BOJANA V. ROSIĆ AND JOBST H. DIEKMANN

STOCHASTIC DESCRIPTION OF AIRCRAFT SIMULATION MODELS AND NUMERICAL APPROACHES

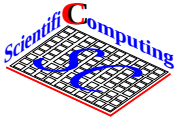
INFORMATIKBERICHT 2014-01
INSTITUTE OF SCIENTIFIC COMPUTING
CARL-FRIEDRICH-GAUSS-FAKULTÄT
TECHNISCHE UNIVERSITÄT BRAUNSCHWEIG

Braunschweig, Germany

This document was created February 2014 using LaTeX 2ε.

Institute of Scientific Computing
Technische Universität Braunschweig
Hans-Sommer-Straße 65
D-38106 Braunschweig, Germany

e-mail: wire@tu-bs.de
url: www.wire.tu-bs.de
e-print: www.digibib.tu-bs.de/?docid=00039000



Stochastic Description of Aircraft Simulation Models and Numerical Approaches

Bojana V. Rosić and Jobst H. Diekmann

Institute of Scientific Computing, TU Braunschweig

wire@tu-bs.de

This paper is concerned with the uncertainty quantification of an aircraft simulation model. Mathematically speaking the aircraft model represents a system of second order differential equations dependent on a set of input parameters related to the mass, aerodynamics and the structure of the aircraft. The input aerodynamic parameters are modelled as random variables and processes whose probability distributions are chosen according to the maximum entropy principle and available experimental data. For a flight dynamics model the evolution of uncertainties in the aircraft state trajectory is estimated with the help of so-called non-intrusive numerical approaches, examples of which are the direct integration method, the stochastic collocation approach and the pseudo-Galerkin method. These numerical methods rely on a set of samples of aircraft state trajectories simply obtained by solving the corresponding systems of deterministic ordinary differential equations.

Contents

Nomenclature

$\mathbf{A}$	=	system matrix
$\mathbf{B}$	=	input matrix
$\mathbf{C}$	=	output matrix
C_{L0}	=	lift coefficient for zero angle of attack
$C_{L\alpha}$	=	linear lift coefficient
C_{D0}	=	parasite drag coefficient
C_{m0}	=	pitch coefficient for zero angle of attack
$C_{m\alpha}$	=	linear pitch coefficient
$\mathbf{D}$	=	feed trough matrix
D_{KL}	=	the Kullback-Leibler divergence
$\mathbb{E}$	=	the mathematical expectation
$\mathscr{F}$	=	σ algebra
F	=	the nonlinear mapping representing ODE
h	=	the altitude
$I(\xi)$	=	the interpolation polynomial
k_1,k_2	=	drag coefficients
$\mathbb{P}$	=	probability measure
q	=	the pitch rate
SU_1	=	the first Sobol' index
t	=	time
$\mathbf{V}$	=	the Vandermode-like matrix
var(x)	=	the variance of variable x
V_{TAS}	=	the true air speed
$\mathbf{x}$	=	the state of the system
$\bar{x}$	=	the mean value of x
$\hat{x}$	=	polynomial chaos approximation of x
$\tilde{x}$	=	the interpolated approximation of x
$\mathbf{y}$	=	the system output
α	=	the angle of attack
Γ_x	=	the functional of x
η	=	elevator impulse
θ	=	the pitch attitude
ξ	=	basic random variable
ξ_j	=	the sample point
$\Psi(\xi)$	=	the multivariate polynomial
ω	=	event in a probability space
Ω	=	the space of all events

1 Introduction

In order to enhance the quality and reliability of aircraft design an important engineering goal is to understand and analyse aircraft behavior under the influence of parameteric uncertainty. The sources of uncertainties can be several: parametric uncertainty (e.g. the description of aerodynamic forces), model uncertainties arising from a lack of scientific knowledge or model simplifications, and the uncertainties appearing due to the measurement noise in experimental results used for the model calibration.

The mathematical description of uncertain aircraft dynamics may be achieved by the theory of stochastic ordinary differential equations (SODE) [25, 35, 33, 2], which is gaining an increasing momentum in scientific work. The numerical solution of an SODE requires several magnitudes higher computational efforts than the solution of the corresponding ODE, but this computational power is presently becoming available. One may note that the probabilistic modelling of uncertainties is not the only possibility [19, 14, 13], other approaches are the fuzzy set theory [17, 15], interval arithmetics [20, 1], etc.

Since the main objective of this paper is the probabilistic study of the uncertainty in flight dynamics, we may refer to the recently developed theory of the stochastic partial differential equations [19, 14, 31] and re-use it for this specific SODEs problem. Several methods exist for propagating uncertainty through some mathematical model. Often they can be classified into intrusive and non-intrusive variants with respect to the use of the deterministic software, i.e. aircraft simulation model, in the following named simulator for simplicity reasons. The non-intrusively implemented method exploits the simulator without or with only slight changes in the code. Typical examples are the direct integration approaches such as the Monte Carlo method and sparse-grid techniques [14, 4, 29, 30, 27, 22], as well as those based on stochastic collocation [3, 18, 23, 4]. On the other hand, intrusively implemented methods require significant modifications, for more information please see [31]. More precisely, both variants differ more in whether they rely on samples of independently computed solutions, or whether they consider a coupled problem. The independent computations may obviously be performed non-intrusively, and the coupled methods can be implemented in both a non-intrusive and in an intrusive manner [31].

Uncertainty quantification of aircraft dynamics is a relatively new area of investigation. In the literature one may find a few papers which describe the numerical methods and the types of the uncertainties appearing in these kinds of problems. For example, the sensitivity of a flight trajectory with respect to the random factors

such as initial conditions, aerodynamic derivatives, thrust and misalignment is investigated in [22] with the help of the Monte Carlo integration. On the other side, [37] studies how to increase the knowledge of the model's validity by sensitivity analysis and the uncertainty sources in aircraft system development. The modelling of the uncertainty in both the rocket dynamics and the atmospheric conditions using stochastic parameters is presented in [5]. Similarly, [24] considers the flight of an unguided, rocket-boosted, aircraft-launched missile with respect to the aleatory uncertainty of the initial mass of the missile and the epistemic uncertainty in the thrust of the rocket motor. However, most of these investigations utilitise the time-costly Monte Carlo method for the propagation of uncertainties. Therefore, the goal of this paper is to show how to overcome these difficulties with other kinds of non-intrusive approaches.

The paper is organized as follows: Section 2 gives the deterministic description of the model problem followed by its stochastic cousin in Section 3. Numerical algorithms for the computation of the time-dependent response are studied in Section 4 and analyzed in Section 5, where the numerical results are presented. Finally, the conclusion is drawn in Section 6.

2 Deterministic Description of the Model Problem

The model considered in this paper is a basic aircraft model (BACM) [28, 7], i.e. a nonlinear flight mechanical model with six degrees of freedom (6 DoF). This model offers the chance to simulate the flight dynamics of a conventional turboprop powered aircraft by describing all forces $\boldsymbol{F}$ and moments (e.g. aerodynamics, gravity and integrated subsystems, for example engines or actuators) $\boldsymbol{M}$ acting on the aircraft. The model structure is shown on a block diagram in Fig. 1. Its upper-left part outlines the subsystems and modules that generate forces and moments which further drive the translational and rotational equations of motion, i.e.

$$\begin{aligned} m\dot{\boldsymbol{V}}_k &= \sum \boldsymbol{F}, \\ I\dot{\boldsymbol{\omega}}_k &= \sum \boldsymbol{M}. \end{aligned} \tag{1}$$

Due to the paper concision and for more clarity, the focus is set here only on one specific part of the diagram, i.e. the aerodynamics module which is playing one of the central roles in the description of acting forces and moments. This module consists of derivative models for the three forces: lift, drag and sideforce, and the three moments: roll-, pitch- and yawing-moment. Each of these forces and moments are modelled

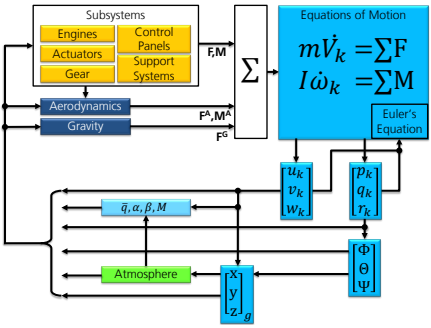


Figure 1: Block diagram of the basic aircraft model structure including submodels for forces and moments, equations of motion and the resulting aircraft motion fed back into the model

as the corresponding sum of dimensionless coefficients for the various aerodynamic influences. For example, knowing the air density (ρ), the airspeed (V) and the wing reference area (S) one may express the lift force F_L in the dimensionless form as

$$C_L = \frac{F_L}{\frac{\rho}{2} \cdot V^2 \cdot S}, \tag{2}$$

and use the sum

$$C_L = C_{L0} + C_{L\alpha} \cdot \alpha + C_{LStall} + \Delta C_{LFlap} + \ldots \tag{3}$$

to model the lift for the aircraft's longitudinal motion. Here, C_{L0} stands for the lift coefficient for zero angle of attack (AoA), $C_{L\alpha}$ is the lift slope with respect to the angle of attack, $C_{L,Stall}$ considers the stall effect and ΔC_L denotes further additional influences described as increments, e.g. for the flap deflection ΔC_{LFlap}. Most of these coefficients are modelled as linear functions except the stall effect, which introduces a nonlinearity.

For the given mean aerodynamic chord l_μ, the dimensionless pitching-moment

$$C_m = \frac{M_m}{\frac{\rho}{2} \cdot V^2 \cdot S \cdot l_\mu} \tag{4}$$

can be designed analogously to the lift model:

$$C_m = C_{m0} + C_{m\alpha} \cdot \alpha + C_{mq} \cdot q^* + \Delta C_{mFlap} + \ldots \tag{5}$$

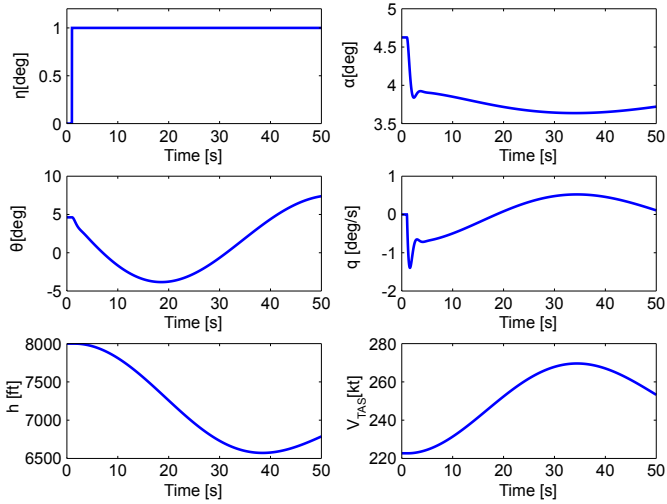


Figure 2: The excitation and response of the aircraft model

Here, the coefficients have a similar meaning as those in Eq. 3, except that the stall effects are neglected, and the term $C_{mq} \cdot q^*$, which describes the aircraft's pitching moment reaction on a pitch rate q^*, is added. This term introduces the dynamic coefficient C_{mq} representing the pitch damping.

The dimensionless drag force

$$C_D = C_{D0} + k_1 \cdot C_L + k_2 \cdot C_L^2 + \Delta C_{DFlap} + ... \tag{6}$$

is modelled as a function in terms of the parasite drag C_{D0} (the drag of the aircraft if no lift is produced), the induced drag given as a second order polynomial function dependent on the aircraft's lift coefficient C_L, and the drag increments for various influences with similar meanings as in the lift model.

These equations introduce some of the important coefficients for the aircraft's longitudinal motion which play the cenral role in the following investigations. The model is implemented in MATLAB-Simulink as a model library [28] and will be considered as a *black-box simulator* code, or just *simulator* in the further text.

Gathering the translational ($\boldsymbol{V}$) and rotational ($\boldsymbol{\varpi}$) degrees of freedom into the state

vector $\boldsymbol{x} := (\boldsymbol{V}, \boldsymbol{\varpi})^T$, the previous model in Eq. 1 may be rewritten to:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \kappa, t), \quad \boldsymbol{x}(0) = \boldsymbol{x}_0, \tag{7}$$

i.e. the initial valued system of ordinary differential equations (ODE). Here, $\boldsymbol{f}$ is a nonlinear function describing the state evolution, $\boldsymbol{x}_0$ is the initial state, and $\boldsymbol{\kappa}$ is the vector of input parameters describing, for example, lift ($C_{L0}, C_{L\alpha}$), drag (C_{D0}, k_1, k_2), pitch ($C_{m0}, C_{m\alpha}$), etc. Besides the described states of the system, the simulation model implies further states for other subsystems, e.g. for actuators. However, they are not the subject of this paper and hence will not be considered further.

In order to define an initial point $\boldsymbol{x}_0$ to start the simulation from, the model is linearized at a predefined working point and transformed to the common state space form

$$\dot{\boldsymbol{x}} = \boldsymbol{A} \cdot \boldsymbol{x} + \boldsymbol{B} \cdot \boldsymbol{u} \tag{8}$$

$$\boldsymbol{y} = \boldsymbol{C} \cdot \boldsymbol{x} + \boldsymbol{D} \cdot \boldsymbol{u} \tag{9}$$

consisting of the state (Eq. 8) and the output (Eq. 9) equations. Here, $\boldsymbol{x}$ is the state, $\boldsymbol{u}$ are the control input values and $\boldsymbol{y}$ are the output values. The system matrix $\boldsymbol{A}$, input matrix $\boldsymbol{B}$, output matrix $\boldsymbol{C}$ and feed through matrix $\boldsymbol{D}$ describe the reactions of the linearized system. Note that the control input values $\boldsymbol{u}$ differ from the input parameters $\boldsymbol{\kappa}$ mentioned before. Namely, the input parameters consider the inputs relevant for this investigation (e.g. C_{L0}), whereas the control input values are the deflections and settings of the primary and secondary flight controls, e.g. rudder deflections or throttle lever positions.

The aircraft motion considered in this paper is an unaccelerated horizontal flight in a predefined altitude and an airspeed for which the corresponding trimming point (i.e. the state and input values) is evaluated using an iterative trimming algorithm. The obtained trim data are then used as the initial data set to start the simulation from. System in Eq. 7 is excited by the elevator pulse of $\eta = 1$ [deg] initiated in time =1 [s], and then integrated in a predefined time interval [0,50] [s]. The resulting trajectory is shown in Fig. 2, where the typical peaks of the well known phugoid motion output values (trajectories), e.g. AoA (α), pitch rate (q), pitch attitude (θ), altitude (h) or true air speed (V_{TAS}), can be seen.

However, being a simulation based on numerical models for aerodynamics, e.g. Computational Fluid Dynamics (CFD, [9]) or even more elementary preliminary design tools (e.g. USAF Digital DATCOM [40]), those predicted trajectories may not behave as they would if experimentally measured. The reasons for this can be several:

- the model in Eq. 1 is simplified compared to reality,
- neither the input parameters are fully known nor the initial state of the system,
- the measurements used for calibration are characterized by noise, etc.

In order to take into account this lack of knowledge, the unknown differences are supplemented by adding the uncertainties into the model. This however requires the reformulation of the model description in Eq. 1 to a more realistic one, as further described.

3 Stochastic Description of the Model Problem

In this paper two kinds of uncertainties are introduced to the aircraft model: one in the parameters $\kappa(\omega)$, and one in the initial $\boldsymbol{x}_0(\omega)$ condition. These uncertainties are modelled as vectors of random variables (RVs) with finite variance, i.e. as mappings [31, 14]:

$$\kappa(\omega) : \Omega \to \mathscr{Q} \quad \text{and} \quad \boldsymbol{x}_0(\omega) : \Omega \to \mathscr{X} \tag{10}$$

from a probability space Ω with the measure $\mathbb{P}$ to a corresponding space of parameters $\mathscr{Q}$, and the state space $\mathscr{X}$, respectively. Introducing Eq. 10 into Eq. 1 the deterministic model transforms to:

$$\begin{aligned} &\dot{\boldsymbol{x}}(\omega) = \boldsymbol{f}(\boldsymbol{x}(\omega), \kappa(\omega), t), \quad \forall t \in \mathscr{T}, \forall \omega \in \Omega \quad (\mathbb{P}.-a.s.), \\ &\boldsymbol{x}_0 = \boldsymbol{x}(0, \omega), \end{aligned} \tag{11}$$

i.e. the system of first order stochastic ordinary differential equations (SODEs) with initial conditions. The model propagates the input uncertainties through the system in Eq. 11 to the output solution $\boldsymbol{x}(t, \omega)$—a stochastic process—on the time interval $\mathscr{T}$. The solution represents a mapping:

$$\boldsymbol{x}(t, \omega) : \ \mathscr{T} \times \Omega \to \mathscr{X} \otimes S, \tag{12}$$

and has values in a tensor-vector space $\mathscr{X} \otimes S$, where $\mathscr{X}$ and $S := L_2(\Omega)$ are the deterministic and stochastic spaces of defintion, respectively. Following this, the numerical computation of Eq. 11 is thus only possible when the full discretisation of the system in both time and in the stochastic sense is performed. For more details please see Section 4.

Table 1: The description of uncertain input parameters for the wing/fuselage combination

Symbol	Interval [%]
C_{L0}	$\pm$ 23.05
$C_{L\alpha}$	$\pm$ 5.27
C_{D0}	$\pm$ 12.4
k_1	$\pm$ 89.16
k_2	$\pm$ 379.74
C_{m0}	$\pm$ 10.46
C_{mq}	$\pm$ 0.53

3.1 Description of input parameters

According to the first available data obtained from the comparison of the measurements and DATCOM analysis (see [40]) one may describe the input parameters as shown in Table 1. The mean values are chosen as the default values, and the intervals (the minimal and maximal values) are prescribed from the data. This further means that the source of uncertainty does not truly originate from the parameters' nature but from the modelling error accompanied by a measurement noise.

Initially, the uncertain parameters $\kappa(\omega)$ in Tab. 1 are modelled as Gaussian with the prescribed mean values and the 99.7% intervals. However, this also means that 0.3% of the normal random variables will not satisfy the limits given in Tab. 1. Therefore, the uncertain parameters $\kappa(\omega)$ are also modelled as uniform random variables, see Fig. 3 and Figs. 4 – 5 for the appropriate comparison analysis. As shown in Fig. 5, the perturbation of the aircraft's true air speed is larger for the uniform than for the normal uncertainty. However, the goal of this paper is not to investigate this difference but rather to describe the methods used for the uncertainty quantification of the system response. Namely, for the numerical computations performed in this paper the uniform assumption of the input uncertainty is not of very high importance, as the parameter descriptions can be improved to more realistic ones in a Bayesian identification setting using existing measurement data [32, 26]. In addition, the coefficents are prescribed for the wing/fuselage combination. Hence, the lift, drag and pitching moments of the horizontal tailplane of the aircraft are not influenced in this investigation.

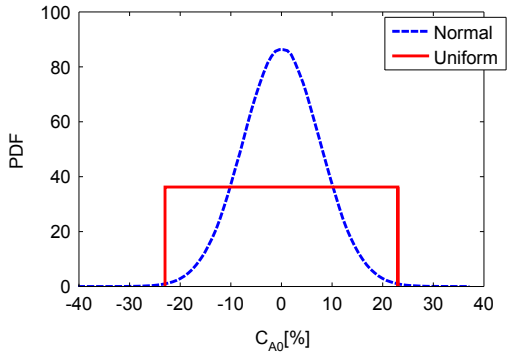


Figure 3: Normal and uniform probability density functions of initial lift coefficient C_{A0}

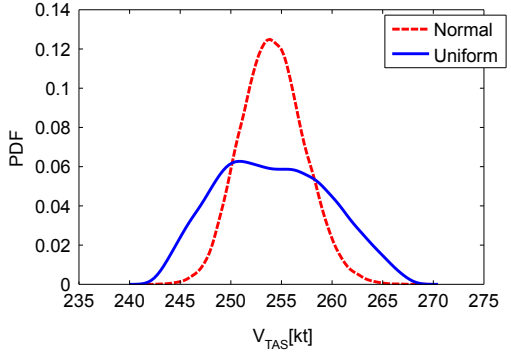


Figure 4: The probability density function of the true air speed V_{TAS} *obtained by assuming the input parameters to be normally and uniformly distributed. The results are obtained by Monte Carlo simulation with* 10^5 *samples.*

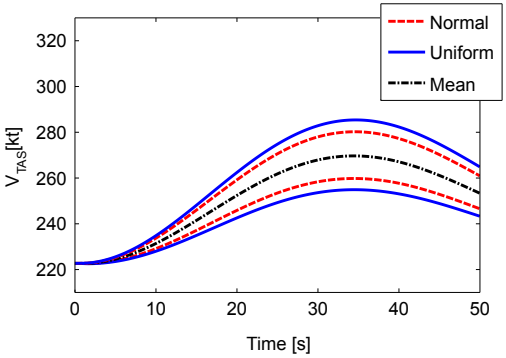


Figure 5: The mean value of the true air speed V_{TAS} and its 99% confidence region: Normal - the 99% confidence region obtained by the Monte Carlo simulation assuming the input uncertainty to be normal, Uniform - the 99% confidence region obtained by the Monte Carlo simulation assuming the input uncertainty to be uniform, Mean- the mean value obtaind by the Monte Carlo simulation assuming the input uncertainty to be normal.

4 Numerical Computation

In order to solve Eq. 11 one has to separately discretise the solution in both time and stochastic space. According to the Basic Aircraft Model [28, 7] the time discretisation is done with the explicit Runge Kutta method of order 4, which formally can be written as:

$$\boldsymbol{x}_{n+1}(\omega) = \boldsymbol{F}(\boldsymbol{x}_n(\omega), \kappa_n(\omega), t_n), \quad \forall \omega \in \Omega, \tag{13}$$

where $\boldsymbol{x}_{n+1}$ and $\boldsymbol{x}_n$ are the unknown and known solutions in the $(n+1)$-st and n-th time step, respectively, and $\boldsymbol{F}$ is the corresponding mapping. However, as the system is only semi-discretised one cannot solve it directly due to the dependance on parameter ω. With respect to the discretisation of the stochastics, i.e. the representation of the RV propagated through the evolutionary model, one may distinguish several numerical approaches for solving Eq. 13. Some possible choices of the RV representation are:

(a) *Sampling*: the RV is described by its evaluation—called *sample*—$\boldsymbol{x}(\omega_j)$ in some set of points $\{\omega_j\}_{j=1}^{N}$ [36].

(b) *Distribution*: the RV is characterised by its distribution [35].

(c) *Moments*: the RV is described by the corresponding moment equations [35]

(d) *Functional approximation*: the RV is approximated by a function of other, simpler RVs [13, 19].

Note that only the representations a) and d) allow to reuse the original simulation code. Choice b) leads to considerate Fokker-Planck types of equations, while c) requires completely new equations to be derived. In this paper we only consider methods from group a), as well as the particular group of methods d), which require only non-coupled computations. Obviously, this choice is made to achieve a non-intrusive implementation of the stochastic algorithm.

4.1 Direct Integration

The main goal of the uncertainty quantification is to compute some functional $\Upsilon_{\boldsymbol{x}}$ of the solution $\boldsymbol{x}$ in a form:

$$\Upsilon_{\boldsymbol{x}} = \mathbb{E}(\Upsilon(t,\omega,\boldsymbol{x}(\omega))) = \int_{\Omega} \Upsilon(t,\omega,\boldsymbol{x}(\omega))\mathbb{P}(d\omega), \tag{14}$$

where $\mathbb{E}(\cdot)$ denotes the mathematical expectation. The simplest examples of $\Upsilon_{\boldsymbol{x}}$ are the mean value

$$\bar{\boldsymbol{x}} = \mathbb{E}(\boldsymbol{x}) = \int_{\Omega} \boldsymbol{x}(\omega)\mathbb{P}(d\omega), \tag{15}$$

or the second moment (i.e. the variance):

$$\mathrm{var}(\boldsymbol{x}) = \mathbb{E}[(\boldsymbol{x}(\omega)-\bar{\boldsymbol{x}})^2] = \int_{\Omega} (\boldsymbol{x}(\omega)-\bar{\boldsymbol{x}})^2\mathbb{P}(d\omega) \tag{16}$$

of the system state $\boldsymbol{x}$.

In practical computation the random processes $\boldsymbol{x}(t,\omega)$ and $\Upsilon(t,\omega,\boldsymbol{x}_{n+1}(\omega))$ in Eq. 14 are approximated by a finite set $\xi := \{\xi_j(\omega)\}_{j=1}^{M}$ of independent RVs, for more details see [14, 4, 29, 30, 27, 22]. This allows the integral in Eq. 14 to be evaluated *numerically* via the finite sum

$$\Upsilon_{\boldsymbol{x}} \approx \sum_{j=1}^{N} \Upsilon(t,\xi_j,\boldsymbol{x}_{n+1}(\xi_j))w_j, \tag{17}$$

where $\{\xi_j\}_{j=1}^N$ and $\{w_j\}_{j=1}^N$ are the sample points and their weights, respectively. This corresponds to integration by a quadrature rule (the set of the deterministic points and weights) or to the Monte Carlo method and its quasi-variants—the points are selected according to the underlying probability measure [31, 14].

Following this, the direct numerical integration of SODEs reduces to a solution of N-decoupled systems of deterministic size, i.e. to N-calls of the black-box BACM simulator:

$$\boldsymbol{x}_{n+1}(\xi_j) = \boldsymbol{F}(\boldsymbol{x}_n(\xi_j), \kappa_n(\xi_j), t_n), \quad \forall j = 1, \ldots N. \tag{18}$$

This concept is the simplest as it only needs— usually very many— evaluations of the deterministic model. However, this makes the method computationally very costly. For example the convergence rate of the pure Monte Carlo method is $\mathcal{O}(N^{-1/2})$, which means that one requires the order of 10^6 BACM runs to compute statistics with accuracy on the third digit. The sparse grid methods [10, 16] are often better because their rate is exponential with respect to the number of calls. However, the rate strongly depends on the regularity of the solution, which makes the sparse integration less robust compared to the Monte Carlo method.

4.1.1 Data compression using thin singular value decomposition

The size of data which have to be stored on a memory disk grows with the increase of the number of BACM-calls in the direct integration run. Namely, storing only 10^5 Monte Carlo samples of important BACM output quantities requires already a memory of 15.2 GB. This can be a problem if the number of calls grows or the simulations have to be repeated with the different values of the input parameters. To avoid this situation, one may apply the thin singular value decomposition (tSVD) on the obtained data set [6] and store only the matrix factorials. However, this is not the optimal solution as one first has to compute all solutions and then to build the tSVD, which reduces the memory costs. As one would not like to store the original results, but only their compressed versions, the tSVD has to be performed on the run with each new coming group of samples. Therefore, the data compression can be realized in the following scenario:

1) specify the size of the preferred group of samples

2) do tSVD on the first group of samples

3) update tSVD with each new coming group.

Table 2 shows results of the updated tSVD for the example of the angle of attack α. The total number of samples is 23000 (in total 300MB), and the size of the update group is 1000 samples. The updated tSVD is performed several times with different precisions (i.e. the number of terms in tSVD) starting from 1e-02 (corresponding to one-term decomposition) up to 1e-12 (corresponding to 183 terms' decomposition). The number of updates equals 22. The procedure start takes the most of the computational time (as one does the decomposition of the initial sample group represented by a matrix), while every following step is computationally cheaper (as one only updates the tSVD). This is plotted in Table 2 in column *mean time*—the time of initial tSVD—and in column *std* — the standard deviation of time necessary for the first up to the last update. As one may notice, the memory saving is huge and the required memory decreases from 300MB to 28MB. The memory improvement factor strictly depends on the prescribed errors in second order statistics (the mean and variance). If one requires a relative error in variance less than 1e-05, then the tSVD with 11 terms is the best choice with respect to the memory requirement.

Table 2: The accuracy of tSVD, compression time and memory requirements for the data set of 23000 *samples with initial storage of 300MB. Tol-the precision of tSVD, No.-number of terms in tSVD, Time- the Mean (time of tSVD of first compression) and the Std (standard deviation of update time), Error-accuracy of second order statistics after compression, Memory- memory requirement of the compressed data set*

Tol	**No.**	**Time** [s]		**Error in**		**Memory**
		Mean	*Std*	*Mean*	*Var*	
1e-2	1	11.93	3.77	1.31e-07	0.99	184 KB
1e-4	2	11.48	4.07	6.36e-11	7.03e-05	405KB
1e-8	11	15.65	2.00	1.94e-14	4.41e-10	2MB
1e-12	183	167.25	39.51	2.28e-14	3.19e-13	28 MB

4.2 Pseudo-Galerkin approach

The pseudo-Galerkin approach projects the solution $\boldsymbol{x}_{n+1}(\xi)$ in a Galerkin manner onto the generalised orthogonal polynomial basis $\{\Psi_\alpha\}_{\alpha\in\mathscr{J}}$ [42, 11, 31], where $\mathscr{J}$ denotes the multi-index set. In other words, one approximates the state $\boldsymbol{x}_{n+1}$ by a

Wiener polynomial chaos expansion (PCE) [39, 34]

$$\hat{\boldsymbol{x}}_{n+1} = \sum_{\alpha \in \mathscr{J}} \boldsymbol{x}_{n+1}^{(\alpha)} \Psi_\alpha(\omega), \tag{19}$$

and evaluates the unknown polynomial coefficients $\boldsymbol{x}_{n+1}^{(\alpha)}$ by projecting the system in Eq. 13 in a Galerkin manner to:

$$\boldsymbol{x}_{n+1}^{(\alpha)} = \mathbb{E}(\boldsymbol{x}_{n+1}\Psi_\alpha) = \int_\Omega \boldsymbol{x}_{n+1}\Psi_\alpha \mathbb{P}(d\omega). \tag{20}$$

Furthermore, the integral in the last relation is evaluated by the numerical integration:

$$\boldsymbol{x}_{n+1}^{(\alpha)} = \sum_{j=1}^{N} \boldsymbol{x}_{n+1}(\xi_j)\Psi_\alpha(\xi_j)w_j, \tag{21}$$

where $\{\boldsymbol{\xi}_j\}_{j=1}^N$ are the sample points and w_j their corresponding weights. In this manner the set of N decoupled deterministic ODEs $\boldsymbol{x}_{n+1}(\xi_j)$ is resolved independently before the final sum is estimated.

Note that for computational reasons the system in Eq. 13 is not directly projected onto the orthogonal basis, but first rewritten to

$$\boldsymbol{x}_{n+1}(\omega) = \boldsymbol{x}_0(\omega) + S(\boldsymbol{x}_0(\omega), \kappa(\omega), t_n), \tag{22}$$

where $\boldsymbol{x}_0(\omega)$ is the initial state and $S(\boldsymbol{x}_0(\omega), \kappa(\omega), t_n)$ is the transition function describing the change of the state in the desired time interval $[0, t_n]$. After a Galerkin projection, the previous equation reads

$$\boldsymbol{x}_{n+1}^{(\alpha)} = \boldsymbol{x}_0^{(\alpha)} + \mathbb{E}(S(\boldsymbol{x}_0(\omega), \kappa(\omega), t_n)\Psi_\alpha(\omega)), \tag{23}$$

which means that the evaluation of the PCE of the system state requires the PCE approximations of the initial condition and the change of the system state in time. The former one is given a priori, while the latter one can be obtained with the help of numerical integration such that

$$\boldsymbol{x}_{n+1}^{(\alpha)} = \boldsymbol{x}_0^{(\alpha)} + \sum_{j=1}^{N} S(\boldsymbol{x}_0(\boldsymbol{\xi}_j), \kappa(\boldsymbol{\xi}_j), t)\Psi_\alpha(\boldsymbol{\xi}_j)w_j \tag{24}$$

holds.

4.3 Stochastic collocation

The stochastic collocation method is another approach to compute the solution of the system of SODEs in Eq. 13 in a non-coupled manner [31]. The method requires the estimation of the solution $\boldsymbol{x}_{n+1}$ in a set of points $\Xi := \{\xi_j\}_{j=1}^{N}$

$$\boldsymbol{x}_{n+1}(\xi_j) = \boldsymbol{F}(\boldsymbol{x}_n(\xi_j), \kappa_n(\xi_j), t_n), \quad \forall j = 1, ..N, \tag{25}$$

similarly to the direct integration approach, see Eq. 18. However, in contrast to the direct integration techniques one does not estimate the value of the integral, but interpolates the solution in a set of points [3, 18, 23, 4], or uses the linear regression model to compute the response surface [41, 4].

4.3.1 Interpolation

The solution is approximated in the form of a polynomial:

$$\tilde{\boldsymbol{x}}_{n+1}(\xi) = \sum_{j=1}^{N} \tilde{\boldsymbol{x}}_{n+1}^{(j)} I_j(\xi), \tag{26}$$

where $I_j(\xi)$ are the multidimensional interpolating functions (e.g. Lagrangian polynomials) satisfying

$$I_j(\xi_i) = \begin{cases} 1 & , i = j \\ 0 & , i \neq j \end{cases} . \tag{27}$$

Polynomial $\tilde{\boldsymbol{x}}_{n+1}(\xi)$ approximates the sample value $\boldsymbol{x}_{n+1}(\xi_j)$ exactly in the interpolation point ξ_j such that

$$\tilde{\boldsymbol{x}}_{n+1}(\xi_j) = \boldsymbol{x}_{n+1}(\xi_j) \tag{28}$$

is satisfied [23, 4]. This is the linear system of equations of type:

$$\mathbf{V}\tilde{\mathbf{x}}_{n+1} = \mathbf{x}_{n+1} \tag{29}$$

where

$$\mathbf{V} := \begin{bmatrix} I_1(\boldsymbol{\xi}_1) & I_2(\boldsymbol{\xi}_1) & \cdots & I_N(\boldsymbol{\xi}_1) \\ I_1(\boldsymbol{\xi}_2) & I_2(\boldsymbol{\xi}_2) & \cdots & I_N(\boldsymbol{\xi}_2) \\ \vdots & \vdots & \ddots & \vdots \\ I_1(\boldsymbol{\xi}_N) & I_2(\boldsymbol{\xi}_N) & \cdots & I_N(\boldsymbol{\xi}_N) \end{bmatrix} \in \mathbb{R}^{N \times N} \tag{30}$$

and

$$\mathbf{x}_{n+1} := (\cdots, \boldsymbol{x}_{n+1}(\boldsymbol{\xi}_j), \cdots)^T \in \mathbb{R}^N. \tag{31}$$

The Vandermode-like matrix $\mathbf{V}$ of coefficients is equal to identity due to the properties of the Lagrange polynomial given in Eq. 27. Hence, there is no need for solving the linear system in Eq. 36, since the interpolating coefficients are equal to the sample values $\tilde{\boldsymbol{x}}_{n+1}^{(j)} = \boldsymbol{x}_{n+1}(\boldsymbol{\xi}_j)$.

Assuming that the sufficient number of interpolating points Ξ is used, one may state that $\boldsymbol{x}_{n+1}(\xi) \approx \tilde{\boldsymbol{x}}_{n+1}(\xi)$, and further compute the first moment of $\boldsymbol{x}_{n+1}$ as

$$\mathbb{E}(\boldsymbol{x}_{n+1}(\xi)) \approx \mathbb{E}(\tilde{\boldsymbol{x}}_{n+1}(\xi)) = \sum_{j=1}^{N} \boldsymbol{x}_{n+1}(\xi_j) \int_{\Omega} I_j(\xi)\mathbb{P}(d\omega), \tag{32}$$

where

$$\int_{\Omega} I_j(\xi)\mathbb{P}(d\omega) = \sum_{k=0}^{N} I_j(\xi_k) w_k = \sum_{j=0}^{N} w_j \tag{33}$$

follows from the identity $I_j(\xi_k) = \delta_{jk}$ and the numerical integration rules. Finally, the first moment can be expressed as:

$$\mathbb{E}(\boldsymbol{x}_{n+1}(\xi)) = \sum_{j=1}^{N} \boldsymbol{x}_{n+1}(\xi_j) w_j, \tag{34}$$

while the higher order moments can be computed in a similar manner.

4.3.2 Linear regression

The interpolation can be used for the computation of the polynomial chaos coefficients given in Eq. 19 by solving:

$$\hat{\boldsymbol{x}}_{n+1}(\xi_j) = \boldsymbol{x}_{n+1}(\xi_j) \tag{35}$$

for the given data set $\{\boldsymbol{x}_{n+1}(\xi_j)\}_{j=1}^{N}$. This is the linear system of equations of type:

$$\mathbf{V}\mathbf{x}_{n+1} = \mathbf{d} \tag{36}$$

where

$$\mathbf{V} := [\Psi_{\alpha,z}] = \begin{bmatrix} \Psi_1(\boldsymbol{\xi}_1) & \Psi_2(\boldsymbol{\xi}_1) & \cdots & \Psi_Z(\boldsymbol{\xi}_1) \\ \Psi_1(\boldsymbol{\xi}_2) & \Psi_2(\boldsymbol{\xi}_2) & \cdots & \Psi_Z(\boldsymbol{\xi}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \Psi_1(\boldsymbol{\xi}_N) & \Psi_2(\boldsymbol{\xi}_N) & \cdots & \Psi_Z(\boldsymbol{\xi}_N) \end{bmatrix} \in \mathbb{R}^{N\times Z} \tag{37}$$

is the Vandermode-like matrix of coefficients, $\mathbf{x}_{n+1} := (\ldots, \boldsymbol{x}_{n+1}^{(\gamma)}, \ldots)^T \in \mathbb{R}^Z$ are the unknown PCE coefficients, and $\mathbf{d} := (\boldsymbol{x}_{n+1}(\xi_1), ..., \boldsymbol{x}_{n+1}(\xi_N))^T \in \mathbb{R}^N$ are the sample values. Note that the system is undetermined when $N < Z$, and hence the number of points has to be chosen with care. If $N = Z$ the solution of Eq. 36 corresponds to the intrpolation, otherwise for $N > Z$ the regression approach is used.

Besides the system presented in Eq. 36 there is one more way of computing the polynomial coefficients. Namely, from Eq. 35 and Eq. 28 one may state that

$$\hat{\boldsymbol{x}}_{n+1}(\xi) \approx \tilde{\boldsymbol{x}}_{n+1}(\xi), \tag{38}$$

i.e.

$$\sum_{\alpha \in \mathscr{J}} \boldsymbol{x}_{n+1}^{(\alpha)} \Psi_\alpha(\boldsymbol{\xi}) = \sum_{j=1}^{N} \boldsymbol{x}_{n+1}(\xi_j) I_j(\xi). \tag{39}$$

By Galerkin projection

$$\boldsymbol{x}_{n+1}^{(\alpha)} = \mathbb{E}(\boldsymbol{x}_{n+1} \Psi_\alpha) \tag{40}$$

this leads to the explicit expression for the unknown coefficients:

$$\boldsymbol{x}_{n+1}^{(\alpha)} = \mathbb{E}(\tilde{\boldsymbol{x}}_{n+1} \Psi_\alpha) = \sum_{j=1}^{N} \boldsymbol{x}_{n+1}(\xi_j) \left(\int_\Omega I_j(\xi(\omega)) \Psi_\alpha(\xi(\omega)) \mathbb{P}(d\omega) \right). \tag{41}$$

Applying the numerical integration, the previous relation reduces to:

$$\boldsymbol{x}_{n+1}^{(\alpha)} = \sum_{j=0}^{N} \boldsymbol{x}_{n+1}(\xi_j) \Psi_\alpha(\xi_j) w_j, \tag{42}$$

i.e. the pseudo-Galerkin approach described in Eq. 24.

4.4 The method of least squares

The method of least squares searches for the function that minimises the mean squares function. In this particular case, the goal of regression is to find the multivariate polynomial expansion which fits the data by minimising the following error [41]:

$$\min_{\hat{\boldsymbol{x}}_{n+1}} \varepsilon \quad = \quad \min_{\hat{\boldsymbol{x}}_{n+1}} \int_\Omega \left[\boldsymbol{x}_{n+1}(\omega) - \hat{\boldsymbol{x}}_{n+1}(\omega) \right]^2 \mathbb{P}(\mathrm{d}\omega), \tag{43}$$

As the distance squared is minimised, this computes the orthogonal projection in the corresponding inner product. The function $\hat{\boldsymbol{x}}_{n+1}$ approximates the solution by

a finite number of uncorrelated and independent RVs $\boldsymbol{\xi} := \{\xi_j\}_{j=1}^{M}$, and hence the integration in Eq. 43 may be evaluated numerically (see direct integration):

$$\min_{\hat{\boldsymbol{x}}_{n+1}} \hat{\varepsilon} = \min_{\hat{\boldsymbol{x}}_{n+1}} \sum_{z=1}^{N} \left[\boldsymbol{x}_{n+1}(\boldsymbol{\xi}_z) - \sum_{\alpha} \boldsymbol{x}_{n+1}^{(\alpha)} \Psi_\alpha(\boldsymbol{\xi}_z) \right]^2 w(\boldsymbol{\xi}_z). \tag{44}$$

Further on the solution of Eq. 44 can be found from the optimality condition:

$$\times_{n+1} := (\boldsymbol{x}_{n+1}^{(\alpha)})_{\alpha \in \mathscr{J}} \in \partial_{\boldsymbol{x}_{n+1}^{(\alpha)}} \hat{\varepsilon} = \mathbf{0}, \tag{45}$$

which in component form becomes:

$$\sum_{\beta} \sum_{\alpha} \sum_{z=1}^{N} \boldsymbol{x}_{n+1}^{(\alpha)} \Psi_\alpha(\boldsymbol{\xi}_z) \Psi_\beta(\boldsymbol{\xi}_z) w_z = \sum_{\beta} \sum_{z=1}^{N} \boldsymbol{x}_z \Psi_\beta(\boldsymbol{\xi}_z) w_z, \tag{46}$$

where $\boldsymbol{x}_z := \boldsymbol{x}_{n+1}(\boldsymbol{\xi}_z)$ denotes the sample points and $w_z := w(\boldsymbol{\xi}_z)$ their integration weights. In matrix notation the previous equation reduces to:

$$\begin{aligned} \mathbf{J}\mathbf{x}_{n+1} &:= \mathbf{V}^T \mathbf{W} \mathbf{V} \mathbf{x}_{n+1} \\ &= \mathbf{V}^T \mathbf{W} \mathbf{d}, \end{aligned} \tag{47}$$

where $\mathbf{J}$ is the Gram matrix, $\mathbf{V}$, $\mathbf{x}_{n+1}$ and $\mathbf{d}$ have the same meaning as before, and

$$\mathbf{W} := \operatorname{diag}[w_k] \in \mathbb{R}^{N \times N}. \tag{48}$$

In the previous equations the coefficient set $\mathbf{x}_{n+1}$ is computed by solving the deterministic residual equation at each independent grid point $\boldsymbol{\xi}_z$. This means that the collocation approach already decouples the system into N smaller independent systems of equations before the linear system is solved. These solutions are obtained by available deterministic solvers, while the solution of the linear system is computed using the Krylov preconditioned techniques or any other type of methods for large systems of linear equations.

5 Results

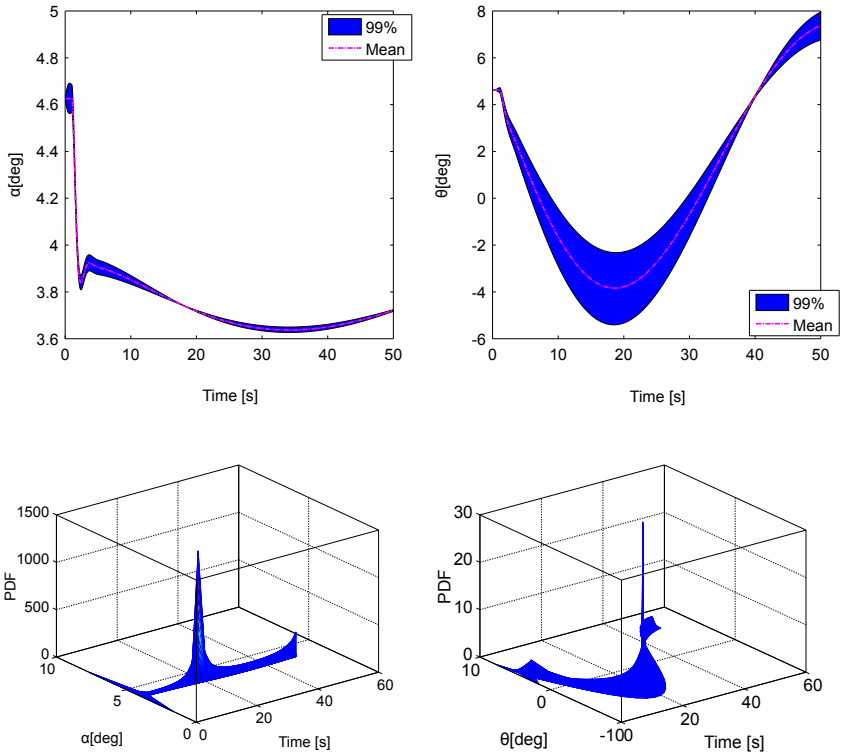
The numerical studies in this section are performed with the help of the stochastic collocation, stochastic response and pseudo-Galerkin approaches, all described in Section 4. For their comparison the result of the Monte Carlo simulation with 100 000 samples is introduced as the reference solution. The numerical investigation

is carried out for the aircraft example in the clean cruise configuration as described in Section 2. The uncertanties are assumed to be present in the description of the aerodynamics according to Tab. 1. Note that these uncertainties originate from the model error, and hence do not represent the parameter nature.

The aicraft motion is taken to be an unaccelerated horizontal flight initiated by a Heaviside function at time $t = 1$ [s], see Section 2. In contrast to the deterministic response in Fig. 2, the aircraft phugoid motion is random due to the aerodynamic uncertainty, and hence represented by many possible trajectories (each corresponding to one MC sample). However, these trajectories are not all characterised by the same probability level of occurance. Some trajectories are more probable while others are less probable to happen. To clarify this, the mean trajectories together with the 99% confidence intervals are plotted in Figs. 6 and 7. To these are added the plots of the corresponding probability density function (PDF) evolutions in time.

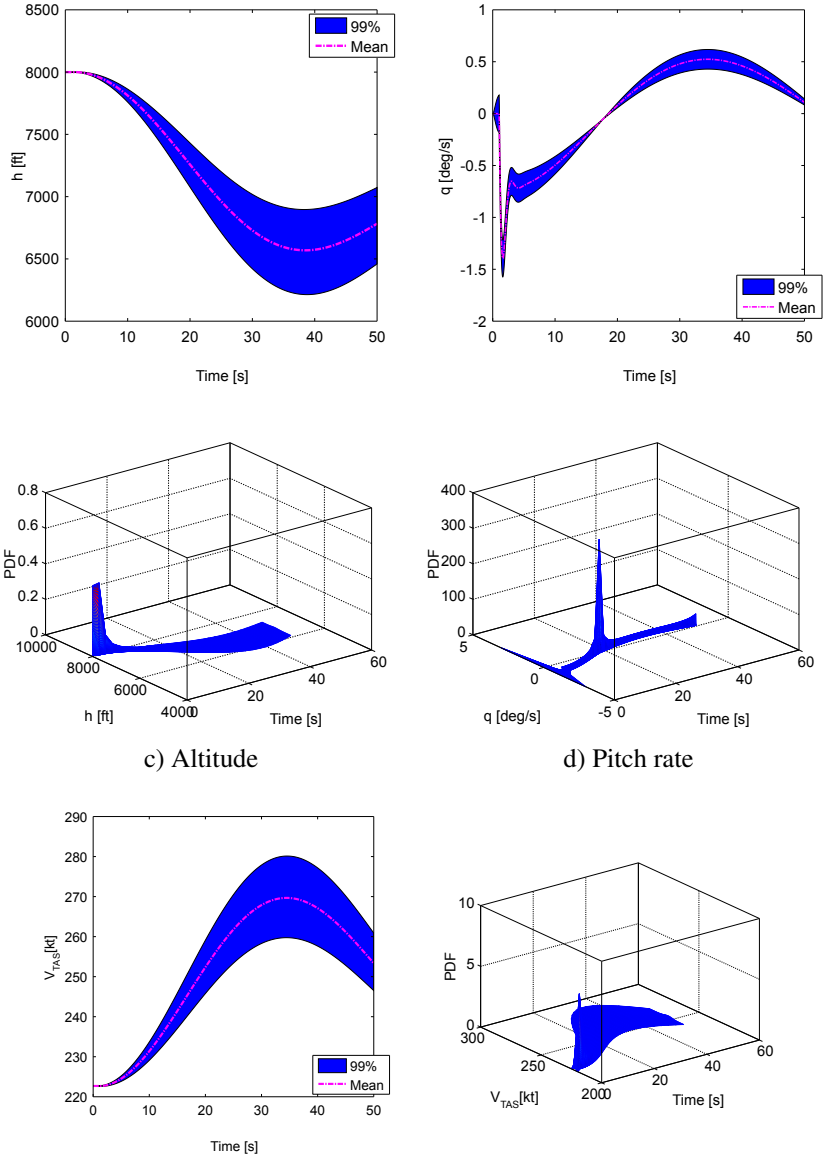
According to Fig. 6a) the angle of attack (AoA) admits sudden changes in the corresponding uncertainty region and hence PDF. Initially, the uncertainty of AoA is quite large, and the PDF is broad. However, after the system is initiated by the Heaviside impulse the uncertainty in AoA starts decreasing. This causes a narrowing of the PDF over time until $t = 20$ [s]. In this moment the 99% region admits the minimum and the pick of PDF occurs such that the response becomes almost deterministic. After $t = 20$ [s] the uncertainty in the response slightly grows following the trend of the AoA mean value trajectory. In contrast to this, the uncertainty in the pitch attitude (see Fig. 6b) grows with time until $t = 20$ [s] when the corresponding 99% region admits maximum. After this, the uncertainty decays over time such that the pitch attitude admits the minimal (almost deterministic) value at 40 [s]. Similarly to these, one may also describe the uncertainties in the altitude, pitch rate and true air speed. For more details please see Fig. 7.

Once the uncertainties are quantified, one may compare the modified coefficients of variation (the ratios of the standard deviation to the mean maximum) for different aircraft response characteristics as presented in Fig. 8. The comparison shows that the uncertainty in the pitch attitude grows until circa 10%, while the second maximal value takes the pitch rate, as expected. The least uncertain is the angle of attack whose coefficient of variation is much smaller than 1%.



a) Angle of attack b) Pitch attitude

Figure 6: The mean aircraft path and the 99% confidence intervals for the angle of attack and the pitch attitude. The results are obtained by the Monte Carlo simulation with 100000 samples



c) Altitude

d) Pitch rate

e) True air speed

Figure 7: The mean aircraft path and the 99% confidence intervals for the altitude, pitch rate and true air speed. The results are obtained by the Monte Carlo simulation with 100000 samples

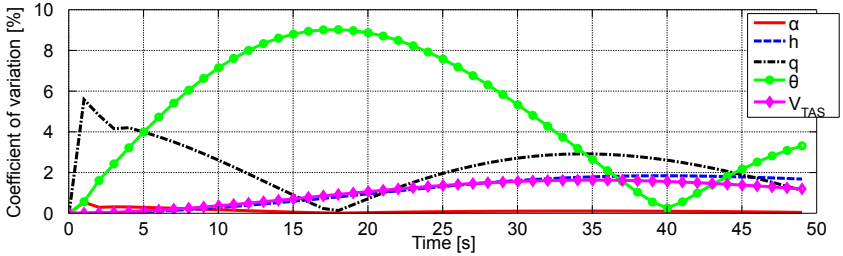


Figure 8: The coefficient of variation for the aircraft response characteristics: α- the angle of attack, h - the altitude, q - the pitch rate, θ- the pitch attitude and V_{TAS} - the true airspeed

5.1 Accuracy analysis

The proxy models are built with the help of non-intrusive numerical approaches: numerical integration by sparse grids (Int), pseudo-Galerkin technique (PG) and stochastic collocation (COL) as described in Section 4. For this purpose are used the nested and non-nested Smolyak Gauss-Hermite sparse grids [12] as shown in Fig. 9. Note that for the same integration level, the nested grid is characterised by a much smaller number of sample points (function calls) than the corresponding non-nested grid.

In order to investigate the accuracy of the non-intrusive methods, the following error estimates

$$\varepsilon_m(\kappa) = \frac{|\mathbb{E}(\kappa(t_i)) - \mathbb{E}(\kappa_{MC}(t_i))|}{\max|\mathbb{E}(\kappa_{MC}(t_i))|}, \quad \varepsilon_v(\kappa) = \frac{|\mathrm{var}\kappa(t_i) - \mathrm{var}\kappa_{MC}(t_i)|}{\max|\mathrm{var}\kappa_{MC}(t_i)|}, \quad \forall t_i \in [0,T] \tag{49}$$

are introduced. The former one, $\varepsilon_m(\alpha)$, is the relative error in the response mean over time, while the latter one, $\varepsilon_v(\kappa)$ is the relative error in variance. To ensure the well-conditioned estimates, the maximal value of the mean or variance are taken into account in previous definitions. Here, $\kappa(t_i)$ respresents the response of the non-intrusive method in time t_i and $\kappa_{MC}(t_i)$ is the reference MC solution at the same time moment.

As expected, the accuracy improves with the number of the sample points being used for interpolation (see Section 4.3.1) or numerical integration (see Eq. 17). In

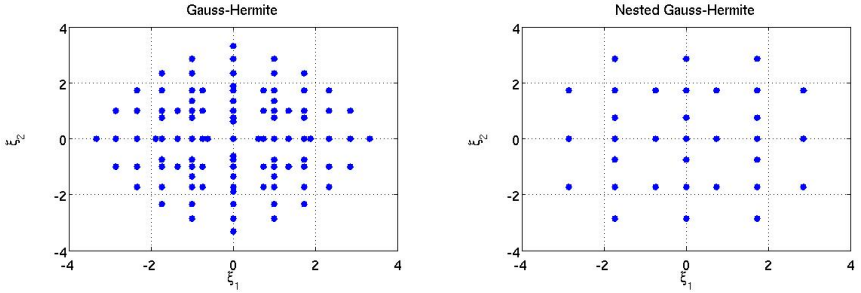
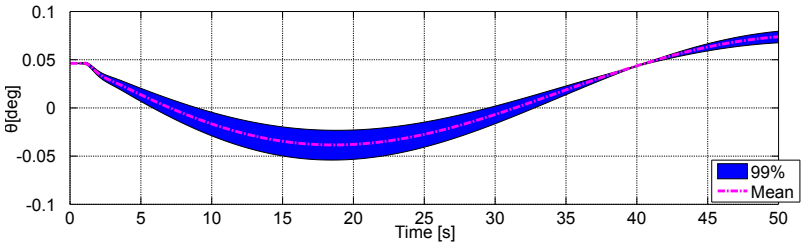


Figure 9: The two dimensional a) non-nested and b) nested Gauss-Hermite sparse grids

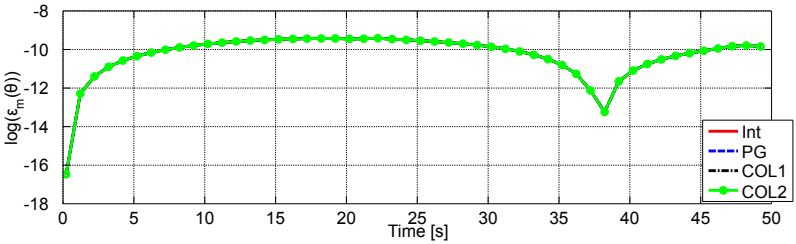
Figs. 10 and 11 are plotted the evolutions of the relative errors $\varepsilon_m(\theta)$ and $\varepsilon_v(\theta)$ for the pitch attitude in time. Based on these results one may notice that already 113 points are enough to achieve the same accuracy in three proxy models: integration by sparse grids (Int), pseudo-Galerkin projection (PG) and stochastic minimisation by least squares (COL1). However, the error in the response obtained by the interpolation/linear regression (COL2) is slightly different than previous ones. Additionaly, the errors continue to change with the increase of the number of sample points. Particularly this happens in the time region where the 99% region of the output uncertainty decreases. This is expected behaviour as at this moment the response is almost deterministic. The same conclusion can be made for the numerical results obtained for the pitch rate (see Figs. 12 and 13) and the angle of attack (see Figs. 14 and 15).

For a more precise investigation, the pseudo-Galerkin method and the interpolation/linear regression are analyzed with respect to the number of sample points, as well as the polynomial order in Fig. 16. As shown, the relative error in the mean and variance of pitch' attitude are slightly oscillating with the increase of the number of the non-nested points in the Gauss-Hermite rule. However, this is not the case when the polynomial order is increased.

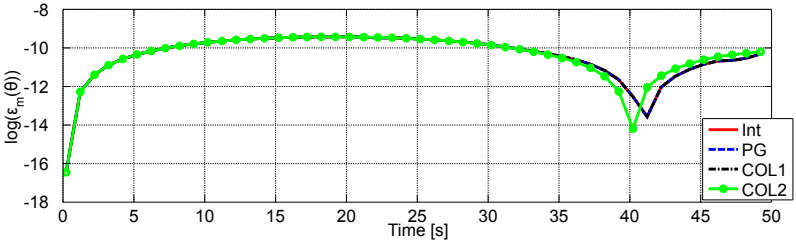
In order to smooth out the error oscillations one may use the nested Gauss-Hermite rule in the numerical integration. This rule is compared to its non-nested cousin in Figs. 17, 18 and Fig. 19 for the same polynomial order and the same grid level. According to these results, the nested rule is shown to be more stable as the error does not drastically change with the increase of the polynomial order, see Figs. 17



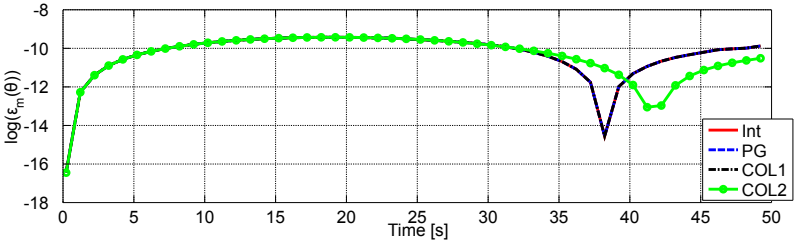
a) Mean value and 99% uncertainty region obtained by MC simulation



b) Results based on non-nested GH sparse grid with 15 points

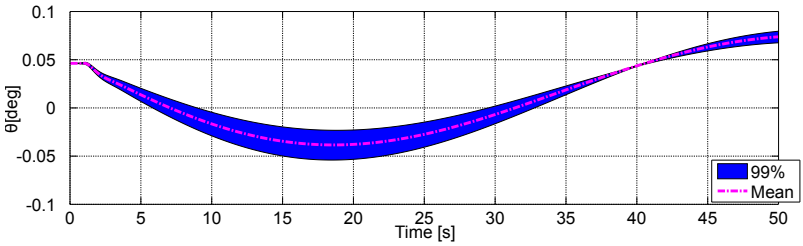


c) Results based on non-nested GH sparse grid with 113 points

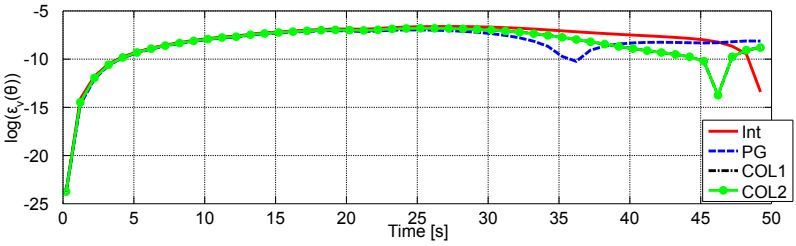


d) Results based on non-nested GH sparse grid with 2437 points

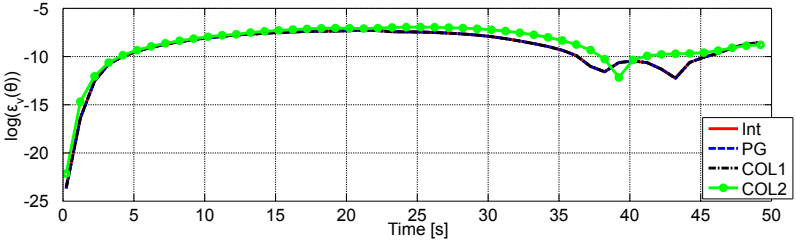
Figure 10: Accuracy comparison of different numerical methods of second polynomial order for the mean pitch attitude: Int-integration, PG-pseudo-Galerkin, COL1-least square, COL2-interpolation/linear regression.



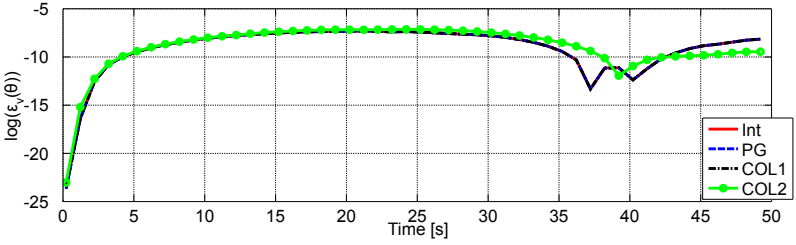
a) Mean value and 99% uncertainty region obtained by MC simulation



b) Results based on non-nested GH sparse grid with 15 points

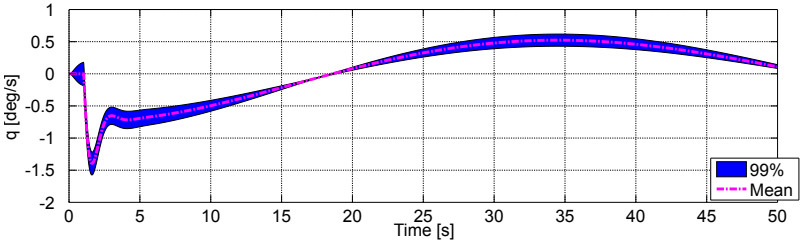


c) Results based on non-nested GH sparse grid with 113 points

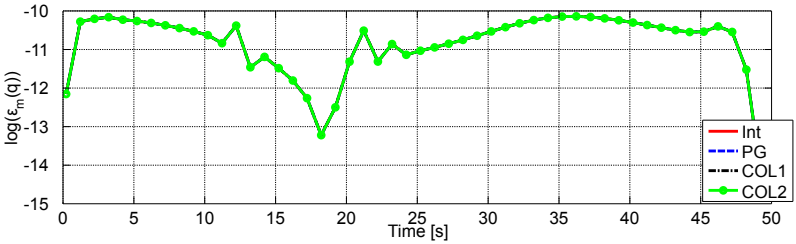


t d) Results based on non-nested GH sparse grid with 2437 points

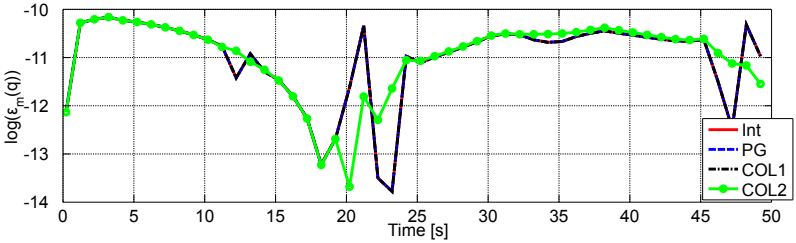
Figure 11: Accuracy comparison of different numerical methods of second polynomial order for the pitch attitude' variance: Int-integration, PG-pseudo-Galerkin, COL1-least square, COL2-interpolation/linear regression



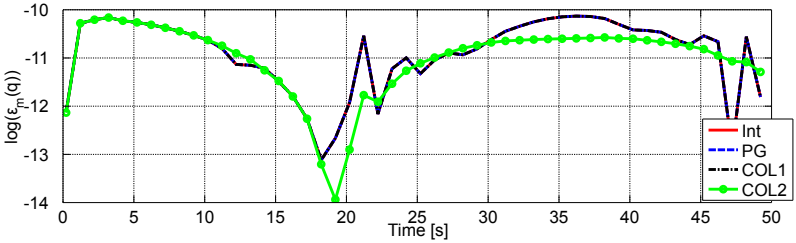
a) Mean value and 99% uncertainty region obtained by MC simulation



b) Results based on non-nested GH sparse grid with 15 points

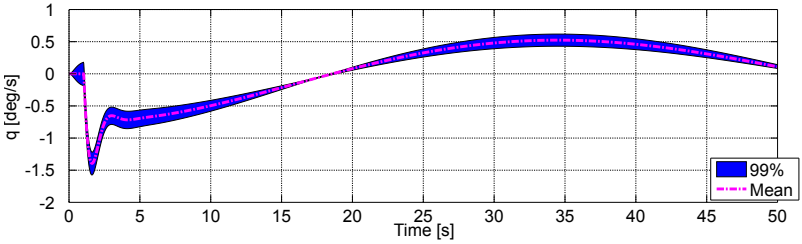


c) Results based on non-nested GH sparse grid with 113 points

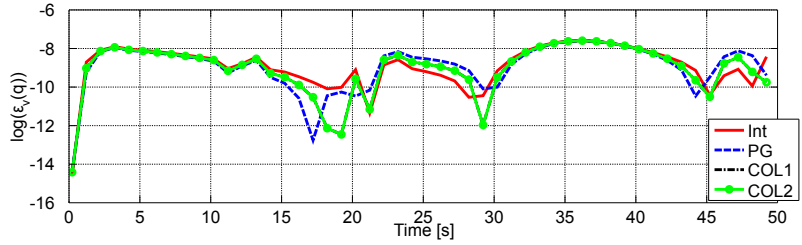


c) Results based on non-nested GH sparse grid with 2437 points

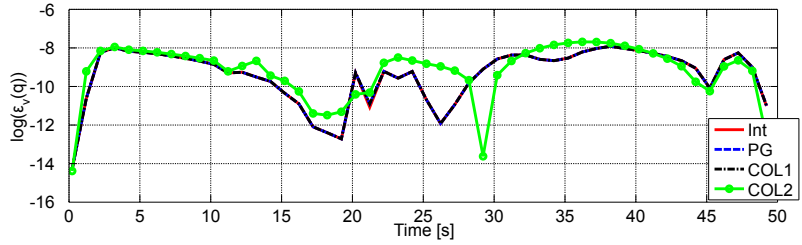
Figure 12: Accuracy comparison of different numerical methods of second polynomial order for the mean pitch: Int-integration, PG-pseudo-Galerkin, COL1-least square, COL2-interpolation/linear regression



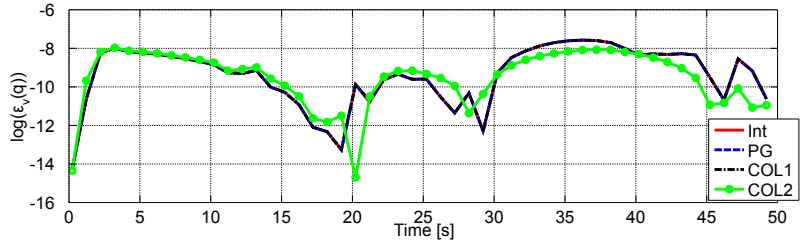
a) Mean value and 99% uncertainty region obtained by MC simulation



b) Results based on non-nested GH sparse grid with 15 points

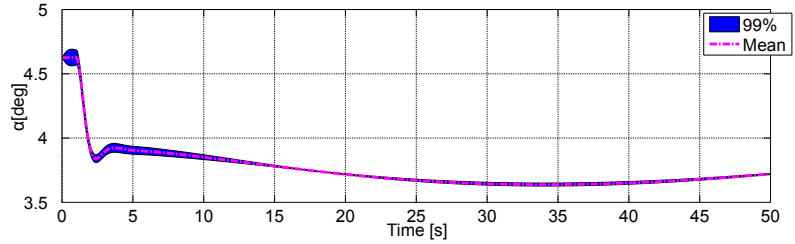


c) Results based on non-nested GH sparse grid with 113 points

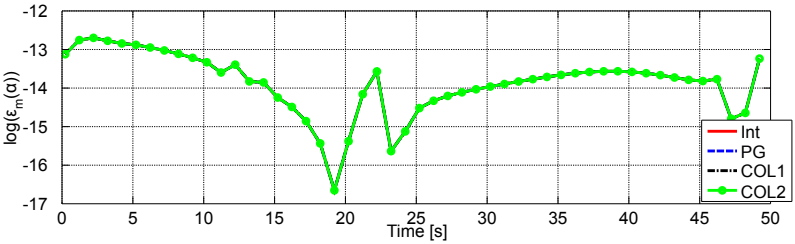


c) Results based on non-nested GH sparse grid with 2437 points

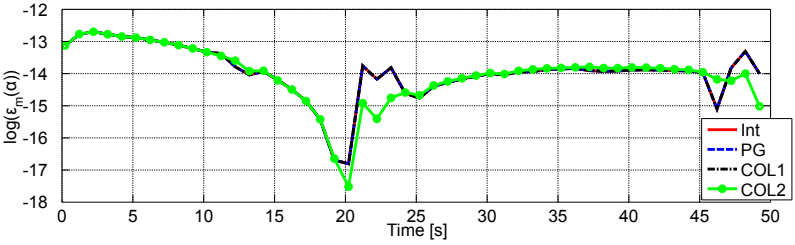
Figure 13: Accuracy comparison of different numerical methods of second polynomial order for the pitch' variance: Int-integration, PG-pseudo-Galerkin, COL1-least square, COL2-interpolation/linear regression



a) Mean value and 99% uncertainty region obtained by MC simulation



b) Results based on non-nested GH sparse grid with 15 points



c) Results based on non-nested GH sparse grid with 113 points

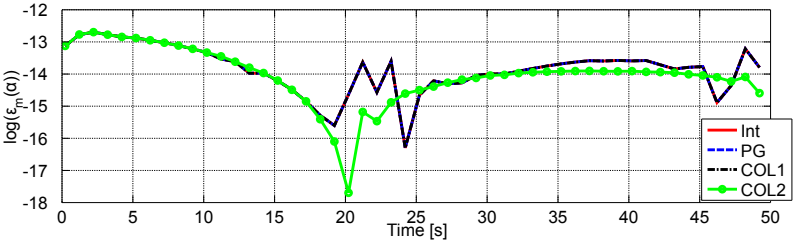
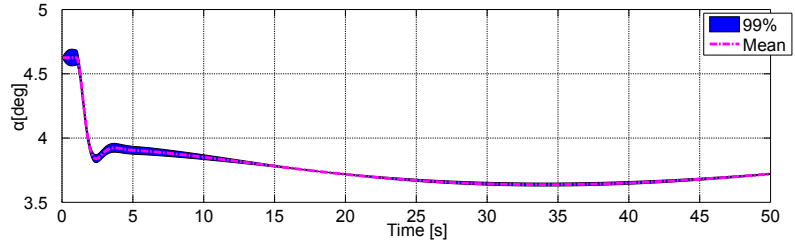
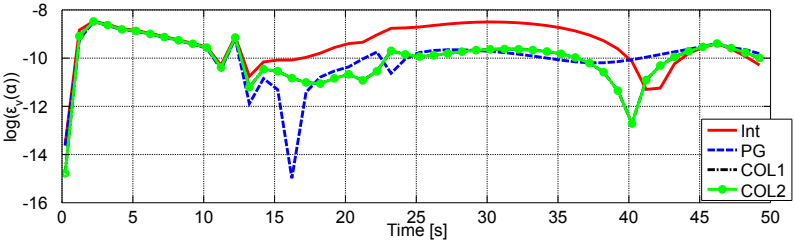


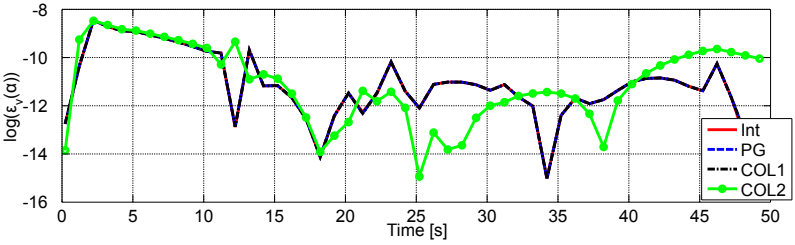
Figure 14: Accuracy comparison of different numerical methods of second polynomial order for the mean AoA: Int-integration, PG-pseudo-Galerkin, COL1-least square, COL2-interpolation/linear regression



a) Mean value and 99% uncertainty region obtained by MC simulation



b) Results based on non-nested GH sparse grid with 15 points



c) Results based on non-nested GH sparse grid with 113 points

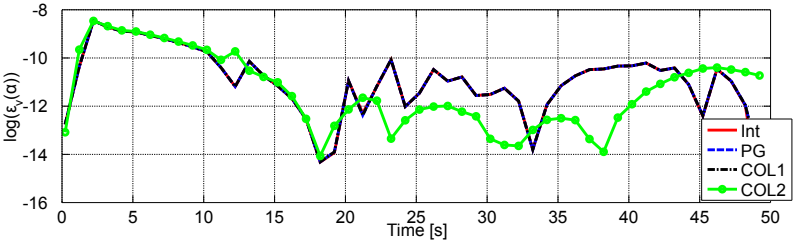
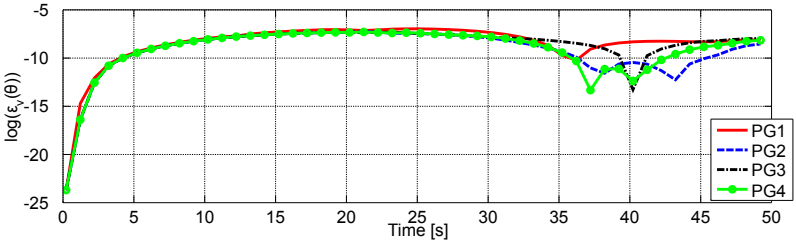
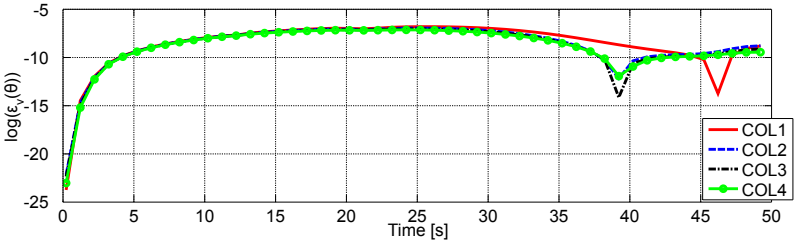


Figure 15: Accuracy comparison of different numerical methods of second polynomial order for the AoA variance: Int-integration, PG-pseudo-Galerkin, COL1-least square, COL2-interpolation/linear regression

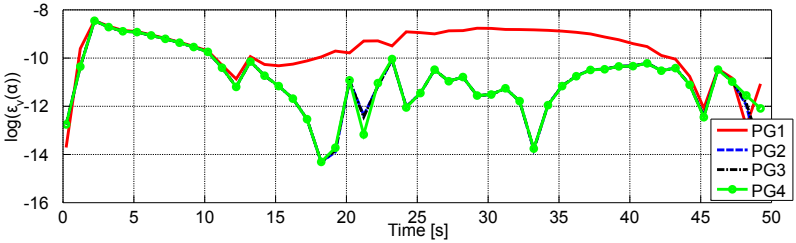


a) PG1= 15 pts, PG2=113 pts, PG3=589 pts, PG4=2437 pts

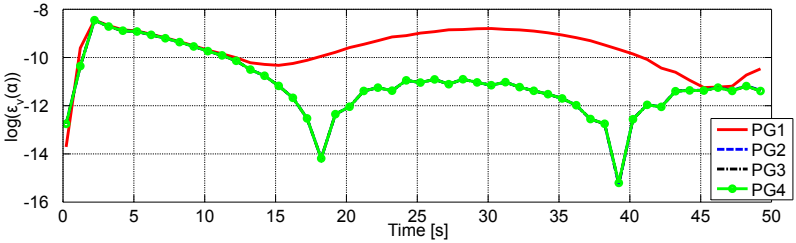


b) COL1= 15 pts, COL2=113 pts, COL3=589 pts, COL4=2437 pts

Figure 16: The evolution of the relative error in pitch's attitude variance for : a) pseudo-Galerkin approach with the non-nested Gauss-Hermite grid , b) stochastic interpolation/linear regression with the non-nested Gauss-Hermite

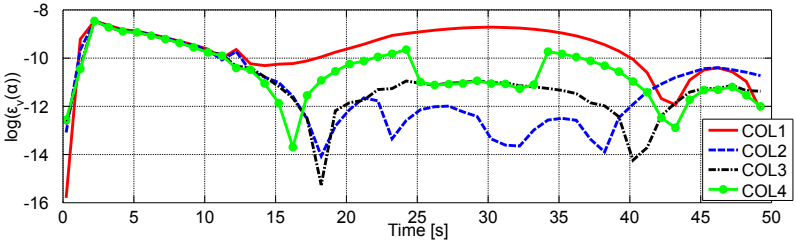


a) The results are obtained with the non-nested rule

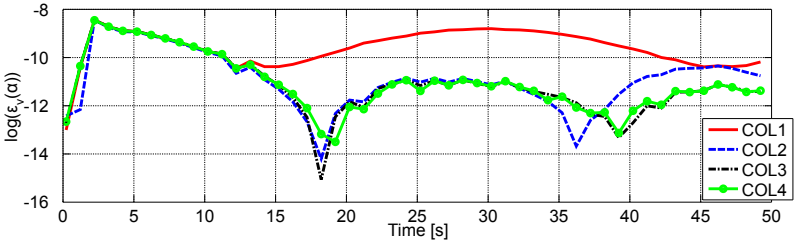


b) The results are obtained with the nested rule

Figure 17: The evolution of the relative error in the AoA variance with respect to the time and the polynomial order of the pseudo-Galerkin projection a) using the non-nested rule b) using the nested-rule. Orders of projections are PG1= 1 order, PG2=2 order, PG3=3 order, PG4=4 order

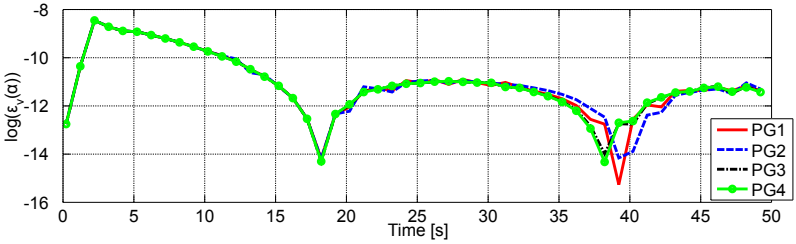


a) The results are obtained with the non-nested rule

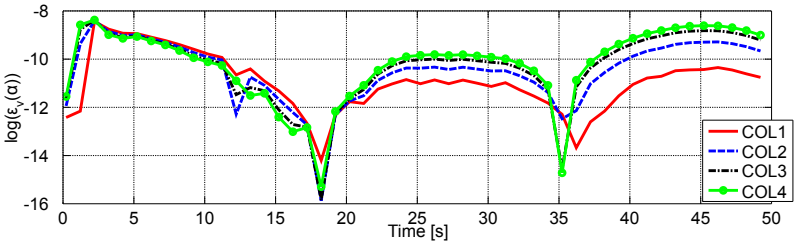


b) The results are obtained with the nested rule

Figure 18: The evolution of the relative error in the AoA variance with respect to the time and the polynomial order used in linear regression (interpolation of polynomial chaos expansion) with a) non-nested rule b) nested rule. The polynomial orders are COL1= 1 order, COL2=2 order, COL3=3 order, COL4=4 order



a) PG1= 15 pts, PG2=113 pts, PG3=589 pts, PG4=2437 pts



b) COL1= 15 pts, COL2=113 pts, COL3=589 pts, COL4=2437 pts

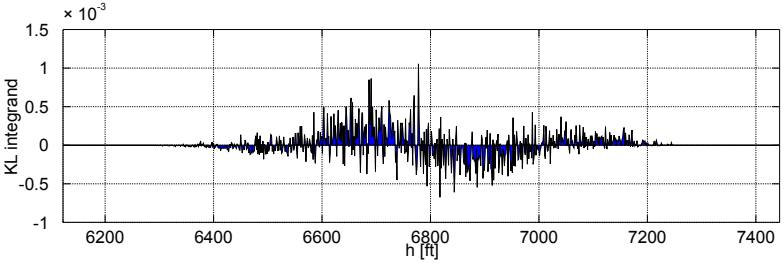
Figure 19: The evolution of the relative error in the AoA variance with respect to time and the number of the non-nested Gauss-Hermite sample points: a) pseudo-Galerkin method b) interpolation

and 18. This is especially the true for the interpolation method. However, the increase of the number of integration points shows that the pseudo-Galerkin method is more stable than the interpolation, see Fig. 19. The instability of the interpolation method happens due to the numerical inaccuracies or the interpolation problem itself. Hence, its application is not very much advised.

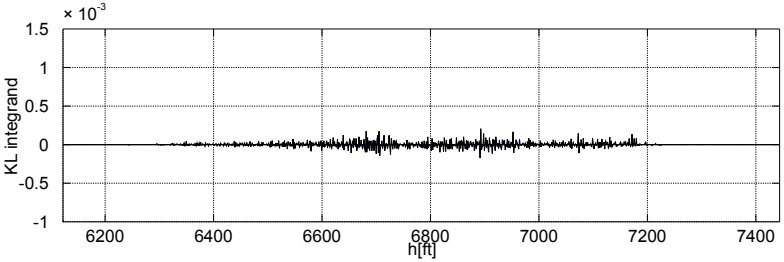
Finally, the numerical comparison can also be done with respect to the Kullback-Leibler divergence (KLD) of the probability density function obtained by numerical integration from the one obtained by the MC simulation. In other words, by computing the integral

$$D_{\mathrm{KL}}(\pi_{NM}\|\pi_{MC}) = \int_{-\infty}^{\infty} \ln\left(\frac{\rho_{NM}(x)}{\rho_{MC}(x)}\right)\rho_{NM}(x)\,\mathrm{d}x, \tag{50}$$

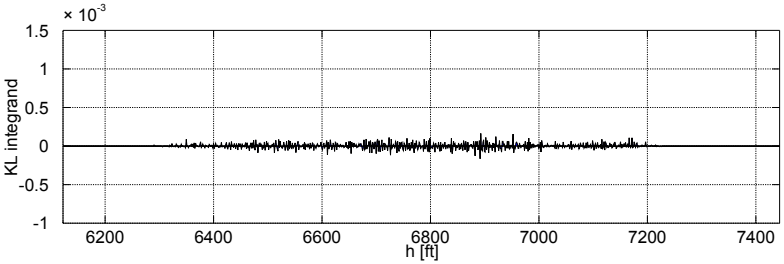
where ρ_{NI} and ρ_{MC} represent the densities of the quantity obtained by a non-intrusive and MC method respectively, one may "measure" how much the PDF of the proxy



a) The KLD difference for 15 points



b) The KLD difference for 113 points



c) The KLD difference for 589 points

Figure 20: The Kullback-Leibler divergence of the PDF for the aircraft altitude

model differs from the original one. In case of the altitude this difference is drawn in Fig. 20 at the end of the time interval. The three plots show that the KLD difference drops down almost to noise with the increase of the number of samples.

5.2 Computation time

Excluding the direct integration approach by simple Monte Carlo simulation, the computation times of numerical schemes presented in this paper are more or less similar. If implemented in a sequential way, the computation cost of the Monte Carlo procedure is:

$$t_{MC} = N \cdot (t_{trim} + t_{sim}), \tag{51}$$

where N is the number of BACM calls, t_{trim} is the time of trimming subroutine and $t_{sim} \approx 60$[s] is the actual BACM simulation time. However, the Monte Carlo method is "embarrassingly parallelisable" and hence the overall computation time can be reduced significantly [21, 8]. Even though the different parallelisation techniques can be applied, the Monte Carlo method is still considered to be inefficient when compared to the other non-intrusive numerical schemes. Namely, for the same accuracy the pseudo-Galerkin and stochastic collocation methods use much smaller number of BACM calls (samples) compared to the Monte Carlo method. Additionally, these algorithms can also be parallelised in a similar manner as Monte Carlo procedure. In this regard, their employment is greatly advised as the computational cost is more than 20 magnitudes smaller than the Monte Carlo simulation time for the same accuracy. However, note that the pseudo-Galerkin and the collocation methods are characterised by similar computation costs. This is due to the small number of random variables describing the problem, i.e. small stochastic dimension.

5.3 Sensitivity analysis

The variance-based sensitivity analysis of the system behaviour on the uncertainty in each of the input parameters is investigated with the help of the first-order Sobol' sensitivity index [38]

$$S_1 = \frac{\mathrm{var}_i(\kappa)}{\mathrm{var}(\kappa)}, \tag{52}$$

where $\mathrm{var}_i(\kappa)$ is the partial variance of the response κ given uncertain input ξ_i, and $\mathrm{var}(\kappa)$ is the total response variance when there are uncertainties in all inputs $\xi_1, \xi_2, \cdots, \xi_7$. Note here that $\xi_1, \xi_2, \cdots, \xi_7$ are the RVs used to describe C_{L0}, $C_{L\alpha}$,

Table 3: The sensitivity analysis of output using the first order Sobol' index SU_1

SU_1	C_{L0}	$C_{L\alpha}$	C_{w0}	k_1	k_2	C_{m0}	C_{mq}
θ [deg]	0.1902	0.8093	1.8e-06	1.8e-06	1.8e-06	1.8e-06	1.8e-06
α [deg]	0.3477	0.6489	1.1e-06	1.1e-06	1.1e-06	1.1e-06	1.1e-06
h [ft]	0.2126	0.7869	9.3e-08	9.3e-08	9.3e-08	9.3e-08	9.3e-08
q [deg/s]	0.2797	0.7172	1.9e-04	1.9e-04	1.9e-04	1.9e-04	1.9e-04
V_{TAS} [kt]	0.2188	0.7808	2.2e-08	2.2e-08	2.2e-08	2.2e-08	2.2e-08

$C_{W0}, k_1, k_2, C_{m0}, C_{m\alpha}$ (see Tab. 1), while κ is the generic variable used to denote one of the system responses presented in previous section.

The index in Eq. 52 is computed with the help of the PCE based proxy model such that:

$$S_1 = \frac{\sum_{\beta \in \mathscr{J}_i} \kappa_\beta^2 \beta!}{\sum_{\gamma \in \mathscr{J}} \kappa_\gamma^2 \gamma!}, \tag{53}$$

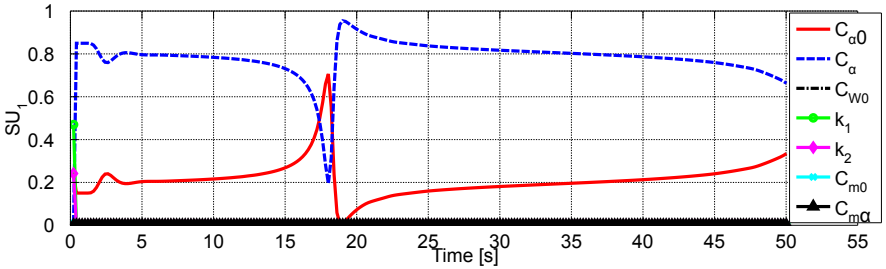
where $\mathscr{J}$ is the full multi-index set determined by all input random variables and corresponding polynomial order, while $\mathscr{J}_i$ is the subset of $\mathscr{J}$ including all polynomials dependent on the i-th random variable only. In other words, one segments the polynomial chaos expansion to

$$\begin{aligned} \kappa(\xi) = \bar{\kappa} \quad &+ \sum_{\beta \in \mathscr{J}_1} \kappa_\beta \Psi_\beta(\xi_1) + \cdots + \sum_{\beta \in \mathscr{J}_7} \kappa_\beta \Psi_\beta(\xi_7) \\ &+ \sum_{\beta \in \mathscr{J}_{12}} \kappa_\beta \Psi_\beta(\xi_1 \xi_2) + \cdots + \sum_{\beta \in \mathscr{J}_{67}} \kappa_\beta \Psi_\beta(\xi_6 \xi_7) + \text{h.o.t}, \end{aligned} \tag{54}$$

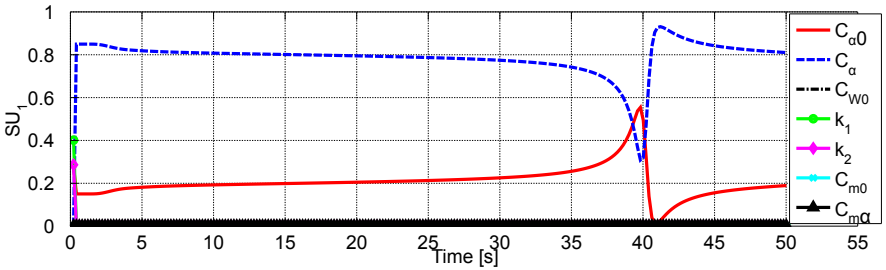
and uses only the first order terms to evaluate Eq. 53. Here, $\mathscr{J}_{ij}$ is the subset of multi-index set $\mathscr{J}$ including the polynomials which depend only on the set of variables (ξ_i, ξ_j), $i \neq j$.

Using the previous formula, the sensitivity of the output variables is studied with respect to each of the seven inputs in Tab. 3. By comparison one may conclude that the parameters describing the lift have the highest impact on the output uncertainty. The perturbations in other input variables such as the drag and pitch moments do not influence the response statistics in a great manner. Thus, their impact can be neglected. The reason for this is based on the fact that the lift compared to drag is the predominating force component (about one order of magnitude difference).

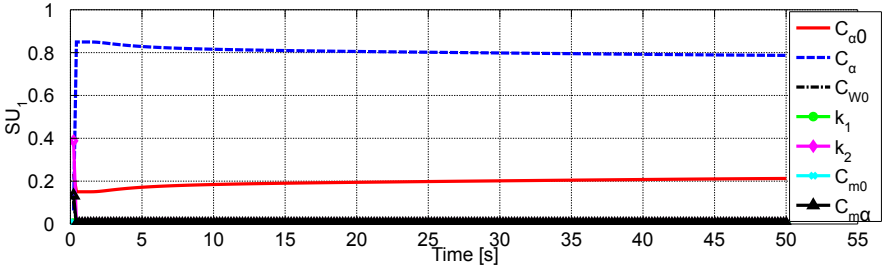
In Fig. 21 the sensitivity evolution of AoA on input uncertainty is shown. Namely,



a) Evolution of sensitivity of the angle of attack on the input uncertainties



b) Evolution of sensitivity of the pitch attitude on the input uncertainties



c) Evolution of sensitivity of the altitude on the input uncertainties

Figure 21: Sensitivity analysis of the angle of attack, altitude and the pitch attitude

Table 4: Comparison of the first sensitivity Sobol' index obtained by polynomial chaos (PCE) and Monte Carlo (MC) approaches

SU_1	θ [deg]	α [deg]	h [ft]	q [deg/s]	V_{TAS} [kt]
PCE	0.1902	0.3477	0.2126	0.2797	0.2188
MC	0.1887	0.3481	0.2099	0.2773	0.2160

before the initial pulse function is applied one may notice the strong correlation between the AoA and k_1 and k_2. However, after the excitation is imposed the aircraft becomes sensitive to the perturbations of the lift parameters C_{L0} and $C_{L\alpha}$. These two are co-dependent such that when one grows the other decays and vice versa. However, the correlation bewteen AoA and C_α is stronger than the correlation between AoA and $C_{\alpha 0}$, except in the moment $t = 20$ [s] when the uncertainty in the output becomes negligible. Similarly, the pitch attitude is also strongly correlated to $C_{L\alpha}$ and moderately to C_{L0}. These two shortly exchange roles in time $t = 40$ when the response becomes almost deterministic. In contrast to this, the sensitivity of the altitude to the presence of uncertainty in the lift coefficient is almost constant over time and takes the value of ca. 0.8 in case of $C_{L\alpha}$, and ca. 0.2 in case of C_{L0}.

In order to check the accuracy of the previous results, the first senstivity index of the output response with respect to the uncertainty in the initial lift coefficient is computed with the Monte Carlo simulation and 100 000 samples. According to the comparison analysis provided in Tab. 4 one may conclude that the sensitivity indices match on the second decimal as expected.

6 Conclusions

The present paper deals with the propagation of uncertainty in input parameters through the aircraft model in clean cruise configuration triggered by the elevator pulse. For the numerical integration the non-intrusive stochastic approaches are used, such as the direct integration, pseudo-Galerkin and stochastic collocation methods. In addition, the paper presents some algorithms used to reduce the memory requirements for the response statistics, which is of high importance when one deals with real problems such as aircraft dynamics.

The numerical study has shown that the perturbations of the lift parameters have a high impact on the system behaviour. This conclusion is verified with the help of the stochastic collocation and stochastic Galerkin method. The comparison of the direct integration and sparse grid algorithms goes in favour to the second approach as the number of calls of the simulator is reduced to more than 20 times for a certain level of accuracy. On the other hand, the comparison of the nested and non-nested grids has shown that the former ones are more suitable for the aircraft model integration as they are more stable with the increase of the number of samples. This allows us to strongly believe that not only the considered problem but also the flight dynamics during the landing phase can be resolved by the most popular stochastic finite element method based on the Galerkin projection or stochastic collocation. However, note that the stochastic interpolation has to be taken with care as the numerical solution can admit strong oscilations with the increase of the polynomial order. As numerically efficient, the non-intrusive methods can also be utilised for the identification of aerodynamical coefficients in the Bayesian framework, which is the future subject of this work.

Acknowledgments

Financial support has been provided by the German Research Foundation (Deutsche Forschungsgemeinschaft – DFG) in the framework of the Sonderforschungsbereich 880.

References

[1] G. Alefeld and J. Herzberger. *Introduction to interval computation*. Academic Press, London, 1983.

[2] A. Barth. *Stochastic partial differential equations: approximations and applications*. PhD thesis, Center of Mathematics for Applications, University of Oslo, Norway, 2009.

[3] N. E. Barthelmann and K. Ritter. High dimensional polynomial interpolation on sparse grids. *Advances in Computational Mathematics*, 12(4):273–288, 2000.

[4] G. Blatman. *Adaptive sparse polynomial chaos expansions for uncertainty propagation and sensitivity analysis*. PhD thesis, Laboratoire de Mécanique et Ingénieries, Institut Français de Mécanique Avancée et Université Blaise Pascal, France, 2009.

[5] S. Box, C. Bishop, and H. Hunt. Stochastic six-degree-of-freedom flight simulator for passively controlled high-power rockets. *Journal of Aerospace Engineering*, 24:31–45, 2011.

[6] M. Brand. Fast low-rank modifications of the thin singular value decomposition. *Linear Algebra and its Applications*, 415:20–30, 2006.

[7] J. Diekmann. Investigation of flight dynamics of a civil aircraft with active high-lift system. Technical report, Institute of Flight Systems, DLR, Braunschweig, Germany, 2012.

[8] V. D. Doan. *Grid computing for Monte Carlo based intensive calculations in financial derivative pricing applications*. PhD thesis, University of Nice, 2010.

[9] T. Gerhold. Overview of the hybrid rans code TAU. In N. Kroll and J. K. Fassbender, editors, *MEGAFLOW – Numerical Flow Simulation for Aircraft Design*, volume 89, pages 81–92. Springer Verlag, Berlin Heidelberg, Germany, 2005.

[10] T. Gerstner and M. Griebel. Numerical integration using sparse grids. *Numer Algorithms*, 18(3–4):209–232, 1998.

[11] L. Giraldi, A. Litvinenko, D. Liu, H. G. Matthies, and A. Nouy. To be or not to be intrusive? The solution of parametric and stochastic equations - the "plain vanilla" Galerkin case. Technical report, Institut für Wissenschaftliches Rechnen, TU Braunschweig, 2013.

[12] F. Heiss and V. Winschel. Likelihood approximation by numerical integration on sparse grids. *Journal of Econometrics*, 144:62–80, 2008.

[13] T. Hida, H. H. Kuo, J. Potthoff, and L. Streit. *White noise analysis-an infinite dimensional calculus*. Kluwer, Dordrecht, 1993.

[14] A. Keese. *Numerical solution of systems with stochastic uncertainties. A general purpose framework for stochastic finite elements*. PhD thesis, Department of Mathematics and Computer Science, TU Braunschweig, Germany, 2003.

[15] F. Khoshnoud. *Novel modal analysis method based on fuzzy sets*. PhD thesis, Brunel University School of Engineering and Design PhD Theses, UK, 2005.

[16] A. Klimke. Sparse grid interpolation toolbox users guide. Technical report, Universität Stuttgart, 2007.

[17] W. A. Klimke. *Uncertainty modeling using fuzzy arithmetics and sparse grids*. PhD thesis, Institut für Angewandte Analysis und Numerische Simulation,Universität Stuttgart, Germany, 2005.

[18] X. Ma and N. Zabaras. An adaptive hierarchical sparse grid collocation algorithm for the solution of stochastic differential equations. *Journal of Computational Physics*, 228(8):3084–3113, 2009.

[19] H. G. Matthies. Stochastic finite elements: computational approaches to stochastic partial differential equations. *Zeitschrift für Angewandte Mathematik und Mechanik (ZAMM)*, 88(11):849–873, 2008.

[20] R. E. Moore. *Interval analysis*. Prentice-Hall, Englewood Cliffs, New Jersey, 1966.

[21] H. Moritsch. *High performance computing in finance–on the parallel implementation of pricing and optimization models*. PhD thesis, Institut für Softwaretechnik und Interaktive Systeme, Technische Universität Wien, 2006.

[22] N. Nassiri, J. Roushanian, and S. Haghighat. Stochastic flight simulation applied to a sounding rocket. In *Proceedings of 55th International Astronautical Congress 2004, Vancouver, Canada*, 2004.

[23] F. Nobile, R. Tempone, and C. G. Webster. A sparse grid stochastic collocation method for partial differential equations with random input data. *SIAM Journal on Numerical Analysis*, 46(5):2309–2345, 2008.

[24] W. Oberkampf, S. DeLand, B. Rutherford B., K. Diegert, and K. Alvin. Error and uncertainty in modeling and simulation. *Reliability Engineering and System Safety*, 75:333–357, 2002.

[25] B. Øksendal. *Stochastic differential equations: an introduction with applications*. Springer Heidelberg Dordrecht London New York, 6th edition, 2003.

[26] O. Pajonk, B. Rosić, and H. G. Matthies. Sampling-free linear Bayesian updating of model state and parameters using a square root approach. *Computers & Geosciences*, 55:70–83, 2013.

[27] A. Prabhakar. Uncertainty propagation in hypersonic flight dynamics and comparison of different methods. Master's thesis, Texas A & M University, USA, 2008.

[28] C. Raab. A versatile and modular architecture for aircraft system simulation and test. Technical report, Institute of Flight Systems, DLR, Braunschweig, Germany, 2006.

[29] C. Restrepo and J. Hurtado. Pattern recognition for a flight dynamics Monte Carlo simulation. *Papers - American Institute of Aeronautics and Astronautics*, 8:6579–6587, 2011.

[30] J. Rosenthal. Parallel computing and Monte Carlo algorithms. *Far East Journal of Theoretical Statistics*, 4:207–236, 2000.

[31] B. Rosić. *Variational formulations and functional approximation algorthms in stochastic plasticity of materials*. PhD thesis, Institute of Scientific Computing, TU Braunschweig, 2012.

[32] B. Rosić, A. Litvinenko, O. Pajonk, and H. G. Matthies. Sampling-free linear Bayesian update of polynomial chaos representations. *Journal of Computational Physics*, 231(17):5761 – 5787, 2012.

[33] S. Särkkä and E. Platen. Applied stochastic differential equations. Technical report, Tampere University of Tehnology, Finland, 2012.

[34] I. E. Segal and R. A. Kunze. *Integrals and Operators*. Springer Verlag, Berlin, 1978.

[35] A. V. Skorokhod. *Studies in the theory of random processes*. Dover Publications, New York, 1982.

[36] C. Snyder, T. Bengtsson, P. Bickel, and J. Anderson. Obstacles to high-dimensional particle filltering. *Monthly Weather Review*, 136:4629–4640, 2008.

[37] S. Steinkellner. *Aircraft vehicle systems modeling and simulation under uncertainty*. PhD thesis, Department of managment and engineering, Linköpings Universitet, Sweden, 2011.

[38] B. Sudret. Global sensitivity analysis using polynomial chaos expansions. *Reliability Engineering and System Safety*, 93:964–979, 2008.

[39] N. Wiener. The homogeneous chaos. *The American Journal of Mathematics*, 60(4):1936–1938, 1938.

[40] J. E. Williams and S. R. Vukelich. The USAF stability and control digital DATCOM. volume i. users manual. Technical report, Wright-Patterson Air Force Base, USA, 1979.

[41] F. Xiong, W. Chen, Y. Xiong, and S. Yang. Weighted stochastic response surface method considering sample weights. *Struct. Multidiscip. Optim.*, 43(6):837–849, 2011.

[42] D. Xiu and G. E. Karniadakis. Modeling uncertainty in flow simulations via generalized polynomial chaos. *Journal of Computational Physics*, 187(1):137–167, 2003.

Technische Universität Braunschweig

Informatik-Berichte ab Nr. 2011-06

2011-06	R. van Glabbeek, U. Goltz, J.-W. Schicke	On Causal Semantics of Petri Nets
2011-07	H. Cichos, S. Oster, M. Lochau, A. Schürr	Extended Version of Model-based Coverage-Driven Test Suite Generation for Software Product Lines
2011-08	W.-B. Pöttner, J. Morgenroth, S. Schildt, L. Wolf	An Empirical Performance Comparison of DTN Bundle Protocol Implementations
2011-09	H. G. Matthies, A. Litvinenko, O. Pajonk, B. V. Rosić and E. Zander	Parametric and Uncertainty Computations with Tensor Product Representations
2011-10	B. V. Rosić, A. Kučerová, J. Sýkora, A. Litvinenko, O. Pajonk and H. G. Matthies	Parameter Identification in a Probabilistic Setting
2011-11	M. Espig, W. Hackbusch, A. Litvinenko, H. G. Matthies and E. Zander	Efficient Analysis of High Dimensional Data in Tensor Formats
2011-12	S. Oster	A Semantic Preserving Feature Model to CSP Transformation
2012-01	O. Pajonk, B. V. Rosić and H. G. Matthies	Deterministic Linear Bayesian Updating of State and Model Parameters for a Chaotic Model
2012-02	B. V. Rosić and H. G. Matthies	Stochastic Plasticity — A Variational Inequality Formulation and Functional Approximation Approach I: The Linear Case
2012-03	J. Rang	An analysis of the Prothero–Robinson example for constructing new DIRK and ROW methods

2012-04	S. Kolatzki, M. Hagner, U. Goltz and A. Rausch	A Formal Definition for the Description of Distributed Concurrent Components - Extended Version
2012-05	M. Espig, W. Hackbusch, A. Litvinenko, H. G. Matthies and P. Wähnert	Efficient low-rank approximation of the stochastic Galerkin matrix in tensor formats
2012-06	S. Mennike	A Petri Net Semantics for the Join-Calculus
2012-07	S. Lity, R. Lachmann, M. Lochau, I. Schaefer	Delta-oriented Software Product Line Test Models - The Body Comfort System Case Study
2013-01	M. Lochau, S. Mennicke, J. Schroeter und T. Winkelmann	Extended Version of 'Automated Verification of Feature Model Configuration Processes based on Workflow Petri Nets'
2013-02	S. Lity, M. Lochau, U. Goltz	A Formal Operational Semantics of Sequential Function Tables for Model-based SPL Conformance Testing
2013-03	L. Giraldi, A. Litvinenko, D. Liu, H. G. Matthies, A. Nouy	To be or not to be intrusive? The solution of parametric and stochastic equations – the "plain vanilla" Galerkin case
2013-04	A. Litvinenko, H. G. Matthies	Inverse problems and uncertainty quantification
2013-05	J. Rang	Improved traditional Rosenbrock-Wanner methods for stiff ODEs and DAEs
2013-06	J. Koslowski	Deterministic single-state 2PDAs are Turing-complete
2014-01	B. Rosić, J. Diekmann	Stochastic Description of Aircraft Simulation Models and Numerical Approaches