge the Air Force of Scientific Research for financial support (FA9550-07-1-0161).

References

Azman, K., & Kocijan, J. (2007). Application of Gaussian processes for black-box modelling of biosystems. *ISA Transactions*, *46*, 443–457.
Blasco, J. A., Fueyo, N., Dopazo, C., & Ballester, J. (1998). Modeling the temporal evolution of a reduced combustion chemical system with an artificial neural network. *Combustion and Flame*, *113*, 38–52.
Brad, R. B., Tomlin, A. S., Fairweather, M., & Griffiths, J. F. (2007). The application of chemical reduction methods to a combustion system exhibiting complex dynamics. *Proceedings of the Combustion Institute*, *31*, 455–463.
Chatterjee, A., & Vlachos, D. G. (2007). An overview of spatial microscopic and accelerated kinetic Monte Carlo methods. *Journal of Computer-Aided Materials Design*, *14*, 258–308.
Cressie, N. (1993). *Statistics for spatial data* (1st ed.). New York: Wiley Interscience.
Deisenroth, M. P., Rasmussen, C. E., & Peters, J. (2009). Gaussian process dynamic programming. *Neurocomputing*, *72*, 1508–1524.
Engel, Y., Mannor, S., & Meir, R. (2005). Reinforcement learning with Gaussian processes. In *ICML 2005: 22nd international conference on machine learning* Bonn, Germany, 7–11 August 2005, (pp. 201–208).
Ghoniem, N. M., Busso, E. P., Kioussis, N., & Huang, H. (2003). Multiscale modelling of nanomechanics and micromechanics: An overview. *Philosophical Magazine*, *83*(31), 3475–3528.
Girard, A., & Murray-Smith, R. (2005). Gaussian processes: Prediction at a noisy input and application to iterative multiple-step ahead forecasting of time-series. In R. Murray-Smith, & R. Shorten (Eds.), *Lecture notes in computer science* (pp. 158–184). Springer-Verlag.
Goldberger, A. S. (1962). Best linear unbiased prediction in the generalized linear regression model. *Journal of the American Statistical Association*, *57*(298), 369–375.

Gray, A. G. (2004). *Fast kernel matrix–vector multiplication with application to Gaussian process learning*. Computer Science Department Technical Report, CMU-CS-04-110, Carnegie Mellon University.

Gregorcic, G., & Lightbody, G. (2009). Gaussian process approach for modelling of nonlinear systems. *Engineering Applications of Artificial Intelligence*, *22*, 522–533.

Gummalla, M., Tsapatsis, M., Watkins, J. J., & Vlachos, D. G. (2004). Multiscale hybrid modeling of film deposition within porous substrates. *AIChE Journal*, *50*(3), 684–695.

Hernandez, A. F., & Grover Gallivan, M. (2008). An exploratory study of discrete-time state-space models using kriging. In *IEEE American Control Conference* Seattle, 2008, (pp. 3993–3998).

Hsu, C. S. (1980). A theory of cell-to-cell mapping dynamical systems. *Journal of Applied Mechanics*, *47*(4), 931–939.

Huang, H.-C., Martinez, F., Mateu, J., & Montes, F. (2007). Model comparison and selection for stationary space–time models. *Computational Statistics and Data Analysis*, *51*, 4577–4596.

Joseph, V. R., Hung, Y., & Sudjianto, A. (2008). Blind kriging: A new method for developing metamodels. *Journal of Mechanical Design*, *130*(3), 031102.

Kevrekidis, I. G., Gear, C. W., & Hummer, G. (2004). Equation-free: The computer-aided analysis of complex multiscale systems. *AIChE Journal*, *50*(7), 1346–1355.

Kitanidis, P. K. (1987). Parametric estimation of covariances of regionalized variables. *Water Resources Bulletin*, *23*(4), 557–567.

Kitanidis, P. K., & Lane, R. W. (1985). Maximum likelihood parameter estimation of hydrologic spatial processes by the Gauss–Newton method. *Journal of Hydrology*, *79*, 53–71.

Kleijnen, J. P. C., & Beers, W. C. Mv. (2004). Application-driven sequential designs for simulation experiments: Kriging metamodelling. *Journal of the Operational Research Society*, *55*, 876–883.

Kleijnen, J. P. C., Beers, W. v., & Nieuwenhuyse, I. v. (2010). Constrained optimization in expensive simulation: Novel approach. *European Journal of Operational Research*, *202*, 164–174.

Koehler, J. R., & Owen, A. B. (1996). Computer experiments. In S. Ghosh, & C. R. Rao (Eds.), *Handbook of statistics* (pp. 261–308). New York: Elservier Science.

Lophaven, S. N., Nielsen, H.B., Sondergaard, J. (2002) *DACE: A Matlab kriging toolbox, version 2.0*. IMM Technical University of Denmark, Lyngby.

Marchant, B. P., & Lark, R. M. (2004). Estimating variogram uncertainty. *Mathematical Geology*, *36*(8), 867–898.

Marchant, B. P., & Larl, R. M. (2007). The Matérn variogram model: Implications for uncertainty propagation and sampling in geostatistical surveys. *Geoderma*, *140*, 337–345.

Martin, J. D., & Simpson, T. W. (2005). Use of kriging models to approximate deterministic computer models. *AIAA Journal*, *43*(4), 853–863.

Pardo-Iguzquiza, E., & Dowd, P. A. (1998). Maximum likelihood inference of spatial covariance parameters of soil properties. *Soil Science*, *163*(3), 212–219.

Pope, S. B. (1997). Computationally efficient implementation of combustion chemistry using in-situ adaptive tabulation. *Combustion Theory and Modelling*, *1*(1), 41–63.

Rasmussen, C. E., & Williams, C. K. I. (2006). *Gaussian processes for machine learning*. The MIT Press.

Ratsch, C., & Venables, J. A. (2003). Nucleation theory and the early stages of thin film growth. *Journal of Vacuum Science and Technology A*, *21*(5), S96–S109.

Sacks, J., Welch, W. J., Mitchell, T. J., & Wynn, H. P. (1989). Design and analysis of computer experiments. *Statistical Science*, *4*(4), 409–435.

Saquing, C. D., Kang, D., Aindow, M., & Erkey, C. (2005). Investigation of the supercritical deposition of platinum nanoparticles into carbon aerogels. *Microporous and Mesoporous Materials*, *80*, 11–23.

Todini, E., & Ferraresi, M. (1996). Influence of parameter estimation uncertainty in kriging. *Journal of Hydrology*, *175*, 555–566.

Vinet, S., & Vazquez, E. (2008). Black-box identification and simulation of continuous-time nonlinear systems with random processes. In *Proceedings of the 17th World Congress, the international federation of automatic control, Seoul, Korea, 2008* (pp. 14391–14396).