UNCECOMP 2017
2nd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15–17 June 2017

ON THE DEVELOPMENT OF THE 3D EULER EQUATIONS USING INTRUSIVE PCE FOR UNCERTAINTY QUANTIFICATION

Kyriakos-Dimitrios Kantarakias, Michail E. Chatzimanolakis, Varvara G. Asouti and Kyriakos C. Giannakoglou

National Technical University of Athens, School of Mech. Eng.,
Parallel CFD & Optimization Unit, Athens, Greece
e-mail: {kyr.kantar,mikechatzimanolakis}@gmail.com,
vasouti@mail.ntua.gr, kgianna@central.ntua.gr

Keywords: Uncertainty Quantification, Intrusive Polynomial Chaos

Abstract. *This paper presents a method for the quantification of uncertainty propagation using intrusive Polynomial Chaos Expansion (iPCE) in CFD. In constrast to commonly implemented non–intrusive methods which take advantage of existing CFD evaluation software in order to quantify the statistical behavior of the flow, an intrusive PCE method is developed and implemented to the 3D Euler equations. Uncertainties are introduced through the flow conditions and their propagation throughout the flow field is quantified. A Probability Density Function (PDF) is assumed for each uncertain flow condition and the generalized PCE inviscid equations for the corresponding coefficient fields of the flow variables are derived. Already known properties of the equations, such as the first order-homogeneity, are found to hold in the new set of the equations. The discretization schemes are adapted to the new set of governing equations while a systematic approach to the corresponding eigenproblem is introduced. The method is applied to 3D inviscid flow cases for which the mean value and the standard deviation of specific flow quantities characterizing the flow are quantified and compared with those computed by the non–intrusive PCE and Monte–Carlo methods.*

1 INTRODUCTION

Computational Fluid Dynamics (CFD) methods ans software are indispensable tools in the analysis and design in various engineering fields. The common practice is to analyze the flow under consideration using a deterministic computational model. However, such ideal situations are rare in real–world applications and the system's performance is sensible to varying conditions; for instance, a compressor's efficiency is affected by variations/uncertainties associated with the inlet and/or outlet conditions etc. In order to take into account these uncertainties, non–deterministic approaches should be used.

During the last decade, polynomial chaos expansion (PCE) methods have been used to model uncertainties in engineering applications [3, 4, 5]. These are based on the idea of homogeneous chaos, [1, 2]. In contrast to sampling techniques such as Monte–Carlo, PCE methods are based on the spectral representation of the uncertain quantities.

PCE methods can be intrusive (iPCE) or non–intrusive (niPCE) depending on whether the governing equations are altered or not, [7, 4]. In the non–intrusive variant the CFD code is used in its standard form as an evaluation software along with an integration formula based on Gauss quadrature and the appropriate weighting functions that correspond to the assumed PDFs. Through this method, the statistical moments of an objective function (such as the lift coefficient of a wing) are computed. On the other hand, the intrusive methodology introduces uncertainties in the mathematical model, the effect of which appears in the flow equations through the PCE of the flow variables. This results in a new set of PDEs governing the PCE coefficient fields of the flow variables to be numerically solved. Contrary to its non–intrusive variant, the iPCE method computes the statistical moments of the flow field and, through post–processing, those of the objective function of interest. In terms of computational cost, both PCE variants are far more efficient than the Monte–Carlo method which requires thousands of CFD evaluations in order to compute the statistical moments of the objective function.

In the present paper, the iPCE of the 3D Euler equations is presented. Emphasis is laid on the derivation of the stochastic flow equations and their discretization. Uncertainties are introduced by the boundary conditions. Other forms, such as uncertainties in the geometry or the properties of the gas are not discussed in this paper, however their effect can be quantified through a similar method.

2 UNCERTAINTY QUANTIFICATION (UQ) USING PCE

Let us assume a set of m uncertain or stochastic variables $\vec{\xi} = (\xi_1, ..., \xi_m)$. Their probability density functions (PDF) w_i are associated with an orthogonal polynomial bases $\psi^{(i)} = \{\psi_0^{(i)}, \psi_1^{(i)}, \dots\}$, where by definition

$$< \psi_j^{(i)}, \psi_k^{(i)} >= \int_{\mathcal{E}_i} \psi_j^{(i)} \psi_k^{(i)} w_i d\xi_i = \delta_{jk} < \psi_j^{(i)}, \psi_j^{(i)} > \tag{1}$$

In eq. 1, no summation for the repeated indices is implied, δ_{jk} is the Kronecker symbol and $\mathcal{E}_i$ denotes the domain of w_i.

The PCE of any quantity $\phi(\vec{\xi})$ can be expressed by using a polynomial basis, defined as the tensor product of $\psi^{(i)}$, $\Psi = \otimes_{i=1}^{m} \psi^{(i)} = \{\Psi_0, \Psi_1, \dots\}$, [1], as follows

$$\phi(\vec{\xi}) = \sum_{i=0}^{\infty} \phi_i \Psi_i(\vec{\xi}) \tag{2}$$

Bases Ψ are also orthogonal, namely

$$<\Psi_i, \Psi_j>= \int_{\mathcal{E}} \Psi_i \Psi_j w d\vec{\xi} = \delta_{ij} < \Psi_i, \Psi_i > \tag{3}$$

In eq. 3, $w(\vec{\xi}) = \prod_{i=1}^{m} w_i(\xi_i)$ and $\mathcal{E} = \cup_{i=1}^{m} \mathcal{E}_i$. Based on the above and after normalizing the Ψ polynomials so that $< \Psi_i, \Psi_i >= 1$, the first two statistical moments, namely the mean value and standard deviation, of any quantity ϕ are given by

$$\mu_\phi = \phi_0 \ , \qquad \sigma_\phi = \sqrt{\sum_{i=1}^{\infty} \phi_i^2} \tag{4}$$

In iPCE methods, one should first apply the PCE to each flow variable, truncated to q+1 terms, introduce these expansions into the flow equations and derive a set of new equations and boundary conditions which must be numerically solved to compute the PCE coefficient fields of all flow quantities. The statistical moments of the quantity of interest (such as lift, drag, losses or any other, usually integral, quantity) are computed at a post-processing level, using eqs. 4.

To determine the most appropriate value of q, a common practice is to retain all polynomials up to a user–defined degree C, which is referred to as the chaos order. For a selected chaos order C and m stochastic variables, q is given by

$$q = \frac{(m+C)!}{m!C!} - 1 \tag{5}$$

The required $q+1$ equations per flow variable can be derived through appropriate Galerkin projections of the governing equations/PDEs. The Galerkin projection of any scalar function $\phi(\vec{\xi})$ to Ψ_i is defined as

$$G_i[\phi] = \int_{\mathcal{E}} \Psi_i \phi w d\vec{\xi} \tag{6}$$

By making use of the properties of the selected orthogonal polynomial bases, the $q+1$ Galerkin projections to the flow PDEs give rise to a new set of PDEs.

3 DERIVATION AND NUMERICAL SOLUTION OF THE iPCE EQUATIONS

3.1 The iPCE Flow Equations

The Euler equations for compressible fluid flows are written, in the standard vector form, as

$$\frac{\partial \vec{U}}{\partial t} + \frac{\partial \vec{f_i}}{\partial x_i} = \vec{0} \tag{7}$$

In eq. 7 $\vec{U} = [\varrho \ \varrho\vec{u} \ E_t]^T$ are the conservative flow variables and $\vec{f_i} = [\varrho u_i \ \varrho u_i \vec{u} + p\vec{\delta_i} \ u_i(E_t+p)]^T$ are the inviscid fluxes in Cartesian coordinates, ϱ is the density, $\vec{u} = [u_1, u_2, u_3]^T$ is the velocity vector, E_t is the total energy per unit mass, p is the pressure and $\vec{\delta_i} = [\delta_{i1}, \delta_{i2}]^T$. Eqs. 7 can be rewritten as

$$\frac{\partial \vec{U}}{\partial t} + A_i \frac{\partial \vec{U}}{\partial x_i} = \vec{0} \tag{8}$$

with $A_i = \frac{\partial \vec{f_i}}{\partial \vec{U}}$ denoting the Jacobian matrices.

The Euler equations solver this work is based upon is an in-house time-marching code, based on the finite volume technique. It is fully parallellized, based on the multi-domain technique, using the MPI protocol. It has been also ported to NVIDIA GPUs (using CUDA C), offering significant speedup (about $\times 60$).

A systematic approach to the derivation of the PCE Euler equations requires the following two definitions, which extend the notion of Galerkin projections to vectors and matrices:

Definition 1 *Let $\vec{X} = [X_1, ..., X_m]^T$ be a vector whose components are $X_j = \sum_{i=0}^{\infty} x_{j,i}\Psi_i(\vec{\xi})$ The Galerkin projected vector $\vec{X}$ of order q is defined as*

$$G_q[\vec{X}] = [g_0[\vec{X}], g_1[\vec{X}], ..., g_q[\vec{X}]]^T$$

where $g_i[\vec{X}] = [G_i[X_1], G_i[X_2], ..., G_i[X_m]]^T$.

Definition 2 *Let A be an $m \times m$ matrix whose components A_{ij} are given by $A_{ij} = \sum_{k=0}^{\infty} a_{ij,k}\Psi_k(\vec{\xi})$. The Galerkin projected matrix A of order q is defined as the block matrix*

$$G_q[A] = \begin{bmatrix} B_{0,0} & B_{0,1} & \dots & B_{0,q} \\ B_{1,0} & B_{1,1} & \dots & B_{1,q} \\ \vdots & \vdots & \vdots & \vdots \\ B_{q,0} & B_{q,1} & \dots & B_{q,q} \end{bmatrix}$$

where the elements of each $m \times m$ block are

$$(B_{k_1,k_2})_{ij} = G_{k_1}[A_{ij}\Psi_{k_2}] = \int_{\mathcal{E}} \Psi_{k_1}\Psi_{k_2}A_{ij}w d\vec{\xi} \;\; 1 \leq i, j \leq m$$

One can verify that the following relation

$$G_q[A\vec{x}] = G_q[A]G_q[\vec{x}] \tag{9}$$

holds for an any vector $\vec{x}$ and matrix A. Applying definitions 1 and 2 to eqs. 7, 8 and taking eq. 9 into consideration the PCE Euler equations are derived (subscript q is omitted hereafter)

$$\frac{\partial G[\vec{U}]}{\partial t} + \frac{\partial G[\vec{f_i}]}{\partial x_i} = \frac{\partial G[\vec{U}]}{\partial t} + G[A_i]\frac{\partial G[\vec{U}]}{\partial x_i} = 0$$
$$G[\vec{U}] = [\varrho_0 \; (\varrho\vec{u})_0 \; E_{t_0} \; \varrho_1 \; (\varrho\vec{u})_1 \; E_{t_1} \dots \varrho_q \; (\varrho\vec{u})_q \; E_{t_q}]^T \tag{10}$$

Eqs. 10 resemble their deterministic counterpart in that they are first–order homogeneous, since $\vec{f_i} = A_i\vec{U}$ implies (based on eq. 9) that $G[\vec{f_i}] = G[A_i]G[\vec{U}]$.

3.2 Numerical Solution of the iPCE Equations

For the sake of simplicity, the Galerkin operator will be omitted in what follows; thus, below, A and $\vec{U}$ will denote $G[A]$, and $G[\vec{U}]$ respectively. Upwind schemes can be applied to the iPCE equations, given that the latter are first order–homogeneous and have the same form as the deterministic ones. According to the vertex–centered finite volume method, each control volume Ω_P is formed around the corresponding grid node P, as illustrated in fig. 1 for a hybrid

2D grid (its extension to 3D is evident). By integrating eqs. 10 over Ω_P, in a steady flow analysis, we get

$$\Omega_P \left(\frac{\vec{U}_P^{\kappa+1} - \vec{U}_P^{\kappa}}{\Delta t_P} \right) + \sum_{Q \in nei(P)} \left[\vec{\Phi}_{PQ} \right] \partial\Omega_{PQ} = 0 \tag{11}$$

where κ is the pseudo–time step counter, $\vec{\Phi}_{PQ}$ is the inviscid numerical flux crossing the interface ($\partial\Omega_{PQ}$) between two adjacent finite volumes (pointing from P to Q); $nei(P)$ stands for the set of neighbouring finite volumes of node P. The inviscid fluxes are computed using the

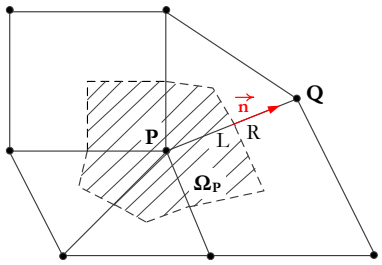

Figure 1: Finite volume Ω_P (hatched area) defined around node P on a hybrid 2D grid comprising triangular and quadrilateral elements. $\vec{n}$ is a vector normal to the part of the interface between P and Q, pointing towards Q, with magnitude equal to the length (2D) or area (3D) of this part. All flow variables associated with Ω_P are collocated at P (in both the deterministic and the iPCE formulations).

flux vector splitting technique [9], applied between P and Q, as follows

$$\vec{\Phi}_{PQ} = A_{PQ}^{-} \vec{U}_{PQ}^{R} + A_{PQ}^{+} \vec{U}_{PQ}^{L} \tag{12}$$

where $A_{PQ} = \frac{\partial(\vec{f}\vec{n})}{\partial \vec{U}} = A_{PQ}^{+} + A_{PQ}^{-}$ and A_{PQ}^{+}, A_{PQ}^{-} are defined using the positive and negative eigenvalues of the Jacobian matrix. For second–order spatial accuracy, $\vec{U}_{PQ}^{L}$ and $\vec{U}_{PQ}^{R}$ (where L and R denote the two states on both sides of the interface between Ω_P and Ω_Q) are computed from $\vec{U}_P$, $\vec{U}_Q$, $\nabla\vec{U}_P$ and $\nabla\vec{U}_Q$ as follows

$$\vec{U}_{PQ}^{L} = \vec{U}_P + \frac{1}{2} (\vec{PQ}) \cdot \vec{\nabla U}_P, \qquad \vec{U}_{PQ}^{R} = \vec{U}_Q - \frac{1}{2} (\vec{PQ}) \cdot \vec{\nabla U}_Q$$

The so–computed fluxes are limited using the van Leer-van Albada limiting function [6]. Spatial gradients are computed using the Green–Gauss integration formula.

The discretized eqs. 11 are solved at each pseudo–time step using the point–implicit Jacobi which is written as

$$\begin{aligned} D_P^{\kappa} \Delta\vec{U}_P^{\kappa+1,\nu} + \sum_{Q \in nei(P)} Z_Q^{\kappa} \Delta\vec{U}_Q^{\kappa+1,\nu} &= -\vec{R}_P^{\kappa,\nu} \\ \vec{U}_P^{\kappa+1} &= \Delta\vec{U}_P^{\kappa+1} + \vec{U}_P^{\kappa} \end{aligned} \tag{13}$$

where κ is the pseudo–time counter, ν the Jacobi internal iteration counter, D_P, Z_Q stand for the diagonal and non–diagonal matrices respectively and $\vec{R}_P^{\kappa,\nu}$ is the residual array. Each Jacobi iteration comprises one iteration to solve the equations corresponding to one of the statistical moments by freezing the other terms.

3.3 Eigen–Decomposition

The implementation of the FVS scheme requires the solution of the corresponding eigenproblem. Starting point is the characteristic equations $\Lambda n_i \frac{\partial \vec{W}}{\partial x_i} = \vec{0}$ where $\vec{W} = [w^A\ w^B\ \ w^C\ w^D\ w^E]^T$ are the characteristic variables, $\Lambda = diag(u^{(n)}, u^{(n)}, u^{(n)}, u^{(n)}+c, u^{(n)}-c)$, $u^{(n)} = u_i n_i$ and c is the speed of sound. By applying the iPCE followed by Galerkin projections, we get

$$G[\Lambda] n_i \frac{\partial G[\vec{W}]}{\partial x_i} = \vec{0} \tag{14}$$

Eqs. 14 can also be written as

$$\begin{bmatrix} d_j G_j[\Psi_0\Psi_0] & d_j G_j[\Psi_0\Psi_1] & \dots & d_j G_j[\Psi_0\Psi_q] \\ d_j G_j[\Psi_1\Psi_0] & d_j G_j[\Psi_1\Psi_1] & \dots & d_j G_j[\Psi_1\Psi_q] \\ \vdots & \vdots & \vdots & \vdots \\ d_j G_j[\Psi_q\Psi_0] & d_j G_j[\Psi_q\Psi_1] & \dots & d_j G_j[\Psi_q\Psi_q] \end{bmatrix} \begin{bmatrix} \vec{W}_0 \\ \vec{W}_1 \\ \vdots \\ \vec{W}_q \end{bmatrix} = \vec{0} \tag{15}$$

where

$$d_j = diag(u_j^{(n)}, u_j^{(n)}, u_j^{(n)}, u_j^{(n)} + c_j, u_j^{(n)} - c_j)$$
$$\vec{W}_i = [w_i^A, w_i^B, w_i^C, w_i^D, w_i^E]^T$$

and $u_j^{(n)}$, w_j^A, c_j denote the j–th term of the PCE of $\vec{u}\cdot\vec{n}$, w^A and of c, respectively. Note that $G_k[\Psi_i\Psi_j]$ is equal to $<\Psi_i, \Psi_j, \Psi_k> = \int_{\mathcal{E}} \Psi_i\Psi_j\Psi_k w d\vec{\xi}$.
The spectral components of c_j can be found through the Galerkin projections of the expression of c, written as a function of the conservative variables, as follows

$$c_j = G_j[c] = \int_{\mathcal{E}} \sqrt{\frac{\gamma R}{c_v \varrho_i \Psi_i} \left(E_{t_k}\Psi_k - \frac{(\varrho_i u_i \Psi_i)(\varrho_k u_k \Psi_k)}{2\varrho_l \Psi_l} \right)} w d\vec{\xi} \tag{16}$$

In eq. 16, γ is the specific heat ratio, R the specific gas constant and c_v the specific heat capacity at constant volume. Note that a similar procedure can be used for the PCE of any quantity expressed in terms of the conservative variables. By re–ordering eqs. 15, we get

$$diag[(Z(u^{(n)}), Z(u^{(n)}), Z(u^{(n)}), Z(u^{(n)}+c), Z(u^{(n)}-c)]\hat{w} = \vec{0} \tag{17}$$

where

$$\hat{w} = [w_0^A \dots w_p^A w_0^B \dots w_p^B w_0^C \dots w_p^C w_0^D \dots w_p^D w_0^E \dots w_p^E]^T$$
$$Z(\lambda) = \begin{bmatrix} \lambda_j G_j[\Psi_0\Psi_0] & \lambda_j G_j[\Psi_0\Psi_1] & \dots & \lambda_j G_j[\Psi_0\Psi_q] \\ \lambda_j G_j[\Psi_1\Psi_0] & \lambda_j G_j[\Psi_1\Psi_1] & \dots & \lambda_j G_j[\Psi_1\Psi_q] \\ \vdots & \vdots & \vdots & \vdots \\ \lambda_j G_j[\Psi_q\Psi_0] & \lambda_j G_j[\Psi_q\Psi_1] & \dots & \lambda_j G_j[\Psi_q\Psi_q] \end{bmatrix} \tag{18}$$

Thus, the solution of an eigenproblem corresponding to a $(5\times(q+1))\times(5\times(q+1))$ matrix is now reduced to one corresponding to the $(q+1)\times(q+1)$ matrix Z. The diagonalization of Z yields the desired eigenvalues and eigenvectors of $G[A_i]$.

4 APPLICATIONS

The iPCE method was programmed and applied to the UQ in an external and an internal aerodynamic case. Comparisons with the niPCE method and/or the (much more expensive) Monte–Carlo technique are shown.

4.1 Flow Around an Aircraft Model

The first problem is considered with the inviscid flow around an aircraft model. The study is carried out around half of the aircraft (due to the symmetric flow conditions) and the unstructured CFD mesh consists of about 45K nodes and 256K tetrahedra. The quantity of interest is the lift coefficient of the aircraft, for which the statistical moments must be computed.

Uncertainties are introduced from the stochastically varying infinite flow angle (a_∞) and/or the infinite Mach number (M_∞). Four cases are studied. Results are presented in Table 1, where $N(\mu, \sigma)$ denotes the normal distribution with mean value μ and standard deviation σ, whereas Ψ_i stand for the Hermite polynomials. Also, $\mathcal{U}(a, b)$ denotes the uniform distribution in the interval $[a, b]$ and, in this case, Ψ_i are the Legendre polynomials.

Flow Conditions		iPCE Chaos order $C=1$	niPCE Chaos order $C=1$	iPCE Chaos order $C=2$	niPCE Chaos order $C=2$	MC
$M_\infty = 0.7$	μ_{C_L}	0.119174	0.119107	0.119183	0.119106	0.118813
$a_\infty \sim N(5^o, 0.5^o)$	σ_{C_L}	0.0095783	0.009657	0.0095786	0.00964	0.0095327
$M_\infty \sim N(0.7, 0.02)$	μ_{C_L}	0.119448	0.119312	0.119485	0.119334	–
$a_\infty = 5^o$	σ_{C_L}	0.001792	0.001894	0.001917	0.001940	–
$M_\infty \sim N(0.7, 0.02)$	μ_{C_L}	0.119326	0.119237	0.119402	0.11925	0.119126
$a_\infty \sim N(5^o, 0.5^o)$	σ_{C_L}	0.0093172	0.0098453	0.0095903	0.009852	0.0097067
$M_\infty \sim N(0.7, 0.02)$	μ_{C_L}	0.119285	–	0.119372	–	0.119089
$a_\infty \sim \mathcal{U}(4.5^o, 5.5^o)$	σ_{C_L}	0.010731	–	0.011010	–	0.011145

Table 1: UQ for the flow around an aircraft model. Statistical moments of the lift coefficient values computed using iPCE, niPCE (with $C=1$ and $C=2$) and the Monte–Carlo method with 2000 replicates in each case.

Figure 2 compares the Mach number's mean and standard deviation fields respectively, in the case uncertainty is only due to the Mach number ($a_\infty = 5^o$; second case in Table 1). It can be seen that the iPCE and niPCE results perfectly match each other all over the aircraft surface. One can also notice the increased variance after the supersonic area of the wing surface which, in the case of niPCE, is extended over a greater area along the wing.

4.2 Flow in the CS10 Compressor Cascade

The second case deals with the transonic flow in a compressor cascade. This is practically a 2D airfoil geometry extruded in the spanwise direction and studied herein as a 3D linear cascade. The compressor profile is that of the Standard Configuration 10 (SC10; [8]) which results by superimposing the thickness distribution of a modified NACA 0006 airfoil on a circular–arc camber line. The blade stagger angle is 45^o and the pitch–to–chord ratio is 1.0. The quantity of interest is the static pressure rise (Δp) and uncertainty is introduced by the inlet flow angle (a_1) and the outlet isentropic Mach number ($M_{2,is}$). Results are presented in Table 2.

Results of iPCE and niPCE are compared in figure 3. Once more, we may notice the higher

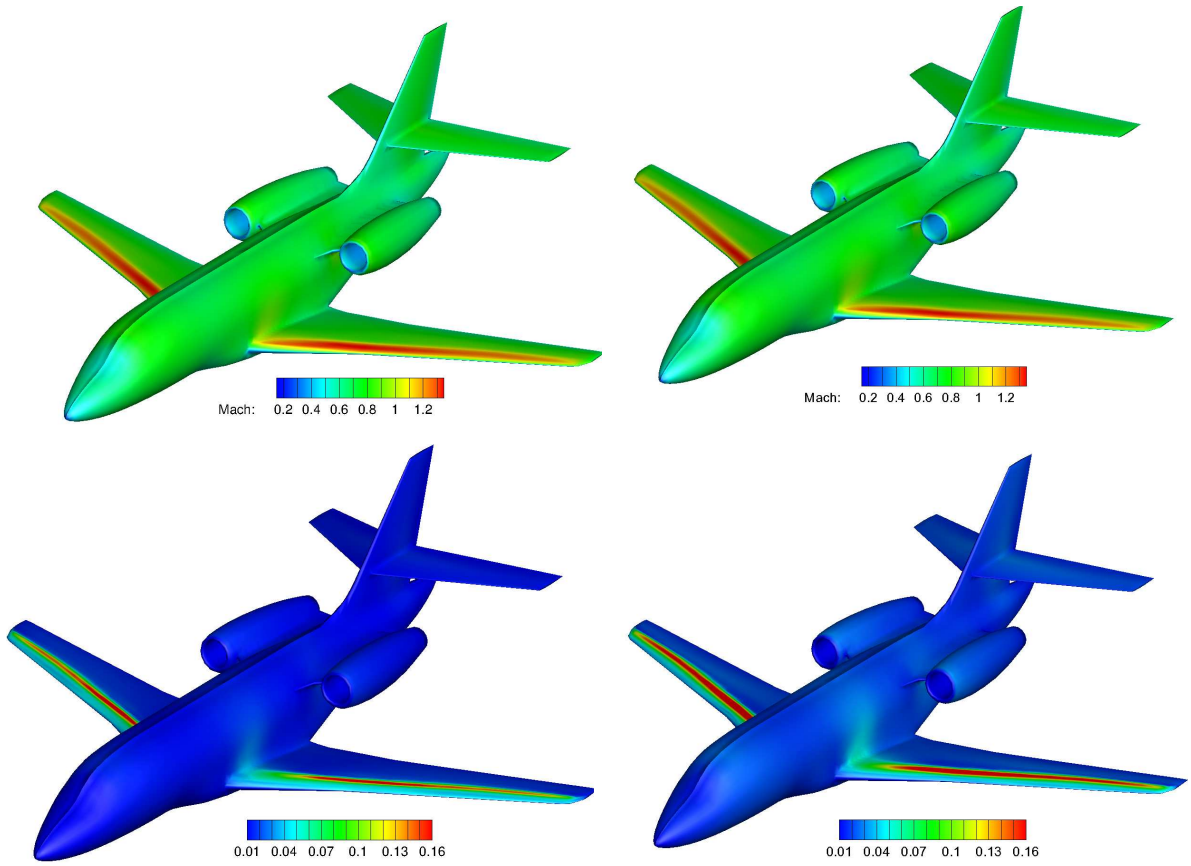

Figure 2: UQ for the flow around an aircraft model ($M_\infty \sim N(0.7, 0.02)$, $a_\infty = 5^o$). Mean Mach number distribution (top) and standard deviation (bottom) on the aircraft surface, computed using the iPCE (left) and niPCE (right), with C=2.

variance of the Mach number field on the blade surface after the shock. The convergence of the spectral continuity equations, for the iPCE with C=2, is plotted in figure 4.

Flow Conditions		iPCE	niPCE	iPCE	niPCE
		Chaos order $C=1$		Chaos order $C=2$	
$M_{2,is} = 0.4425$	$\mu_{\Delta p}$	1.32930	1.33021	1.32944	1.32998
$a_1 \sim N(58^o, 1^o)$	$\sigma_{\Delta p}$	0.02675	0.02519	0.03179	0.02956
$M_{2,is} \sim N(0.4425, 0.005)$	$\mu_{\Delta p}$	1.33754	1.33748	1.33728	1.33786
$a_1 = 58^o$	$\sigma_{\Delta p}$	0.009685	0.009135	0.009786	0.009346

Table 2: CS10 Compressor Cascade. Statistical moments of Δp computed using iPCE and niPCE, with $C=1$ and $C=2$.

5 CONCLUSIONS

This paper presents the development of the intrusive polynomial chaos expansion of the 3D Euler equations through a systematic procedure that ensures the unimpeded numerical solution of the resulting equations. For a single uncertain quantity the iPCE equations ask for about $1.5\times$ more time than the baseline Euler equations which makes it far more efficient than any Monte–Carlo sampling technique and comparable with the niPCE with chaos order equal to one. In any other case, i.e. for more uncertain variables and/or higher PC order, the iPCE vastly

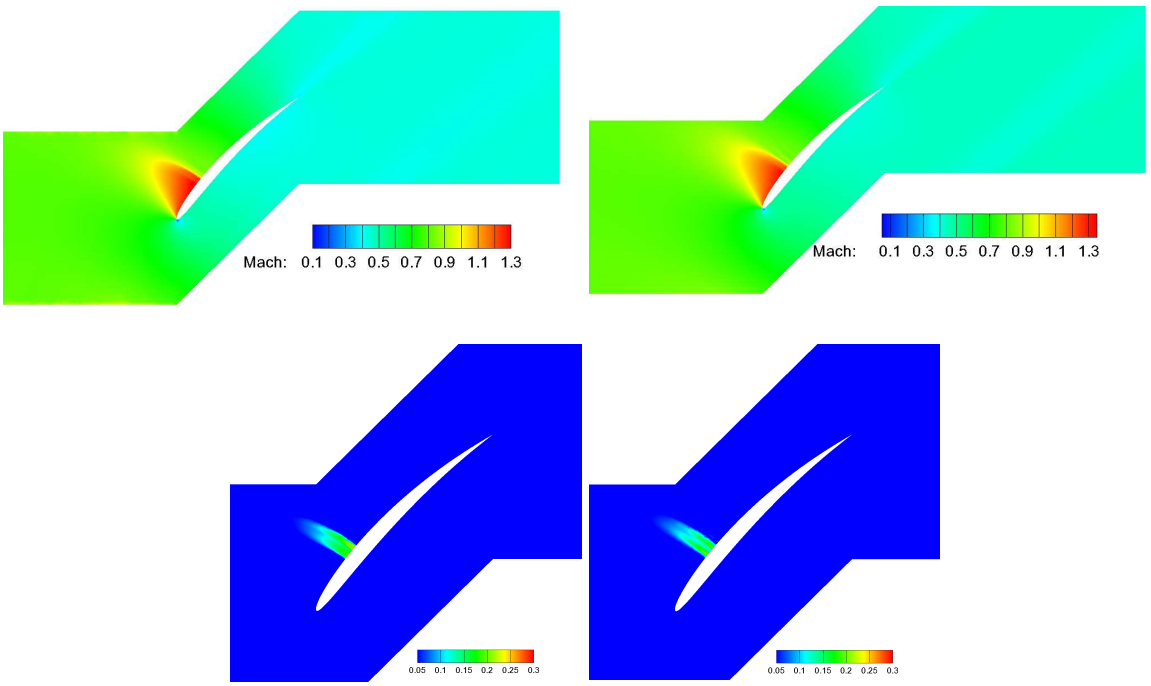

Figure 3: CS10 Compressor Cascade ($M_{2,is} \sim N(0.4425, 0.005)$, $a_1 = 58^o$). Mean (top) and standard deviation (bottom) fields of the Mach number computed using the iPCE (left) and niPCE (right), with C=2.

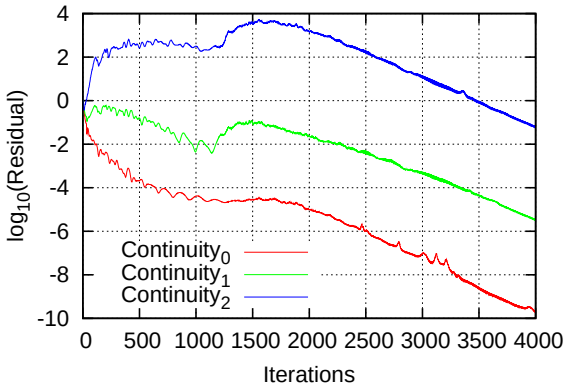

Figure 4: CS10 Compressor Cascade ($M_{2,is} \sim N(0.4425, 0.005)$, $a_1 = 58^o$). Convergence of the residual of the spectral continuity equations (iPCE, C=2).

outperforms the niPCE.

Though, in the cases studied herein, uncertainties are introduced through the flow conditions, the implementation of any other type of uncertainty, such as uncertainties related to the geometry, is straightforward by means of the presented development. Regarding future work, the extension to viscous/turbulent flows along with the corresponding continuous adjoint method, for use in optimization under uncertainties, is foreseen.

REFERENCES

[1] N. Wiener, The Homogeneous Chaos. *American Journal of Mathematics*, **60**, 897-936, 1938.

[2] D. Xiu, G.E. Karniadakis, Modeling Uncertainty in Flow Simulations via Generalized Polynomial Chaos. *Journal of Computational Physics*, **187**, 37-167, 2003.

[3] S. Hosder, R. W. Walters, Non-Intrusive Polynomial Chaos Methods for Uncertainty Quantification in Fluid Dynamics, AIAA 2010-129, *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, Orlando, FL, 4-7 January 2010.

[4] C. Dinescu, S. Smirnov, C. Hirsch, C. Lacor, Assessment of Intrusive and Non-Intrusive Non-Deterministic CFD Methodologies Based on Polynomial Chaos Expansions. *International Journal of Engineering Systems Modeling and Simulations*, **2**, 87-98, 2010.

[5] S. Smirnov, C. Lacor, Non-Deterministic Compressible Navier-Stokes Simulations Using Polynomial Chaos. *5th European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS 2008)*, Venice, Italy, June 30 - July 5, 2008.

[6] G.D. van Albada, B. van Leer, W.W. Roberts, A Comparative Study of Computational Methods in Cosmic Gas Dynamics. *Astronomy and Astrophysics*, **108**, 76-84, 1982.

[7] G. Onorato, G.J.A. Loeven, G. Ghorbaniasl, H. Bijl, C. Lacor, Comparison of Intrusive and non-intrusive Polynomial Chaos Methods for CFD Applications in Aeronautics. *5th European Conference on Computational Fluid Dynamics, (ECCOMAS 2010)*, Lisbon, Portugal, 14-17 June 2010.

[8] Standard Configurations for Unsteady Flow Through Vibrating Axial-Flow Turbomachine-Cascades (STCF) *http://www.energy.kth.se/proj/projects/Markus Joecker/STCF/STCF1to10/stcf10.asp*

[9] B. van Leer, Flux Vector Splitting for the Euler Equations. *8th International Conference on Numerical Methods in Fluid Dynamics*, Aachen, Germany, June 28 - July 2, 1982.

[10] V.G. Asouti, X.S. Trompoukis, I.C. Kampolis, K.C. Giannakoglou, Unsteady CFD Computations Using Vertex-Centered Finite Volumes for Unstructured Grids on Graphics Processing Units. *International Journal for Numerical Methods in Fluids*, **67**, 232-246, 2011.