

# Nonlinear Robust Performance Analysis using Gradient-Based Optimisation - An Aeroelastic Case-Study

Jongrae Kim, Declan G. Bates, Ian Postlethwaite

**Abstract**— This paper considers the problem of computing lower bounds on the worst-case performance of nonlinear systems using gradient-based optimisation. Four different gradient-based local optimisation methods are applied to a robust performance analysis problem for a nonlinear aeroelastic system. The first method formulates the optimisation problem in the classical Euler-Lagrange setting and computes the gradient by backward integration of the resulting adjoint system. The second method also uses the Euler-Lagrange formulation, but uses complex perturbations to calculate the gradient. The third method employs Sequential Quadratic Programming, while the fourth method considered is Simultaneous Perturbation Stochastic Approximation (SPSA), which uses a stochastic approach to decide the search direction. The performance of all four methods is evaluated in terms of computational complexity, numerical accuracy, and ease of implementation, and compared with a standard industrial approach based on a gridding of the uncertain parameter space.

## I. INTRODUCTION

The robustness analysis of linear time invariant (LTI) systems subject to parametric and/or nonparametric uncertainty is now a relatively mature subject, and many powerful tools are available which can readily be applied to practical problems - see for example [1], [2] and [3] for an overview of recent aerospace applications and case-studies. The robustness analysis of nonlinear systems is, of course, a much more difficult problem, and, so far, most developments in this area have been at a theoretical level. Methods for computing upper bounds on robust stability or performance generally rely on generalisations of Lyapunov or Small-Gain theories, and are often computationally intractable and prone to conservatism. Due to the inherent non-convexity of the problem, relatively few methods are available with which to compute lower bounds on either robust stability or performance - the standard approach currently employed by the aerospace industry, for example, is exhaustive off-line simulation using either stochastic (Monte-Carlo) or deterministic (gridding the uncertain parameter space) approaches, [4].

In [5], a promising new approach to nonlinear robust performance analysis was presented, which formulated the problem in the classical Euler-Lagrange optimisation setting [6]. This algorithm was applied successfully to the robust performance analysis of a ducted fan experimental test rig in [7] and an F-16 autopilot simulation in [8]. In [9], the algorithm was modified to improve its convergence properties, by using *Armijo's Rule*, [10], instead of a simple steepest ascent method, to update the estimate of the worst-case uncertain parameters at each iteration. In this paper, we consider another modification to the algorithm, i.e. we

use the recently introduced complex perturbation approach [11], to numerically approximate the gradient (in the standard Euler-Lagrange framework, gradient information is calculated by backward integration of an adjoint system, [7]). Two other gradient-based optimisation methods are also considered in this study - Sequential Quadratic Programming (SQP), [12] [13], and Simultaneous Perturbation Stochastic Approximation (SPSA) [14].

In [4], the potential usefulness of optimisation methods for the robustness analysis of complex nonlinear industrial systems was highlighted. Many industrial contributors to [4], however, felt that more computational experience was required in applying optimisation methods to realistic problems. The main purpose of this study (and the main contribution of this paper) is to add to that computational experience, and to gain some insight into the trade-offs between reliability, computational complexity and ease of implementation among the different optimisation algorithms, when applied to a realistic benchmark robustness analysis problem. In addition, the performance of the optimisation-based approaches described above is compared with a standard industrial approach based on a gridding of the uncertain parameter space. Various advantages and disadvantages of the different optimisation-based approaches are revealed in the study - in general, however, their superiority (both in terms of reliability and efficiency) to the current industrial state-of-the-art is clearly demonstrated.

The paper is organised as follows: In Section II, the aeroelastic system considered in the paper is described. In Section III, the robust performance analysis problem for this system is formulated as an optimisation problem. Section IV briefly describes the various methods used to solve this problem. The results of the application of each method are presented and evaluated in Section V. Section VI contains some conclusions.

## II. A NONLINEAR AEROELASTIC SYSTEM

The aeroelastic system considered in this study consists of a nonlinear model of a NACA 0012 airfoil with two degrees of freedom, i.e., pitch angle,  $\alpha$ , and plunge displacement,  $h$ . The equations of motion for the system are given by

$$\begin{bmatrix} m_T & m_W x_\alpha b \\ m_W x_\alpha b & I_\alpha \end{bmatrix} \begin{Bmatrix} \ddot{h} \\ \ddot{\alpha} \end{Bmatrix} + \begin{bmatrix} c_h & 0 \\ 0 & c_\alpha \end{bmatrix} \begin{Bmatrix} \dot{h} \\ \dot{\alpha} \end{Bmatrix} + \begin{bmatrix} k_h & 0 \\ 0 & k_\alpha(\alpha) \end{bmatrix} \begin{Bmatrix} h \\ \alpha \end{Bmatrix} = \begin{Bmatrix} -L \\ M \end{Bmatrix} \quad (1)$$

where  $m_T$  is the total mass,  $m_W$  is the mass of the wing only, and  $I_\alpha$  is the moment of inertia about the elastic axis. The terms  $a$  and  $x_\alpha$  represent the non-dimensionalised elastic axis and center of mass locations by the length of midchord,  $b$ , respectively. The location of the elastic axis,

This work was supported by EPSRC research grant GR/S61874/01  
J. Kim, D. G. Bates and I. Postlethwaite are with the Department of Engineering, University of Leicester, Leicester, LE1 7RH, United Kingdom  
jrk7, dgb3, ixp@le.ac.uk

TABLE I  
UNCERTAIN PARAMETERS

Nominal Value	Uncertainty	Unit
$\bar{U} = 16.0$	$-2.00 \leq \Delta U \leq 2.00$	[m/sec]
$\bar{a} = -0.6$	$-0.15 \leq \Delta a \leq 0.15$	[-]
$\bar{k}_{\alpha_0} = 6.833$	$-0.68 \leq \Delta k_{\alpha_0} \leq 0.68$	[N·m/rad]
$\bar{k}_{\alpha_1} = 0.0$	$9.00 \leq \Delta k_{\alpha_1} \leq 11.00$	[N·m/rad <sup>2</sup> ]
$\bar{k}_{\alpha_2} = 0.0$	$600.92 \leq \Delta k_{\alpha_2} \leq 734.45$	[N·m/rad <sup>3</sup> ]
$\bar{k}_{\alpha_3} = 0.0$	$23.91 \leq \Delta k_{\alpha_3} \leq 29.23$	[N·m/rad <sup>4</sup> ]
$\bar{k}_{\alpha_4} = 0.0$	$-4579.14 \leq \Delta k_{\alpha_4} \leq -5596.72$	[N·m/rad <sup>5</sup> ]
$\bar{m}_T = 12.387$	$-0.1239 \leq \Delta m_T \leq 0.1239$	[kg]
$\bar{m}_W = 2.0490$	$-0.02049 \leq \Delta m_W \leq 0.02049$	[kg]
$\bar{b} = 0.135$	$-0.00135 \leq \Delta b \leq 0.00135$	[m]
$\bar{\rho} = 1.225$	$-0.01225 \leq \Delta \rho \leq 0.01225$	[kg/m <sup>3</sup> ]
$\bar{c}_{\alpha} = 0.036$	$-0.0018 \leq \Delta c_{\alpha} \leq 0.0018$	[kg·m <sup>2</sup> /sec]
$\bar{c}_{l\alpha} = 6.28$	$-0.0628 \leq \Delta c_{l\alpha} \leq 0.0628$	[-]
$\bar{c}_{l\beta} = 3.358$	$-0.03358 \leq \Delta c_{l\beta} \leq 0.03358$	[-]
$\bar{c}_{m\beta} = -1.94$	$-0.195 \leq \Delta c_{m\beta} \leq 0.195$	[-]
$\bar{c}_h = 27.43$	$-1.3715 \leq \Delta c_h \leq 1.3715$	[kg/sec]
$\bar{k}_h = 2844.4$	$-284.44 \leq \Delta k_h \leq 284.44$	[N/m]

TABLE II  
DEPENDENT PARAMETERS

Parameter	Value	Unit
$c_{m\alpha}$	$(0.5 + a) c_{l\alpha}$	[-]
$x_{\alpha}$	$[0.0873 - (b + a \cdot b)] / b$	[m]
$I_{\alpha}$	$m_W x_{\alpha}^2 b^2 + 0.0517$	[kg·m <sup>2</sup> ]

$a$ , has a significant role in determining the overall stability of the system, however, its exact location is very difficult to determine accurately. To reflect this fact, it is represented in the model as

$$a = \bar{a} + \Delta a \quad (2)$$

where  $\bar{a}$  is a nominal value and  $\Delta a$  is the predicted level of uncertainty. The terms  $c_h$  and  $c_{\alpha}$  are the plunge and pitch structural damping coefficients, and the structural stiffness for the plunge and pitch motions is given by  $k_h$  and  $k_{\alpha}$ , respectively. The term  $k_{\alpha}(\alpha)$  is a nonlinear function of  $\alpha$ , given by [15]:

$$k_{\alpha}(\alpha) = \sum_{i=0}^{\infty} k_{\alpha_i} \alpha^i \quad [\text{N} \cdot \text{m} / \text{rad}] \quad (3)$$

where the  $k_{\alpha_i}$ 's are constants. For numerical simulation purposes, the following 4<sup>th</sup>-order approximation is used for  $k_{\alpha}(\alpha)$  [15]:

$$k_{\alpha}(\alpha) = k_{\alpha_0} + k_{\alpha_1} \alpha + k_{\alpha_2} \alpha^2 + k_{\alpha_3} \alpha^3 + k_{\alpha_4} \alpha^4 \quad (4)$$

where each of the coefficients is given by

$$k_{\alpha_i} = \bar{k}_{\alpha_i} + \Delta k_{\alpha_i} \quad (5)$$

for  $i = 1, 2, \dots, 4$  and  $\Delta k_{\alpha_i}$  represents a bounded level of uncertainty for each coefficient. As shown in [15], the above approximation closely matches experimental results for deviations in  $\alpha$  up to  $\pm 11.49^\circ$ . In addition, the following quasi-steady aerodynamic model for the lift,  $L$ , and the moment,  $M$  are used [15]:

$$L = \rho U^2 b (c_{l\alpha} f_{\alpha} + c_{l\beta} \beta) \quad (6a)$$

$$M = \rho U^2 b^2 (c_{m\alpha} f_{\alpha} + c_{m\beta} \beta) \quad (6b)$$

where  $\rho$  is air density,  $U$  is the freestream velocity,  $c_{l\alpha}$  and  $c_{m\alpha}$  are the aerodynamic lift and moment coefficients, respectively,  $\beta$  is the flap deflection, and

$$f_{\alpha} = \left[ \alpha + \frac{\dot{h}}{U} + \left( \frac{1}{2} - a \right) \frac{b \dot{\alpha}}{U} \right] \quad (7)$$

See [15] for the definitions of the other terms.

There are seventeen uncertain parameters in the model, and the nominal value and the uncertainty bound for each of these parameters are given in Table I. Three other parameters in the model, which are functions of the uncertain parameters, are given in Table II. In [15], a control law to regulate pitch angle by adjusting the flap deflection angle  $\beta$  was designed for the nominal system. Using feedback linearisation, the desired control input for the flap deflection angle is computed. In [15] an adaptive control law is also designed to estimate the nonlinear torsional spring constants,  $k_{\alpha_i}$ . However, only the fixed gain part of the controller is used in the present study. For this control law, the resulting zero dynamics are Hurwitz stable in the range of  $-1 \leq a \leq 1$  and  $0 < U \leq 30$  [m/sec], [15]. Due to physical limitations, the actual flap deflection is restricted to  $\pm 12^\circ$ .

### III. ROBUST PERFORMANCE ANALYSIS PROBLEM

Following [5], we formulate the robust performance analysis problem using the following finite  $L^2$  gain cost function:

$$\max_{x_{\delta} \in \Delta_{\delta}} J = \frac{1}{2} \int_{t_0}^{t_f} g^T(x) g(x) dt \quad (8)$$

where  $t_0$  and  $t_f$  are fixed,  $g(x)$  is a piecewise continuous function, the cost function is subject to

$$\dot{x} = f(x, x_{\delta}) \quad (9a)$$

$$\dot{x}_{\delta} = 0 \quad (9b)$$

where the initial condition is given by  $x_0 = x(t_0)$ ,  $\Delta_{\delta}$  is a hyperbox in  $\mathcal{R}^p$ , and  $p$  is the number of uncertain parameters. Bounds for the values of the uncertain parameters  $x_{\delta_i}$  are given by

$$\underline{x}_{\delta_i} \leq x_{\delta_i} \leq \bar{x}_{\delta_i} \quad (10)$$

for  $i = 1, 2, \dots, p$ , where  $\underline{x}_{\delta_i}$  and  $\bar{x}_{\delta_i}$  are constants. Hence, the problem is to find the optimal initial condition  $x_{\delta}$  to maximize the cost function.

For the aeroelastic system given in (1), the vector of uncertain parameters is given by

$$x_{\delta} = [\Delta U, \Delta a, \Delta k_{\alpha_0}, \Delta k_{\alpha_1}, \Delta k_{\alpha_2}, \Delta k_{\alpha_3}, \Delta k_{\alpha_4}, \Delta m_T, \Delta m_W, \Delta b, \Delta \rho, \Delta c_{\alpha}, \Delta c_{l\alpha}, \Delta c_{l\beta}, \Delta c_{m\beta}, \Delta c_h, \Delta k_h]^T \quad (11)$$

where the bound for each parameter is given in Table I. To avoid numerical problems, each uncertain parameter is normalised so that it is bounded as follows:  $-1 \leq x_{\delta_i} \leq 1$ , for  $i = 1, 2, \dots, 17$ .

Since the main control objective in the aeroelastic system is regulation of the pitch angle, an appropriate cost function for robust performance analysis is

$$\max_{x_{\delta} \in \Delta_{\delta}} J = \frac{1}{2} \int_{t_0}^{t_f} Q_1 \alpha^2(t) + Q_2 \dot{\alpha}^2(t) dt \quad (12)$$

where  $Q_1$  and  $Q_2$  are positive scaling factors, given by 10 and 1, respectively, which are used to (approximately) relatively equally penalise the values of  $\alpha$  and  $\dot{\alpha}$ , and  $t_0$  and  $t_f$  are set equal to 0 and 5 sec, respectively. The initial condition for (1) is given by  $\alpha(t_0) = 0.0483$  [rad],  $\dot{\alpha}(t_0) = 3.1819$  [rad/sec],  $h(t_0) = 0.0135$  [m],  $\dot{h}(t_0) = 0.2485$  [m/sec].

In the following, the different approaches used to solve the above robust performance analysis problem are briefly described.

#### A. Gridding The Uncertain Parameter Space

For the purposes of comparison with standard methods currently employed by industry, the worst-case value of the cost function in (12) was evaluated for all possible combinations of the extreme points of the uncertain parameters. This required 131072 ( $2^{17}$ ) cost function evaluations. The maximum value of the cost function found was 15.888 and the corresponding worst-case combination of uncertain parameters is given by

$$x_\delta = [-1, -1, -1, +1, -1, +1, +1, -1, +1, -1, +1, -1, +1, -1, +1]^T \quad (13)$$

These results can be considered as the current industrial benchmark for this type of problem [4]. Note that the exponential increase in computation time places severe limits on the number of uncertain parameters that can be considered under this approach. In addition, since only the vertices of the uncertain parameter space are checked, worst-cases that occur in the interior of the parameter space are guaranteed to be missed a priori.

#### B. Euler-Lagrange Optimisation Framework

In the classical Euler-Lagrange framework, the augmented cost is given by

$$J = \int_{t_0}^{t_f} g^T g + \lambda^T [f - \dot{x}] + \lambda_\delta^T [0 - \dot{x}_\delta] dt \quad (14)$$

where  $\lambda$  and  $\lambda_\delta$  are the Lagrange multipliers. Taking the first variation of this cost,  $\delta J$ , gives the following adjoint system (the gradient is a row vector in this paper):

$$\dot{\lambda} = - \left( \frac{\partial f}{\partial x} \right)^T \lambda - \left( \frac{\partial g}{\partial x_2} \right)^T y \quad (15a)$$

$$\dot{\lambda}_\delta = - \left( \frac{\partial f}{\partial x_\delta} \right)^T \lambda \quad (15b)$$

with boundary conditions:

$$\lambda(t_f) = 0 \quad (16a)$$

$$\lambda_\delta(t_f) = 0 \quad (16b)$$

Thus,  $\delta J$  becomes:

$$\delta J = \lambda_\delta^T(t_0) \delta x_\delta \quad (17)$$

The initial value of  $\lambda_\delta$  can, therefore, be interpreted as the gradient of the cost function with respect to the uncertain parameters, i.e.,

$$\lambda_\delta^T(t_0) = \left. \frac{\partial J}{\partial x_\delta} \right|_{t=t_0} \quad (18)$$

$\lambda_\delta(t_0)$  can be obtained at each numerical iteration by backward integration of the adjoint system, (15), with the final condition,  $\lambda_\delta(t_f) = 0$ . Considering the Lagrange multiplier term by term

$$\lambda_{\delta_i}(t_0) = \left. \frac{\partial J}{\partial x_{\delta_i}} \right|_{t=t_0} \quad (19)$$

for  $i = 1, 2, \dots, p$  there are three possible values for the initial condition:

$$x_{\delta_i}^* = \begin{cases} x_{\delta_i} = \bar{x}_{\delta_i} \text{ and } \lambda_{\delta_i}(t_0) > 0 \\ x_{\delta_i} = \underline{x}_{\delta_i} \text{ and } \lambda_{\delta_i}(t_0) < 0 \\ \underline{x}_{\delta_i} \leq x_{\delta_i} \leq \bar{x}_{\delta_i} \text{ where } \lambda_{\delta_i} = 0 \end{cases} \quad (20)$$

More details of the Euler-Lagrange framework can be found in [6]. The original update law used in [5], [7], and [8] is given by

$$x_{\delta_i}^{\text{uncnst}} = x_{\delta_i}^{\text{current}} + \lambda_{\delta_i}(t_0) \quad (21)$$

for  $i = 1, 2, \dots, p$ , so that

$$x_{\delta_i}^{\text{updated}} = \begin{cases} \underline{x}_{\delta_i}, & \text{for } x_{\delta_i}^{\text{uncnst}} < \underline{x}_{\delta_i} \\ \bar{x}_{\delta_i}, & \text{for } x_{\delta_i}^{\text{uncnst}} > \bar{x}_{\delta_i} \\ x_{\delta_i}^{\text{uncnst}}, & \text{otherwise} \end{cases} \quad (22)$$

for  $i = 1, 2, \dots, p$ . Since the update law, (21), is an unscaled steepest ascent direction, it may jump all over the uncertain space when the gradient is too large and it may converge very slowly when the gradient is too small. Indeed, very slow convergence of the original algorithm was reported in [16], where the algorithm was applied to a flight control law robustness analysis problem for a Cessna Citation aircraft. To avoid this problem, the  $x_{\delta_i}^{\text{uncnst}}$  can be updated in the following way:

$$x_{\delta_i}^{\text{uncnst}} = x_{\delta_i}^{\text{current}} + \gamma^k \lambda_{\delta_i}^k(t_0) \quad (23)$$

where  $k$  is the iteration step, and  $\gamma^k$  is the step-size to be determined. The step size  $\gamma^k$  can be chosen using the *Maximisation Rule* (In minimisation problems this is known as the *Minimisation rule*.) or *Armijo's Rule*. In this paper Armijo's rule is adopted -  $\sigma$  and  $\zeta$  are first chosen in the range of (0, 1) and then  $\gamma^k$  is set to

$$\gamma^k = \zeta^m \gamma_{\max} \quad (24)$$

where  $m$  is the smallest integer such that

$$J [x_{\delta_i}^{\text{current}} + \gamma^k \lambda_{\delta_i}^k(t_0)] - J (x_{\delta_i}^{\text{current}}) \geq \sigma \gamma^k \|\lambda_{\delta_i}^k(t_0)\|_2 \quad (25)$$

The design parameters are given by  $\sigma = 10^{-10}$ ,  $\zeta = 1/2$ , and  $\gamma_{\max} = 1$ . More details of this scheme can be found in [10]. Finally, the stop condition is given by [12]

$$\left| J (x_{\delta_i}^{\text{updated}}) - J (x_{\delta_i}^{\text{current}}) \right| \leq \epsilon \quad (26)$$

where  $\epsilon$  is equal to  $10^{-6}$ .

### C. Gradient Approximation using Complex Perturbations

Finite difference methods for gradient approximation usually try to reduce the step size as much as possible in order to minimise the round off error. Below a certain (problem dependent) value of the step size, however, subtraction cancellation error will make the estimate of the gradient equal to zero. As pointed out in [17], this problem can be overcome if the cost function is perturbed in a complex direction as follows:

$$\left. \frac{\partial J(x_\delta)}{\partial x_{\delta_i}} \right|_{x_\delta=x_\delta^*} \approx \frac{\text{Im}[J(x_\delta + jh_i)]}{h} \quad (27)$$

for  $i = 1, 2, \dots, p$ , where  $\text{Im}(\cdot)$  is the imaginary part of the argument,  $j$  is the imaginary number  $\sqrt{-1}$ ,  $h_i$  is a vector of the same dimension as  $x_\delta$  whose elements are all zero except the  $i$ -th element which is  $h$ , and  $h$  is a real number. The approximation error is  $O(h^2)$ , which is the same as for the central difference method. Note, however, that since for this formulation there is no subtraction cancellation error,  $h$  can now be made arbitrarily small (as long as it remains inside the numerical range for real numbers of the computer). In this paper,  $h$  is set to  $10^{-50}$  and the expected numerical error is then only  $O(10^{-100})$ . The gradient of  $J(x_\delta)$  can be approximated by applying the above operation  $p$ -times for each  $i$ .

In [11] the above complex perturbation method is applied to approximate the gradient of a cost function  $J(x_\delta)$  for a dynamical system. By substituting  $x_\delta$  in (9a) by  $x_\delta + jh_i$ , (9a) becomes  $\dot{x} = f(x, x_\delta + jh_i)$ . Then, the following two sets of differential equation are obtained:

$$\dot{x}_R(t) = \text{Re}[f(x, x_\delta + jh_i)] \quad (28a)$$

$$\dot{x}_I(t) = \text{Im}[f(x, x_\delta + jh_i)] \quad (28b)$$

where  $\text{Re}(\cdot)$  is the real part of the argument. Note that the above operation, i.e., calculating the real and the imaginary parts of  $f(x, x_\delta + jh_i)$ , is easy in most cases. Finally, the solution,  $x(t) = x_R(t) + jx_I(t)$  is obtained by solving the differential equation, (28), with the initial condition,  $x_R(t_0) = x(t_0)$  and  $x_I(t_0) = 0$ . After obtaining the solution,  $x(t)$ , it is substituted into the cost function, (12). Taking the imaginary part, the approximate differential can then be calculated using (27). By repeating this step for each of the  $x_{\delta_i}$ , the approximate gradient is obtained. The complex perturbation approach can be used to calculate the gradient in any gradient-based optimisation algorithm - in this paper we used it as an alternative method to calculate the gradient in the Euler-Lagrange algorithm described in the previous section.

### D. Sequential Quadratic Programming

Sequential Quadratic Programming (SQP) is a powerful nonlinear programming method which has been applied to a wide variety of problems. In each iteration of SQP, the nonlinear optimisation problem is approximated as a Quadratic Programming (QP) problem and the QP is solved using the Lagrange method, active set method, etc [12] [13].

The QP problem corresponding to (8) is given by

$$\min_{s^k \in \mathbb{R}^p} \frac{1}{2} s^{kT} H^k s^k + \left. \frac{\partial J}{\partial x_\delta} \right|_{x_\delta=x_\delta^k} s^k \quad (29)$$

where  $H^k$  is the Hessian to be approximated and the gradient of  $J(x_\delta)$  is computed using a finite-difference approximation. Then, the solution for  $s^k$  is determined using some QP algorithm. Since the search direction is determined in the above equation, the  $x_\delta$  is updated as follows:

$$x_{\delta_i}^{\text{uncnst}} = x_{\delta_i}^{\text{current}} + \gamma^k s^k \quad (30)$$

where  $\gamma^k$  is determined by some maximisation rule. In this paper, the SQP implementation in “fmincon” of the MATLAB optimisation toolbox is used. More details about the algorithm can be found in [18].

### E. Simultaneous Perturbation Stochastic Approximation

Simultaneous Perturbation Stochastic Approximation (SPSA) is a stochastic optimisation method. Unlike with algorithms that use finite difference based gradient approximation, the computational cost to approximate the gradient does not increase with the dimension of the parameter space. The stochastic approximation for the gradient is given by [14]

$$g^k = \frac{J(x_\delta^{\text{current}} + c_k \Delta_k) - J(x_\delta^{\text{current}})}{2c_k} \Delta_k^{-1} \quad (31)$$

where

$$\Delta_k = [\Delta_{k1}, \Delta_{k2}, \dots, \Delta_{kp}]^T \quad (32a)$$

$$\Delta_k^{-1} = [\Delta_{k1}^{-1}, \Delta_{k2}^{-1}, \dots, \Delta_{kp}^{-1}]^T \quad (32b)$$

and  $c_k$  is equal to  $c/(k+1)^{m_1}$ . The distribution of each  $\Delta_{ki}$  for  $i = 1, 2, \dots, p$  is the Bernoulli  $\pm 1$  distribution with a probability of 1/2 for each outcome. Several other possible distribution functions are given in [14]. The update law is

$$x_{\delta_i}^{\text{uncnst}} = x_{\delta_i}^{\text{current}} + \gamma^k g^k \quad (33)$$

where  $\gamma^k$  is equal to  $\gamma/(k+1+A)^{m_2}$ . The optimal values for the parameters in the algorithm were chosen by following the method suggested in [14], and they are given as follows:  $m_1 = 0.101$ ,  $m_2 = 0.602$ ,  $c = 10^{-9}$ ,  $A = 30$ ,  $\gamma = (1+A)^{0.602} \times 10^{-3}$ . A more detailed description of how to choose each design parameter can be found in [14]. Finally, the algorithm stops when (26) is satisfied for 20 consecutive iterations.

## IV. RESULTS AND DISCUSSION

Since the nonlinear optimisation problem considered in this paper is likely to have multiple local optima, each optimisation algorithm was run 100 times, each time starting from a different randomly chosen initial guess for the uncertain parameters. Results for the 100 trials of each method, together with those for the gridding approach using 131072 grid points, are given in Tables III and IV. From Table III, the maximum value of the cost function (i.e. the best estimate of worst-case performance) was computed by the SQP algorithm to be 17.7535, with the corresponding worst-case uncertain parameter combination given as:

$$x_\delta^* = [-0.30, -0.75, -1.0, +1.0, -1.0, +1.0, +1.0, -1.0, +1.0, -1.0, -1.0, +1.0, -1.0, +1.0, -1.0, +1.0] \quad (34)$$

TABLE III  
THE COST FUNCTION VALUES FOR 100 TRIALS

Method	Min	Max	Average	Standard Deviation
Gridding	1.2216	15.8881	3.7590	2.8988
Euler-Lagrange	2.5249	17.6606	16.9590	1.4720
Complex Perturbation	16.8927	17.6971	17.3164	0.1739
SPSA <sup>1</sup>	7.6930	15.8592	12.6520	1.8124
SPSA <sup>2</sup>	1.5476	16.8531	12.6339	3.1783
SQP	2.0214	17.7535	16.2245	4.0672

TABLE IV  
NUMBER OF COST FUNCTION EVALUATIONS FOR 100 TRIALS

Method	Min	Max	Average	Standard Deviation	Total
Gridding	131072	131072	131072	0	131072
Euler-Lagrange	42	454	129.50	62.56	12950
Complex Perturbation	190	1718	586.19	213.95	53619
SPSA <sup>1</sup>	78	2041	380.04	444.38	38004
SPSA <sup>2</sup>	66	22212	382.39	883.88	3786010
SQP	151	2181	630.27	342.21	63027

Since the SPSA is a stochastic approach, to evaluate the performance it runs 100 times for each 100 trials.

<sup>1</sup> the values in this row are in average sense from 100 runs for each trial.

<sup>2</sup> the values in this row are in average sense from the total runs of all trial, i.e. 100×100 runs.

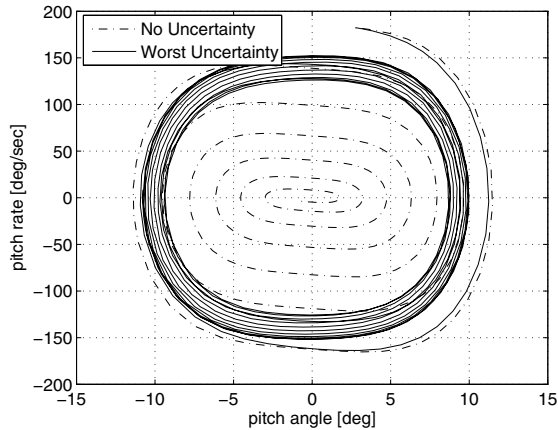


Fig. 1. Pitch angle phase-plane plots for the nominal and the worst-case

Note that  $\Delta U^*$  and  $\Delta a^*$  are not on the boundary of their possible range of variation (their actual values are given by -0.61 m/sec and -0.11, respectively), and thus the true optimal solution can never be found by a gridding approach unless it has a very fine gridding, which is computationally prohibitive in practice. Figure 1 shows the phase-plane plot for the pitch angle for the nominal system (all uncertainties equal to zero) and for the worst-case parameter combination. Whereas for the nominal system the controller quickly regulates the pitch angle to zero, the phase-plane plot for the worst-case uncertainty combination shows the presence of a limit cycle.

As can be seen from the tables, although each optimisation method was able to find an optimal or close to optimal solution, there were significant differences in terms of computational performance between the different approaches. Note that since the most computationally expensive part for this type of problem is the simulation of the model, the computational cost is measured by the

number of function evaluations. For the gridding approach, the number of function evaluations required is significantly larger than for any of the optimisation methods - this is of course inevitable for any problem with a reasonably large number of uncertain parameter, given the exponential growth in computation time for this approach. The SQP method computed the maximum value of the cost function, however, it was computationally less efficient than the other optimisation methods. In addition, the standard deviation of the cost found by SQP is the worst of all the methods, i.e. it seems to be more susceptible to getting trapped in local maxima. The Euler-Lagrange approach is clearly the most efficient algorithm of the five considered - it requires less than a quarter as many function evaluations as the SQP algorithm. In terms of reliability, the average optimal cost computed by the Euler-Lagrange approach is also significantly closer to the true maximum value than that computed by the SQP algorithm and the standard deviation of the cost for 100 trials is the second smallest after the complex perturbation.

Although the above results for the Euler-Lagrange formulation are very promising, there are a number of problematic issues with the practical implementation of this approach. Due to the fact that this approach calculates gradient information through the backward integration of the adjoint system, the quality of the results is particularly sensitive to numerical integration errors. This is the reason why the Euler-Lagrange algorithm didn't find the exact optimal solution - because of inaccuracy in the integration, the gradients point in the wrong direction at the final stage of the optimisation and the resulting convergent points are scattered around the global optimum. The integration of the adjoint system can also take significantly more time than is required for integration of the original system, since all the time histories of the states for the original system have to be stored and interpolated to solve the adjoint system backwards in time. In addition, we note that for complex systems, deriving the adjoint system (15) could be a very difficult process.

TABLE V  
WORST PARAMETERS COMBINATION FROM EACH OPTIMISATION : NORMALISED VALUE

Method	$\Delta U$	$\Delta a$	$\Delta k_{a1}$	$\Delta k_{a2}$	$\Delta \rho$	$\Delta c_{m\beta}$	Distance From SQP
Gridding	-1.0000	-1.0000	1.0000	1.0000	1.0000	1.0000	2.9234
Euler-Lagrange	-0.3711	-0.7749	0.9970	0.4707	-0.6983	-1.0000	0.6134
Complex Perturbation	-0.3525	-0.7570	0.6795	0.3566	-1.0000	-0.9931	0.7205
SPSA	-1.0000	-1.0000	0.6696	0.6696	0.8162	1.0000	3.6063
SQP	-0.3048	-0.7483	1.0000	1.0000	-1.0000	-1.0000	0

The complex perturbation approach found the second highest value of the cost function after that found by SQP. The number of function evaluations is approximately 7% lower than that required by SQP. However, the consistency of convergence for this method is much better than for all other methods - the standard deviation of the cost for 100 trials is around 10% of the values given by the other methods. This result is not surprising because the gradient approximation is extremely accurate with this method. However, since the complex approach doubles the dimension of the differential equations in the model, it increases the computation time for the cost function evaluation at each iteration. This could represent a significant extra computational burden for systems where the dimension of the original differential equations is already very high.

Since SPSA is a stochastic approach, to evaluate the cost for each trial, i.e., each initial  $x_\delta$ , the SPSA algorithm was run 100 times to get some probabilistic performance values. The maximum value of the cost from the SPSA is the worst among all the optimisation methods. The number of function evaluations required on average, however, is quite small. Overall, the performance of SPSA for this problem was not very impressive. Based on the results, it would seem to be more useful for problems where the probability that the maximum can be obtained in a small number of function evaluations is very high. However, the average value of the cost obtained with this method is still much better than that achieved by the gridding approach. Hence, as pointed out in [14], the real power of SPSA will be shown when the number of uncertain parameters is extremely large, for example around several hundreds instead of seventeen. In that case, other methods will suffer from the curse of dimensionality but the SPSA may give some reasonable answers with reduced computational cost.

The worst parameter combination found by each method is shown in Table V. The parameters which are not shown in the table are the same as (34) for all methods. In the table, the distances from the various parameter combinations to the worst-case found by the SQP method are shown. The distances from the Euler-Lagrange and the complex perturbation worst-cases are closer than the others.

## V. CONCLUSIONS

The problem of analysing the worst-case performance of a nonlinear aeroelastic system was formulated as an optimisation problem and solved using gradient-based local optimisation methods. A "brute-force" approach based on a gridding of the uncertain parameter space was also applied. Four different gradient-based local optimisation methods were applied and evaluated. The first method formulates the optimisation problem in the classical Euler-Lagrange setting and computes the gradient by backward

integration of the resulting adjoint system. The second method also uses the Euler-Lagrange formulation, but uses complex perturbations to calculate the gradient. The third method employs Sequential Quadratic Programming, while the fourth method considered is Simultaneous Perturbation Stochastic Approximation (SPSA), which uses a stochastic approach to decide the search direction. The study highlighted the various advantages and disadvantages of each method, in terms of convergence, reliability, computational cost, and implementation difficulty. In general, gradient-based optimisation methods were seen to outperform the current industrial state-of-the-art for this problem.

## ACKNOWLEDGMENTS

This work was carried out under EPSRC research grant GR/S61874/01.

## REFERENCES

- [1] G. Ferreres, *A Practical Approach To Robustness Analysis With Aeronautical Applications*. Kluwer Academic/Plenum Publishers, 1999.
- [2] D. G. Bates and I. Postlethwaite, *Robust Multivariable Control Of Aerospace Systems*. DUP Science, 2002.
- [3] K. A. Wise, "Comparison of six robustness tests evaluating missile autopilot robustness to uncertain aerodynamics," *Journal of Guidance, Control, and Dynamics*, vol. 15, no. 4, pp. 861–870, 1992.
- [4] C. Fielding, A. Varga, S. Bannani, and M. S. (Eds.), *Advanced Techniques for Clearance of Flight Control Laws*. Springer, 2002.
- [5] J. E. Tierno, R. M. Murray, and J. C. Doyle, "An efficient algorithm for performance analysis of nonlinear control systems," in *American Control Conference*, June 21–23, Seattle, WA, USA, 1995.
- [6] A. E. Bryson, Jr., and Y.-C. Ho, *Applied Optimal Control: Optimization, Estimation, and Control*. John Wiley & Sons, 1975.
- [7] J. E. Tierno, "A computational approach to nonlinear system analysis," Ph.D. dissertation, California Institute of Technology, Pasadena, CA, USA, 1996.
- [8] I. M. Gregory and J. E. Tierno, "A new approach to aircraft robust performance analysis," in *AIAA GN&C Conference & Exhibit*, July 29–31, San Diego, CA, USA, AIAA-96-3860, 1996.
- [9] J. Kim, D. G. Bates, and I. Postlethwaite, "Nonlinear robust performance analysis of an aeroelastic system," in *The 16th IFAC World Congress*, July 4–8, Prague, Czech Republic, 2005.
- [10] D. P. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.
- [11] J. Kim, D. G. Bates, and I. Postlethwaite, "Complex-step gradient approximation for robustness analysis of nonlinear systems," in *The 16th IFAC World Congress*, July 4–8, Prague, Czech Republic, 2005.
- [12] R. Fletcher, *Practical Methods Of Optimization : Unconstrained Optimization*. John Wiley & Sons, 1980, vol. 1.
- [13] ———, *Practical Methods Of Optimization : Constrained Optimization*. John Wiley & Sons, 1981, vol. 2.
- [14] J. C. Spall, *Introduction To Stochastic Search And Optimization : Estimation, Simulation, and Control*. John Wiley & Sun, 2003.
- [15] T. W. Strganac, J. Ko, and D. E. Thompson, "Identification and control of limit cycle oscillations in aeroelastic systems," *Journal of Guidance, Control, and Dynamics*, vol. 23, no. 6, pp. 1127–1133, November–December 2000.
- [16] M. Ledegang, "Worst-case tracking performance using the worst-algorithm," Master's thesis, Delft University of Technology, Postbus 5, 2600 AA Delft, The Netherlands, July 1999.
- [17] J. R. R. A. Martins, P. Sturdza, and J. J. Alonso, "The complex-step derivative approximation," *ACM Transactions on Mathematical Software*, vol. 29, no. 3, pp. 245–262, Sept. 2003.
- [18] MathWorks, *Optimization Toolbox (Version 2) For Use With MATLAB*. The MathWorks, Inc., 2003.