Clearance of Nonlinear Flight Control Laws Using Hybrid Evolutionary Optimization

Prathyush P. Menon, Jongrae Kim, Declan G. Bates, and Ian Postlethwaite, *Fellow, IEEE*

***Abstract*—The application of two evolutionary optimization methods, namely, differential evolution and genetic algorithms, to the clearance of nonlinear flight control laws for highly augmented aircraft is described. The algorithms are applied to the problem of evaluating a nonlinear handling quality clearance criterion for a simulation model of a high-performance aircraft with a delta canard configuration and a full-authority flight control law. Hybrid versions of both algorithms, incorporating local gradient-based optimization, are also developed and evaluated. Statistical comparisons of computational cost and global convergence properties reveal the benefits of hybridization for both algorithms. The differential evolution approach in particular, when appropriately augmented with local optimization methods, is shown to have significant potential for improving both the reliability and efficiency of the current industrial flight clearance process.**

***Index Terms*—Evolutionary algorithms, flight control, nonlinear systems, robustness analysis, simulation.**

I. Introduction

MODERN high-performance aircraft are often designed to be naturally unstable due to performance reasons and, therefore, can only be flown by means of a flight control system which provides artificial stability. As the safety of the aircraft is dependent on the controller, it must be proven to the clearance authorities that the controller functions correctly throughout the specified flight envelope in all normal and various failure conditions, and in the presence of all possible parameter variations.

This task is a very lengthy and expensive process, particularly for high-performance aircraft, where many different combinations of flight parameters (e.g., large variations in mass, inertia, centre of gravity positions, highly nonlinear aerodynamics, aerodynamic tolerances, air data system tolerances, structural modes, failure cases, etc.) must be investigated so that guarantees about worst case stability and performance can be made [1].

The aircraft models used for clearance purposes describe the actual aircraft dynamics, but only within given uncertainty bounds. One reason for this is the limited accuracy of the aerodynamic data set determined from theoretical calculations and wind tunnel tests. These parameters can even differ between two aircraft of the same type, due to production tolerances. Moreover, the employed sensor, actuator, and hydraulic models are usually only approximations, where the nonlinear effects are not fully modeled because they are either unknown or would make the model unacceptably complex.

The goal of the clearance process is to demonstrate that a set of selected criteria expressing stability and handling requirements are fulfilled. Typically, criteria covering both linear and nonlinear stability, as well as various handling and performance requirements are used for the purpose of clearance. The clearance criteria can be grouped into four classes: 1) linear stability criteria; 2) aircraft handling/pilot induced oscillation (PIO) criteria; 3) nonlinear stability criteria; and 4) nonlinear handling criteria. This paper focuses on the evaluation of a nonlinear handling criterion, which is described in detail in the next section. Details of the other clearance criteria can be found in [1].

In the clearance process, for each point of the flight envelope, for all possible configurations and for all combinations of parameter variations and uncertainties, violations of the clearance criteria and the worst case result for each criterion must be found. Based on the clearance results, flight restrictions are imposed where necessary. Faced with limited time and resources, the current flight clearance process employed by the European aerospace industry uses a gridding approach, whereby the various clearance criteria are evaluated for all combinations of the extreme points of the aircraft's uncertain parameters [1]. This process is then repeated over a gridding of the aircraft's flight envelope. Clearly, the effort involved in the resulting clearance assessment increases exponentially with the number of uncertain parameters. Another difficulty with this approach is the fact that there is no guarantee that the worst case uncertainty combination has in fact been found, since 1) it is possible that the worst case combination of uncertain parameters does not lie on the extreme points and 2) only a few selected points in the aircraft's flight envelope can be checked. This paper presents a new approach to the clearance problem based on the use of hybrid optimization techniques, which will be shown to have the capability to significantly improve both the reliability and efficiency of the current flight clearance process.

This paper is organized as follows. Section II describes the aircraft simulation model and flight clearance criterion used in this study. Section III considers the use of local gradient-based optimization methods for the problem of flight clearance. In Section IV, the results of applying two evolutionary algorithms to the flight clearance problem are described. Hybrid versions of both algorithms are developed and applied in Section V. Some conclusions are presented in Section VI. Preliminary results from this study were first presented in [2].

II. Aero-Data Model in a Research Environment (ADMIRE)—Aircraft Model

The aircraft model used in the present study is the Aero-Data Model in a Research Environment (ADMIRE) [3], a nonlinear,

Manuscript received March 17, 2005; revised November 2, 2005. This work was supported in part by the Engineering and Physical Sciences Research Council (EPSRC) under Research Grant GR/S61874/01.

The authors are with the Department of Engineering, University of Leicester, Leicester LE1 7RH, U.K. (e-mail: ppm6@le.ac.uk; jrk7@le.ac.uk; dgb3@le.ac.uk; ixp@le.ac.uk).

Digital Object Identifier 10.1109/TEVC.2006.873220

TABLE I
AIRCRAFT MODEL UNCERTAIN PARAMETERS [5]

Parameter	Bound	Description
Δ_{mass}	[-0.1, +0.1]	variation in aircraft mass from nominal one (9100 kg) [%]
Δ_{xcg}	[-0.075, +0.075]	variation in position of center of mass [m]
$\Delta_{C_{m\delta_e}}$	[-0.05, +0.05]	uncertainty in pitching moment due to elevator deflection [1/rad]
$\Delta_{I_{yy}}$	[-0.2, +0.2]	the inertia uncertainty from nominal one (81000 kg·m^2) [%]
$\Delta_{C_{m\alpha}}$	[-0.05, +0.05]	uncertainty in pitching moment due to AoA [1/rad]

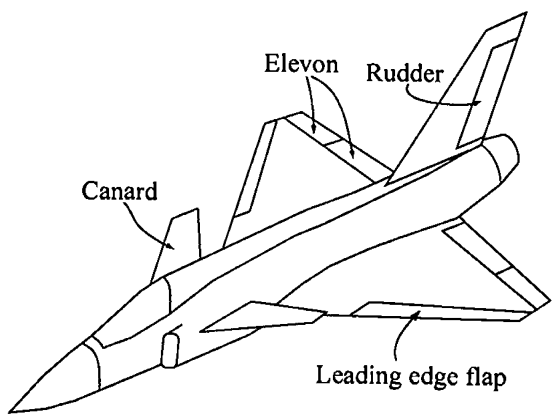

Fig. 1. ADMIRE—Aircraft model and control surfaces (not to scale).

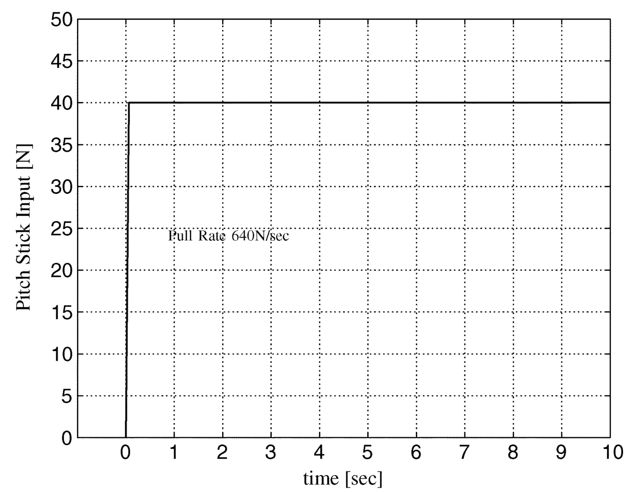

Fig. 2. Pitch stick pull command.

six degree-of-freedom simulation model developed by the Swedish Aeronautical Research Institute (FOI) using aero data obtained from a generic single seated, single engine fighter aircraft with a delta-canard configuration. A (not to scale) schematic of the aircraft is shown in Fig. 1. ADMIRE is augmented with a full-authority flight control system and includes engine dynamics and detailed nonlinear actuator models. The model includes a large number of uncertain aerodynamic, actuator, sensor, and inertia parameters, whose values, within specified ranges, can be set by the user.

The aircraft dynamics are modeled as a set of twelve first order coupled nonlinear differential equations, given as follows:

$$\dot{x}(t) = f\left(x(t), u(t), \Delta\right) \quad (1a)$$
$$y(t) = h\left(x(t), u(t)\right) \quad (1b)$$

where $x(t)$ is the state vector with 12 components, i.e., velocity, angle-of-attack (AoA or α), sideslip angle, and angular rate, attitude, and position vectors. Δ represents the uncertain aircraft parameters—Table I shows the uncertain parameters considered in this study. $y(t)$ is the output vector, and $u(t)$ is the control input vector, whose components are left and right canard deflection angle, left and right inboard/outboard elevon deflection angle, leading edge flap deflection angle, rudder deflection angle, landing gear status (extract/retract), and vertical and horizontal thrust vectoring. The control input is determined by

$$u(t) = g\left(x(t), y_{\mathrm{REF}}(t)\right) \quad (2)$$

where $g(\cdot,\cdot)$ is an industry standard flight control law, which is provided with the ADMIRE model, and $y_{\mathrm{REF}}(t)$ is the reference demand that consists of the pilot inputs, i.e., pitch stick demand, roll stick demand, rudder pedal demand, and thrust demand. Equations (1) and (2) together represent the closed loop dynamics of the aircraft with the flight control law in the loop.

The augmented ADMIRE operational flight envelope is defined up to Mach 1.2 and altitude 6000 meters [3]. The longitudinal control law is gain scheduled over the whole flight envelope with respect to Mach and altitude variations and is designed to ensure robust stability and handling performance over the entire flight envelope. The model also contains rate limiting and saturation blocks [4], as well as nonlinear stick shaping elements in its forward path.

A. Nonlinear Clearance Criterion

The clearance criterion considered in this study is the AoA limit exceedence criterion [1], [5], [6]. For this criterion, it is required to identify the flight cases where, for the pull-up manoeuvre defined in Fig. 2, the maximum overshoot occurs in AoA. In particular, the combination of uncertainties that yields the largest exceedence of the defined limits must be identified. The test aims to assess the effectiveness of the AoA limiting scheme in the flight control system, in terms of the peak overshoot in AoA that occurs in response to the specified manoeuvre. Fig. 2 shows the specified pitch stick command, a rapid pull in longitudinal stick to a defined level (40 N) at a 640 N/s stick rate

TABLE II
RESULTS FOR LOCAL OPTIMIZATION ALGORITHM

Starting Point $\left[\Delta^0_{mass}, \Delta^0_{xcg}, \Delta^0_{Cm_{\delta_e}}, \Delta^0_{I_{yy}}, \Delta^0_{Cm\alpha}\right]$	Convergent Point $\left[\Delta^*_{mass}, \Delta^*_{xcg}, \Delta^*_{Cm_{\delta_e}}, \Delta^*_{I_{yy}}, \Delta^*_{Cm\alpha}\right]$	Number of Simulations
[0, 0, 0, 0, 0]	[0.100, 0.0750, 0.050, 0.06084, 0.050]	375
[0.100, 0.0750, 0.050, 0.200, 0.050]	[0.100, 0.0750, 0.050, 0.18309, 0.050]	366
[-0.100, -0.0750, -0.050, -0.200, -0.050]	[0.100, 0.0750, 0.050, -0.12634, 0.050]	322

with stick hold for 10 s. The present analysis aims to estimate the clearance criterion [1]

$$\alpha_{\max} = \max(\alpha(t)) \quad \text{for} \quad t \leq 10\,[\text{sec}] \tag{3}$$

for all possible combinations of aircraft parametric uncertainty.

B. Optimization Based Flight Clearance

In this paper, the flight clearance problem defined above is formulated as an optimization problem and solved using a number of different approaches. The optimization problem itself is to find the combination of real parametric uncertainties that gives the worst value of the criterion defined in (3). Since this and many other clearance criteria must be checked over a huge number of envelope points and aircraft configurations, it is imperative to find the most computationally efficient approach to the problem. Previous efforts to apply optimization methods to this problem [1, Ch. 7] have revealed that the nonlinear optimization problems arising in flight clearance, while having relatively low order, often have multiple local optima and expensive function evaluations. Therefore, the issue of whether to use local or global optimization, and the associated impact on computation times is a key consideration for this problem.

In [1, Ch. 21], local optimization methods such as Sequential Quadratic Programming (SQP), and limited memory Broyden–Fletcher–Goldfarb–Shanno method with bounded constraints (L-BFGS-B) were used to evaluate a range of linear clearance criteria for the High Incidence Research Model (HIRM+) aircraft model. In [1, Ch. 22], global optimization schemes such as genetic algorithms (GA), adaptive simulated annealing (ASA), and multicoordinate search (MCS) were also applied to evaluate nonlinear clearance criteria for the same aircraft model. In [5] and [6], global optimization methods such as GA and ASA were applied to the ADMIRE model with a different flight clearance criterion. The contributions of this paper are as follows. We demonstrate conclusively, for a realistic, industry-standard aircraft simulation model, the necessity of using global optimization approaches in order to avoid getting trapped in local solutions to the flight clearance problem. We also show, however, that incorporation of local optimization methods into global algorithms using hybrid switching schemes can drastically reduce computation times *and* improve convergence to the true global solution. Finally, we compare the performance of two evolutionary optimization algorithms (and their hybrid versions) on a realistic problem which is of significant interest to the aerospace industry.

All the results presented in this paper were generated with the ADMIRE model trimmed at Mach 0.4 and altitude 3000 meters in straight and level flight. Once the trim is achieved, the pull-up manoeuvre shown in Fig. 2 is applied and the cost function is given by (3), i.e., maximum AoA.

III. Local Optimization

A local optimization method based on SQP is first considered to solve the above problem. The implementation provided in [7] (specifically the function *"fmincon"*) was used to find the constrained minimum of a scalar function of several variables starting at an initial estimate. A medium scale optimization scheme is chosen, where the gradients are estimated by the function itself using the finite difference method [7].

Local optimization methods can, of course, get locked into local optima in the case of multimodal surfaces, however, they are also much more computationally efficient than global optimization approaches. Whether a local method converges to a local optimum or not completely depends on the initial starting point in the search space. Crucially however, in typical flight clearance problems, very little information is available as to where to start the optimization—the number of uncertain parameters and strong nonlinearity of the system mean that even advanced knowledge of flight mechanics provides little insight into how to choose initial values for the uncertain parameters. In the present analysis, constraints are due only to the upper and lower bounds of the uncertainty in the variables.

The starting point for *"fmincon"* could be simply the nominal values of the uncertain parameters. The optimization algorithm calls the simulation model to evaluate the cost function for a particular point over the search space. The uncertain parameter values are supplied by the optimization algorithm at each iteration and the cost function is evaluated and returned. The iterations continue until the specified termination criterion is met. Typical calculation results for our problem are shown in Table II—note that in Table II the last column shows the total number of simulations, i.e., the number of cost function (fitness) evaluations. Later, it will be shown, via exhaustive global optimization trials, that the parameter combination in the second row is (as far as can be established) the global solution. As expected, however, for each different initial guess for the values of the uncertain parameters, the local optimization algorithm converges to a different point in the uncertain parameter space. These results show, therefore, that using local optimization methods in isolation allows very little confidence to be established that the true worst case violation of the clearance criterion has been found.

IV. Global Optimization

The global optimization methods to be applied to the flight clearance problem in this paper belong to the class of evolutionary optimization algorithms [8]. Genetic algorithms (GAs) are amongst the best known and most widely used evolutionary optimization algorithms in the field of control engineering [9], [10]. An interesting new subclass of this method, differential evolution (DE) [11], is also investigated in this study, as results in the recent literature indicate that DE can offer improved convergence and reduced computational overheads, and these issues are of particular interest for the problem of flight clearance. Many other powerful optimization algorithms based on evolutionary principles exist, e.g., particle swarm optimization [12], ant colony optimization [13], and self organizing migrating algorithms (SOMA) [14], and their suitability for flight clearance problems could certainly be profitably investigated in future studies.

A. Genetic Algorithms (GAs)

The first global optimization method we consider in this study is GA, which are general purpose stochastic search and optimization procedures, based on genetic and evolutionary principles [15]. This approach assumes that the evolutionary process observed in nature can be simulated on a computer to generate a population of fittest candidates. In a genetic search technique, a randomly sourced population of candidates undergoes a repetitive evolutionary process of reproduction through selection for mating according to a fitness function, and recombination via crossover with mutation. A complete repetitive sequence of these genetic operations is called a generation. To use this evolutionary method, it is necessary to have a method of encoding the candidate as an artificial chromosome as well as a means of discriminating between the fitness of candidates. A fitness function is defined to assign a performance index to each candidate—this function is specific to the problem and is formed from the knowledge domain. GA have become a popular, robust search, and optimization technique for problems with large as well as small parameter search spaces. The recent survey paper [9] reports that GAs have also become a very popular search and optimization technique for problems in control engineering. Due to their stochastic nature, global optimization schemes such as GA can be expected to have a much better chance of converging to a global optimal. The price to be paid for this improved performance is a dramatic increase in computation time when compared with local methods. In the sequel, the genetic operators employed to generate and handle the population in the GA for the clearance problem are described. The reader is referred to [15] for more details of different operators, binary coding schemes, and the theory of genetic search.

1) Variable Representation: The genetic representation, i.e., the chromosome, for the clearance problem considered here, is the real uncertain parameter set. Each of the uncertainties corresponds to one gene. A binary coded string is generated to represent the chromosome, where each of the parametric uncertainties is bounded as shown in Table I. The level of accuracy for each parameter was chosen to be 10^{-6}, so that a change in the least weighted bit will give an accuracy level of 10^{-6} to each uncertain parameter. The number of bits required to represent a variable depends on the upper and lower bounds of the optimization variables and the accuracy level required. All the variables are represented in a binary vector format. The length of the chromosome is 105 bits, consisting of 5 genes each of 21 bits. The binary values for each uncertain parameter are converted into real values and these real values are assigned to the respective uncertain parameter variables in the ADMIRE model immediately after the trim condition is achieved, and prior to applying the stick command shown in Fig. 2. After simulation, each chromosome is assigned a fitness value, and the fitness function is the nonlinear response criterion given in (3).

2) Initialization: The GA search starts from an initial random number of candidates of a given size N_{size}. For the present study, the number N_{size} is kept fixed at 50. If the population size is reduced below a certain level, the population loses diversity over the search space [15] and the quality of the final solution falls or takes longer to compute.

3) Selection: The manner in which the candidates in the current iteration (generation) are qualified for producing the successive generations depends on the selection scheme. There are many different selection schemes available [15], [16]. In this analysis, use is made of the roulette wheel selection scheme. Parents are selected according to their fitness. The fitter the chromosomes are, the more chance they have to be selected. This is analogous to a roulette wheel containing all the chromosomes in the population. The size of a section in the roulette wheel is proportional to the value of the fitness function of each chromosome. The selection depends on the probability factor of selection which is assigned a value 0.6 in this study.

4) Crossover: Crossover is a recombination operator that ensures the mixing up of the information content in two binary coded chromosomes. Usually, two parent chromosomes are selected randomly to interchange the information content, and thereby produce new offspring that contain information content from both the parents. A probability of crossover is defined which determines the maximum allowed number of pairs for crossover operation. In general, the probability of crossover is kept high. A simple single-point crossover scheme is employed in this study with a probability of 0.9. The information between the parents is exchanged at a randomly chosen crossover point over the length of bits.

5) Mutation: Mutation introduces random variations in the population over the search space, by randomly flipping a bit value in the case of binary coded GA. In this study, the operation is done with a very low probability of 0.005. A binary uniform mutation scheme is used, which randomly selects an individual and sets it to a random value by flipping a randomly selected single bit.

6) Replacement Strategy: An elitist strategy is followed such that over each generation the best candidate in the current population moves into the new generation population by replacing the worst candidate of the population. This ensures the presence of a better candidate in the new generation and thereby increases the average fitness of the population over generations.

7) Termination Criterion: Many different termination criteria can be employed. In the present study, an adaptive termination criterion is used that is dependent on improvement in the solution accuracy over a finite number of successive generations. The algorithm terminates the search if there is no improvement on the best solution achieved (above a defined accuracy level,

TABLE III
GLOBAL OPTIMIZATION COMPARISON STATISTICS: NUMBER OF SIMULATIONS

Optimisation Method	Trails	Average	Minimum	Maximum	Std. Deviation	Prob. of Success
GA	100	4485	2400	7500	828.364	65%
DE	100	3086	1152	4176	567.57	90%

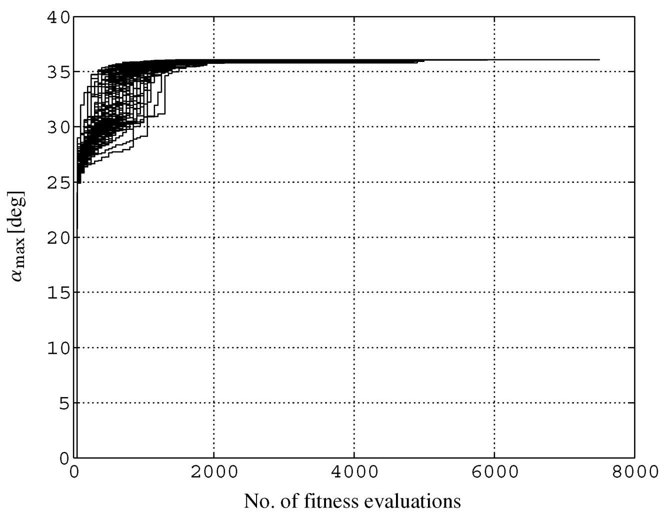

Fig. 3. GA—Number of fitness evaluations versus best fitness.

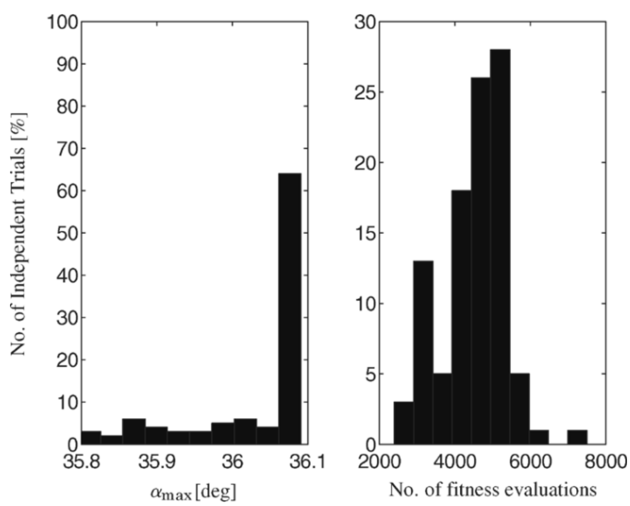

Fig. 4. GA results histogram.

here chosen as 10^{-6}) for a defined successive number of generations. This number of generations is fixed at 15.

8) Results: Fig. 3 shows the number of fitness evaluations versus the best fitness for 100 GA trials. The statistics of the results, from the 100 independent trials, are given in Table III. The number of fitness evaluations corresponds to the number of simulations, which, for consistency is used to compare the computational overheads of the different algorithms. The left histogram of Fig. 4 shows the percentage distribution of the maximum value of AoA achieved over the 100 trials. The right histogram of Fig. 4 shows the percentage distribution of the total number of fitness evaluations required to obtain the solution over the 100 independent trials. A large number of fitness evaluations, an average of 4485 simulations in this case, is required to obtain the global, or near global solution. The probability of success in attaining the true global solution is also rather low, at only 65%. The global solution found in this example is the following:

$$\left[\Delta^*_{\text{mass}}, \Delta^*_{xcg}, \Delta^*_{C_{m_{\delta_e}}}, \Delta^*_{I_{yy}}, \Delta^*_{C_{m_\alpha}}\right] = [0.1000,\ 0.0750,\ 0.0500,\ 0.183\,09,\ 0.0500] \quad (4)$$

and $\alpha_{\max}$ is 36.0908°. Note that four of the uncertain parameters in this case are on their upper bounds and $\Delta^*_{I_{yy}}$ is inside its bound. A sensitivity analysis is performed about the solution and is shown in Fig. 5, where the x axis is normalized. As different allowable minimum and maximum bounds are defined for each of the uncertain parameters, for the purposes of comparison the uncertain parameters are normalized to have a variation between -1 and $+1$ in the sensitivity plots. As shown in the figure, the uncertain parameter $\Delta_{I_{yy}}$ has many local maxima.

Tuning GA optimization parameters, such as the different GA-operator probabilities may, of course, improve the above results to a certain extent. However, there are few available guidelines as to how to do this tuning. Another possible approach would be to use alternate selection schemes and scaling and ranking procedures, such as those described in [15, Ch. 4]. However, for the present problem the advantage to be gained from these techniques is not expected to be significant. Finally, we note that in the context of the current flight clearance process, the computational cost of the number of fitness evaluations required by the above approach would be likely to prove prohibitive to its widespread adoption by industry [1, Ch. 1].

B. Differential Evolution (DE)

The second global optimization method considered in this study is DE, a relatively new global optimization method, introduced by Storn and Price in [11]. This method belongs to the same class of evolutionary global optimization techniques as GA, but unlike GA it does not require either a selection operator or a particular encoding scheme. Essentially, a subtype of GA, despite its apparent simplicity, the quality of the solutions computed using this approach has been claimed to be often better than that achieved using other evolutionary algorithms, both in terms of accuracy and computational overhead [11].

The DE method has recently been applied to several problems in different fields of engineering design, with promising results. In [17], for example, it was applied to find the optimal solution for a mechanical design example formulated as a mixed integer discrete continuous optimization problem. In [18], DE was successfully applied in system design application, in particular, handling the nonlinear design specification constraints. In [10], the DE method was applied and compared with other local and global optimization schemes in an aerodynamic shape optimization problem for an aerofoil. The DE method consists of the following four main steps: 1) random initialization; 2) mutation

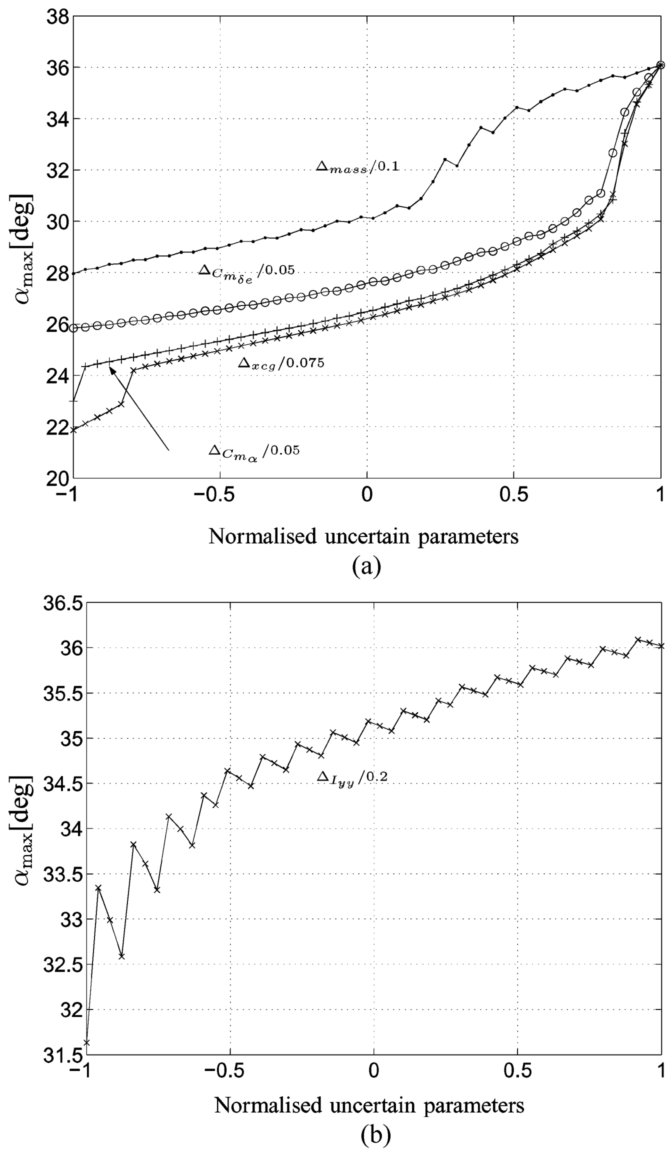

Fig. 5. Sensitivity plots about the global solution.

3) crossover; and 4) evaluation and selection. There are different schemes of DE available based on the operators. The one used in the present studies is referred as "$DE/rand/1/bin$." The steps of this scheme will be described in detail in the sequel.

1) Random Initialization: Like other evolutionary algorithms, DE works with a fixed number, N_p, of potential solution vectors, initially generated at random according to

$$\mathbf{x}_i = \mathbf{x}^L + \rho_i(\mathbf{x}^U - \mathbf{x}^L), \quad i = 1, 2, \ldots, N_p \tag{5}$$

where $\mathbf{x}^U$ and $\mathbf{x}^L$ are the upper and lower bounds of the parameters of the solution vector and ρ_i is a vector of random numbers in the range [0 1]. N_p is fixed at 12 in the current study. Each $\mathbf{x}_i$ consists of elements $(x_{1i}, x_{2i}, \ldots, x_{di})$, which are the uncertain parameters defined in Table I. The dimension d of the optimization problem considered is, therefore, 5. The fitness of each of these N_p solution vectors is evaluated using the cost function given in (3).

2) Mutation: The scaled difference vector $F_m D_{ij}$ between two random solution vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ is added to another randomly selected solution vector $\mathbf{x}_k$ to generate the new mutated solution vector $\bar{\mathbf{x}}_n^{G+1}$, i.e.,

$$\bar{\mathbf{x}}_n^{G+1} = \mathbf{x}_k^G + F_m D_{ij}, \quad D_{ij} = \mathbf{x}_i^G - \mathbf{x}_j^G \tag{6}$$

where F_m is the mutation scale factor, a real valued number in the range [0, 1], (fixed at 0.8 in this study), and G represents the iteration number. Fig. 6 shows a simple two-dimensional example of the mutation operation used in the DE scheme. The difference vector D_{ij} determines the search direction and F_m determines the step size in that direction from the point $\mathbf{x}_k^G$.

3) Crossover: During crossover, each element of the nth solution vector of the new iteration $\mathbf{x}^{G+1}$, is reproduced from the mutant vector $\bar{\mathbf{x}}_n^{G+1}$ and a chosen parent individual $\mathbf{x}_n^G$ as given in (7)

$$x_{ji}^{G+1} = \begin{cases} x_{ji}^G, & \text{if a generated random number} > \rho_c \\ \bar{x}_{ji}^{G+1}, & \text{otherwise} \end{cases} \tag{7}$$

where $j = 1, 2, \ldots, d$ and $i = 1, 2, \ldots, N_p$. Note that $\bar{\mathbf{x}}_n^{G+1}$ has elements $(\bar{x}_{1n}^{G+1}, \bar{x}_{2n}^{G+1}, \ldots, \bar{x}_{dn}^{G+1})$ and $\mathbf{x}_n^G$ has elements $(x_{1n}^G, x_{2n}^G, \ldots, x_{dn}^G)$. $\rho_c \in [0, 1]$ is the crossover factor, which is fixed at 0.8 in the present study.

4) Evaluation and Selection: After crossover, the fitness of the new candidate $\mathbf{x}_n^{G+1}$ is evaluated using (3). If the new candidate $\mathbf{x}_n^{G+1}$ has a better fitness than the parent candidate $\mathbf{x}_n^G$, then $\mathbf{x}_n^{G+1}$ is selected to become part of the next iteration. Otherwise, $\mathbf{x}_n^G$ is selected and subsequently identified as $\mathbf{x}_n^{G+1}$.

5) Termination Criterion: The same termination criterion as that chosen for the GA trials was used.

6) Results: Fig. 7 shows the number of fitness evaluations versus the best fitness for 100 DE trials. 90 trials converged to the true global solution given in (4), giving the maximum AoA overshoot. Seven trials converged to solutions very close to the global solution, and three trials gave different solutions. Compared with the GA results, DE can be seen to offer significantly improved convergence properties, while the reduced number of initial random starting points (only 12 initial random points against 50 random initial points for the GA) means that the total number of fitness evaluations required in each trial was also significantly reduced. Table III provides the statistics of the results obtained from the 100 trials of the DE algorithm, and also compares them to those from the GA. The average number of fitness evaluations required for DE, 3086 in this case, is 31% less than required by the GA. The probability of success of the DE algorithm is also much higher, at 90%. The left subplot in Fig. 8 shows the distribution of the maximum value of AoA achieved. The right subplot shows the distribution of the number of fitness evaluations over 100 independent trials of the DE algorithm. Note that, in addition to the improved results, another advantage of this method compared with that of GA is the reduced number of optimization parameters that must be adjusted by the user.

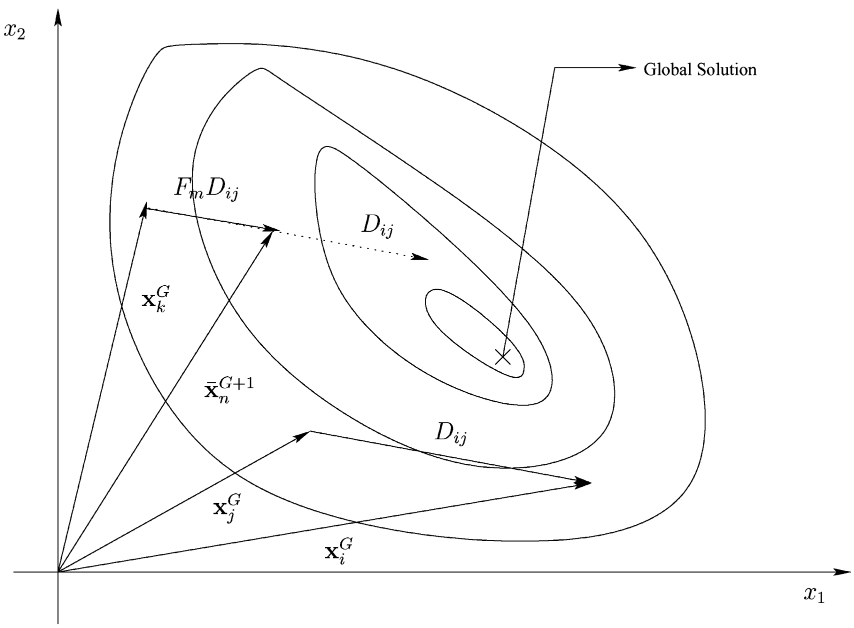

Fig. 6. DE mutation strategy.

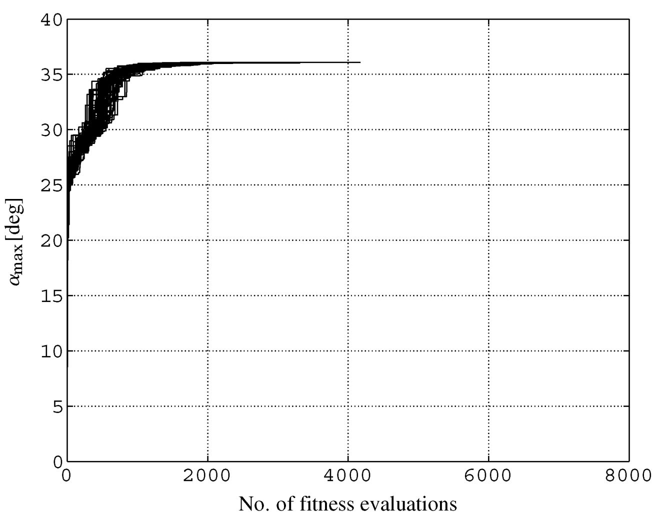

Fig. 7. DE—Number of fitness evaluations versus best fitness.

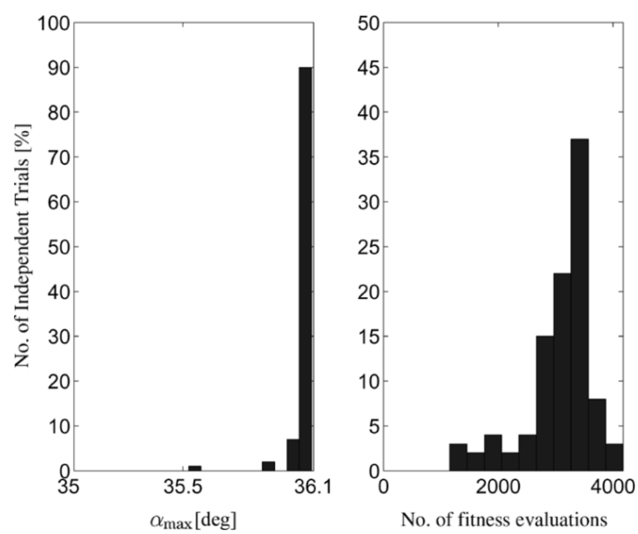

Fig. 8. DE results histogram.

V. Hybrid Optimization

Global optimization methods based on evolutionary principles are generally accepted as having a high probability of converging to the global or near global solution, if allowed to run for a long enough time with sufficient initial candidates and reasonably appropriate probabilities for the evolutionary optimization parameters. As shown by the preceding results, however, the rate of convergence can be very slow, and moreover, there is still no guarantee of convergence to the true global solution. Local optimization methods, on the other hand, can very rapidly find optimal solutions, but the quality of those solutions entirely depends on the starting point chosen for the optimization routine. In order to try to extract the best from both schemes, several researchers have proposed combining the two approaches [16], [19], [20]. In such hybrid schemes, there is the possibility of incorporating domain knowledge, which gives them an advantage over a pure blind search based on evolutionary principles. In [2], a hybrid GA (HGA) scheme was developed using a switching strategy originally proposed in [20], and applied to a nonlinear flight clearance problem. In the next section, we compare the performance of this HGA scheme with a novel hybrid DE (HDE) scheme developed for this study. For a recent comprehensive overview of similar other approaches to hybrid optimization (also known as memetic algorithms), the reader is referred to [21].

A. Hybrid GA (HGA)

The HGA scheme is based on the idea of associating with both the global and local methods, a reward or gain. The reward associated with a method is a measure of how well the method helped in providing a solution which is better than the

TABLE IV
HYBRID OPTIMIZATION COMPARISON STATISTICS: NUMBER OF SIMULATIONS

Optimisation Method	Trails	Average	Minimum	Maximum	Std. Deviation	Prob. of Success
HGA	100	2011	1357	4468	547.42	92%
HDE	100	1106	477	1434	192.42	98%

TABLE V
HYBRID GENETIC ALGORITHM

1) Initialize $W_{\mathrm{GA}}^0 = 0.9$, $W_{\mathrm{Local}}^0 = 0.1$, $c = 0.3$, $k = 1$, set the calculation mode "Search", set the number of confirmations to zero, and generate initial population for GA
2) While the confirmation number is less than a certain number (e.g. 20)
   a) Calculate P_{GA}^k, (9)
   b) (Flip Coin) = a random number between zero and one
   c) If (Flip Coin) $< P_{\mathrm{GA}}^k$ then run GA and update W_{GA}^k, (8)
   d) else choose the local algorithm with the following initial guess
      i) If the calculation mode is "Search", choose one randomly from two best in the population,
      ii) else choose one randomly from the subset of population where the distance of each element from the current best is out of 1σ (standard deviation of the population from the current best)
      iii) Update W_{Local}^k, (8)
   e) If the cost does not improve,
      i) Initialize the following every five confirmations: population, $W_{\mathrm{GA}}^0 = 0.5$, $W_{\mathrm{Local}}^0 = 0.5$, $c = 0.6$ and set calculation mode equal to "Confirm"
      ii) Increase the number of confirmation
   f) else set the number of confirmation equal to zero and set calculation mode equal to "Search"
3) end of While

one previously found. The reward associated to each optimization scheme will determine the probability for that optimization scheme to be chosen at each iteration. The reward for each optimization scheme thus keeps varying depending on how well it is performing. A simple way to assign a reward is with a weighted geometric average. The following equation is used to update the weighted reward for each optimization scheme [20]:

$$W_{\mathrm{GA/Local}}^{k+1} = W_{\mathrm{GA/Local}}^{k}(1-c) + cR_{\mathrm{GA/Local}}^{k} \quad (8)$$

where W^k and R^k are the weighted reward and the improvement in the solution at the iteration k, respectively, and c is a constant in [0, 1]. R^k is computed based on the improvement in the best solution attained over each iteration/generation. In case no improvement occurs, the value of R^k is set equal to zero. If one knows at each time step which optimization method is going to give most improvement toward the global solution, that particular method can be chosen to accelerate the convergence. When it is not known beforehand, a decision has to be taken based on the previous reward and by calculating the associated probability. The algorithm for the hybrid switching scheme is summarized in Table V.

Due to the frequent occurrence of local maxima in flight clearance problems, it is desirable that, initially, the GA should have a higher probability of being chosen than the local algorithm. Hence, initially the weights for GA and the local algorithm are given as 0.9 and 0.1, respectively. The local algorithm used in the present study is the implementation of the SQP method [7], described in Section III.

Due to the improved convergence properties of the HGA algorithm (see below), it was possible to reduce the size of the initial population to 40 candidates. The initial guess for the local algorithm is taken from the population depending on the calculation mode. There are two modes in the algorithm, search and confirm. In search mode, the initial guess is chosen from the two best in the population. In confirm mode, the initial guess is chosen from a subset of the population, chosen to be far away from the current best. From here onwards the decision-making is done based on probability matching depending on the rewards associated with each of the optimization schemes. The probability of selecting the GA at any iteration can be calculated from the following equation [20]:

$$P_{\mathrm{GA}}^k = \frac{W_{\mathrm{GA}}^k}{\left(W_{\mathrm{GA}}^k + W_{\mathrm{Local}}^k\right)}. \quad (9)$$

A random number generator simulates a coin toss and depending on the result one of the optimization schemes is chosen. If the scheme chosen is global optimization, it proceeds with only one generation. If the local scheme is chosen, then the optimization runs until it either converges or reaches the defined maximum number of cost function evaluations. At the end of a run of either of the optimization schemes, the improvement achieved above the value of the best solution prior to the optimization run is checked. The reward for a particular, local or global, optimization is assigned, the probabilities are updated and the sequence is repeated until no improvement occurs from either of the two methods.

Fig. 9 shows the number of fitness evaluations versus the best fitness for 100 trials of the HGA. Table IV provides the statistical results. The average number of cost function evaluations required was 2011, an improvement of 55% when compared with the standard GA. The success rate in finding the true global solution is also dramatically improved, from 65% to 92%. The left

TABLE VI
HYBRID DIFFERENTIAL EVOLUTION

1) Initialize random candidate solutions in search space
2) Evaluate the fitness of each solution and choose the best fitness
3) Apply DE for few initial iterations (e.g., 10); Update the best fitness value in each iteration
4) While the termination criteria satisfied, calculate the improvement in best fitness
    a) If *Improvement* in best fitness, then continue DE
    b) else
        i) Choose a random solution from the set, say X_0 and apply local optimisation "*fmincon*" with X_0 as initial point
        ii) If *Improvement* in best fitness, then replace X_0 in the set with the new solution
        iii) else keep X_0 in the set.
5) end of While

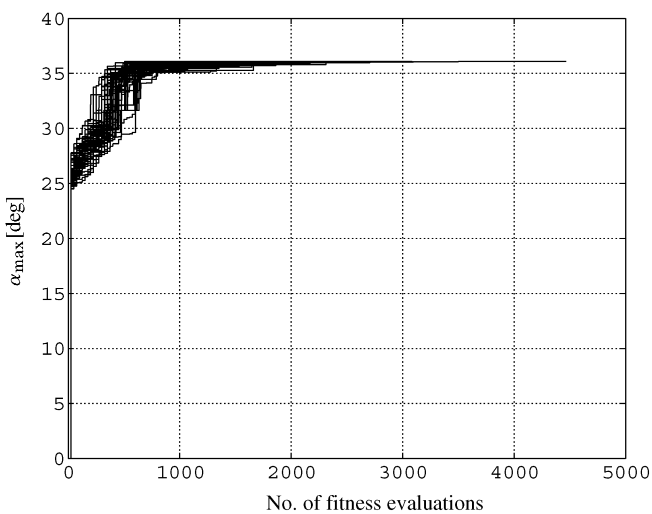

Fig. 9. HGA—Number of fitness evaluations versus best fitness.

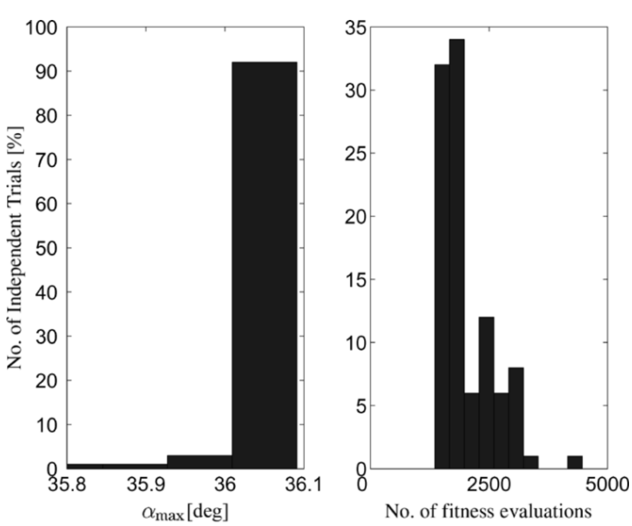

Fig. 10. HGA histogram.

and right subplots of Fig. 10 show the histogram distributions of maximum AoA obtained and the number of fitness evaluations taken, respectively, over the 100 independent trials.

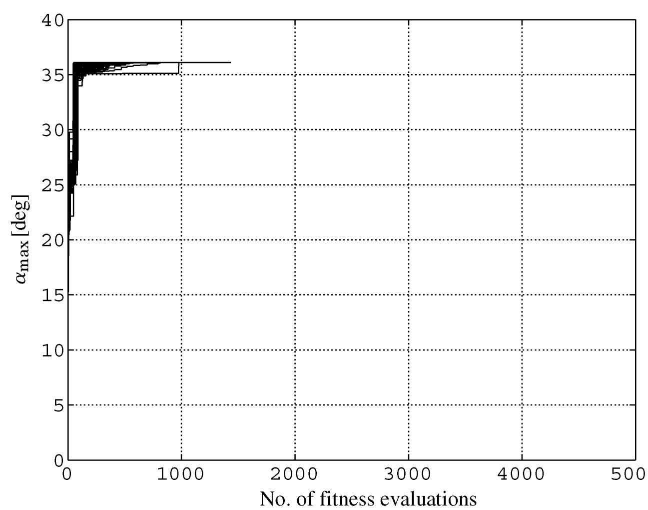

Fig. 11. HDE—Number of fitness evaluations versus best fitness.

B. Hybrid DE (HDE)

In [22], the conventional DE methodology was augmented by combining it with a downhill simplex local optimization scheme. This hybrid scheme was applied to an aerofoil shape optimization problem and was found to significantly improve the convergence properties of the method. At each iteration, local optimization was applied to the best individual in a current random set. The hybrid DE scheme employed in this study applies gradient-based local optimization, again using *"fmincon"*, to a solution vector *randomly* selected from the current set—for our problem, this was seen to give better results than using the *best* solution vector, as proposed in [22]. When the local scheme is chosen, the optimization starts from the given initial condition and continues until it either converges or reaches a defined maximum number of cost function evaluations. The algorithm is simple, and tries to search for the global optimum in a "greedy" way, demanding improvement in the achieved optimum value in every iteration. A pseudocode for the hybrid DE algorithm is given in Table VI.

Fig. 11 shows the number of fitness evaluations (function evaluations) versus the best fitness for 100 trials of the HDE algorithm. Table IV provides the statistical results and compares them with the results of the HGA. The average number of cost function evaluations required was 1106, an improvement of 64% when compared with the standard DE algorithm, and

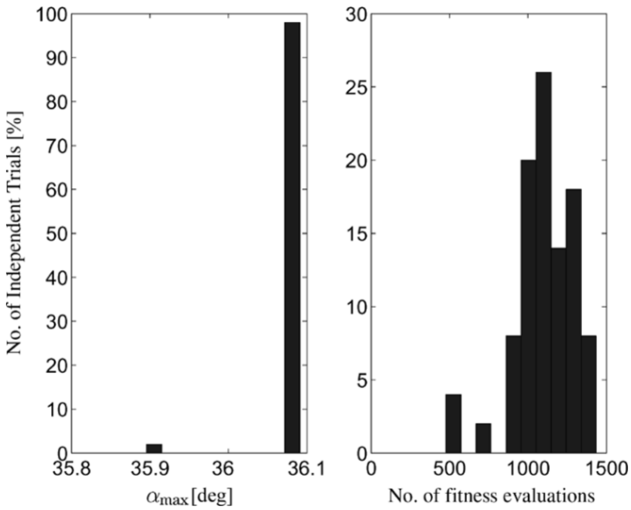

Fig. 12. HDE results histogram.

45% when compared with the HGA. The success rate in finding the true global solution is also extremely high, at 98%. The left and right subplots of Fig. 12 show the histogram distributions of maximum AoA obtained and the number of fitness evaluations taken, respectively, over the 100 independent trials.

VI. Conclusion

This paper has compared the performance of two different evolutionary optimization algorithms, namely, GAs and DE, on a nonlinear flight control law clearance problem. The necessity of using global optimization methods for flight clearance was clearly demonstrated by the very different results returned by gradient-based optimization starting from different initial points in the parameter space. The GA method, on the other hand, converged to the exact global solution in 65 out of 100 different trials, while the DE algorithm converged in 90 out of 100 trials. Particularly, striking is the fact that DE achieves this improved accuracy in tracking the global solution with a reduced computational overhead—taking an average of 3086 simulations, 31% faster than the average of 4485 simulations required by GA.

Hybrid versions of both algorithms incorporating local gradient-based optimization were shown to offer significant advantages in terms of both reduced computational complexity and improved global convergence properties. The hybrid version of the GA employing the SQP local optimization scheme converged to the global solution in 92 out of 100 individual trials, with an average of 2011 simulations. The hybrid version of the DE method outperformed all other schemes considered in this study by converging to the global solution in 98 out of 100 independent trials with an average of only 1106 simulations—45% faster than the hybrid GA. These results indicate that the recently introduced DE approach in particular, when appropriately augmented with local optimization methods, has significant potential to improve both the reliability and efficiency of the current industrial flight clearance process.

Acknowledgment

The authors are grateful to the members of GARTEUR Action Groups FM-AG11 and FM-AG17 for many helpful discussions during the course of this research.

References

[1] C. Fielding, A. Varga, S. Bennani, and M. Selier, Eds., *Advanced Techniques For Clearance of Flight Control Laws*. Berlin, Germany: Springer-Verlag, 2002.
[2] P. P. Menon, J. Kim, D. G. Bates, and I. Postlethwaite, "Improved clearance of flight control laws using hybrid optimization," in *Proc. IEEE Conf. Cybern. Intell. Syst.*, Singapore, Dec. 2004, pp. 676–681.
[3] L. S. Forssell, G. Hovmark, Å. Hyden, and F. Johansson, The aero-data model in a research environment (ADMIRE) for flight control robustness evaluation, GARTUER/TP-119-7, Aug. 1, 2001. [Online]. Available: http://www.foi.se/admire/main.html
[4] L. Rundqwist, K. Stahl-Gunnarsson, and J. Enhagen, "Rate limiters with phase compensation in JAS39 GRIPEN," in *Proc. Eur. Control Conference*, Jul. 1997, pp. 2451–2457.
[5] L. S. Forssell and Å. Hyden, "Flight control system validation using global nonlinear optimization algorithms," in *Proc. Eur. Control Conf.*, Cambridge, U.K., Sep. 2003, CD-ROM.
[6] L. S. Forssell, "Flight clearance analysis using global nonlinear optimization based search algorithms," in *Proc. AIAA Guidance, Navigation, and Control Conf.*, Austin, TX, Aug. 2003, CD-ROM.
[7] *"Optimization Toolbox User's Guide,"* ver. 2, The MathWorks, Sep. 2000.
[8] T. Back, U. Hammel, and H. P. Schwefel, "Evolutionary computation: Comments on the history and current state," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 3–17, Apr. 1997.
[9] P. J. Fleming and R. C. Purshouse, "Evolutionary algorithms in control systems engineering: A survey," *Control Engineering Practice*, vol. 10, pp. 1223–1241, 2002.
[10] T. Rogalsky, R. W. Derksen, and S. Kocabiyik, "Differential evolution in aerodynamic optimization," *Canadian Aeronautics and Space Inst. J.*, vol. 46, pp. 183–190, 2000.
[11] R. Storn and K. Price, "Differential evolution: A simple and efficient heuristic for global optimization over continuous space," *J. Global Optimization*, vol. 11, pp. 341–369, 1997.
[12] M. Clerc and J. Kennedy, "The particle swarm—explosion, stability, and convergence in a multidimensional complex space," *IEEE Trans. Evol. Comput.*, vol. 6, no. 1, pp. 58–73, Feb. 2002.
[13] M. Dorigo and L. M. Gambardella, "Ant colony system: A cooperative learning approach to the traveling salesman problem," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 53–66, Apr. 1997.
[14] I. Zelinka and J. Lampinen, "SOMA—self-organizing migrating algorithm," in *Proc. 6th Int. Conf. Soft Computing*, Brno, Czech Republic, 2000, 80-214-1609-2.
[15] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Reading, MA: Addison-Wesley, 1989.
[16] L. Davis, Ed., *Handbook of Genetic Algorithms*. New York: Van Nostrand Reinhold, 1991.
[17] J. Lampinen and I. Zelinka, "Mechanical engineering design by differential evolution," in *New Ideas in Optimization*, D. Corne, M. Dorigo, and F. Glover, Eds. London, U.K.: McGraw-Hill, 1999, pp. 127–146.
[18] R. Storn, "System design by constraint adaptation and differential evolution," *IEEE Trans. Evol. Comput.*, vol. 3, no. 1, pp. 22–34, Apr. 1999.
[19] J. Yen, J. C. Liao, D. Randolph, and B. Lee, "A hybrid approach to modeling metabolic systems using genetic algorithm and simplex method," in *Proc. 11th IEEE Conf. Artif. Intell. Appl.*, Los Angeles, CA, Feb. 1995, pp. 277–283.
[20] F. G. Lobo and D. E. Goldberg, Decision making in a hybrid genetic algorithm, IlliGAL Rep. 96009, Sep. 1996.
[21] N. Krasnogor and J. Smith, "A tutorial for competent memetic algorithms: model, taxonomy, and design issues," *IEEE Trans. Evol. Comput.*, vol. 9, no. 5, pp. 474–488, Oct. 2005.
[22] T. Rogalsky and R. W. Derksen, "Hybridization of differential evolution for aerodynamic design," in *Proc. 8th Annu. Conf. Comput. Fluid Dynamics Society of Canada*, 2000, pp. 729–736.

Prathyush P. Menon was born in Kerala, India, in 1976. He received the B.Tech degree in electrical and electronics engineering from Mahatma Gandhi University, Kerala, in 1997 and the M.Tech degree in aerospace engineering from the Indian Institute of Technology, Bombay, in 2003.

He was Research Engineer at the Systems Integration Laboratory, Aircraft Research Design Center (ARDC), Hindustan Aeronautics Limited, Bangalore, India. Since January 2004, he is a research student in the Control and Instrumentation Research Group, Department of Engineering, University of Leicester, Leicester, U.K. His main current research interests are in the area of robust and nonlinear control, flight dynamics and control, and optimization techniques.

Jongrae Kim was born in Jeju, Korea, in 1969. He received the B.Sc. and M.Sc. degrees in aerospace engineering from the Inha University, Incheon, Korea, in 1991 and 1996, respectively, and the Ph.D. degree in aerospace engineering from Texas A&M University, College Station, in 2002.

From September 2002 to December 2003, he was a Postdoctoral Researcher at the University of California, Santa Barbara. Since January 2004, he has been a Research Associate in the Control and Instrumentation Research Group, Department of Engineering, University of Leicester, Leicester, U.K. His main research interests are in the area of robustness analysis, optimal control and estimation, optimization, and dynamics of aerospace and bionetwork systems.

Declan G. Bates was born in Clonmel, Ireland, in 1970. He received the B.Eng. degree in electronic engineering and the Ph.D. degree in control engineering from Dublin City University, Dublin, Ireland, in 1992 and 1996, respectively.

He subsequently joined the Control and Instrumentation Research Group, University of Leicester, Leicester, U.K., first as a Research Associate and later as a Lecturer in control engineering. He has held visiting lecturer positions at the Technical University of Delft, Holland, and the University of Cranfield, U.K. He is the coauthor of over 50 technical publications, including two research monographs. His research interests are in the area of robust and nonlinear control, with particular emphasis on robustness analysis problems in aerospace and biological systems.

Ian Postlethwaite (M'79–SM'92–F'01) was born in Wigan, U.K., in 1953. He received the B.S.E. degree (First Class) from Imperial College, London University, London, U.K., in 1975 and the Ph.D. degree from Cambridge University, Cambridge, U.K., in 1978.

From 1978 to 1981, he was a Research Fellow at Cambridge University and spent six months in 1980 at General Electric Company, Schenectady, NY. In 1981, he was appointed to a Lectureship in Engineering Science at Oxford University. In 1988, he moved to a Chair of Engineering at the University of Leicester, where he was Head of the Department from 1995 to 2004, and in 2005, he became a Pro-Vice-Chancellor. He has held visiting research positions at the University of California at Berkeley and the Australian National University. He is a coauthor of *Multivariable Feedback Control* (Wiley, 1996 and 2005) with S. Skogestad. His research involves theoretical contributions to the field of robust multivariable control and the application of advanced control system design to engineering systems.

Dr. Postlethwaite is a Fellow of the IEE and the InstMC. In 1991, he received the IEE FC Williams Premium. In 2001, he was awarded the Sir Harold Hartley Medal of the InstMC, and in 2002, he received a Best Paper Prize from the *IFAC Journal of Control Engineering Practice*.