IBPSA-ENGLAND BSO12

First Building Simulation and Optimization Conference
Loughborough, UK
10-11 September 2012

ACCELERATION OF BUILDING DESIGN OPTIMISATION THROUGH THE USE OF KRIGING SURROGATE MODELS

Es Tresidder[1], Yi Zhang[1], and Alexander I. J. Forrester[2]
[1]Institute of Energy and Sustainable Development, De Montfort University, Leicester, United Kingdom
[2]School of Engineering Sciences, University of Southampton, United Kingdom

ABSTRACT

This paper describes an experiment to test the performance of Kriging surrogate modelling optimisation techniques on a building design problem with discrete design choices. Surrogate modelling optimisation offers advantages over traditional optimisation techniques on design problems with expensive (time consuming) performance evaluation models. The techniques are tested for both single and multi-objective optimisation problems with the objective of minimising both annual CO_2 emissions predicted by a dynamic simulation and construction cost. The estimated CO_2 emissions and costs of all possible designs were first established through comprehensive analysis using a multi-processor computer, enabling the performance of the optimisation to be assessed precisely against a known single optimum or Pareto front. The performance is compared against an evolutionary algorithm (EA) searching the dynamic simulation model on the same design problem. The results show that for this design problem, Kriging surrogate modelling optimisation is effective at finding estimates of optimum designs. In the case of the single-objective optimisation it is able to find the optimum in fewer simulation calls than the stand-alone EA. In the case of the multi-objective optimisation it is capable of finding a better Pareto front if the total number of simulations is restricted, although the time cost associated with Kriging does not always mean it is worth using.

INTRODUCTION

The design of low energy buildings is a non-linear optimisation problem. Many variables interact with each other, meaning the optimum assignation for each variable depends partly on the value chosen for other variables. The best combination of variables will also depend on the use intended for the building, the climatic zone in which it is located, and site specific conditions such as shading and exposure to wind. The situation is further complicated by the addition of cost minimisation as a second objective. Many different variables are important to the energy performance, but each additional variable taken into consideration makes the set of all possible designs (the design space) exponentially larger.

The time required to estimate energy performance using a dynamic simulation model may mean that the total simulation budget is severely limited. With many thousands of possible designs, and a limited simulation budget, efficient methods of searching the design space become especially valuable.

Several methods for exploring this design space have been proposed to try and find optimal building designs (Verbeeck and Hens, 2007; Coley and Schukat, 2002). One of the most promising of these, evolutionary algorithms (EAs) use Darwinian concepts of evolution by natural selection to improve a population of building designs. While this method is effective at finding optimum designs, even with complex, multi-modal and discontinuous design problems (Hasan et al., 2008; Jin, 2005), 'convergence' on an estimated optimum design or Pareto front can still take many hundreds or even thousands of design samples. This may be infeasible if each sample is a time-consuming dynamic simulation.

Various approaches have been taken to tackle this problem of limited time, a vast design space and time-consuming performance estimation. These have included 'tuning' the various parameters of the evolutionary algorithm (Wright and Alamji, 2005), reducing the complexity of the building models to allow more simulations to be run (Wang et al, 2005) and seeding the starting population with known good designs (Hamdy et al., 2011).

Another approach to this problem is to build a *surrogate model* after an initial sample of the dynamic simulation model (the main model), and use the EA to repeatedly search the surrogate model instead of the main model. After each search of the surrogate model the design suggested is checked on the main model. The surrogate may be much faster to interrogate than the main simulation model, meaning that more interogations can be made in the available time. Surrogate models typically incur a time cost of their own (they are time consuming to build), but the reduction in time due to fewer main-model evalutions can more than make up for this if the main model is time consuming to interrogate (Brownlee et al, 2010). The use of surrogate models has been suggested for building design optimisation (Wetter and Wright, 2004).

This paper tests the efficacy of one sort of surrogate model, the Kriging model, for use in optimising cost-effective, low-energy building designs. Kriging optimisation has been shown to be effective in other engineering desgn problems (Huang et al., 2011; Forrester et al., 2007 and 2008). However, most of these examples have been on design problems with continuous variables (De Guido et al, 2011). This paper tests the effectiveness of Kriging optimisation on a single and multi-objective, highly discrete, building design problem. Its performance is compared to that of EAs searching the main model directly, with the performance measure being the total number of main-model simulations required to find the known optimum (for the single-objective optimisation), or the quality of the estimate of the Pareto front after 200 main-model simulations.

SURROGATE MODELS AND KRIGING

After first sampling the main model, the output can be approximated using a surrogate model. Instead of making calls to the building model for all results, we can fill in the gaps between the set of observed data by fitting a surrogate model, i.e. a model which can be used in lieu of direct calls to the building model by the EA. Such models are, essentially, curve fits through known data and may take many forms, including polynomial regression (Box and Draper, 1987), neural networks, radial basis functions (Broomhead and Lowe, 1988), support vector regression (Smola and Schölkopf, 2004) and the method we concentrate on here: Kriging (named after its developer, Danie Krige, see, e.g. Sachs et al., 1989). A review of surrogate models of fitness functions in evolutionary computation is made by Jin, 2005.

As an example of how powerful the Kriging method can be, Figure 1 shows the contours of the Branin function (an analytic function often used to test optimisation algorithms) and a Kriging prediction based on 20 observations. The similarity between the two contour maps is remarkable; the surrogate has captured all the key features of the true Branin function. Clearly, if the true function is time-consuming to compute, an optimisation process could save time and resources by calling on the surrogate model for data rather than the true function.

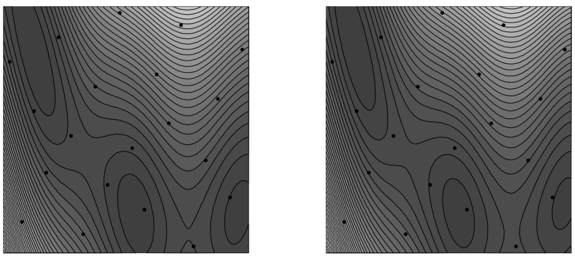

Figure 1. contours of the true Branin function (left) compared with a Kriging prediction (right) based on 20 sample points (dots) (from Forrester and Keane [2008]).

EXPERIMENT DESIGN

Building model

A building model was built using DesignBuilder[1] and exported to run in EnergyPlus[2]. The building model was of a simple rectangular residential building, air-change rates were fixed and solar reflections not modelled in order to keep calculation times as short as possible. This allowed comprehensive analysis of all solutions so that the performance of the optimisation methods could be compared against known optima. The design problem is described in more detail in a previous paper by the authors (Tresidder et al., 2011). Each analysis of the model took around 50 seconds to run in EnergyPlus. Using such a simple model was judged to be beneficial because the aim of the paper was to compare optimisation techniques, rather than to answer specific building-design questions, and this comparison is much more accurate if the true optimum is known (Hawe and Sykulski, 2007). Ten design variables were chosen with three discrete choices available for each one.

Discrete variables

The design variables used in this study were discrete, with three choices for each one. The Kriging model, on the other hand, assumes that variables are continuous with values ranging from 0 to 1. To turn the designs suggested by the EA on the Kriging model into designs that could be tested on the main simulation model the following simple formula was used;
If the value is below 1/3 choose option 1.
If the value is between 1/3 and 2/3 choose option 2.
If the value is above 2/3 choose option 3.

Objective functions

The objective function for the single-objective optimisation was the total CO_2 emissions from energy used to heat, light and cool the building as described in Tresidder et al. (2011). The multi-objective optimisation also included construction-cost minimisation as an objective. The cost model was relatively simplistic and included only material costs. These costs were based on cost estimates provided by industry.

Kriging surrogate modelling

The surrogate modelling routines used here are from the freely available Matlab toolbox that accompanies Forester et al. (2008).
The initial set of observed data is chosen according to a sampling routine based on evolutionary programming that searches for the Latin hypercube design which maximizes the minimum distance between sample locations, i.e. the set of experiments

[1] http://www.designbuilder.co.uk/
[2] http://apps1.eere.energy.gov/buildings/energyplus/

that most evenly cover the design space (Morris and Mitchell, 1995).

Based on these samples a Kriging model is built, with its hyper-parameters chosen by maximising the likelihood of the data[3] using the EA provided in the Kriging package[4]. The Kriging model can then be used to predict the expected improvement (EI) at un-sampled points in the design space. An EA is then used to search the Kriging model for designs with a high EI. After each run of the EA on the Kriging model the suggested design is tested on the main-model and the information gained is incorporated in the Kriging model for subsequent optimisations. In this way, areas of the model that erroneously suggest promising designs are not allowed to distort the optimisation process.

The process of Kriging model optimisation used in this study is summarised in Figure 2 opposite.

Single-objective optimisation algorithms and testing

The algorithm used to search the Kriging model can instead be used to search for better designs directly on the main simulation model. In this way, it was possible to construct a like-for-like test in which, starting with the same initial sample population, the EA was used to search for optimum designs either with or without the surrogate model in place.

The single-objective optimisations with a surrogate model were run with starting sample sizes of 5, 10, 20 and 50. The algorithm was then set to run until it found the known optimum. From the same starting populations, optimisations were run without the surrogate model on population sizes that equalled the starting population size. For each sample/population size a total of 10 optimisations were run to establish the average performance of the algorithm. The performance of the two processes was compared based on the total number of main model simulations required to find the true optimum design.

Multi-objective optimisation algorithms and testing

The expected improvement used in Kriging single-objective optimisation can be reformulated to work with more than one objective. The EI becomes an estimate of the probability that a design suggested by the EA will improve on the current pareto front. Combining more than one objective into a single objective can be problematic, since it requires the user to pre-assign relative importance to each objective (Wang et al. 2005). However, this is not the case with a multi-objective EI, meaning we can keep a single-objective algorithm without pre-assigning weightings to each objective.

Rather than combining the two objectives (and thus having to assign weightings to each objective) to enable the same EA to be used *without* the surrogate model, it was decided to compare the surrogate optimisation method against a well established multi-objective EA.

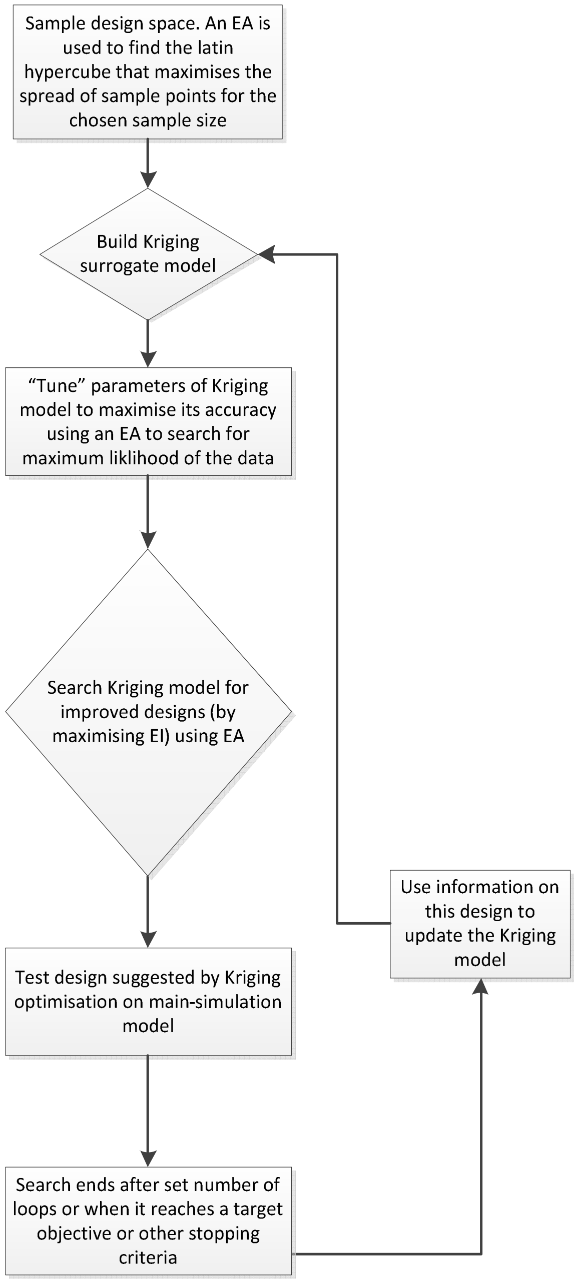

Figure 2. A summary of the process of Kriging model optimisation used in this study

The algorithm jEPlus+EA, an efficient, multi-objective EA that uses a modified version of the well known NSGAII algorithm (Deb et al. 2002) was used as the benchmark against which the performance of Kriging optimisation was compared. jEPlus+EA has previously shown very good performance (Zhang, 2012) on building optimisation problems. The

---

[3] "Maximising the data" means trying to find the Kriging model that maximises the likelihood that the known data points (those sampled from the main simulation model) could have come from the Kriging model.

[4] Forrester et al. 2008

modifications made to the standard NSGAII algorithm were to preserve all members of the current estimate of the Pareto front from one generation to the next (meaning the population tends to grow as the generations progress), to use integer encoding and to avoid duplicate designs.

Settings for the jEPlus+EA were left at the default values present in the package, and the settings for the surrogate modelling optimisation were left as described in Tresidder et al. 2011, apart from the encoding length, which was altered as described below.

For the surrogate model optimisation the total simulation budget was set at 200 samples. Of this budget, 60 samples were used to build the initial Kriging model, with a further sample taken after each run of the EA on the Kriging model for a further 140 samples (to make a total of 200 samples).

The optimisations were each run 10 times with a different random seed. This allowed an estimate of the average performance of the algorithm on this design problem.

Because Kriging assumes that variables are continuous, when it is used on a discrete design problem many points with a high expected improvement will in fact turn out to be duplicates of previously sampled designs. A database was built while the optimisation ran which stored information on all previously sampled designs. This could be checked for duplicates before calling the main model. Several ideas for how the performance of Kriging optimisation might be improved on discrete design problems were tested based on trying to reduce the number of duplicate sample points. These were:

- Discarding duplicate samples. If the EA suggests a design that has in fact already been sampled this sample point is not included in the updated Kriging model and the EA is told to start searching over again.
- Including duplicate samples. If the EA suggests a design that has already been sampled then the performance of this design is called from memory and included in the Kriging model. The total number of unique samples (and therefore simulation calls) is kept to 200.
- Reducing the encoding length. The default encoding length in the Kriging modelling package is 20bits, encoding over 10,000 possible numbers, but since there are only three different designs this means that many genetic changes will not result in phenotypic changes (leading to duplicates). Encoding lengths of 2bits and 3bits were also tested in addition to the default encoding length.

The performance of the multi-objective algorithms was compared based on several metrics. Because the true Pareto front was known, the first metric was the mean Euclidian distance between the estimated Pareto points and the nearest true Pareto point. The second metric was the standard deviation in the mean Euclidian distance for each test. This gives an idea of the variability in distance from the pareto front. For both of these metrics a lower number was considered superior. The final metric was the number of unique designs on the estimated Pareto front. More designs on the Pareto front was held to be superior.

RESULTS AND ANALYSIS

Single-objective optimisation

With the same starting population, one optimisation with a Kriging surrogate model and one without can be considered a pair. In the case of every pair tested, for all different population sizes, the Kriging surrogate optimisation always found the optimum design before the optimisation without the surrogate model. The average performance of the optimisations (from 10 runs of each) is shown in Table 1, below.

Initial sample/ population size	Average number of main-model samples required to find optimum. Mean (± SD)		Two-sample t-test, assuming unequal variance, P (T<=t) two-tail
	Without Kriging model	With Kriging model	
5	1154 (±887)	84 (±38)	0.0013**
10	1325(±1612)	68 (±36)	0.024*
20	584 (±348)	88 (±59)	0.0003**
50	625 (±409)	100 (±45)	0.0008**

*Table 1. A summary of results for the single-objective optimisation, comparing the number of main-model samples required to find the optimum for an evolutionary algorithm with and without a Kriging surrogate model.(*P<0.05; **P<0.01).*

The differences observed between the performances of the two optimisation methods were significant at the 99% confidence level for all except the initial sample size of 10, which was significant at the 95% confidence level.

Furthermore, comparing the performance of the best performing Kriging optimisation (initial sample size of 10) with the best performing non-Kriging optimisation (population size of 20) the difference between these two populations is significant at the 99.9% confidence level.

However, knowing simply that one methodology is able to find the optimum design faster than another doesn't necessarily mean it is a lot better – they might both get very close to the optimum design very quickly, and the user may not be concerned about the last fraction of a percent of improvement. It is perhaps better to evaluate the performance visually:

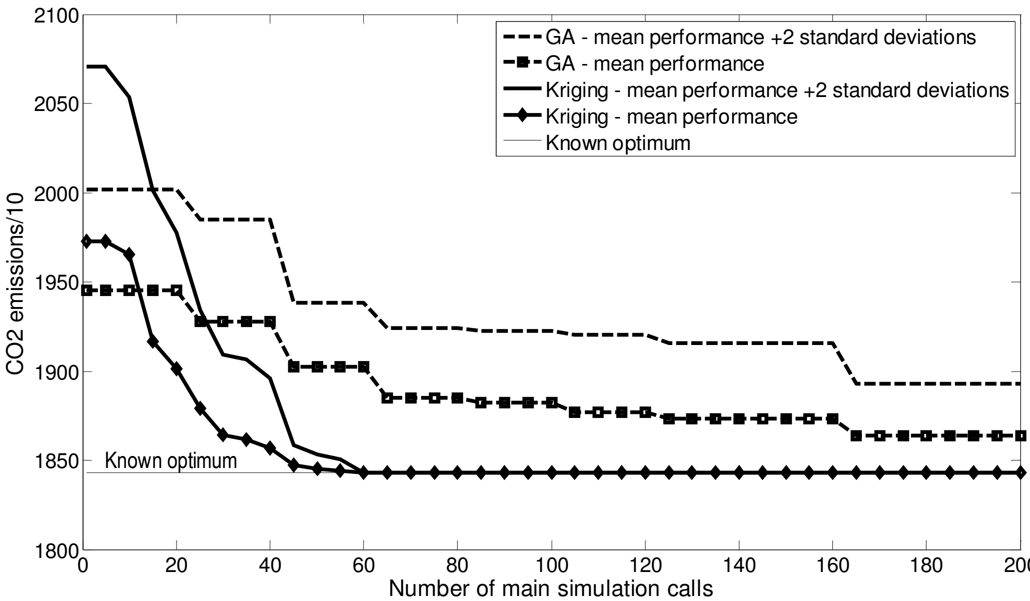

Figure 3. The progression of the mean and mean+2 standard deviations for optimisations using ga.m with and without a Kriging surrogate model. The known optimum point is shown as a horizontal line for comparison.

Figure 3, above, compares the performance of the best-performing Kriging optimisation (initial sample size of ten) with the best-performing stand-alone EA (population size of 20). The lines with markers are the average (mean) performance, the unmarked lines the mean + 2 standard deviations (i.e. approximately 95% of optimisations could be expected to perform better than this). It can be seen that in addition to finding the optimum design faster, the Kriging modelling optimisation is also considerably faster at getting close to the optimum. Based on these results, 95% of Kriging optimisations could be expected to outperform the mean performance of the EA operating on the main model.

Multi-objective optimisation

The performance of the different types of multi-objective optimisation are summarised in Table 2, opposite.

It can be seen that if the sampling budget is limited to 200 samples, the best performing of the Kriging optimisations is better, on average, on all metrics of performance than jEPlus+EA; the estimated Pareto points are closer to the true Pareto points, there is less variability in the distance of the estimated Pareto points from the true Pareto points and there are more designs on the estimated Pareto front. The improvement in performance based just on the Euclidian distance was tested using a T test and found to be significant with 99.9% confidence (P value = 0.0002).

Optimisation method	Euclidian distance Mean (±mean SD in Euclidian distances)	Ave number of Pareto solutions
Kriging, duplicates included, encoding 20-bit	7.4(±11.6)	29.5
Kriging, duplicates discarded, encoding 20-bit	9.3(±15.6)	30
Kriging, duplicates included, encoding 3-bit	4.6 (±9.5)	38.2
Kriging, duplicates included, encoding 2-bit	10.2 (±18)	26.3
jEPlus+EA	12.6(±11.2)	25.8

Table 2. Comparing the performance of all multi-objective optimisation algorithms tested with a sampling budget of 200 samples of the main simulation model. Algorithms that used a Kriging surrogate model are shaded in grey.

Similarly to the single-objective case, merely examining the results in a table doesn't give us all the information we want – we cannot see how well spread out the estimated Pareto designs are, or understand how close they are to the true Pareto front compared to the rest of the design space. In order to visualise the results, the performance of the median (fifth best) performing (in terms of mean Euclidian distance), Kriging and jEPlus+EA optimisations is plotted alongside all known designs and the known Pareto front (Figure 4).

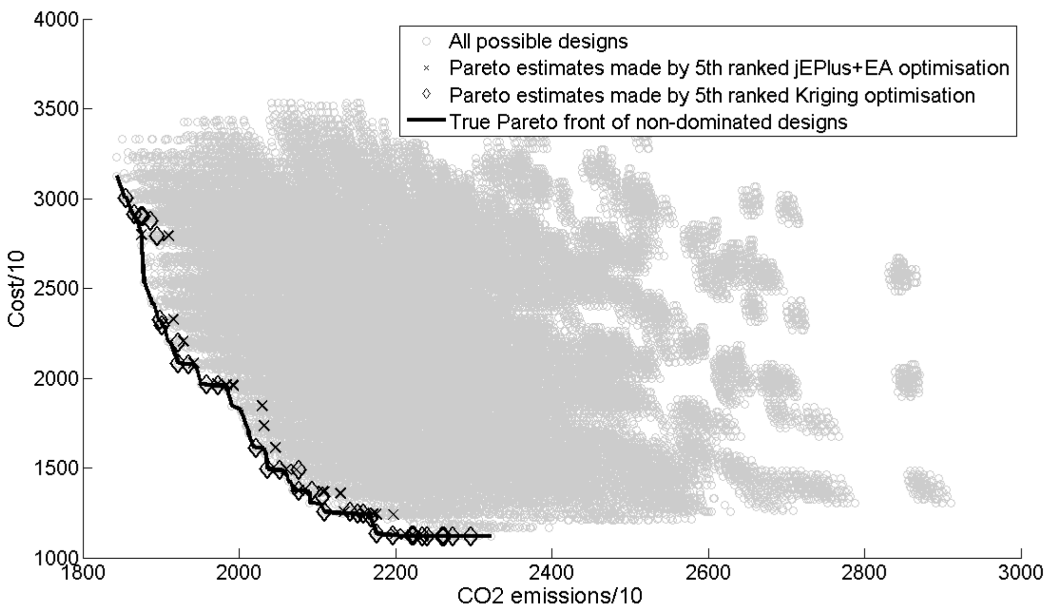

Figure 4. A comparison of the estimations made by the 5th best performing Kriging (diamonds) and jEPlus+EA (crosses) optimisation runs with a sampling budget limited to 200 runs of the main simulation model. The true Pareto front is marked with a solid line and all known designs are shown as light grey circles.

Examining Figure 4, it can be seen that not only do the Kriging estimations (marked with diamonds) tend to be closer to the true Pareto front than the jEPlus+EA estimations (marked with crosses), they are also cover a greater extent of the true Pareto front.

The performance of the Kriging model when duplicates were included was better than when they were excluded, and the performance was further improved by reducing the encoding length to 3 bits rather than 20 bits. However, these improvements were not solid enough to be statistically significant with 95% confidence. Shortening the encoding length further still reduced the performance on all metrics.

This calculation of which optimisation method performs best when the simulation budget is restricted is an over-simplification. Because of the time penalty associated with using the Kriging model it is useful to know how many samples are required for jEPlus+EA to match the performance of the Kriging optimisation, and then to calculate whether or not this means the Kriging model has been beneficial on this optimisation problem.

Because the performance is being measured on three different metrics, it is impossible to say exactly when the performance of one method equals or betters the other (Zitzler et al., 2004). However, looking only at the Euclidian distance metric, jEPlus+EA got as close to the known Pareto front as Kriging optimisation after on average 505 main model samples.

DISCUSSION

In the case of the single-objective optimisation, the addition of a Kriging model enabled the optimum design to be found with dramatically fewer samples of the main simulation model. Based on the performance of the algorithms on the single-objective design problem posed in this paper, the time cost associated with Kriging becomes worth "paying" if the main-model simulation takes more than 14 seconds. Since we know that the main-model took around 50 seconds to interrogate for the building design used in this study we can conclude that, for the parameters tested, Kriging offered an advantage in single-objective optimisations.

The performance advantage offered by Kriging in a multi-objective optimisation when tested against the established algorithm jEPlus+EA was much less dramatic. Kriging did offer an advantage in the case described if the sampling budget was limited to 200 samples. However, jEPlus+EA got equally close to the Pareto front in, on average, 2.5 times the number of main-model simulation calls (Table 2). With a time penalty of approximately 15 hours[5] for running

[5] With the following computer set-up: Intel E5530 CPU (2.4GHz), 16GB memory, SUSE Linux 64bit (Linux version 2.6.16), Matlab version R2011b.

the Kriging model on this optimisation problem, it would actually be faster to run jEPlus+EA for 500 or more main-model samples than to run the Kriging model for 200 main-model samples. For the parameters described here, on the multi-objective optimisation problem the time “cost” of Kriging is not worth paying.

If the relationship of jEPlus+EA requiring ~500 main model interrogations to equal the performance of Kriging after 200 main-model interrogations holds for more complex building models then Kriging would start to show an advantage for optimisations on designs with main-model simulations taking more than three minutes. However, a more complex design problem may show very different behaviour under the two optimisation methods than that shown in this study. The results shown here are encouraging that an advantage might be offered by Kriging on more complex building design optimisations, but more work is required before solid conclusions can be drawn.

Running jEPlus+EA for many more generations always resulted in a better Pareto estimate on all metrics than achieved by the Kriging optimisation after 200 main-model runs. If computing power allows, and the performance of the designs is very important, running jEPlus+EA for many generations, appears to offer a better estimate of the Pareto front.

The performance advantage offered by Kriging on optimisations with restricted main-model simulations was much greater for the single-objective optimisation than for the multi-objective optimisation. This suggests that jEPlus+EA performs considerably better than the standard EA provided with the Surrogate modelling optimisation package. It may be possible to improve the performance of Kriging optimisation by changing the EA that searches the Kriging model to include some of the characteristics of the EA used in jEPlus+EA.

Potential users of the surrogate modelling techniques described in this paper should also note that this method does suffer from limitations in terms of the number of variables that can be considered. These limitations tend not to be present when using an EA searching the main model. For each additional variable that is added the Kriging model becomes exponentially more complex (and therefore more time-consuming to build). The upper limit for the number of variables that can be included is was not examined in this paper, but it may be that it is not possible to use this method on many more than the 10 variables optimised in this study.

The improvement in performance made by reducing the encoding length from 20 bits to 3 bits requires more investigation, first to establish whether this result holds true generally, and if so to investigate exactly why it results in better performance.

CONCLUSIONS

Kriging surrogate modelling has been shown to work on building design problems with highly discrete design choices.

In the single-objective design problem posed in this paper use of the kriging surrogate model enabled the optimum design to be found with fewer calls to the main simulation model compared to using the same evolutionary algorithm with no surrogate model. For the single-objective optimisation the time penalty associated with using Kriging was more than compensated for by the reduced time required due to fewer main-model interrogations.

For the multi-objective design problem posed in this paper the use of a surrogate model allows a significantly better approximation of the Pareto front to be made if the number of calls to the main simulation is limited to 200. Using jEPlus+EA without a Kriging model, a Pareto front of similar quality took approximately 500 calls to the main simulation engine. However, with the building model used in this study the reduced number of calls to the main-model did not justify the increased time cost associated with using a Kriging model.

Further work is required to establish the likely reduction in main-model simulations offered by Kriging on more complex and time-consuming building models and whether this reduction justifies the increased time cost associated with building and interrogating the Kriging model.

References

Box E. P and Draper N. R (1987). Empirical Model Building and Response Surfaces. John Wiley & Sons.

Broomhead D. S and Lowe D (1988). Multivariate functional interpolation and adaptive networks. Complex Systems, 2:321–355.

Brownlee A. I, Regnier-Coudert O, Mccall J. A. W, Massie S (2010). Using a Markov network as a surrogate fitness function in a genetic algorithm. IEEE Congress on Evolutionary Computation, 1-8.

Coley D and Schukat S (2002). Low-energy design: combining computer-based optimisation and human judgement. Building and Environment 37(12):1241-1247

De Guido S, Geijselaers H. J. M and de Boer A (2011). Continuous-discrete variable optimization on composites using Kriging surrogate model. 16th international conference on composite structures.

Deb K, Pratap A, Agarwal S and Meyarivan T (2002). A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Transaction on Evolutionary Computation, 6(2), 181-197.

Forrester A. I. J, Sóbester A, Keane A. J (2007). Multi-fidelity optimization via surrogate modelling. Proc R Soc A 463(2088)):3251–3269

Forrester A. I. J, Sóbester A, and Keane A. J (2008). Engineering Design via Surrogate Modelling: A Practical Guide. John Wiley & Sons, Chichester,

Hamdy M, Hasan A and Siren K (2011) Applying a multi-objective optimization approach for Design of low-emission cost-effective dwellings. Building and Environment 46(1): 109-123

Hasan A, Vuolle M and Siren K (2008) Minimisation of life cycle cost of a detached house using combined simulation and optimisation. Building and Environment 43(12): 2022-2034

Huang Z, Chengen W, Chen J and Hong T (2011). Optimal design of aeroengine turbine disc based on kriging surrogate models. Computers and Structures 89(1-2): 27-37

Hawe G and Sykulski J (2007) Science, Measurement and Technology 1(1): 37-47

Jin Y (2005). A comprehensive survey of fitness approximation in evolutionary computation. Soft computing. Volume 9, pages 3-12.

Morris M. D and Mitchell T. J (1995). Exploratory designs for computational experiments. Journal of Statistical Planning and Inference, 43:381–402.

Sacks J, Welch W. J, Mitchell T. J, and Wynn H (1989). Design and analysis of computer experiments. Statistical Science, 4(4):409–423.

Smola A. J and Schölkopf B (2004). A tutorial on support vector regression. Statistics and Computing, 14: 199–222.

Tresidder E.G, Zhang, Y, Forrester, A. I. J (2011). Optimisation of low-energy building design using surrogate models, BS2011 - 12th Conference of International Building Performance Simulation Association, Sydney, 14-16 November, 2011

Verbeeck G and Hens H (2007) Life Cycle Optimisation of Extremely Low Energy Dwellings. Journal of Building Physics 31(2):143-177

Wang W, Zmeureanu R and Hugues R (2005). Applying multi-objective genetic algorithms in green building design optimization. Building and Environment 40(11): 1512-1525

Wetter M and Wright J (2004) A comparison of deterministic and probabilistic optimization algorithms for nonsmooth simulation-based optimisation. Building and Environment 39(8): 989-999.

Wright J and Alamji A (2005). The robustness of genetic algorithms in solving unconstrained building optimization problems. 9th international IBPSA conference, Montreal, Canada

Zhang, Y. 2012. Use jEPlus as an efficient building design optimisation tool, CIBSE ASHRAE Technical Symposium, London, 18-19 April, 2012

Zitzler E, Laumanns M, Bleuler S (2004). A tutorial on Evolutionary Multiobjective Optimization. Meataheuristics for Multiobjective Optimization. Springer, Berlin. Volume 535. Pages 3-37.