Metamodelling of multivariable engine models for real-time flight simulation.

P. Stewart, P. J. Fleming and S. A. MacKenzie*

September 18, 2003

Abstract

Sophisticated real-time distributed flight simulation environments may be constructed from a wide range of modelling and simulation tools. In this way accuracy, detail and model flexibility may be incorporated into the simulator. Distributed components may be constructed by a wide range of methods, from high level environments such as *Matlab*, through coded environments such as *C* or *Fortran* to hardware-in-the-loop. In this paper the Response Surface Methodology is combined with a hyper-heuristic (evolutionary algorithm) and applied to the representation of computationally intensive non-linear multivariable engine modelling. The paper investigates the potential for metamodelling (models of models) dynamic models which were previously too slow to be included in multi-component, high resolution real-time simulation environments. A multi-dimensional gas turbine model with five primary control inputs, six environmental inputs and eleven outputs is considered. An investigation has been conducted to ascertain to what extent these systems can be approximated by response surfaces with experiments which have been designed by hyper-heuristics as a first step towards *automatic* modelling methodology.

Keywords

Response Surface Methodology, Evolutionary Algorithms, Hyper-Heuristics, Real-Time.

Introduction

The roots of model reduction theory go back several decades, the main motivating factor in this study is to approximate computationally intensive models to achieve wall-clock real-time processing. The generalised form of this approach is known as *Metamodelling*, meaning the creation of *a model of a model*. The key feature of any method applied to achieve metamodelling is the retention of the dominant modes of the system, a method also known as dominant pole approximation. Standard methods tend to be successfully applied to linear or quasi-linear systems, however high-order non-linear complex systems tend to be far more problematical. Response Surface Methodology (RSM) [1] has been successfully applied to certain real-time approximation [2] and control design [3] problems. The system dominant pole approximations were low order and constituted a three dimensional response surface. Successful metamodelling relies on the excitation of the dominant modes of the system, along with adequate population of the multidimensional operational space of the system. Population of this space has hitherto been performed according to factorially designed experiments, however in the multidimensional case under consideration here, a strict factorial approach can result in an extremely large number of experiments being performed. Evolutionary algorithms are hence applied to the optimisation problem of minimising the number of experiments, and maximising the accuracy of the derived metamodel.

*Paul Stewart is with the Electrical Machines and Drives Group in the Department of Electrical Engineering, University of Sheffield, Mappin Street, Sheffield S1 3JD U.K. E-mail: p.stewart@sheffield.ac.uk, Tel: +44 (0)114 2225841. Peter J. Fleming is with the Department of Automatic Control and System Engineering, University of Sheffield, Sheena A MacKenzie is with QinetiQ Ltd. U.K.

1 The Response Surface Methodology

The response surface methodology is a technique designed to optimise process control by the application of designed experiments in order to characterise a system . The relationship between the response variable of interest (y), and the predictor variables ($\xi_1, \xi_2, ..., \xi_k$) may be known exactly allowing a description of the system of the form

$$y = g\left(\xi_1, \xi_2, ..., \xi_k\right) + \epsilon \tag{1}$$

where ϵ represents the model error, and includes measurement error, and other variability such as background noise. The error will be assumed to have a normal distribution with zero mean and variance σ^2 In general, the experimenter approximates the system function g with an empirical model of the form

$$y = f\left(\xi_1, \xi_2, ..., \xi_k\right) + \epsilon \tag{2}$$

where f is a first or second order polynomial. This is the empirical or response surface model. The variables are known as *natural variables* since they are expressed in physical units of measurement. In the response surface methodology (RSM), the natural variables are transformed into *coded variables* $x_1, x_2, ...x_k$ which are dimensionless, zero mean, and the same standard deviation. The response function now becomes

$$\eta = f\left(x_1, x_2, ..., x_k\right) \tag{3}$$

The successful application of RSM relies on the identification of a suitable approximation for f. this will generally be a first order model of the form

$$\eta = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_k x_k \tag{4}$$

or a second order model of the form

$$\eta = \beta_0 + \sum_{j=1}^{k} \beta_j x_j + \sum_{j=1}^{k} \beta_{jj} x_j^2 + \sum_{i<j}\sum \beta_{ij} x_i x_j \tag{5}$$

It may be neccessary to employ an approximating function greater than an order of two, based on the standard Taylor series expansion. The response surface methodology is intimately connected to *regression analysis*. For example when considering the first order model, the β terms comprise the unknown parameter set which can be estimated by collecting experimental system data. This data can either be sourced from physical experiments, or from previously designed dynamic computer models. The parameter set can be estimated by regression analysis based upon the experimental data. The method of least squares is typically used to estimate the regression coefficients. With $n > k$ on the response variable available, giving $y_1, y_2, ..., y_n$, each observed response will have an observation on each regressor variable, with x_{ij} denoting the ith observation of variable x_j. Assuming that the error term ϵ has $E(\epsilon) = 0$ and $Var(\epsilon) = \sigma^2$ and the (ϵ_i) are uncorrelated random variables. The model can now be expressed in terms of the observations

$$\begin{aligned} y_i &= \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_k x_{ik} + \epsilon_i \\ &= \beta_0 + \sum_{j=1}^{k} \beta_j x_{ij} + \epsilon_i, \\ i &= 1, 2, ..., n \end{aligned} \tag{6}$$

The β coefficients in equation (6) are chosen such that the sum of the squares of the errors (ϵ_i) are minimised via the least squares function

$$\begin{aligned} L &= \sum_{i=1}^{n} \epsilon_i^2 \\ &= \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{n} \beta_j x_{ij} \right)^2 \end{aligned} \tag{7}$$

The model can be more usefully expressed in matrix form as

$$\mathbf{y} = \mathbf{X}\beta + \epsilon \tag{8}$$

where

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ . \\ . \\ . \\ y_n \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} 1 & x_{11} & x_{12} & \dots & x_{1k} \\ 1 & x_{21} & x_{22} & \dots & x_{2k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n1} & x_{n2} & \dots & x_{nk} \end{bmatrix},$$

$$\beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ . \\ . \\ . \\ \beta_n \end{bmatrix}, \quad \epsilon = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ . \\ . \\ . \\ \epsilon_n \end{bmatrix}. \tag{9}$$

It is now necessary to find a vector of least squares estimators $\mathbf{b}$ which minimises the expression

$$L = \sum_{i=1}^{n} \epsilon_i^2 = \epsilon^{'} \epsilon = (\mathbf{y} - \mathbf{X}\beta)^{'} (\mathbf{y} - \mathbf{X}\beta) \tag{10}$$

and yields the least squares estimator of β which is

$$\mathbf{b} = \left(\mathbf{X}^{'}\mathbf{X}\right)^{-1} \mathbf{X}^{'}\mathbf{y} \tag{11}$$

and finally, the fitted regression model is

$$\hat{\mathbf{y}} = \mathbf{X}\mathbf{b}, \quad \mathbf{e} = \mathbf{y} - \hat{\mathbf{y}} \tag{12}$$

where $\mathbf{e}$ is the vector of residual errors of the model.

2 Multiobjective optimisation by evolutionary algorithm

Evolutionary algorithms are global parallel search and optimisation methods based around Darwinian principles, working on a population of potential solutions to a problem. Every individual in the population represents a particular solution to the problem, often expressed in binary code. The population is evolved over a series of generations to produce better solutions to the problem. The general multiple objective optimisation problem is described as [5]:

$$min\{f_1(\mathbf{x}) = z_1, ..., f_j(\mathbf{x}) = z_j\} \tag{13}$$

where

$$\mathbf{x} \in D, \tag{14}$$

The solution of $\mathbf{x} = [x_1, ..., x_i]$ is a vector of *decision variables*, and D is the set of feasible solutions. If each decision variable takes discrete values from a finite set, then the problem is combinatorial. The image of solution $\mathbf{x}$ in the objective space is a *point*

$$\mathbf{z}^x = \lfloor z_1^x, ..., z_j^x \rfloor = \mathbf{f}(\mathbf{x}), \tag{15}$$

such that

$$z_j^x = f_j(\mathbf{x}), j = 1, ..., J. \tag{16}$$

Point $\mathbf{z}^1$ *dominates* $\mathbf{z}^2$, $\mathbf{z}^1 \succ \mathbf{z}^2$, if $\forall j$ $z_j^1 \leq z_j^2$ and $z_j^1 < z_j^2$ for at least one j. Solution $\mathbf{x}^1$ dominates $\mathbf{x}^2$ if the image of $\mathbf{x}^1$ dominates the image of $\mathbf{x}^2$. A solution $\mathbf{x} \in D$ is *efficient (Pareto-optimal)* if there is no $\mathbf{x}' \in D$ that dominates $\mathbf{x}$. The point which is an image of an efficient solution is *nondominated*. The set of all efficient solutions is called the efficient set. The image of the efficient set in the objective space is called the *nondominated set* or *Pareto front*. An *approximation to the nondominated set* is a set A of points (and corresponding solutions) such that $\neg\exists \mathbf{z}^1, \mathbf{z}^2 \in A$ such that $\mathbf{z}^1 \succ \mathbf{z}^2$, that is set A is composed of mutually nondominated points. The point $\mathbf{z}^*$ composed of the best attainable objective function values is called the *ideal point*. At every generational step, each individual of the population is run on the hardware, and its performance evaluated and ranked via a cost function. Individual performance is

indicated by a fitness value, an expression of the solution's suitability in the solution of the problem. The relative degree of the fitness value determines the level of propagation of the individual's genes to the next generation. In the multi-objective evolutionary algorithm (MOGA) in use here, the rank of a certain individual corresponds to the number of individuals in the current population by which it is dominated. All nondominated individuals are assigned rank 1, while dominated ones are penalized according to the population density of the corresponding region of the tradeoff surface. Fitness assignment is performed as follows [6];

- Sort population according to rank.
- Assign fitness to individuals by interpolating from the best (rank 1) to the worst (rank $n \leq M$), the *Pareto ranking assignment process* [6], according to a (usually) linear function.
- Average the fitness of individuals with the same rank, so that all of them will be sampled at the same rate. This keeps the global population fitness constant while maintaining appropriate selective pressure.

Evolution is subsequently performed by a set of genetic operators which stochastically manipulate the genetic code. Most genetic algorithms include operators which select individuals for mating, and produce a new generation of individuals. *Crossover* and *Mutation* are two well-used operators. The crossover operator exchanges genetic material between parental chromosomes to produce offspring with new genetic code. The mutation operator makes small random changes to a chromosome. Trade-offs occur between competing objectives with the consequence that it is very rare to find a single solution to a particular problem. In reality a family of *non-dominated* solutions will exist. These *Pareto-optimal* [7,8] solutions are those for which no other solution can be found which improves on a particular objective without a detrimental effect on one or more competing objectives. The designer then has the opportunity to select an appropriate compromise solution from the trade-off family based on a subjective engineering knowledge of the required performance. Individuals which represent candidate solutions to the optimisation problem are encoded as either binary or real number strings, producing an initial population of chromosomes by randomly generating these strings. The population of individuals is evaluated using an objective function which characterises the individual's performance in the problem domain. The experimental system is run iteratively with each individual's set of parameters. The objective function determines how well each individual performs based on experimental data, and is used as the basis for selection via the assignment of a fitness value.

3 Gas turbine engine model

The system under consideration here is a Rolls-Royce Spey Mk202 engine model which has been supplied in the form of a *Simulink* S-function. The vector of primary control inputs are shown in table 1 These inputs are *demanded* values, since the engine model includes representations of the actuator dynamics. There is also a vector of environmental inputs shown in table 2 Of the 11 input variables, 8 can be considered as independent variables while the remaining three are scheduled as a function of HP spool speed. Although the only direct control available is the main engine fuel flow assisted by the reheat fuel flow rate, the model must be excited and adequately populated in all 8 dimensions to accurately produce the metamodel. It can at the moment be assumed that the internal model states which are scheduled will also be adequately excited. This at the moment is not fully answered and represents a separate additional field of study. The engine model constructs a vector of 11 outputs shown in table 3 The model also includes a number of time varying parameters. Of prime consideration is *Heat soak* which refers to slow transients of small but significant magnitude associated with the transfer of heat between the gas streams through the engine and the solid material of the engine's turbine blades, casing etc. These cause changes in the mechanical clearances which temporarily peturb the efficiency and flow capacities of the turbines from their steady-state values. The final input signal is a vector of deterioration components which allow the effects of deviations of the model from its nominal condition to be simulated (table 4)

3.1 Discussion

For true functionality of the metamodel, an approximation must be made to the excited system in 2 primary inputs, 6 environmental inputs, 1 heat soak variable and 12 deterioration variables, making a total of 21 variable dimensions leading to an 11 dimensional output. This level of detail and accuracy

Name	Units	Notes
Main engine fuel flow	kg/s	Range: 0.1 to 1.5 kg/s
Reheat fuel flow	kg/s	Range: 0.0 to 6.0 kg/s
Sine of the nozzle petal angle	Dimensionless	Scheduled with HP spool speed
HP compressor guide vane angle	Degrees	Scheduled with HP spool speed
HP compressor handling bleed valve position	%	Scheduled with HP spool speed

Table 1: Engine primary input vector

Name	Units	Notes
Altitude	km	Full flight envelope
Flight mach no.	Dimensionless	Full flight envelope
Temperature deviation from standard	K	Full flight envelope
Intake pressure loss factor	Dimensionless	Range: 1 to 0.95
Cooling bleed	kg/s	Range: 0 to 0.3 kg/s
LP shaft power offtake	kW	Range 0 to 100 kW

Table 2: Vector of environmental inputs

Name	Description	Typical value (for nh=90%)
nl (%)	LP spool speed	79
nh (%)	HP spool speed	90
t2c (K)	HP compressor inlet temperature	368
dp2b (kPa)	Bypass duct inlet differential pressure	23
t2b (K)	Bypass duct inlet total temperature	368
p3 (kPa)	HP compressor exit total pressure	1437
t3 (K)	HP compressor exit temperature	667
t6 (K)	Core exit mean temperature	852
dp3 (kPa)	HP compressor exit differential pressure	74
ps4 (kPa)	HPT stage 2 static pressure	495
Te (kN)	Engine thrust	

Table 3: Model output vector

Description	Suggested value
LP compressor efficiency (inner part)	-.01
LP compressor efficiency (outer part)	-.01
HP compressor efficiency	-.01
HP turbine efficiency	-.01
LP turbine efficiency	-.01
LP compressor flow function	-.05
HP compressor flow function	-.04
HP turbine flow function	-.01
LP turbine flow function	-.03
HP shaft power offtake	0 to 150kW
Cooling bleed no. 11	0 to -.3 kg/s
Combustion efficiency	-.002

Table 4: Vector of deterioration components

is the culmination of intense modelling developments along with elements of *ad hoc* tuning to fine tune the underlying differential equations and non linear elements which describe the system. It is desired to design a *fast and efficient* algorithm to convert such a model into and accurate and efficient real-time representation. The first salient feature is the sheer complexity in terms of dimensions of the original model. Secondly it is obvious that the original model must be excited in all these dimensions in order to provide a good spread of data to which a new model can be fitted. What is not obvious however is a method of determining the number, position or spread of excitations (*experiments*) in each of the dimensions. At the very least, some form of optimisation must be applied to keep the number of experiments down to a manageable level. Even if each of the dimensions and the interactions were linear, a minimum of 2^{21} experiments would have to be conducted and in the non linear case, no sampling theorem exists. This makes the application of designed experiments [4] unusable due to the complexity and non linearity of the system. Consequently the *stochastic* optimal search characteristics of Evolutionary algorithms will be applied to the task of optimising the experimental set.

4 Polynomial metamodelling

In order to assess the potential of this approach to metamodelling, we will consider initially the variables *altitude* (km) and *flight mach number*. The input variable of interest is the *main engine fuel flow* and the output variable is *main engine thrust*. A *factorial* approach[4] to populating the three dimensional response surface would be to quantize altitude from 0 to 10 km in steps of (say) 2km and to quantize the flight mach number from 0 to 1 in steps of (say) 0.2. This gives a total of 6^2 experiments and without *a priori* knowledge of the system response, there is no method initially of knowing whether the model will be over or under fitted. It is generally a method of trial and error to ascertain the structure of the response surface which is fitted to the experiments. We will first examine the factorial approach in order to later assess the performance of the automatically designed model.

4.1 Factorial modelling

The factorial approach described in the previous section was applied to the Spey gas turbine model. The thirty six operating points were excited by zero mean white noise excitations (a total of 36 experiments). A polynomial fit to the to the response was performed by applying an Auto Regressive Exogenous (ARX) model which is a linear difference equation that relates the input $u(t)$ to the output $y(t)$ as follows:

$$y(t) + a_1 y(t-1) + ... + a_n a y(t-na) = b_1 u(t-nk) + ... + b_n b u(t-nk-nb+1) \tag{17}$$

The structure is thus entirely defined by the three integers na, nb, and nk. na is the number of poles, $nb+1$ is the number of zeros, and nk is the pure time delay (the dead time) in the system. For a sampled data control system, typically $nk = 1$ if there is no dead time.For multi-input systems nb and nk are row vectors, where the *i'th* element gives the order/delay associated with the *i'th* input. The output of this procedure is a discrete-time model of the form

$$A(q)y(t) = B(q)u(t) + e(t) \tag{18}$$

For example with the model running at a height of $2km$ and a speed of $mach0.2$, then a polynomial fit of the form

$$\begin{aligned} A(q) = 1 \quad &- \quad 0.3637(+-0.05092)q^{-1} + 0.1236(+-0.006121)q^{-2} \\ &- \quad 0.06397(+-0.006127)q^{-3} + 0.01066(+-0.001762)q^{-4} \end{aligned} \tag{19}$$

$$\begin{aligned} B(q) \quad &= \quad 11.26(+-0.736)q^{-1} + 11.26(+-0.736)q^{-2} \\ &+ \quad 2.11(+-0.08076)q^{-3} - 0.7587(+-0.145)q^{-4} \end{aligned} \tag{20}$$

is obtained. The system can now be characterised according to the polynomial coefficients and be described by a response surface. In order to prepare the data for this fitting method, the natural units ξ_1 (altitude in km) and ξ_2 (flight mach number) of the experimental data is first transformed into the corresponding normalised coded variables x_1 and x_2, such that

$$x_{i1} = \frac{\xi_{i1} - [max(\xi_{i1}) + min(\xi_{i1})]/2}{[max(\xi_{i1}) - min(\xi_{i1})]/2} \tag{21}$$

and

$$x_{i2} = \frac{\xi_{i2} - [max(\xi_{i2}) + min(\xi_{i2})]/2}{[max(\xi_{i2}) - min(\xi_{i2})]/2} \tag{22}$$

A response surface model is fitted to each of the polynomials such that the second order model (order and structure ascertained by trial and error) to be fitted to the data is

$$\begin{aligned} y &= \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_{11} x_1^2 \\ &+ \beta_{22} x_2^2 + \beta_{12} x_1 x_2 + \epsilon \end{aligned} \tag{23}$$

It is now possible to compare the response of the polynomial based model with the original *simulink* model. The system was excited via a zero mean white noise demand into the main fuel flow (figure 1) During

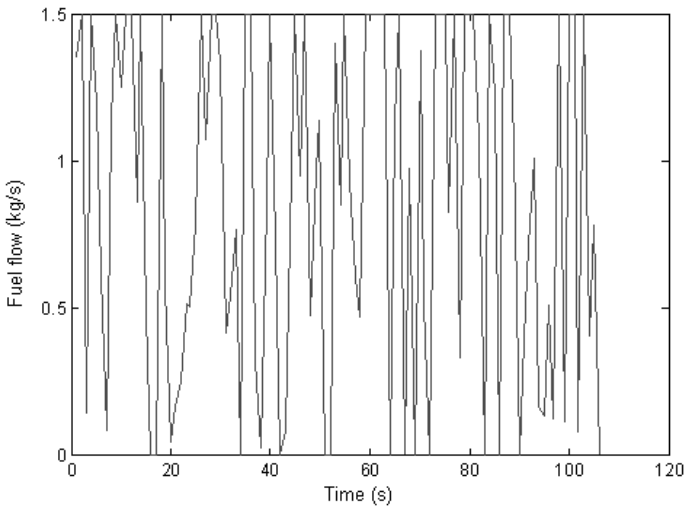

Figure 1: White noise excitation of the main engine fuel flow

the simulation, the altitude was allowed to ramp linearly from 0 to 2km, and the airspeed was allowed to ramp linearly from 0 to mach 0.5. The resulting output is shown in figure 2. With thrust varying between 25 and $65kN$, the output error is bounded by a maximum range of $0.4kN$, a value which was repeated during similar experiments across the operating range. The polynomial metamodelling approach using response surfaces fitted by least squares appears to show considerable promise in producing a compact and elegant fast model. However several steps in the modelling procedure such as the choice of response surface models and the quantisation of the factorial experiments is at best performed by trial and error. In order to derive a tractable approach to metamodelling these complex systems, a more generic and optimised approach will be investigated with the aid of evolutionary algorithms.

5 An evolutionary approach to experimental design

A multiobjective evolutionary algorithm was first applied to the task of optimising the number of factorial experiments involved in the metamodelling of the engine in terms of airspeed and altitude. A two objective problem was posed, to maximise the accuracy of the model, while at the same time minimising the number of experiments conducted.

5.1 Evaluation of model accuracy by the objective function

An autocorrelation function was introduced as one of the elements in the objective function to minimise model error. Whether the shift operator λ is forwards or backwards, the auto-correlation [9] function is introduced as

$$\phi_{uu}(\lambda) = \frac{1}{T} \int_0^{0+T} u(t-\lambda) u(t) dt \tag{24}$$

Thus the auto-correlation is the time average of the product of the signal at a specified time and its value at a time shifted by λ seconds. It gives a measure of how much the value of u at time t depends on the value of time at $t - \lambda$

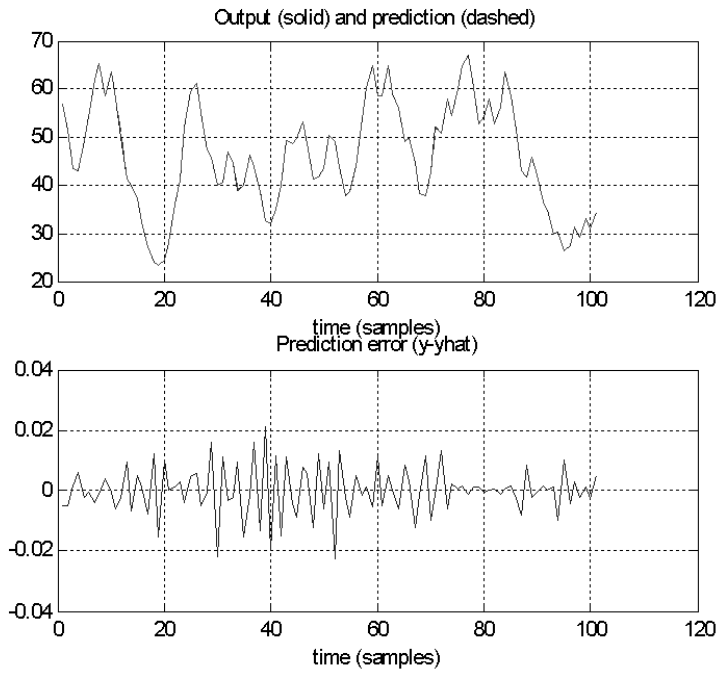

Figure 2: Comparison of predicted and actual thrust with a measurement of prediction error.

The cross-correlation function is a measure of the correlation existing between two functions as

$$\phi_{uy}(\lambda) = \frac{1}{T}\int_0^{0+T} u(t-\lambda)y(t)dt \tag{25}$$

giving a measure of how much the value of y at time t depends on the value of u at time $t-\lambda$.

In theory, in order to investigate the accuracy of the model approximation, one should check if the residuals are uncorrelated with all nonlinear and linear combinations of past data. However in practice, such a test is unrealistic. Norgaard [10] suggests (as is adopted here) a small set of functions for auto-correlation between residuals, and cross correlation functions. The correlation functions are supported by a function seeking to minimise the sum of the residuals with respect to time. This the part of the objective function which seeks to minimise model error can be decomposed into three sub-components, namely 99% confidence regions around zero limits for auto-correlation and cross-correlation functions, and minimisation of sum of residuals.

5.2 Minimisation of model complexity by objective function

Application of evolutionary algorithms to the objective of minimising the model complexity attempts to produce an accurate well behaved model with a smaller population than the 36 produced under the factorial approach. The objective function is implemented into the evolutionary algorithm (EA) such that it seeks to minimise the number of elements n in the $2n$ matrix which specifies the number of decision variables representing the experiments undertaken to characterise the system.

5.3 Selection of response surface model order and structure

Once the optimal number and position of experiments has been identified, the response surface model order and structure is selected by EA. Decision variables were defined as the model order (up to 5th order, the number of interactions between the variables, and the placement of interactions. The objective function seeks to minimise the sum of squared errors between the identified response surface and a randomly chosen verification set.The algorithm also tries to achieve the minimum model order and interaction numer possible.

The automatic algorithm thus takes the original gas turbine model, searches for the most accurate metamodel which is constructed from the minimum number of experiments. A response surface which has been optimised in terms of model order and parameters is then fitted to this data.

6 RESULTS

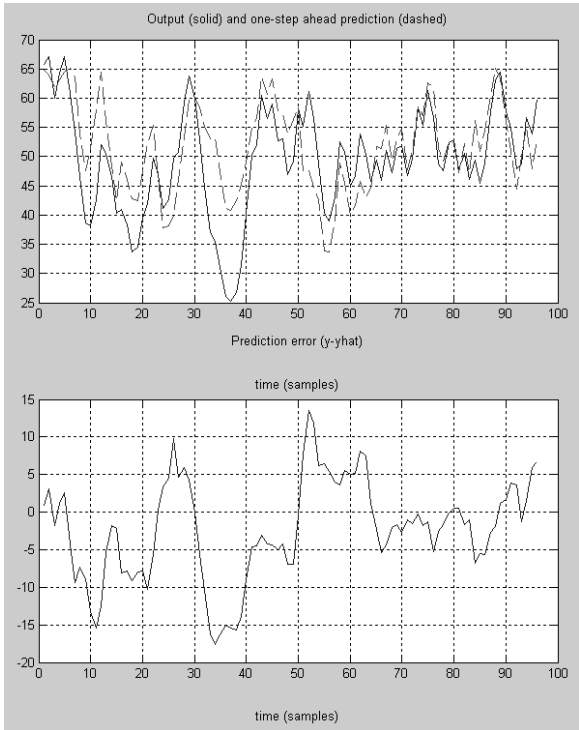

Figure 3: Actual and predicted thrust response from unsuccessful model approximation

Figure 3 shows the output of the metamodel excited by white noise demand to the main engine fuel flow, rising from 2 to $4km$ altitude and from mach 0.3 to 0.6. This was an unsuccessful metamodel chosen to illustrate an underfitted model. That is, the product of only 5 experiments to characterise the system. The sum of squared error is extremely large and is consequently unsatisfactory. A tradeoff is made between the number of experiments conducted and the accuracy of the metamodel. Figure 4 Shows the auto-correlation and cross-correlation output of the successful candidate chosen for the best balance between model accuracy and model complexity. The 99% confidence level bounds are shown. The predicted and actual output of the turbine model are shown in figure 5. Again, the output of the metamodel is excited by white noise demand to the main engine fuel flow, rising from 2 to $4km$ altitude and from mach 0.3 to 0.6. The number of experiments to characterise the system had fallen to 9 in this case.

7 Conclusions

An initial investigation has been conducted to assess the feasibility of multidimensional metamodelling for multivariable complex systems. By applying evolutionary algorithms to the task of minimising the number and position in the searchspace of experiments to characterise the system, a minimum number is found, subject to constraints on model accuracy. Further, an evolutionary algorithm searches for the optimal response surface suject to the constraint of minimising complexity. A simple polynomial metamodel can then be derived which characterises the system and is computationally extremely economical. The derivation of the simple, accurate metamodel has to some extent been automated. The task ahead is to extend this approach to encompass all the other variable in the system to derive a full multivariable metamodel. Work is currently in progress using neural networks to learn the effects of the deterioration components and the effects of heatsoak on the multivariable metamodel

Acknowledgments

The author would like to acknowledge the Department of Trade and Industry (DTI) CARAD programme, and the Ministry of Defence's (MOD) Corporate Research Programme for funding.

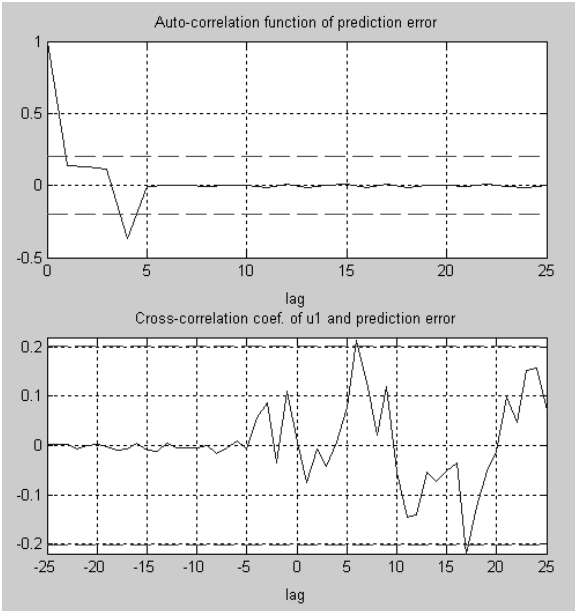

Figure 4: Auto-correlation and cross-correlation output of the successful candidate.

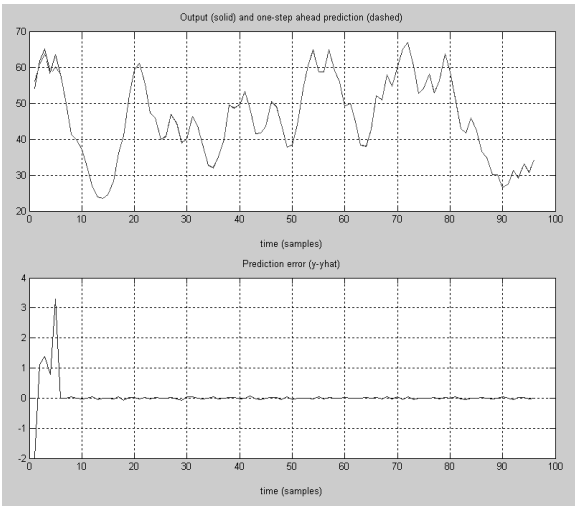

Figure 5: Actual model output compared to metamodel predicted output.

References

[1] Myers R.H., and Montgomery D.C., (1995) *"Response surface methodology: process and product optimization using designed experiments"*, John Wiley and Sons Inc. USA, ISBN 0-471-58100-3.

[2] Stewart P and Fleming P., *"The Response Surface Methodology for real-time distributed simulation"*, IFAC Conference on New Technologies for Computer Control, Hong Kong, China. November 19-22, pp.128-133, 2001.

[3] Stewart P and Fleming P.J., *"The Response Surface Methodology for rapid prototyping of real-time control systems"* 2002 IEEE American Control Conference, Anchorage, Alaska, USA. Pp.3343-3348, May 8-10, 2002.

[4] Hicks C.R. and Turner K.V., *"Fundamental concepts in the design of experiments"* Oxford University Press, New York. ISBN 0-19-512273-9.

[5] Jaszkiewicz A., Hapke M. and Kominek P., *Performance of multiobjective evolutionary algorithms*, Evolutionary Multi-criterion Optimsation, Zitzler E. (Ed.), pp.241-255, ISBN 3-540-41745-1, Springer-Verlag, 2001.

[6] Fonseca C.M. and Fleming P.J., *Genetic algorithms for multiobjective optimisation: formulation, discussion and generalisation* In Stephanie Forrest, editor, Proceedings of the Fifth International Conference on Genetic Algorithms, pp.416-423, San Mateo, California, 1993. University of Illinois at Urbana-Champaign, Morgan Kauffman Publishers.

[7] Fonseca C.M. and Fleming P.J., *An overview of evolutionary algorithms in multiobjective optimisation* Evolutionary Computation, vol.3(1), pp.1-16, 1995,.

[8] Fonseca C.M. and Fleming P.J., *Multiobjective optimisation and multiple constraint handling with evolutionary algorithms - Part 1: A unified formulation and Part 2: Application example* IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans, vol.28(1), pp.26-37 and 38-47, 1998.

[9] Anthony M. and Bartlett P.L., *Neural network learning: Theoretical foundations* Cambridge University Press, Cambridge, UK, 1999.

[10] Norgaard M., Ravn O., Poulsen N.K. and Hansen L.K., *Neural networks for modelling and control of dynamic systems* Springer-Verlag London Ltd., UK, 2000.