

# A Constraint Mapping Approach to the Structural Optimization of an Expensive Model using Surrogates

Stephen Leary, Atul Bhaskar and Andy Keane

Computational and Engineering Design Centre  
Mechanical Engineering  
School of Engineering Sciences  
The University of Southampton  
Highfield  
Southampton  
SO17 1BJ  
United Kingdom

## Abstract

The use of response surface methods are well established in the global optimization of expensive functions, the response surface acting as a surrogate to the expensive function objective.

In structural design however, the change in objective may vary little between the two models: it is the constraints that change with models of varying fidelity. Here approaches are described whereby the coarse model constraints are mapped so that the mapped constraints more faithfully approximate the fine model constraints. The shape optimization of a simple structure demonstrates the approach.

## 1 Introduction

The design optimization of an engineering system typically requires hundreds of analyses of that system. This process is computationally very expensive and as a result approximation methods have become popular as a means of reducing this cost. More recently, the use of models of varying fidelity have been considered.

In the aero-engines industry there are two main types of analysis - low precision whole engine modelling and high precision component modelling. Run times are high and not well suited to optimization (personal communication with McMillan, Wiseall, Rolls-Royce Plc [1]). The objective here is to develop a third type of analysis specifically targeted at structural optimization. This could incorporate models of varying degrees of complexity, and models which are submodels of the whole. The present work is motivated by this industrial need.

The selective use of models of varying fidelity throughout the design process is considered. The approach combines the expensive model together with approximations based on a cheaper surrogate.

Traditionally surrogate modelling approaches consider the multilevel optimization of some response which differs slightly when considering models of varying fidelity. In structural applications however, the change in response can vary little if at all. More usually it is the constraints which change with changing model complexity. In this case it is suggested that the surrogate modelling be applied to the relevant constraint and not to the response we wish to optimize. Typically a constraint arises from ensuring the level of stress in a material remains below some certain prescribed value.

The two main areas of interest considered here are response surface methods and space mapping algorithms.

Response surface methods are a class of techniques which approximate the value of some response at unsampled values of some independent variables based on a limited number of responses given certain input values of the independent variables. As a result, response surface methods are particularly applicable to investigations where there are only a few observations available because the evaluation of a response is computationally very expensive.

Large scale repetitive analysis of such computationally expensive problems (e.g. optimization) may then be replaced by a large scale repetitive analysis of the response surface which acts as a surrogate.

Examples of response surface approximations include polynomial regression [2], Shepard weighting functions [3, 4] and more advanced statistical methods such as kriging [5, 6]. In this paper we concentrate on kriging.

Space mapping algorithms [7, 8, 9, 10, 11] provide an alternative technique to using surrogates throughout the optimization process. They concern the selective use of models of varying fidelity. The basic idea is to optimize an accurate, expensive, non-linear model by utilizing the approximate behaviour of a less accurate but cheaper model through a parameter mapping.

The space mapping approach has been successfully applied to multilevel optimization in microwave circuit design in [7, 8, 9, 10].

We consider approaches which aim to establish a mapping between constraints given a limited number of cheap and expensive model evaluations and compare this to approaches based on kriging.

The constraint mapping approaches are applied to a simple structural problem where we wish to minimize the weight of a beam subject to constraints. Two models are described with different mesh densities, here the weight remains the same but the calculated stress (which forms one of the constraints) differs in the two models. Initial results are encouraging.

## 2 Response surface methods

In a typical response surface method the relationship between observations (responses) and independent variables is defined as

$$y = f(\mathbf{x}) \tag{1}$$

where  $y$  is the observed response,  $\mathbf{x}$  is a vector of  $k$  independent variables

$$\mathbf{x} = [x_1, x_2, \dots, x_k] \tag{2}$$

and  $f(\mathbf{x})$  is some unknown function. The response surface of the unknown function is defined as

$$\hat{y} = \hat{f}(\mathbf{x}), \quad (3)$$

an approximation based here on kriging. A brief description of its implementation now follows.

The response is expressed as

$$y(\mathbf{x}) = \mu + \epsilon(\mathbf{x}) \quad (4)$$

where  $\mu$ , a constant, is the mean of the input responses and  $\epsilon(\mathbf{x})$  is a Gaussian random function with zero mean and variance  $\sigma^2$ . The term  $\epsilon(\mathbf{x})$  is used to build up localized deviations from a global model.

We do not assume that these errors  $\epsilon$  are independent, as in regression, but that the errors are correlated, the correlation between errors being related to some distance measure between corresponding points. A popular distance measure [6] is

$$d(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \sum_{h=1}^k \theta_h |x_h^{(i)} - x_h^{(j)}|^{p_h} \quad (\theta_h \geq 0, 1 \leq p_h \leq 2) \quad (5)$$

where  $\theta_h$  and  $p_h$  are some as yet undetermined parameters. Using this distance function the correlation between the errors at  $\mathbf{x}^{(i)}$  and  $\mathbf{x}^{(j)}$  is defined by the exponential correlation function

$$\mathbf{R}(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \exp[-d(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})]. \quad (6)$$

This particular function has the desirable property that the correlation is near to one if  $\mathbf{x}^{(i)}$  is close to  $\mathbf{x}^{(j)}$ . However, if  $\mathbf{x}^{(i)}$  and  $\mathbf{x}^{(j)}$  are far apart, then the correlation will be small.

The hyperparameters  $\theta_h$ ,  $p_h$ ,  $h = 1, \dots, k$  come from maximizing the likelihood of the sample. This is defined as

$$\frac{1}{(2\pi)^{\frac{N}{2}} (\sigma^2)^{\frac{N}{2}} |\mathbf{R}|^{\frac{1}{2}}} \exp \left[ \frac{-(\mathbf{y} - \mathbf{1}\mu)^T \mathbf{R}^{-1} (\mathbf{y} - \mathbf{1}\mu)}{2\sigma^2} \right] \quad (7)$$

where  $\mathbf{y}$  is a vector of responses at the sampled points,  $\mathbf{1}$  is a vector of ones and  $\mathbf{R}$  is the correlation matrix defined in (6).

The mean  $\mu$  is given by

$$\mu = \frac{\mathbf{1}^T \mathbf{R}^{-1} \mathbf{y}}{\mathbf{1}^T \mathbf{R}^{-1} \mathbf{1}} \quad (8)$$

and the variance is given by

$$\sigma^2 = \frac{(\mathbf{y} - \mathbf{1}\mu)^T \mathbf{R}^{-1} (\mathbf{y} - \mathbf{1}\mu)}{N}. \quad (9)$$

Note that the dependence on  $\theta_h$ ,  $p_h$  is through the correlation matrix  $\mathbf{R}$ .

Another term of interest is the correlation vector  $\mathbf{r}(\mathbf{x})$  between the response at a location  $\mathbf{x}$  and the  $N$  previously sampled points  $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(N)}$ . This correlation vector is expressed as

$$\mathbf{r}(\mathbf{x}) = \mathbf{R}(\mathbf{x}, \mathbf{x}^{(i)}) = [\mathbf{R}(\mathbf{x}, \mathbf{x}^{(1)}), \dots, \mathbf{R}(\mathbf{x}, \mathbf{x}^{(N)})] \quad (10)$$

and is important since it is used in the prediction of  $y$  at  $\mathbf{x}$ .

The prediction at an unsampled point is given by

$$\hat{y}(\mathbf{x}) = \mu + \mathbf{r}^T \mathbf{R}^{-1} (\mathbf{y} - \mathbf{1}\mu), \quad (11)$$

and it is clear that this approximation interpolates the data.

The mean squared error of prediction

$$s^2(\mathbf{x}) = \sigma^2 \left[ 1 - \mathbf{r}^T \mathbf{R}^{-1} \mathbf{r} + \frac{(1 - \mathbf{1}^T \mathbf{R}^{-1} \mathbf{r})^2}{\mathbf{1}^T \mathbf{R}^{-1} \mathbf{1}} \right] \quad (12)$$

gives a measure of accuracy in our approximation at  $\mathbf{x}$ . This is obviously very useful and could be used when we consider model fusion.

## 2.1 Applications to model fusion

As highlighted earlier, one approach is to use the response surface as an approximation to the expensive model. However in many cases additional information is available via cheaper models. Utilizing these cheaper models can yield improvements in prediction and it is to this approach that we now turn.

Model fusion concerns the selective use of models of varying fidelity throughout the optimization process. For instance, we may be interested in minimizing the values of a certain response in some finite element code. Various models can exist, for example we may have a cheap model using beam or shell elements and a low mesh density and an expensive model based on 3D elements and using a high mesh density, or even combinations of the two. The cheap models are computationally inexpensive but may lack accuracy, a key question is: when can we trust them? On the other hand, expensive models are more accurate but a minimization based on these models alone would prove too computationally intensive. The obvious approach is to use the cheap model where the two models agree and switch to the expensive model when required.

One way to do this was introduced by El-Beltagy [12], where the cheap model response  $f_a$  is included in the input data, that is, we input  $[\mathbf{x}_i, f_a(\mathbf{x}_i)]$ ,  $i = 1, \dots, N$ , where we have  $N$  sampled points, and use this information to train our model based on an expensive model output  $f_e(\mathbf{x}_i)$ ,  $i = 1, \dots, N$ .

The strategy used in this paper is to build a response surface of the differences between the two models and add this to the cheap model in order to approximate the expensive model, that is, we consider  $N$  expensive function evaluations  $f_e(\mathbf{x}_i)$  as well as  $N$  cheap function evaluations at the same inputs  $f_a(\mathbf{x}_i)$ . We then form a difference

$$d(\mathbf{x}_i) = f_e(\mathbf{x}_i) - f_a(\mathbf{x}_i) \quad (13)$$

and consider using a response surface  $\hat{d}$  based on this difference. Throughout the optimization process we consider

$$f_a(\mathbf{x}) + \hat{d}(\mathbf{x}) \quad (14)$$

which approximates  $f_e(\mathbf{x})$  and interpolates  $f_e(\mathbf{x}_i)$ .

Our interest here is in constrained optimization. Two possible approaches may be considered:

- Include the constraints in the objective using penalty functions [13].
- Treat the objective and any constraints separately.

The penalty function approach seems appealing as it requires only one response surface, however, this approach introduces sharp changes into the objective which are difficult to model using a limited amount of data.

The second approach uses separate response surfaces for the objective and constraints. This adds some cost to the algorithm but may be worth it as the gain in accuracy could outweigh the gain in cost. In any case, in a general problem not all constraints will be active: we only need apply the response surface to any active constraints. This of course requires some knowledge of the problem. Such knowledge may come from the cheaper model. It is this approach which interests use here.

The algorithm may be represented as

- Consider  $N$  sample points  $\mathbf{x}_i$ ,  $i = 1, \dots, N$  spread throughout the design space
- Evaluate the expensive model objective  $f_e$  and constraints  $\sigma_e^1, \dots, \sigma_e^J$  ( $J$  is the number of (active) constraints) at these points
- Evaluate the cheap model objective  $f_a$  and constraints  $\sigma_a^1, \dots, \sigma_a^J$  at these points
- Evaluate the differences between the models  $d_f, d_{\sigma^1}, \dots, d_{\sigma^J}$  at the sampled points
- Calculate the response surfaces  $\hat{d}_f, \hat{d}_{\sigma^1}, \dots, \hat{d}_{\sigma^J}$
- Optimize using the response surfaces.

### 3 Space Mapping Algorithms

The space mapping algorithm was introduced by Bandler in [7]. In this section we describe a simple implementation of this approach highlighting the principles involved. We then go on to describe how the algorithm can be adapted to include problems with constraints.

Consider a mathematical model  $f_e(\mathbf{x}) : \mathcal{R}^n \rightarrow \mathcal{R}$  representing measurements of some system under consideration. We wish to find the optimal set of parameters  $\mathbf{x}^*$  which minimizes  $f_e(\mathbf{x})$ , that is, we wish to find

$$\mathbf{x}^* = \arg \min_{\mathbf{x}} f_e(\mathbf{x}). \quad (15)$$

The problem of interest again is the case when  $f_e(\mathbf{x})$  is so expensive that solving (15) is considered intractable.

A standard approach when confronted with a problem such as (15) is to consider a cheaper model  $f_a(\mathbf{z})$  which, while approximating  $f_e(\mathbf{x})$ , is faster to evaluate but also less accurate. The coarse model  $f_a(\mathbf{z})$  is used to gain

information about the expensive model  $f_e(\mathbf{x})$ , by obtaining a correspondence between the parameters  $\mathbf{x}$  and  $\mathbf{z}$ .

Since  $f_a(\mathbf{z})$  is cheap to evaluate,

$$\mathbf{z}^* = \arg \min_{\mathbf{z}} f_a(\mathbf{z}) \quad (16)$$

is much easier to solve than (15). However  $\mathbf{z}^*$  is likely to be different to  $\mathbf{x}^*$  due to the differences between the models.

Since  $f_a(\mathbf{z})$  and  $f_e(\mathbf{x})$  are approximating the same physical system, it is reasonable to assume some similarity between the models. Therefore, it should be possible to use the coarse model to gain information about the fine model.

A function mapping  $\mathbf{p}$  between the two parameter spaces is sought,  $\mathbf{p}$  relates a set of fine model parameters  $\mathbf{x}$  to a set of coarse model parameters  $\mathbf{z}$ . This is achieved by finding  $\mathbf{z}$  such that we have the best similarity between the expensive response  $f_e(\mathbf{x})$  and the coarse response  $f_a(\mathbf{z})$ . The functional relationship between the parameter spaces is  $\mathbf{z} = \mathbf{p}(\mathbf{x})$ . The similarity is measured through the residual

$$r(\mathbf{x}, \mathbf{z}) = \|f_e(\mathbf{x}) - f_a(\mathbf{z})\|. \quad (17)$$

The mapping function is defined as

$$\mathbf{p}(\mathbf{x}) = \arg \min_{\mathbf{z}} r(\mathbf{x}, \mathbf{z})^2. \quad (18)$$

Each evaluation of the mapping function requires at least one expensive function evaluation, the principle of the mapping is sketched in figure 1.

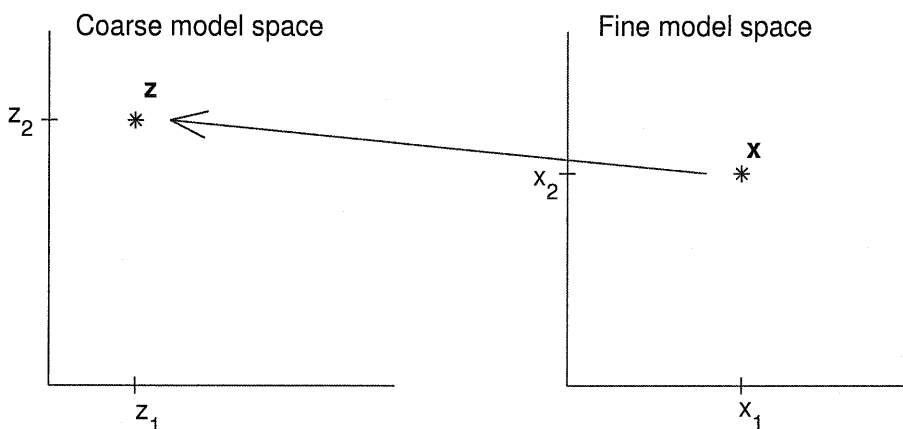


Figure 1: A two dimensional mapping from  $\mathbf{x}$  to  $\mathbf{z}$ .

From the definition of the mapping function it follows that

$$f_a(\mathbf{p}(\mathbf{x})) \approx f_e(\mathbf{x}), \quad (19)$$

that is, the coarse model with the mapped parameters approximates the fine model. Thus minimizing  $f_a(\mathbf{p}(\mathbf{x}))$  provides an approximation to our problem (15).

A perfect mapping is defined as the case where the optimal coarse model parameters  $\mathbf{z}^*$  satisfy  $\mathbf{z}^* = \mathbf{p}(\mathbf{x}^*)$ .

## 4 Parameter Extraction

In the examples presented here, an approach using derivatives is used. This approach is based on the fact that the cheap approximation is a relatively faithful approximation to the expensive problem.

Here a term

$$w \left\| \frac{\nabla f_e(\mathbf{x})}{\|\nabla f_e(\mathbf{x})\|} - \frac{\nabla f_a(\mathbf{z})}{\|\nabla f_a(\mathbf{z})\|} \right\| \quad (20)$$

is added to the right side of equation (17) where  $w$  is some weight reflecting the fact that it is more important to get the function values right. If it is the case that the extracted parameter is not unique, then this derivative term identifies the most likely contender.

The derivatives here are evaluated using finite differences, hence in this parameter extraction approach extra expensive function evaluations are required. However, in severe cases this approach may not work.

Parameter extraction has also been addressed by Bakr *et al.* [14] who consider the problem in the context of space mapping theory. Their approach, “multipoint parameter extraction”, uses several points  $\mathbf{x} + \Delta \mathbf{x}_i$  around the point of interest and essentially minimizes

$$r(\mathbf{x}, \mathbf{z})^2 + \sum_i r(\mathbf{x} + \Delta \mathbf{x}_i, \mathbf{z} + \Delta \mathbf{z}_i)^2 \quad (21)$$

in order to obtain the correct parameter.

It must be noted that this approach adds expense to the algorithm as an  $n$  point parameter extraction requires  $n$  expensive function evaluations.

## 5 Local Space Mapping Algorithms

In conventional space mapping algorithms the mapping  $\mathbf{p}$  is sequentially approximated using linear approximations  $\mathbf{p}_k$  around the current set of parameters  $\mathbf{x}_k$ .

The approximation is given by

$$\mathbf{p}_k(\mathbf{x}) = \mathbf{z}_k + \mathbf{B}_k(\mathbf{x} - \mathbf{x}_k). \quad (22)$$

$\mathbf{B}_k$  is an approximation to the unknown matrix  $\mathbf{B}$ , the Jacobian of the mapping function and is calculated using Broydens update [15]

$$\mathbf{B}_{k+1} = \mathbf{B}_k + \frac{\mathbf{z}_{k+1} - \mathbf{z}_k - \mathbf{B}_k \mathbf{h}_k}{\mathbf{h}_k^t \mathbf{h}_k} \mathbf{h}_k \quad (23)$$

where  $\mathbf{h}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$ . The mapped parameters  $\mathbf{z}_k$  come from (18), that is, they satisfy  $\mathbf{z}_k = \mathbf{p}(\mathbf{x}_k)$ .

The algorithm initially starts with no information about the mapping, that is, we set  $\mathbf{B}_0 = \mathbf{I}(n, n)$  and  $\mathbf{z}_0 = \mathbf{x}_0$ .

Since the linearization (22) is only valid in a neighbourhood of  $\mathbf{x}_k$  we define a trust region in which the approximation error

$$\|\mathbf{p}(\mathbf{x}) - \mathbf{p}_k(\mathbf{x})\| \quad (24)$$

is acceptable.

The linearization  $\mathbf{p}_k$  is only accepted for the set

$$\{\mathbf{x} : \|\mathbf{x} - \mathbf{x}_k\| < \delta_k\}, \quad (25)$$

where  $\delta_k$  is the size of the trust region at step  $k$ , which forms a confined region around  $\mathbf{x}_k$ . In its simplest form, the algorithm may be represented as (c.f. [11]):

```

Given  $\mathbf{x}_0, \delta_0$ 
 $\mathbf{B}_0 = \mathbf{I}(n, n), \mathbf{z}_0 = \mathbf{x}_0, k = 0$ 
do until convergence
     $\mathbf{x}_{k+1} = \arg \min_{\mathbf{x}} f_a(\mathbf{p}_k(\mathbf{x}))$  (subject to  $\{\mathbf{x} : \|\mathbf{x} - \mathbf{x}_k\| < \delta_k\}$ )
    evaluate  $f_e(\mathbf{x}_{k+1})$ 
    perform a parameter extraction (18) to obtain  $\mathbf{z}_{k+1}$ 
    update  $\delta, \mathbf{B}$ 
     $k = k + 1$ 
end do.
```

The initial size of the trust region is problem dependent. An approach for updating the trust region as the iteration proceeds can be found in [11]. In [11] the algorithm is also modified to overcome potential problems with imperfect maps. In this approach the algorithm switches between space mapping optimization and direct expensive optimization as the iteration proceeds.

## 5.1 Constrained Optimization

In many structural examples constraints relating to say maximum stress are also present. We consider how these constraints could be treated in space mapping optimization.

The first approach is through the use of penalty functions. The previous strategy is employed with minor alterations: here the functions  $f_a$  and  $f_e$  are replaced by penalty functions  $g_a$  and  $g_e$  incorporating any constraints. The algorithm may thus be represented as:

```

Given  $\mathbf{x}_0, \delta_0$ 
 $\mathbf{B}_0 = \mathbf{I}(n, n), \mathbf{z}_0 = \mathbf{x}_0, k = 0$ 
do until convergence
     $\mathbf{x}_{k+1} = \arg \min_{\mathbf{x}} g_a(\mathbf{p}_k(\mathbf{x}))$  (subject to  $\{\mathbf{x} : \|\mathbf{x} - \mathbf{x}_k\| < \delta_k\}$ )
    evaluate  $g_e(\mathbf{x}_{k+1})$ 
    perform a parameter extraction (18) to obtain  $\mathbf{z}_{k+1}$ 
    update  $\delta, \mathbf{B}$ 
```



$k = k + 1$   
end do.

One problem here may be in the parameter extraction stage where the penalty functions might increase the non-uniqueness of the solution of (18). Another problem is that it introduces a sharp change into the mapping function  $\mathbf{p}(\mathbf{x})$ .

The other possibility is to separately map the objective and any constraints, this would simplify the parameter extraction stage but increase the number of mappings required. In its simplest form the algorithm could be represented as:

```

Given  $\mathbf{x}_0, \delta_0$ 
 $\mathbf{B}_0^{obj} = \mathbf{I}(n, n), \mathbf{B}_0^{con1} = \mathbf{I}(n, n), \dots, \mathbf{B}_0^{conJ} = \mathbf{I}(n, n), \mathbf{z}_0 = \mathbf{x}_0, k = 0$ 
do until convergence
   $\mathbf{x}_{k+1} = \arg \min_{\mathbf{x}} f_a(\mathbf{p}_k^{obj}(\mathbf{x}))$  subject to
     $\sigma_{min}^1 \leq \sigma_a^1(\mathbf{p}_k^{con1}(\mathbf{x})) \leq \sigma_{max}^1$ 
     $\sigma_{min}^2 \leq \sigma_a^2(\mathbf{p}_k^{con2}(\mathbf{x})) \leq \sigma_{max}^2$ 
     $\sigma_{min}^J \leq \sigma_a^J(\mathbf{p}_k^{conJ}(\mathbf{x})) \leq \sigma_{max}^J$ 
     $\{\mathbf{x} : \|\mathbf{x} - \mathbf{x}_k\| < \delta_k\}$ 
  evaluate  $f_e(\mathbf{x}_{k+1}), \sigma_e^1(\mathbf{x}_{k+1}), \dots, \sigma_e^J(\mathbf{x}_{k+1})$ 
  perform a parameter extraction (18) to obtain  $\mathbf{z}_{k+1}^{obj}, \mathbf{z}_{k+1}^{con1}, \dots, \mathbf{z}_{k+1}^{conJ}$ 
  update  $\delta, \mathbf{B}^{obj}, \mathbf{B}^{con1}, \dots, \mathbf{B}^{conJ}$ 
   $k = k + 1$ 
end do.
```

Here  $f_a$  represents the cheap objective,  $f_e$  represents the expensive objective,  $\sigma_a^i$  represents the  $i^{th}$  cheap constraint and  $\sigma_e^i$  represents the  $i^{th}$  expensive constraint. Each mapping function  $\mathbf{p}^{obj}, \mathbf{p}^{con1}, \dots, \mathbf{p}^{conJ}$  is approximated locally as before and defined on a given trust region.  $\delta$  is then taken as the smallest trust region, that is  $\delta = \min\{\delta^{obj}, \delta^{con1}, \dots, \delta^{conJ}\}$ .

## 6 Global Space Mapping Algorithms

Rather than a local approach where  $\mathbf{p}_k(\mathbf{x})$  is linear about  $\mathbf{x}$  and defined on a given trust region, we could consider a global approach where  $\mathbf{p}(\mathbf{x})$  is approximated by  $\hat{\mathbf{p}}(\mathbf{x})$ , an approximation coming from an artificial neural network [9] or some other response surface.

This global mapping approach can be defined as follows

- Given  $N$  training samples  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$ .  
Obtain  $\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_N (= \mathbf{p}(\mathbf{x}_1), \mathbf{p}(\mathbf{x}_2), \dots, \mathbf{p}(\mathbf{x}_N))$  by parameter extraction.
- Using this data form a response surface  $\hat{\mathbf{p}}$ .
- Then  $f_a(\hat{\mathbf{p}}(\mathbf{x})) \approx f_e(\mathbf{x})$ , hence  $f_a(\hat{\mathbf{p}}(\mathbf{x}))$  can be used for optimization.

The approach can again be applied to constrained optimization. One may consider a penalty function approach whereby  $f_e$  and  $f_a$  in the above are replaced by their respective penalty functions incorporating any constraints. Again this will probably increase the non-uniqueness of the parameter extraction step as well as introducing a sharp change in the shifts which is difficult to model - especially with a limited number of sampled points.

The alternative is to separately map the objective and constraints, in this case, for an objective and  $N$  constraints, the following strategy is proposed:

- Given  $N$  training samples  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$ .  
 Obtain  $\mathbf{z}_1^{obj}, \mathbf{z}_2^{obj}, \dots, \mathbf{z}_N^{obj} (= \mathbf{p}^{obj}(\mathbf{x}_1), \mathbf{p}^{obj}(\mathbf{x}_2), \dots, \mathbf{p}^{obj}(\mathbf{x}_N))$   
 $\mathbf{z}_1^{con1}, \mathbf{z}_2^{con1}, \dots, \mathbf{z}_N^{con1} (= \mathbf{p}^{con1}(\mathbf{x}_1), \mathbf{p}^{con1}(\mathbf{x}_2), \dots, \mathbf{p}^{con1}(\mathbf{x}_N))$   
 $\vdots$   
 $\mathbf{z}_1^{conJ}, \mathbf{z}_2^{conJ}, \dots, \mathbf{z}_N^{conJ} (= \mathbf{p}^{conJ}(\mathbf{x}_1), \mathbf{p}^{conJ}(\mathbf{x}_2), \dots, \mathbf{p}^{conJ}(\mathbf{x}_N))$   
 by parameter extraction.
- Using this data form the response surfaces  $\hat{\mathbf{p}}^{obj}, \hat{\mathbf{p}}^{con1}, \dots, \hat{\mathbf{p}}^{conJ}$ .
- Then  $f_a(\hat{\mathbf{p}}^{obj}(\mathbf{x})) \approx f_e(\mathbf{x}), \sigma_a^1(\hat{\mathbf{p}}^{con1}(\mathbf{x})) \approx \sigma_e^1(\mathbf{x}), \dots, \sigma_a^J(\hat{\mathbf{p}}^{conJ}(\mathbf{x})) \approx \sigma_e^J(\mathbf{x})$ .  
 Hence  $f_a(\hat{\mathbf{p}}^{obj}(\mathbf{x})), \sigma_a^1(\hat{\mathbf{p}}^{con1}(\mathbf{x})), \dots, \sigma_a^J(\hat{\mathbf{p}}^{conJ}(\mathbf{x}))$  can be used for optimization.

Because of the difficulty envisaged with the penalty function approach, the approach considering a separate treatment of objective and constraints will only be considered.

## 7 Results

### Illustrative Example

A beam of length 20 mm, height  $h$  mm and width  $w$  mm is clamped at one end (see figure 2) and is subjected to a uniformly distributed load of 0.05 N/mm. We wish to find the optimal values of  $h$  and  $w$  such that the weight is minimized subject to

$$\sigma_x < 250 \text{ N/mm}^2, \quad (26)$$

and

$$0.5 \text{ mm} \leq h, w \leq 1 \text{ mm}. \quad (27)$$

The problem was solved using a simple finite element beam model. Two levels of complexity were considered, a coarse model consisting of just 2 elements and a fine model consisting of 100 elements. In these two models the objective  $V$  (volume is proportional to weight) remains the same whereas the stress, which forms the constraint, varies. It is the variation in this constraint that is mapped here.

Results for the cheap model optimization, a response surface based on kriging the expensive data points alone (RSM), a response surface method mapping

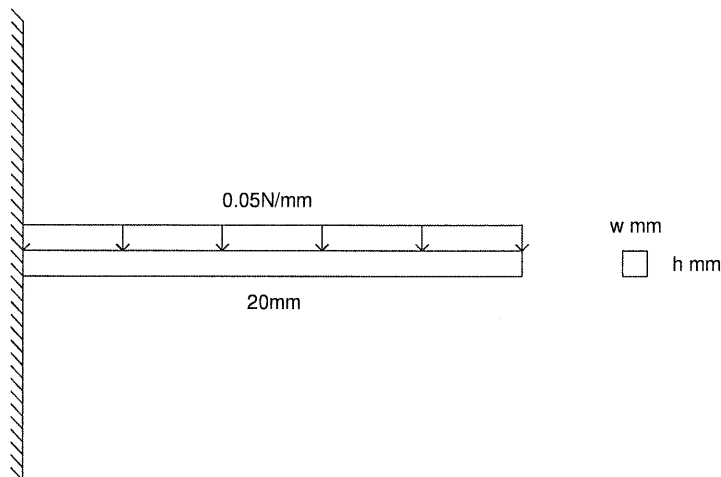


Figure 2: The problem.

based on model fusion (MFRSM), a local space mapping constrained optimization (LSMCO), global space mapping constrained optimization (GSMCO) and expensive model optimization are shown in table 1. The expensive model result is shown for comparison purposes only, in general this much information about the fine model is unavailable.

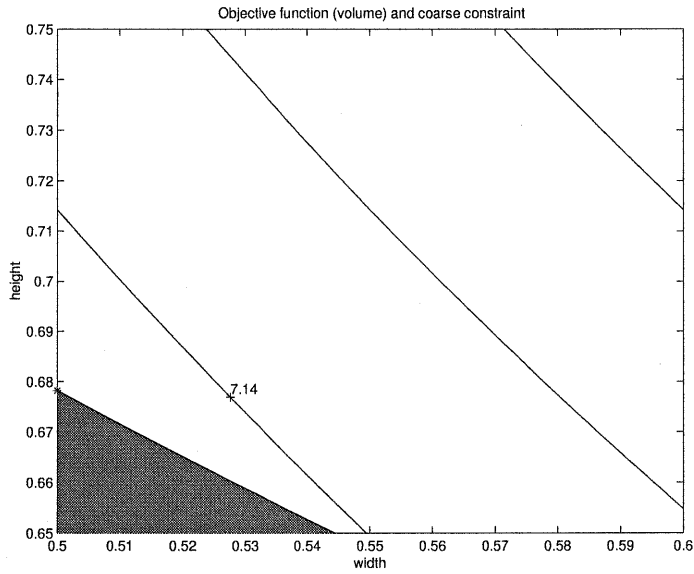
The following figures represent the objective function (weight) as well as the stress constraint. Figure 1 (a) represents the cheap model (coarse mesh), figure 1 (b) the expensive model (fine mesh) and figure 1 (c) the mapped model (using the response surface approach - space mapping gives a very similar result).

The expensive model objective and constraint are included for comparison purposes and are much more expensive to compute. The mapped model provides excellent agreement with the expensive model and this is achieved at a fraction of the cost.

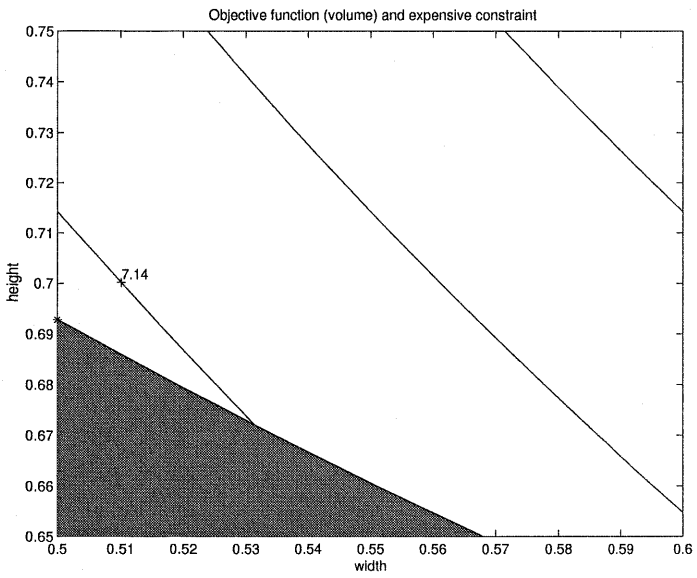
Model	width	height	model stress	actual stress	V
Cheap	0.5	0.67823	250.00	260.86	6.7823
RSM	0.5	0.71391	250.00	235.44	7.1391
MFRSM	0.5	0.69411	250.00	249.07	6.9411
LSMCO	0.5	0.69270	250.00	250.08	6.9270
GSMCO	0.5	0.69410	250.00	249.08	6.9410
Expensive	0.5	0.69281	250.00	250.00	6.9281

Table 1: Results

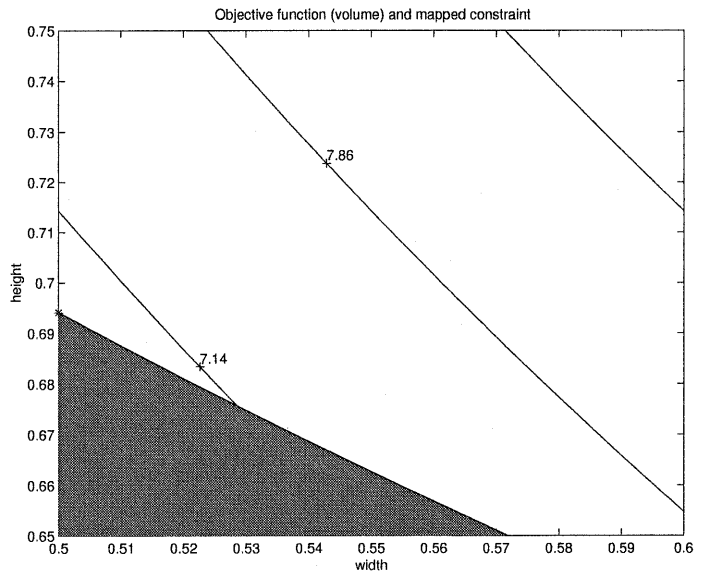
The cheap model is very inexpensive but the approximation of the constraint lacks accuracy. The minimum here produces an analytical stress  $> 250$ , and as a result the design is infeasible.



(a)



(b)



(c)

Figure 3: Objective and constraint with optimum (\*).

The response surface methods use nine expensive function evaluations at the points  $((0.5, 0.5), (0.75, 0.5), (1, 0.5), (0.5, 0.75), (0.75, 0.75), (1, 0.75), (0.5, 1), (0.75, 1), (1, 1))$ . Kriging the expensive model alone (RSM) gives a feasible design in this case, however, using the nine expensive function evaluations alone does not supply accurate information on the constraint surface.

The fused model (MFRSM), by utilizing information from the cheap model, together with the nine expensive function evaluations has led to a much improved approximation of the expensive model minima.

In the above, the LSMCO approach used  $\mathbf{x}_0 = \mathbf{z}_0 = 1$  and  $\delta_0 = 0.05$ . The algorithm converged in eight iterations requiring a total of 24 calls to the expensive model.

The GSMCO approach used the same nine data points as the response surface methods thus requiring a total of 27 expensive function evaluations.

The results presented here highlight the fact that fusion strategies provide an effective way of optimizing expensive models. At this stage we are not suggesting that any fusion strategy is better than any other. Which of these algorithms works best will generally be problem dependent. The purpose of the work described here is to illustrate how response surface methods and space mapping can be modified to incorporate constraints.

## 8 Conclusions

The use of models of varying fidelity throughout the optimization process provides an approach for optimizing computationally expensive problems. Usually the response we wish to optimize is considered. Here approaches which map the constraints are described. This is useful if the actual objective varies little with the different models but the constraints alter significantly.

The use of response surfaces for fusing models of varying fidelity subject to constraints is introduced. The problem of modeling constraints in the space mapping approach has also been addressed. The separate treatment of objective function and constraints is suggested as a means of overcoming potential problems with non-uniqueness of the parameter extraction stage and the difficulty in modelling sharp changes in the mapping. In the simple example given, encouraging results are obtained. They agree closely with the analytical solution.

The strategies described here show how it is possible to make real gains in computational efficiency through the use of surrogate modelling in engineering design. Of course, the example described here is rather simple.

## Acknowledgements

We would like to thank Rolls-Royce Plc for the financial support received as a part of the University Technology Partnership (UTP). We also thank Dr. Alison McMillan and Dr. Steve Wiseall of Rolls-Royce Plc for several useful discussions and for their industrial feedback.

## References

- [1] A. J. McMillan and S. Wiseall, Personal communication, Rolls-Royce Plc, Jan-Aug 2000.
- [2] R. F. Gunst and R. L. Mason, Regression analysis and its application: a data-oriented approach, 1980.
- [3] D. Shepard, A two dimensional interpolation function for irregularly spaced data, In Proceedings of the 23rd National Conference, ACM, New York, pp.517-523, 1968.
- [4] R. J. Renka, Multivariate interpolation of large sets of scattered data, ACM Transactions on Mathematical Software, Vol. 14, No. 2, pp.139-148, 1988.
- [5] D. G. Krige, A statistical approach to some basic mine valuation problems on the Witwatersrand, Journal of Chemical, Metallurgical and Mining Society of South Africa, Vol. 52, pp.119-138, 1951.
- [6] D. R. Jones, M. Schonlau and W. J. Welch, Efficient global optimization of expensive black-box functions, Journal of Global Optimization, Vol. 13, No. 4, pp. 455-492, 1998.
- [7] J. W. Bandler, R. M. Biernacki, S. H. Chen, P. A. Grobelny and R. H. Hemmers, Space mapping technique for electromagnetic optimization, IEEE Transactions on Microwave Theory and Techniques, Vol. 42, No. 12, pp.2536-2544, 1994.
- [8] M. H. Bakr, J. W. Bandler, R. M. Biernacki, S. H. Chen and K. Madsen A trust region aggressive space mapping algorithm for EM optimization, IEEE Transactions on Microwave Theory and Techniques, Vol. 46, No. 12, pp.2412-2425, 1998.
- [9] J. W. Bandler, M. A. Ismail, J. E. Rayas-Sanchez and Q. J. Zhang, Neuro-modeling of microwave circuits exploiting space mapping technology, Vol. 47, No. 12, pp.2417-2427, 1999.
- [10] M. H. Bakr, J. W. Bandler, N. Georgieva and K. Madsen A hybrid aggressive space mapping algorithm for EM optimization, IEEE Transactions on Microwave Theory and Techniques, Vol. 47, No. 12, pp.2440-2449, 1999.
- [11] Jacob Sondergaard, Non-linear optimization using space mapping, MSc. Thesis, Technical University of Denmark, 1999.

- [12] M. A. El-Beltagy, A natural approach to multilevel optimization in engineering design, PhD Thesis, University of Southampton, 1999.
- [13] R. L. Fox, Optimization methods for engineering design, Reading, Mass. : Addison-Wesley, 1971.
- [14] M. H. Bakr, J. W. Bandler and N. Georgieva, An aggressive approach to parameter extraction, IEEE Transactions on Microwave Theory and Techniques, Vol. 47, No. 12, pp.2440-2449, 1999.
- [15] C. G. Broyden, A class of methods for solving non-linear simultaneous equations, Mathematics of Computation, Vol. 19, pp.577-593, 1965.