

Construction of device performance models using adaptive interpolation and sensitivities

David Tsao

Department of Electrical and Computer Engineering
McGill University, Montreal

June 2004

A thesis submitted to McGill University in partial fulfillment of the
requirements of the degree of Master of Engineering

© David Tsao, 2004



Library and
Archives Canada

Bibliothèque et
Archives Canada

Published Heritage
Branch

Direction du
Patrimoine de l'édition

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file *Votre référence*

ISBN: 0-494-06591-5

Our file *Notre référence*

ISBN: 0-494-06591-5

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.


Canada

ABSTRACT

The performance of a device, for certain design parameters, can be modeled using finite element analysis (FEA); however, this can be computationally complex and time consuming. But because this performance can be modeled as a continuous function over the parameter ranges, it is efficient to combine FEA with interpolation to rapidly estimate the performance for any variables. While there exists many interpolation techniques, FEA can provide sensitivities at little extra cost, so these methods should take advantage of this information. In this thesis, a new adaptive interpolation scheme is proposed which uses radial basis functions (multiquadrics) while making use of sensitivities. Results demonstrate the greater accuracy of the new scheme compared to a previous multiquadric algorithm without sensitivities. Test cases include artificial functions, and an example combining FEA and the interpolation of the reflection coefficient for a rectangular waveguide containing a partial height metallic post.

RÉSUMÉ

L'exécution d'un dispositif, pour certains paramètres de conception, peut être modélisée en utilisant l'analyse d'élément finie (AEF); cependant, ceci peut être computationnellement complexe et prendre beaucoup de temps. Mais parce que cette exécution peut être modélisée comme une fonction continue par-dessus les gammes de paramètre, c'est efficace pour combiner AEF avec l'interpolation pour rapidement estimer l'exécution pour n'importe quelles variables. Il existe beaucoup de techniques d'interpolation, mais puisque le coût d'obtenir des sensibilités par AEF est peu, ces méthodes devraient profiter de cette information. Dans cette thèse, un nouvel algorithme d'interpolation adaptatif est proposé qui utilise les fonctions de base radiales (multiquadriques) en utilisant les sensibilités. Les résultats démontrent que le nouveau méthode est plus précis en comparaison avec un algorithme de multiquadrique précédent sans sensibilités. Les tests incluent des fonctions artificielles, et un exemple combinant AEF avec l'interpolation du coefficient de reflet pour un guide d'ondes rectangulaire contenant une poste métallique d'hauteur partielle.

ACKNOWLEDGEMENTS

We would like to thank my supervisor, Professor Jonathan P. Webb, for his guidance and always being receptive to my ideas for how to accomplish this thesis. I would also like to thank him on his advice on how to best write and present my report. Finally, I am grateful for the constant enthusiasm he showed in my work and his confidence in this project to present it at the Conference on Electromagnetic Field Computation.

In addition, I would like to thank all my colleagues in the CAD lab for providing a friendly environment in which to do the research. Notably, I would like to thank Vineet Rawat for always being supportive when I needed help debugging my program and Dileep Nair for his advice on coupling my program to the Finite Element software using VB. They all contributed to my time as a Master's student of McGill being a very enjoyable experience.

TABLE OF CONTENTS

1	Introduction.....	1
1.1	Problem Statement.....	1
1.2	Literature Review.....	2
1.2.1	Inverse distance weighted methods	3
1.2.2	Radial basis function methods	4
1.2.3	Using gradient information.....	5
1.2.4	Adaptive Sampling.....	6
1.3	Outline.....	8
2	Gradient Influenced Multiquadric (GIMQ).....	9
2.1	Introduction.....	9
2.2	Using derivative information	10
2.2.1	Pseudo-data points	10
2.2.2	Proposed basis functions and solving the unknowns.....	12
2.3	Optimizing the shift parameter	14
2.3.1	Leave out one.....	15
2.3.2	Golden search algorithm	15
3	Error estimation and adaptive interpolation.....	16
3.1	Error estimation	16
3.2	Adaptive sampling	18
4	Results.....	20
4.1	Artificial Test Functions	20
4.1.1	1-D Example	20
4.1.2	2-D examples	23
4.1.3	3-D Example	31
4.2	Practical Examples – Waveguides.....	32
4.2.1	Full-Height Post in Rectangular Waveguide	32
4.2.2	Partial-Height Post in Rectangular Waveguide	35
5	Conclusion	39

1 Introduction

1.1 Problem Statement

Numerous multivariate problems in science and engineering have a continuous behavior over the range of their parameters. Often however, when studying these problems, obtaining the function value at any given point is not practical or too costly. Instead, the user is limited to having only a discrete amount of information on the variables and the corresponding behaviors at those points.

A Response Surface Model (RSM) is a function capable of estimating the behavior of an objective function, anywhere over the parameter range, given the limited amount of data available. This can be accomplished either through interpolation or data fitting. Furthermore, this data is often irregularly spaced or “scattered” and the function must be able to handle this.

Some examples of fields where scattered data interpolation arises are “medical imaging, meteorological or geological modeling, cartography, and computer aided geometric design” [1]. In the case of computer aided design, a good interpolation technique can be used to model circuits and microwave structures, by combining the accuracy of EM simulators with the speed of circuit simulators. The data points are first obtained from often costly EM simulations then interpolated over desired ranges to construct a RSM which estimates the function for any values of the design parameters. This can be used for optimization, and replace expensive simulations saving time and money.

In this thesis a new adaptive interpolation technique for scattered data is introduced. It takes advantage of partial derivatives at the data points, where most other interpolation techniques do not include these, since they are generally not available. The algorithm also includes a method for adaptively sampling new data. The report will show,

through theoretical and practical examples, that this new method is more accurate than previous solutions. Some aspects of the work recently appeared in a conference publication [2].

1.2 Literature Review

There is a large literature on the many methods of interpolating scattered data. This is to be immediately distinguished from data fitting. Interpolation requires that the surface pass through the data points. Data fitting, on the other hand, is primarily concerned with the smoothness of the overall surface which need only pass near the data points, like “best-fit” lines.

Scattered data interpolation can be categorized into two general techniques: global methods and local methods. In global methods, the interpolating function over the whole space depends on the entire set of given data points: this can prove inefficient for a very large number of given data. In local methods, the interpolated space is divided into several functions, each only using “nearby” data points. Two excellent literature review papers are offered by Franke [3] and Amidror [1]. These papers review on the following methods for scattered data interpolation:

- Inverse distance weighted method
- Radial basis function (RBF) methods
- Rectangle based blending method
- Triangle (or Tetrahedrization) based methods

Of these four groups, only the first two are of real concern in this paper. Both are applicable to higher dimensions: a necessity for modeling in design space. The other two, rectangle and triangle based methods, in addition to being more complex and having greater storage requirements, are only really useful for up to two dimensions, thus not suitable for our purposes.

Sections 1.2.1 and 1.2.2 discuss the first two methods, for problems where the data set is composed of n points. For clarity, all equations in these sections will be

presented for two dimensions only, where the i -th point has coordinates (x_i, y_i) , and objective function $f(x, y)$ at that point is $f_i = f(x_i, y_i)$.

These equations can nevertheless, clearly, be generalized for P -dimensions.

1.2.1 Inverse distance weighted methods

Also known as Shepard's methods, inverse distance weighted methods are some of the most common forms of scattered data interpolation. Shepard's method [4] represents a particular case of the more general technique known as Interpolating Moving Least Squares (IMLS) [5].

Shepard's method is a global technique using weights: interpolated points are more dependent on nearby data, and less affected by data points far away. The original formula is a weighted average of values of the mesh points.

$$f(x, y) = \frac{\sum_{i=1}^N w_i(x, y) f_i}{\sum_{i=1}^N w_i(x, y)}, \quad (1)$$

where $w_k = d_k^{-\mu}$, with μ having typical value of 2, and $d_i = \sqrt{(x - x_i)^2 + (y - y_i)^2}$.

A major problem with this technique however is the resulting flatness at the data points. Some variations to Shepard's methods include changing the value of μ to manipulate the flatness: certain mesh points can even be given more influence by changing μ individually for each point. The method can also be made local by only using nearby data points to build the function, eliminating the effect of distant points.

To resolve the problem of flatness at the data points, Shepard proposed a way to take advantage of partial derivatives, and even estimate these when they are not available. To do this, the value of f_i in Eq. (1) is replaced by:

$$g_i = f_i + a_i(x - x_i) + b_i(y - y_i) \quad (2)$$

where a_i and b_i are the x - and y - partial derivatives at the mesh point. The result is a much more pleasing and continuous function.

Shepard's method is one of the most popular techniques. It is also easily extendable to higher dimensions. In its most basic form, there are certain shortcomings, perhaps the most important being the zero derivatives at every data point. The potential improvement using partial derivatives is shown in [6]. In Franke's tests [1], he did not examine Shepard's method using derivatives, but he did determine radial basis function methods to be the best technique when gradient information is not available. The question must then arise of how the "best" technique can be improved using partial derivatives.

1.2.2 Radial basis function methods

First introduced in the 1960's by Hardy [7], radial basis function methods have proven effective at interpolating scattered data. Franke showed that Hardy's multiquadric (HMQ) approach was the most impressive in terms of smoothness and fitting ability. The basic idea of RBF is to choose some function $G_k(x,y)$, so that for each data point we get:

$$f_i = \sum_{i=1}^n a_i G_i(x, y) \quad (3)$$

It then remains to find the coefficients a_i . As the name implies, the functions G_i will actually be associated with the distances between points (x, y) and (x_i, y_i) .

In Hardy's original method, G_i is a hyperboloid centered at (x_i, y_i) :

$$G_i = \sqrt{(x - x_i)^2 + (y - y_i)^2 + h^2} \quad (4)$$

The overall surface response can therefore be seen as summing circular two-sheet hyperboloids. The shape of the surface around any point is influenced by the sharpness or flatness of every conical surface around each data point.

Here, h is a so-called "shift parameter" to be determined and optimized by the user. Though the method proves quite stable with respect to h , in terms of maintaining smoothness, its optimal value is not easily found. Choosing too small a value will make the hilltops "sharper", whereas giving h too large a value may cause it to overshadow the hyperboloid itself. Alotto et al. [8] used a bootstrapping technique to optimize the shift parameter: for this, they applied a "leave out one" method, to estimate the average

interpolation error. This method removes the user intervention, making the choice of h automatic.

There are other possibilities for the basis function such as using paraboloids:

$$G_i = (x_i - x)^2 + (y_i - y)^2 + h^2 \quad (5)$$

However, according to Hardy, using quadrics other than cones might displace the maxima and minima of the interpolated surface. It is therefore useful to know the position of some points having zero slopes. To take advantage of this new information, one suggestion made by Hardy would be to add a simple polynomial series to the multiquadric. The example given, in 1-D, is:

$$z = \sum_{i=1}^n a_i [(x - x_i)^2 + h^2]^{1/2} + \sum_{i=1}^m b_i x^i, \quad (6)$$

where n is the total number data points, and there are m points having zero slope. This can then be solved by differentiating the function to obtain:

$$\sum_{i=1}^n a [(x - x_i)^2 + h^2]^{-1/2} (x - x_i) + \sum_{i=1}^m i b_i x^{i-1} = 0 \quad (7)$$

The coefficients a_i and b_i can then be solved for, since we have a system of $n + m$ equations and $n + m$ unknowns.

1.2.3 Using gradient information

In Section 1.2.1, the term IMLS was introduced. This technique is applied in [9] where a P -dimensional quadratic is used for the basis functions with $1 + 2P + (P^2 - P)/2$ terms (this differs from Shepard's method, where there is only one term). Weights ensure that data points far away from the sample point have minimal effect. With partial derivatives available, the author shows that with n data points and a large P , approximately $P/2$ data

points are required to get the same number of equations as unknowns. This is as opposed to $P^2/2$ which would be required if the derivatives were known.

IMLS using gradients is shown to have a better RMS error estimate when compared to HMQ without gradients. However, based on Fig. 1 of [9], it does appear to have an undesirable oscillatory behavior, compared to the multiquadric RSM, which does not.

In [10], another approach taking advantage of partial derivatives is taken in which the RSM uses inverse polynomial models. In addition to these inverse polynomials, pseudo-points are extrapolated at the extremes of the input ranges, to be used as extra data. These pseudo-points are only associated, however, with the limiting values of the parameters: in a P -dimension, there are only $2P$ pseudo-points.

In [11], the authors examine a curve-fitting technique which uses gradients. The authors use cubic functions for the fitting of the force and flux surfaces of an electromagnetic device. Though not an interpolation technique, this is yet another paper demonstrating how computation time is greatly reduced through the use of gradients.

Few papers have tackled the idea of using partial derivatives to improve interpolation, probably since these are often not available. But when they can be obtained at little extra cost to the finite element analysis, the benefits of including them in interpolation should be explored. References [9] and [10] propose techniques which appear to show improvements; but at the same time they have their own drawbacks, such as an oscillatory behavior or not associating pseudo-points to the data points themselves.

The purpose of this thesis is to explore a new technique that uses gradient information, and is more stable than previous solutions. Since Hardy's multiquadric technique seems to be the most impressive method when gradients are not available, it will be the basis of this new algorithm.

1.2.4 Adaptive Sampling

Because EM simulators are so computationally intensive, it is necessary to selectively sample data points to make the process more efficient. These points should be sampled adaptively so as to reduce the error between the interpolation function and scattering parameter behavior, while using the minimal amount of data.

There are several papers on adaptive sampling for interpolating S-parameters [12], [13], [14]. These papers focus on frequency as the main variable, but the concept should apply for multivariate problems. The S-parameters are first represented using rational functions then, using a suitable error estimate, the new sample point is determined.

In [13] the author uses the Model Based Parameter Estimation (MBPE) and in [12] the frequency response is represented using Multidimensional Cauchy Rational Functions (MCRF). Some ways of estimating the fitting error are using two models that have different data sets, or two models which have different orders for the numerator and denominator. Once these functions are built, the point of maximum deviation between the two can be used at the new sampling point. Points are added until the two models converge to an acceptable value.

When using different orders of the rational functions as an error estimate, the difference between these degrees is determined heuristically or fixed *a priori*. A solution to this is presented in [15], where a genetic algorithm determines this difference. The algorithm chooses those models which are closest together, so most likely to best represent the true function.

In [16], the authors propose another adaptive technique for building P -dimensional models for microwave structures, called multidimensional adaptive parameter sampling (MAPS). In this paper, the new sampling points are chosen by taking the maximum deviation between the “best fit” and “second best fit” interpolation functions.

In addition to adding new sample points to improve the overall RSM, they can also be added to explore the behavior of the RSM around its global extreme [17], [18], [19], [20], for optimization purposes.

By progressively improving the accuracy of the optimum region, through the addition of several new points, the user can determine the optimum electromagnetic device. There are several ways to optimize the objective function, such as using an Evolution Strategy or Simulated Annealing. An important note is that the main purpose of these optimization techniques is to explore the global extreme. Also, using these methods, there is a danger of becoming trapped in a local minimum. Resolving this

requires a further optimization technique, such as adding a point in the most un-explored area. In [17], a cluster feature associated with their genetic algorithm allows the user to add new points around local minima that are potentially masking the global minimum. This way, not only the best point is considered and the overall Response Surface is improved.

A three stage sampling process is used in [21] to search for the optimum. The first stage involves using Simulated Annealing to find minima of several HMQ interpolations that have different shift parameters. The second stage adds a point at the highest deviation between the HMQ interpolation having the smallest shift parameter and another HMQ with a given shift parameter. And finally, the third stage adds points in regions where the sample points are more widely spaced apart.

In addition to taking advantage of partial derivatives, the algorithm presented implements a technique for adaptively sampling new data points. As will be shown, the method for doing this is inspired from the technique applied in [16]. The goal of adding these new sample points will be to improve the overall RSM, so not just exploring the optimum region.

1.3 Outline

In Chapter 2, we will examine the proposed interpolation technique. Similarly to [10], the algorithm uses pseudo-data points, but improves upon it by associating these points with every data point.

In Chapter 3, the adaptive aspect of the algorithm will be explored, including an error estimate technique involving the original multiquadric method.

Chapter 4 will present the results for several interesting examples, proving the effectiveness of the algorithm. These will include some theoretical and practical examples of waveguides. One practical example will involve coupling to a Finite Element solver to determine new sample points.

Chapter 5 will conclude this thesis with final observations and remarks on potential further analysis of various aspects of the algorithm.

2 Gradient Influenced Multiquadric (GIMQ)

2.1 Introduction

As mentioned in section 1.2.4, two papers [9][10] have tackled the problem of how to take advantage of gradient information. These previous solutions, however, have demonstrated their own drawbacks. [9] and [10] use variations of IMLS and inverse polynomial models respectively, but in Franke's paper [3], it was determined that Hardy's multiquadric technique was the most solid form of interpolation. For this reason, this will be our starting point. We will take the best technique and try to include the gradient information to make it better.

In Hardy's method, we have n data points, along with their respective value f_i . We can then apply Eq. (3) to each point, resulting in n linear equations with n unknown coefficients a_i . In matrix notation, we are left with:

$$\mathbf{f} = \mathbf{M}\mathbf{a} \tag{8}$$

Now, if we want to include partial derivatives, there is a new set of information. For each new point, in a P -dimensional problem, there are P partial derivatives. If we add these to the vector \mathbf{f} , we get a new vector \mathbf{F} of size $n(P+1)$: for each of the n points, there is the function value and P partial derivatives.

We must now find new basis functions that exploit the benefits of having partial derivative information. These new basis functions will yield a fully determined $n(P+1)$ by $n(P+1)$ matrix and add flexibility and accuracy to the interpolation itself.

2.2 Using derivative information

2.2.1 Pseudo-data points

In [10], Malik et al. generate pseudo-points only at the extremes of the input ranges. This means there are only $2P$ pseudo-points in a P -dimensional problem, regardless of the number of data points, which in turn limits their contribution.

Another approach suggested in the same paper is to generate several pseudo-data points around the actual data points. For example, in two dimension, a point (x, y) will have 8 pseudo-points generated around it: $f(x - \Phi_1, y - \Phi_2)$, $f(x, y - \Phi_2)$, $f(x + \Phi_1, y - \Phi_2)$, $f(x - \Phi_1, y)$, ... The authors do not suggest a way of finding the optimal step sizes for these extra points, but these must be restricted so they don't overlap.

From their tests, the authors of [10] concluded that for their particular examples generating pseudo-points around the extremes resulted in the most accurate model. An argument can be made, however, that having these virtual points associated with each data point is more reasonable and beneficial, in general. In addition to providing more weight around the data points, there would be new pseudo-points for each new sample point added to the data set, be it for optimization or improving the overall RSM.

We suggest having one pseudo-point along each dimension at each point. Therefore, in two dimensions instead of having 8 extra points, we would instead generate only two. Taking our previous example of a point (x, y) , these extra points would find themselves at coordinates: $(x - \Phi_1, y)$ and $(x, y - \Phi_2)$.

Immediately, the problem of overlapping pseudo-point is much easier to solve since there are fewer of them. In order to reduce the chance of overlapping, the algorithm will use step sizes which vary for each point.

One possibility is to set the step size at a fix ratio of the distance between a point \mathbf{a} and its nearest neighbor \mathbf{b} : the step size Φ may be $\|\mathbf{a} - \mathbf{b}\|/\beta$. For our results, we will use a β value of 5: of course this number will have an influence on the RSM.

Figure 1 illustrates an example of data points in two dimensions, with their pseudo-points.

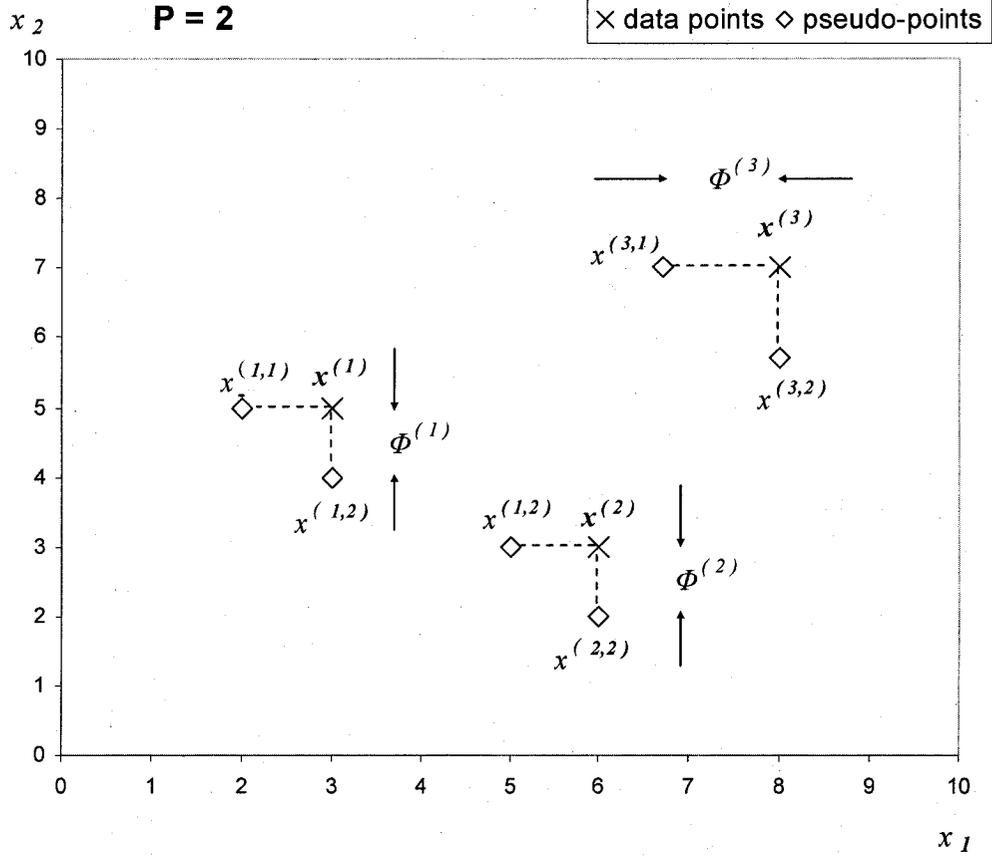


Figure 1 - Position of data points and their pseudo-points

Remark the following change in notation from Section 1.2.1 (used for the remainder of the report):

- x_w is the w -th component of vector \mathbf{x} .
- $\mathbf{x}^{(i)}$ is the i -th point.
- $\Phi^{(i)}$ is the distance between the i -th point and its pseudo-points.
- $x_k^{(i,w)} = x_k^{(i)} - \delta_{k,w} \Phi^{(i)}$: k -th component of $\mathbf{x}^{(i,w)}$

The pseudo-points are represented as $\mathbf{x}^{(i,w)}$, and are vectors having the component $x_k^{(i,w)}$.

From our tests, as will be seen in Chapter 4, it appears that having a smaller step size between the parent points and their pseudo-points results in a smoother and more

accurate surface. This might be explained by the fact that all the data points (parent and pseudo) are concentrated around real information – the objective function and partial derivatives at the coordinates of the parent point.

Interpolated points around the original data points will be more accurate than those further away. Therefore, pseudo-points which are closer to the parent point will be more precisely interpolated than if they are spread further away. Of course, pseudo-points are not evaluated as true data values nevertheless, but making them closer to the parent point does seem to have this effect on the surface.

The optimal step size, though, is not necessarily as small as possible. Firstly, it must not be so small so as to render the matrix singular. Also, the pseudo-points should still be spread out enough so that the overall surface can make good use of them.

The optimal value of the step size will vary depending on the RSM. And, the above explanation has not been mathematically proven to be valid, only hypothesized as reasonable. Further investigation would be required to find an optimum value of β .

2.2.2 Proposed basis functions and solving the unknowns

As stated in Section 2.1, if we use partial derivatives there are many more givens than unknowns. Since \mathbf{F} now has $n(P+1)$ elements, the matrix M must be a square matrix of the same size. But with the addition of pseudo-data points, there are now $P+1$ points for each of the n data points. Recall the original multiquadric, in two dimensions:

$$f = \sum_{i=1}^n a_i \sqrt{(x_1 - x_1^{(i)})^2 + (x_2 - x_2^{(i)})^2 + h^2} \quad (9)$$

The new basis functions added are also two-sheet hyperboloid, but now centered at the pseudo-data points. In two dimensions, this results in the surface being re-defined as:

$$f = \sum_{i=1}^n a_{3(i-1)+1} \sqrt{(x_1 - x_1^{(i)})^2 + (x_2 - x_2^{(i)})^2 + h^2} + \quad (10)$$

$$a_{3(i-1)+2} \sqrt{(x_1 - x_1^{(i)} + \Phi^{(i)})^2 + (x_2 - x_2^{(i)})^2 + h^2} +$$

$$a_{3(i-1)+3} \sqrt{(x_1 - x_1^{(i)})^2 + (x_2 - x_2^{(i)} + \Phi^{(i)})^2 + h^2}$$

Now since our new \mathbf{M} matrix has $n(P+1)$ columns, the extra $P = 2$ rows per data point are obtained by taking the derivatives of the basis functions with respect to each dimension:

$$\frac{df}{dx_1} = \sum_{i=1}^n a_{3(i-1)+1} \frac{x_1 - x_1^{(i)}}{\sqrt{(x_1 - x_1^{(i)})^2 + (x_2 - x_2^{(i)})^2 + h^2}} \quad (11)$$

$$+ a_{3(i-1)+2} \frac{x_1 - x_1^{(i)} + \Phi^{(i)}}{\sqrt{(x_1 - x_1^{(i)} + \Phi^{(i)})^2 + (x_2 - x_2^{(i)})^2 + h^2}}$$

$$+ a_{3(i-1)+3} \frac{x_1 - x_1^{(i)}}{\sqrt{(x_1 - x_1^{(i)})^2 + (x_2 - x_2^{(i)} + \Phi^{(i)})^2 + h^2}}$$

$$\frac{df}{dx_2} = \sum_{i=1}^n a_{3(i-1)+1} \frac{x_2 - x_2^{(i)}}{\sqrt{(x_1 - x_1^{(i)})^2 + (x_2 - x_2^{(i)})^2 + h^2}} \quad (12)$$

$$+ a_{3(i-1)+2} \frac{x_2 - x_2^{(i)}}{\sqrt{(x_1 - x_1^{(i)} + \Phi^{(i)})^2 + (x_2 - x_2^{(i)})^2 + h^2}}$$

$$+ a_{3(i-1)+3} \frac{y - y_i + \Phi_i}{\sqrt{(x_1 - x_1^{(i)})^2 + (x_2 - x_2^{(i)} + \Phi^{(i)})^2 + h^2}}$$

So we have $n(P+1)$ linear equations and $n(P+1)$ unknowns. Equations (10), (11), (12) can be re-written in their general form for P -dimensions as:

$$f = \sum_{i=1}^n \left\{ a_{(P+1)(i-1)+1} \cdot g(\underline{x} - \underline{x}^{(i)}) + \sum_{q=1}^P a_{(P+1)(i-1)+1+q} \cdot g(\underline{x} - \underline{x}^{(i,q)}) \right\} \quad (13)$$

and

$$\frac{df}{dx_w} = \sum_{i=1}^n \left\{ a_{(P+1)(i-1)+1} \cdot g'_w(\underline{x} - \underline{x}^{(i)}) + \sum_{q=1}^P a_{(P+1)(i-1)+1+q} \cdot g'_w(\underline{x} - \underline{x}^{(i,q)}) \right\} \quad (14)$$

where,

- $g(\underline{r}) = \sqrt{|\underline{r}|^2 + h^2}$
- $g'_w \equiv \frac{\partial g}{\partial r_w} = \frac{r_w}{\sqrt{|\underline{r}|^2 + h^2}}$

Inserting the locations (x_i, y_i) into the above equations and knowing their values f_i as well as partial derivatives, we can solve the coefficients:

$$\mathbf{a} = \mathbf{M}^{-1} \mathbf{F} \quad (15)$$

The surface model built from these coefficients is what we will call the Gradient Influenced Multiquadric.

2.3 Optimizing the shift parameter

As in Hardy's multiquadric, the proposed new algorithm also contains a shift parameter h that can affect the overall RSM; however, because the new algorithm is built using additional information, it is probable that the surface will be less sensitive to h .

Without having access to the true function, there is no way to determine the optimum value of h . It is desirable though to automatically calculate it in a suitable way,

to remove any intervention from the user. In [22], several ways of choosing h for HMQ are reviewed, as well as the “leave out one” method which is implemented.

2.3.1 Leave out one

In [8], the authors use a bootstrapping technique to determine h . Their approach uses an error estimate technique known as “leave out one”, similar to [22].

This error estimate is evaluated by rebuilding the surface n times, leaving out one of the n data points each time. For each surface, using a left-out point, the difference is taken between the omitted true value and the interpolated value at the coordinates of that omitted point. The idea is that the average error around the area of the omitted point is approximately the error between the true and interpolated value.

The overall error of the surface, using all data points, is the root-mean-square of these individual differences.

We apply this technique to optimize the shift parameter in the new algorithm. But each surface built using a reduced sampling set, still makes use of the partial derivatives of the remaining points. Note that in this case, for each point left out we also omit the P partial derivatives. Once again, the error of the overall RSM is the root-mean-square of the individual errors found for each “reduced” interpolated surface.

2.3.2 Golden search algorithm

Once we have a method for estimating error for a given h , the next step is to find the value of h that will minimize this error. The golden search algorithm [23] is a common method that can be found in many textbooks.

The method involves evaluating the error for two different values of h . The range between them is progressively reduced in the direction of the smaller error, until the bracketing interval is sufficiently small, resulting in a quasi-optimum h .

Note that unlike the HMQ method, the h of GIMQ may not take on the value of zero. This would lead to some elements within the matrix being divided by zero.

3 Error estimation and adaptive interpolation

3.1 Error estimation

The algorithm presented in Chapter 2 improves upon Hardy’s multiquadric by making use of gradient information, so-called the GIMQ method. But once the model has been built, it is useful to have an error measure, to estimate the accuracy of the RSM.

Section 2.3.1 describes a potential way of doing this, called “leave out one”, used to optimize the shift parameter h . While this method certainly is a possible candidate for our error estimate, there is reason to explore other possibilities.

Firstly, the “leave out one” estimates the error of the surface based on a very limited number of points. Also, it does not distinguish any point as having more or less importance in its calculation, leaving the potential for a very large over-estimate of the error. As a crude example of this, observe Figure 2.

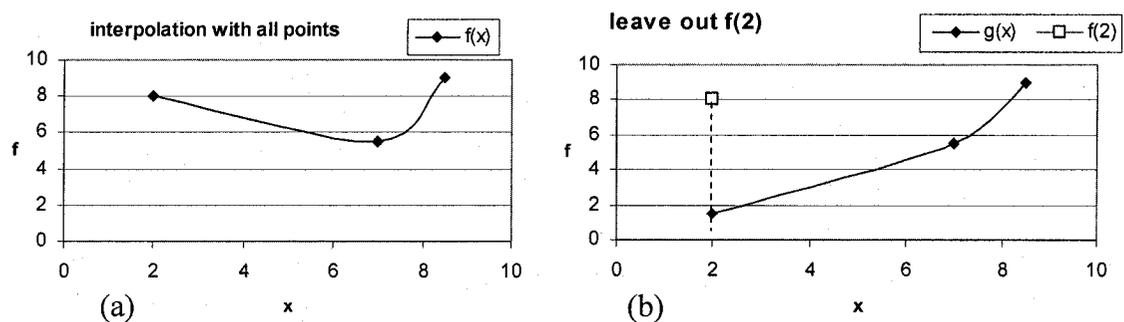


Figure 2 - Leave out one overestimating error

From the figure above, we see that the point having coordinate $x = 2$ is the most isolated. Figure 2 (a) shows the model $f(x)$ that includes all data points. Figure 2 (b) shows what the reduced model $g(x)$ without that point might look like. As we can see the

error at that point can be very large. However that point actually is available in the data set, so the contribution of this difference in the error estimate will be more significant than it should be.

Though it is more desirable to over-estimate the error than under-estimate the error, this over-estimate should still be relatively close to the true error. Because the “leave out one” uses the n sample points of the data set, it is difficult to ensure accuracy over the entire range of the surface. A better error measure would be to analyze the contribution of many points over the range of the surface.

For this reason, we have come up with an alternative method which we believe to be more accurate, and will call the “Mean Quadric Difference” (MQD). As we will see, another important benefit is this error estimate takes into consideration a large number of points, rather than the limited number of sample points: this will prove to be very useful for adaption.

In 1.2.4, we found that one way to estimate the fitting error is to take the difference between two interpolation models representing the same objective function. Since our algorithm is based on HMQ, it is convenient to use this as the second model with which to compare.

To calculate the MQD, a regular grid is built having density proportional to the number of sample points, n . This density is also dependent on the number of dimensions, P . The number of regular grid points along each direction should be greater than approximate number of sample points along that direction: to determine the number of points along each direction of the grid, we used the following formula:

$$k\sqrt[P]{n} \tag{16}$$

Here, k is an integer greater than 1 (e.g. 3, 5, 10...) which manages the number of grid points along each direction. To reduce computation time, a smaller k can be used for a larger number of dimensions.

The objective function is evaluated at the grid points using both HMQ and GIMQ. The root-mean-square of all these differences estimates the average fitting error of our new model.

Using analytical functions, it was found that the MQD represents a better measure than the “leave out one”.

Another aspect of the MQD is that because it is an RMS of the errors at many points, we remove points that do not fit within the possible constraints of the problem (e.g. $x_1 < x_2$): behavior in un-desirable regions of the surface can be discarded as irrelevant to the interpolation and the error estimation.

In addition to providing the user with an overall error measure of the interpolated model, this method can be used to estimate the error of any interpolated point: instead of taking the RMS over the surface, the difference between HMQ and GIMQ is taken at a specific point. This clearly would not have been possible with the “leave out one” method.

3.2 Adaptive sampling

The user is able to estimate the overall fitting error of the interpolated surface using the MQD. But as shown in Section 1.2.4, this type of method may also be used for adaptive sampling.

Since the MQD calculates differences over a regular grid, the coordinates where the largest deviation is found can be used as a new point to be sampled. Note that this point will not necessarily be in the region of the surface where few points exist, but will more likely be in a region having a steep gradient. Also, the coordinates of the point should fit within the constraints of the problem as mentioned in 3.1.1. This is somewhat similar to the second stage of the sampling process done in [24], as mentioned in 1.2.4. The algorithm is searching for the point which is the most “unreliably” interpolated.

We will call the error estimate at that particular point the “Peak Quadric Difference” (PQD). If one considers that the RSM is only as good as its worst point, this might be used as an alternative error measure to the MQD. The benefit of this is the user knows that no interpolated point on the regular grid, at least, will have an error greater than the PQD: it becomes a “comfort zone”. Of course, there may exist a point where the error estimate is larger than PQD, but off of the grid. Therefore, the grid should be

fine enough to find reliable coordinates of the PQD, but not so much so as to make computation time too great.

There are benefits to each error measure: while the MQD better represents the fitting error of the overall surface, the Peak Quadric Difference may be more practical to the user.

Individually, the shift parameters of the HMQ and GIMQ are still optimized using the “leave out one” method. The benefit of using the “leave out one” to optimize the shift parameter of the HMQ and GIMQ interpolations separately is that they remain independent of each other. If we were to use the MQD to optimize the shift parameter of GIMQ, the two surfaces would converge, but the GIMQ might be deviating away from the true function, making this method less accurate. Also, the GIMQ is likely to be more accurate on average, but it is conceivable that in some regions the HMQ is closer to the true objective function. Keeping the optimization of each interpolation independent allows us to remove any assumptions on the accuracy of the RSM.

Because the two RSMs are built independently, the Peak Quadric Difference is more likely to be found in the most “unreliable” region.

Finally, instead of using a fine grid, in the future we might also consider running an optimization algorithm over the parameter space to find the point having maximum quadric difference: this could potentially be a less expensive search method.

4 Results

4.1 Artificial Test Functions

4.1.1 1-D Example

In order to demonstrate the superiority of GIMQ over HMQ, let us first examine the 1-dimensional function:

$$\frac{1}{5}e^x \sin(x^2) \quad (17)$$

This function is interesting because it has some regions with rapid variation, and other regions which are flatter.

Using four scattered data points to construct the GIMQ and HMQ, we can see how including the gradients greatly improves the interpolation. The function and RSMs are plotted in Figure 3.

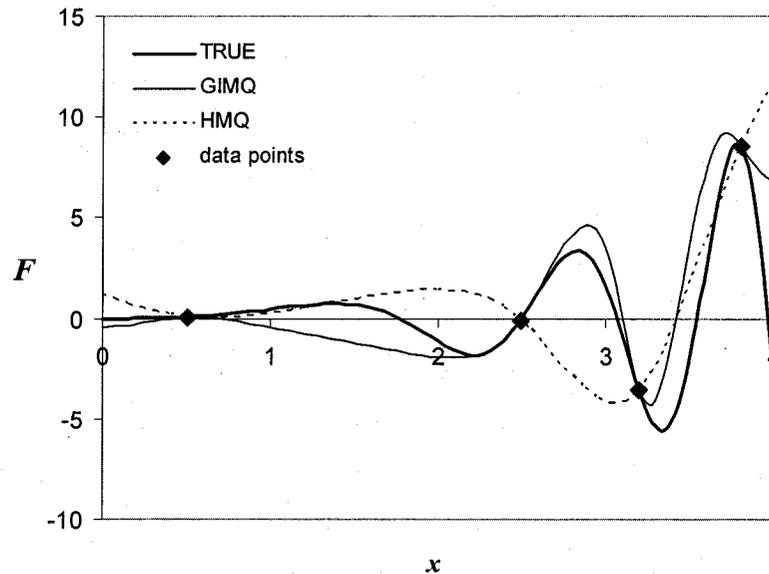


Figure 3 - HMQ and GIMQ of simple 1-D test function

From the graph, it is clear how GIMQ better approximates the true function, compared to HMQ, because it not only matches the function at the data points, but the gradients as well.

The GIMQ interpolation is able to capture the peak between data points $x = 2.5$ and $x = 3.2$ because it considers the derivatives at these coordinates. Meanwhile, HMQ completely misses this peak.

Also notable is the behavior of the RSMs beyond sample point $x = 3.8$. As we can see, because of the inclusion of the slope, the GIMQ will begin to descend, similar to the true function. HMQ, on the other hand is only trying to connect the points $x = 3.2$ and $x = 3.8$, so that it has an ascending slope: it has no information that the objective function should be decreasing.

For the adaption aspect of the algorithm, observe the behavior of the RSMs when the next four points are added one at a time at the PQD, in Figures 4 and 5.

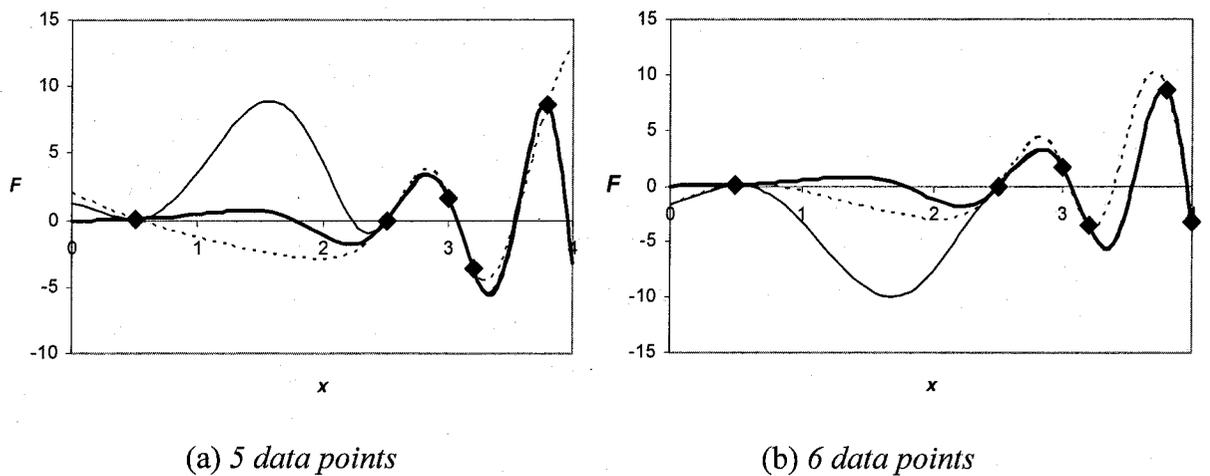
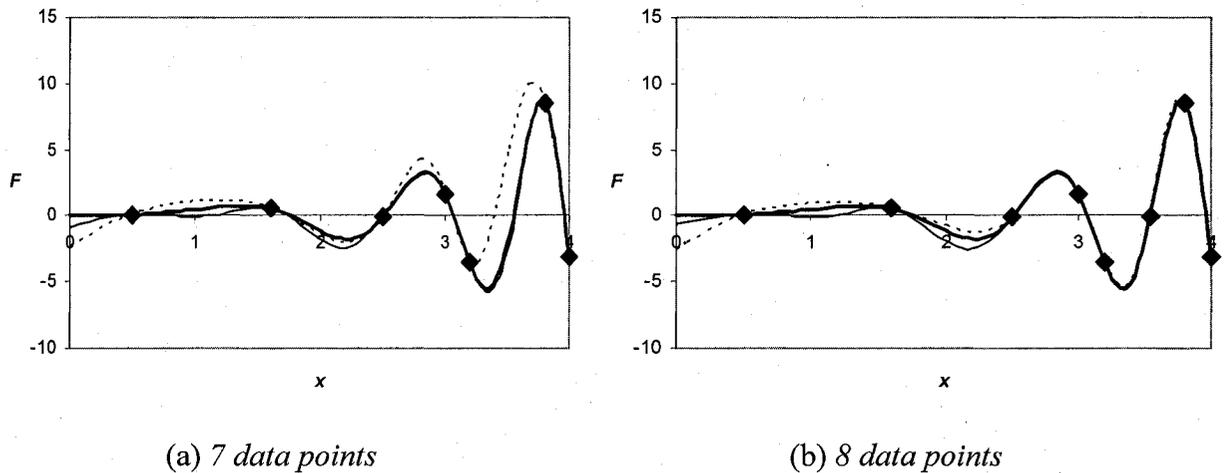


Figure 4 - RSMs with 1 and 2 points added adaptively to original four data points of Fig. 3



**Figure 5 – RSMs with 3 and 4 points added adaptively
to original data points of Fig. 3**

In Figure 3, the PQD is found at $x = 3$ (added in Fig. 4 (a)). It appears a better position for adding the new point would be around $x = 1.5$ or $x = 4$. But of course the true function is not available, and more importantly the PQD is still found in a region of rapid variation. Recall that the PQD is the point where the difference between the HMQ and GIMQ is greatest. In this case, HMQ has a greater effect on the location of the first adaptively sampled new point.

As we can see in Figures 4 and 5, most of the new points were added in regions of rapid variation. Of the four points, three were added in the upper half of the spectrum (at $x = 3, 4, 3.54$) and only one was added in the bottom half (at $x = 1.6$). Also, the GIMQ is almost indistinguishable for values above $x = 2.5$.

At this point, it is important to note the importance of the shift parameter h . In Figure 4 (a), the HMQ is closer to the true function overall, especially between $x = 0.5$ and $x = 2.5$. The GIMQ still matches the gradients at the sample points, but apparently, using the leave-one-out method has failed to generate the optimal h . We have already discussed the limitations of using the “leave-out-one” method, but due to its relative simplicity, we decided to use it. It would be interesting to explore other optimization methods for h in future research.

Ultimately, the goal of this algorithm is to use the gradients in an efficient way, and not the optimization of the shift parameter. However, to prove the superiority of GIMQ given the proper h , experimental values of the shift parameter were tested manually for the same set of data points in Figure 4 (a). We tried to manually optimize h for both the HMQ and GIMQ as best we could.

Clearly, from Figure 6, the GIMQ has more potential in providing a better interpolation of the true function.

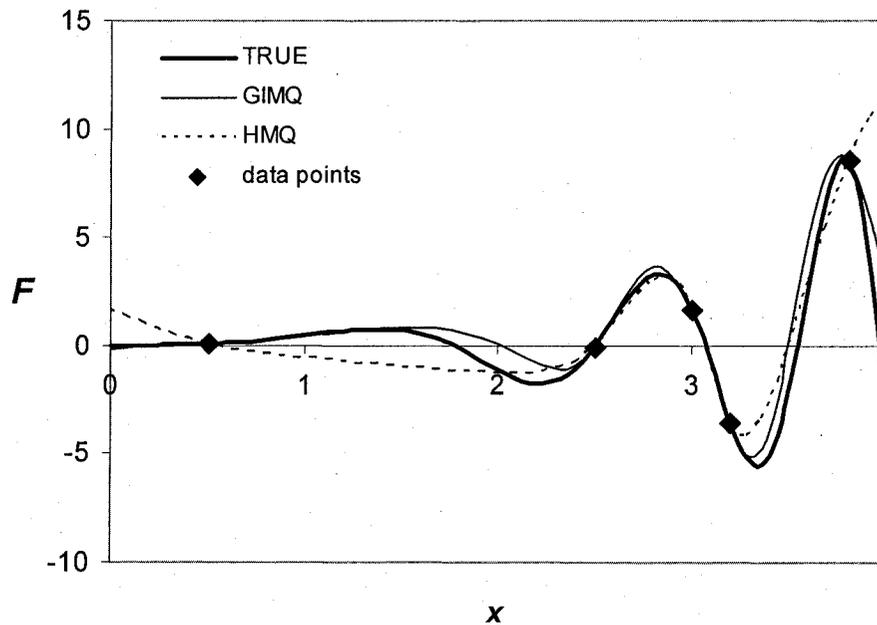


Figure 6 - RSMs for 5 data points and manually optimized shift parameters

4.1.2 2-D examples

The first 2-D objective function used to test the algorithm is one having some regions of changing variation, similar to the 1-D example. The surface is plotted in Figure 7 and represents the objective function:

$$2 \sin(xy), \quad (18)$$

between $0 < x < \pi$ and $0 < y < \pi$.

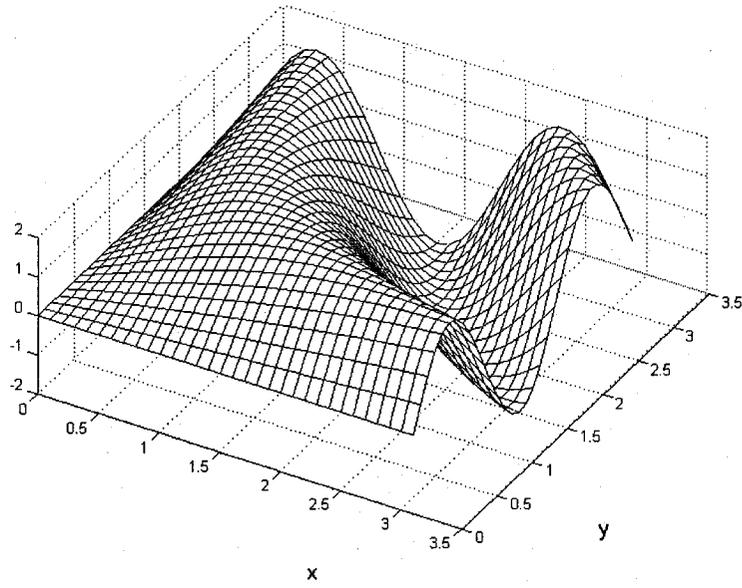


Figure 7 - Visualization of a 2-D objective function defined by Eq. (18)

Using regular 3 x 3 and 4 x 4 grids as sample points (totals of 9 and 16 data points) to build the RSMs, we can compare the difference between the HMQ and GIMQ. As is clear from figures 8 and 9, the RSM that includes gradient information is able to better represent the objective function compared to HMQ.

Perhaps the most obvious difference in Figure 8, is that the GIMQ is able to capture the peak around point (2.8, 2.8), whereas the HMQ completely misses it. Similar to our 1-D example, this is due to the gradient information at (π, π) indicating a steep descending slope.

In Figure 9 (a), we can see that using 16 points evenly spaced data points already gives a very good approximation of the objective function surface. HMQ using the same 16 points, however, is still “unrecognizable” as the true objective function, as is evident in Figure 9 (b).

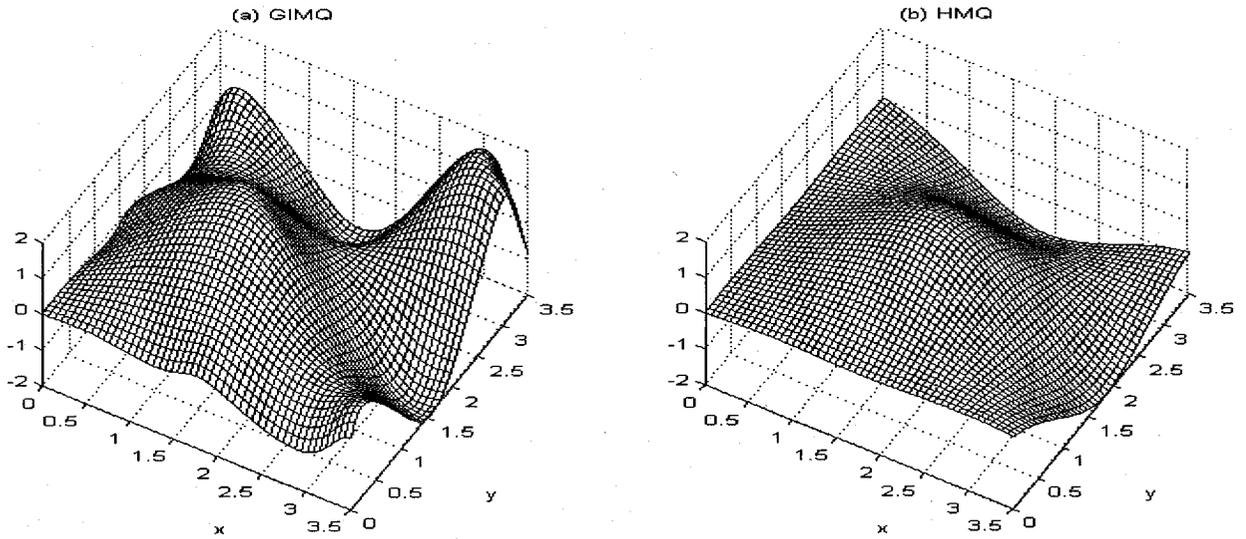


Figure 8 - RSMs with sample points spaced in 3x3 even grid

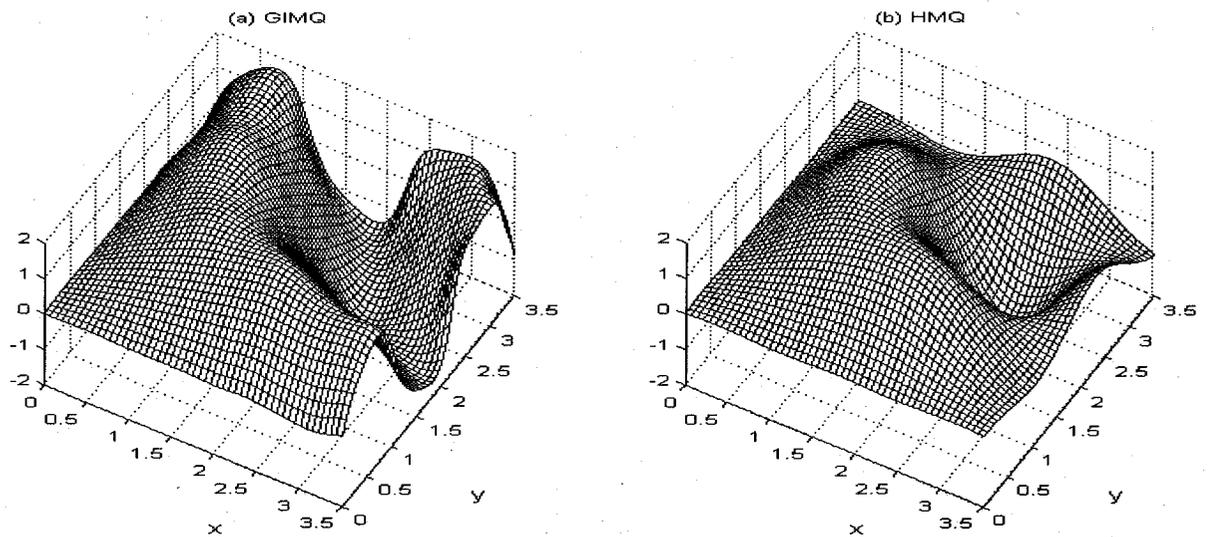


Figure 9 - RSMs with sample points are spaced in 4x4 even grid

The effect of the adaptive algorithm is shown by plotting the contour lines of the 2-D function in Figure 10. The program begins by building the RSM with only four data points, represented as squares in the corners of the sample space. Twenty points are then

adaptively added. They are indicated by circles in Figure 10, and are mostly located in regions of rapid variation (lines close together).

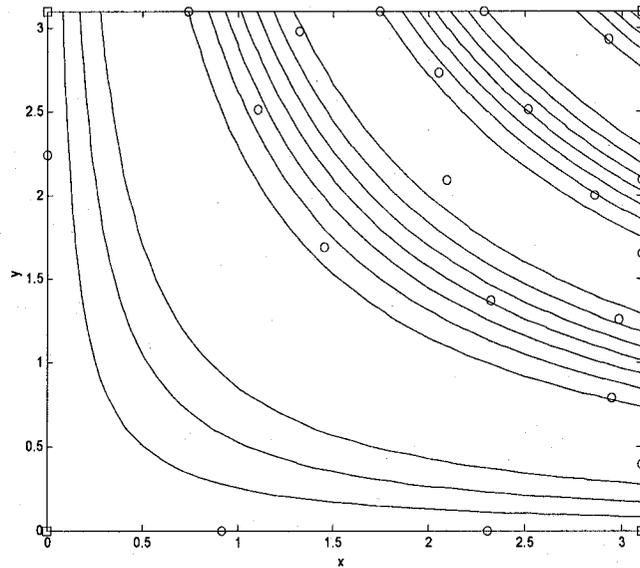


Figure 10 - A contour plot of a 2D test function. Adaptively added points indicated as circles, and initial points indicated as squares in the corners.

We now compare the errors of HMQ and GIMQ, by evaluating the RMS error of the surfaces using a 50×50 grid: this grid size was chosen so that there is not much difference in the estimation of the true error, if we were to use a finer grid.

At the same time, we can also compare these results to our errors measures, the Mean Quadric Difference (MQD) and Peak Quadric Difference (PQD). The results are shown in Figure 11. Note that the MQD is not evaluated using the 50×50 grid, but rather a grid where the number of points along each axis is defined by Equation (16) and for a k value 5: this is because the MQD results are those that would be returned to the user from the implemented program.

The RSMs use the same four initial points as in Figure 10, and the errors are evaluated with the RSMs adding an adaptively sampled point at each stage.

The first thing we notice in Figure 11 is that in the first few stages, there are still too few points to construct a reliable Response Surface. Therefore, although the RMS error of HMQ at 4 points is smaller than the GIMQ error, neither interpolation can be

deemed useful. After two or three points have been added, the Response Surfaces begin to take shape, and we can see the GIMQ out-performing HMQ. The errors also appear to diverge, as more points are added: GIMQ is converging to the true function faster than HMQ. Also, it is possible that sometimes adding a point will decrease the quality of the interpolation in areas far away from the data points. The result is an increase in the overall interpolation error.

If we observe the MQD in Figure 11, it is relatively close to the true GIMQ error; however, in the early stages, it under-estimates it. After six points have been added to the initial four, we can see the MQD over-estimating the true error, which is better. At this point, GIMQ is providing a fairly good interpolation, and more error is introduced into the Mean Quadric Difference due to HMQ.

Since it is difficult to predict when the MQD will over-estimate the error, observe the behavior of the Peak Quadric Difference in Figure 11. As we can see, the PQD is not as close to the true error of GIMQ as MQD, but at least it over-estimates this error at every stage.

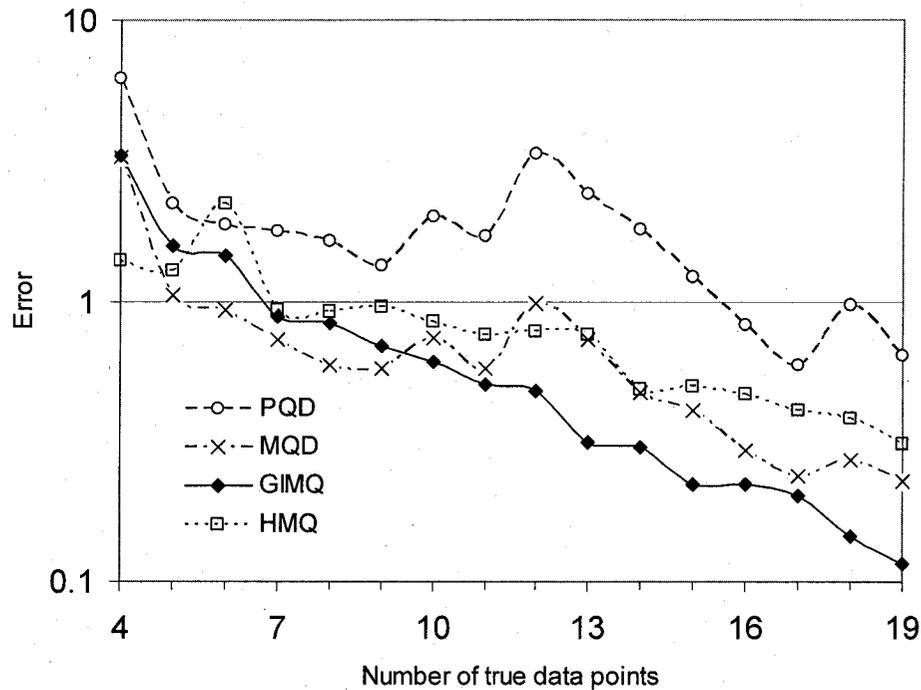


Figure 11 - Comparison of HMQ and GIMQ errors and error measures for $2\sin(xy)$

Having already demonstrated the effect of having an optimal shift parameter for our interpolation, we can now examine another aspect of the GIMQ: the positioning of the pseudo-points relative to their parent points.

In section 2.2.1, we referred to a parameter β , representing the denominator of the fixed ratio of the distance between a sample point and its nearest neighbor. For the results of this thesis, we used a β value of 5, but as we explained this may not necessarily be the optimal number for all functions, and its value will have an influence on the RSM.

To demonstrate this, we consider another objective function, still having regions of rapid variation, but also having regions flatter than those of Equation 18. The function we will use is:

$$0.2e^x \sin(y^2) \quad (19)$$

The range of the sample space is $0 < x < 4$ and $0 < y < 4$. Also, the GIMQ is built using 25 points evenly spaced in a 5×5 grid, over these ranges. The true function is plotted in Figure 12. Figure 13 shows the GIMQ using $\beta = 5$.

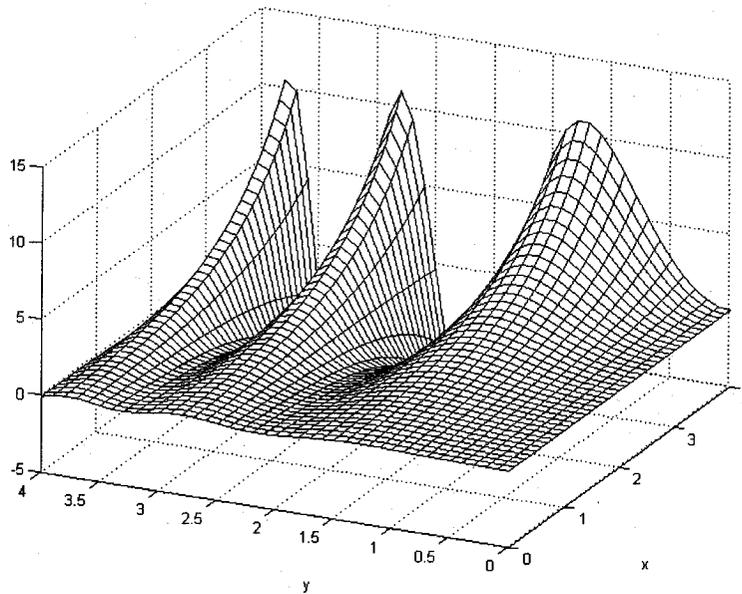


Figure 12 - Visualization of objective function defined by Eq. (16)

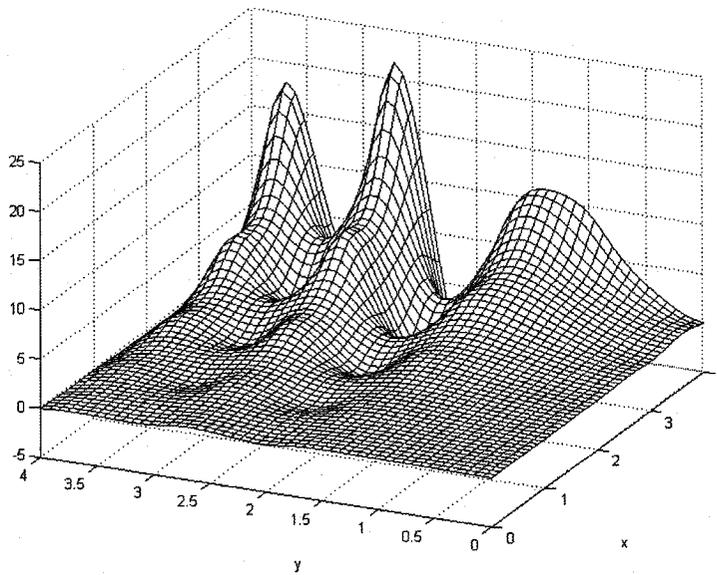


Figure 13 - GIMQ using 5x5 sample point grid and $\beta = 5$

Clearly, the RSM is capturing the general shape of the objective function, having the appearance of three peaks, with valleys in between, and a flatter region for low values of x .

We now examine the effect of setting $\beta = 3$, in Figure 14.

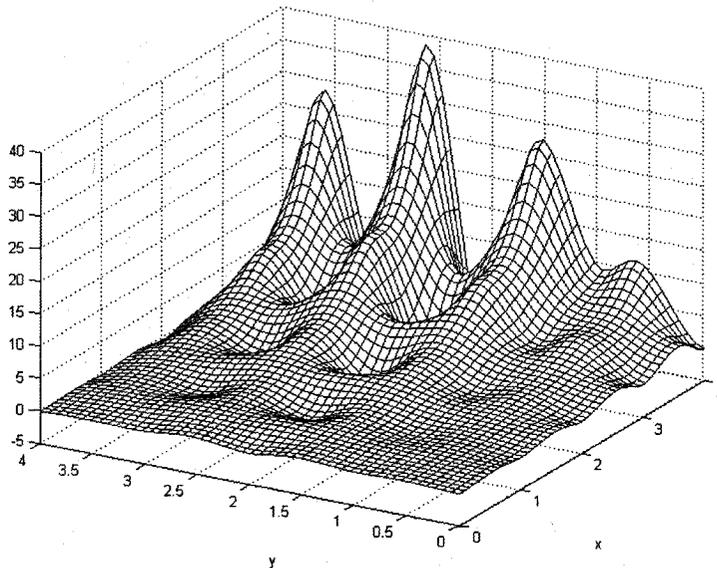


Figure 14 - GIMQ using 5x5 sample point grid and $\beta = 3$

Immediately notice this is a worse interpolation than that of Figure 13. Firstly, the peaks at $x = 4$ reach much higher values: the highest peak in the middle almost reaches 40, whereas in Figure 13, this was around $f = 22$. Its actual value in the true function should be around $f = 12$.

Also, the RSM which uses $\beta = 5$ is a much smoother surface. Figure 14 even appears to have four peaks at $x = 4$, due to this unevenness.

Finally, a β value of 10 is used in Figure 15. This is smoother than Figure 13, though less obvious than the difference between $\beta = 5$ and $\beta = 3$. Also, the peaks are even closer to the true objective function.

Using a 50 x 50 grid of points, we can calculate the RMS error the three Response Surfaces. Results are summarized in Table 1.

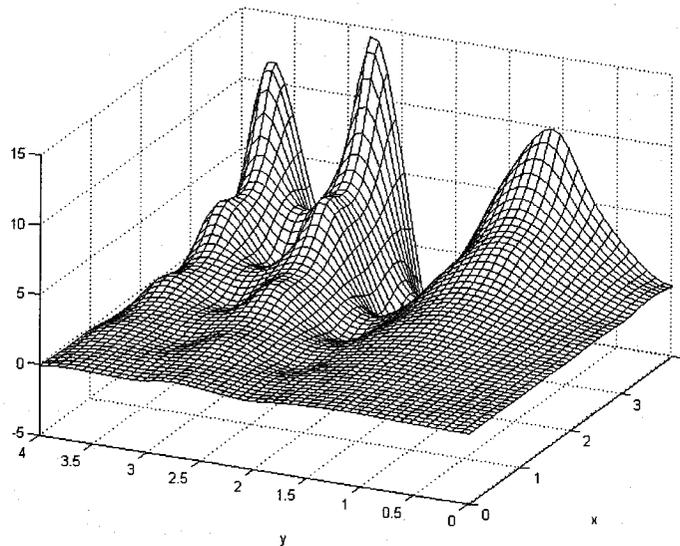


Figure 15 – GIMQ using 5x5 sample point grid and $\beta = 10$

Table 1 - RMS error of GIMQ for different values of β

β	3	5	10
<i>Error</i>	6.01668	2.89568	1.79147

As theorized in Section 2.2.1, it appears that having all data points (pseudo-points and parent points) concentrated around real information – the objective function and its gradients – results in a more accurate interpolation. As a reminder, the pseudo-points cannot be placed too close to the parent points, or the matrix will become ill-conditioned.

4.1.3 3-D Example

The final artificial test function was a function containing three variables. Using a regular grid with 3, 4 and 5 sample points along each axis (for a total of 27, 64 and 125 true points), the RSMs were built for the function:

$$5 \sin(2x + 1) \sin(2y + 1) \sin(2z + 1) \quad (20)$$

Using a 50 x 50 x 50 grid of points to sample the HMQ and GIMQ, the RMS errors were obtained. The results are shown in Figure 16.

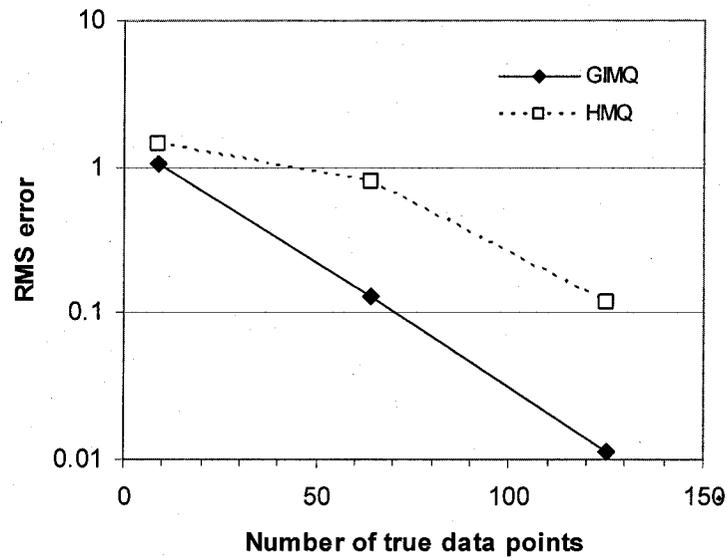


Figure 16 - RMS error of HMQ and GIMQ for 3D test function, using 3 regular grids of data points

Clearly, GIMQ provides a much better interpolation of the objective function than HMQ, and is converging faster. Also notice that at 125 points, the GIMQ error is about an order of magnitude smaller than HMQ.

4.2 Practical Examples – Waveguides

Having studied the various aspects of GIMQ and HMQ for artificial test functions, we now apply the algorithm to real problems.

As stated in the introduction, interpolation can be used to model microwave structures in a multi-dimensional space. Once the RSM has been built, it is easy to obtain a good approximation of the objective function at any coordinate. Finite Element Analysis (FEA), on the other hand, can be time consuming and inconvenient. However, FEA is able to provide gradient information at little extra cost [25]. Using all this information at several data points, a good representation of the S_{11} surface of a microwave device can be built.

We will examine two waveguide examples. The first will use theoretical formulas for the S_{11} and its gradients, in a 2-D problem. Next, the algorithm will be coupled to a Finite Element solver to study the S_{11} of a complex 3-D device.

4.2.1 Full-Height Post in Rectangular Waveguide

Consider a full-height metallic post with circular cross section inside a rectangular waveguide. The axis of the post is perpendicular to the electrical field and in the middle of the waveguide wall.

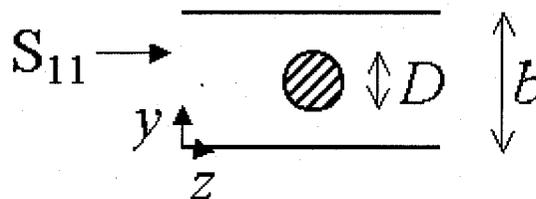


Figure 17 - Side view of metallic post in waveguide

Marcuvitz [26] provides formulas for the normalized susceptances of the equivalent pi network. Using these, it is possible to derive then equations for the magnitude and gradients of reflection coefficient S_{11} . This objective function will depend on the height of the waveguide, b , and diameter D of the post – these are normalized by the wavelength and so are dimensionless.

The derived equation for the magnitude of S_{11} is:

$$|S_{11}| = \frac{A^2 + 3}{\sqrt{A^4 + 6A^2 + \frac{4}{A^2} + 9}} \quad (21)$$

where $A = \frac{\pi^2 D^2}{2b}$.

Using Equation (21) and a very fine grid over the design space (50x50), we calculate the RMS error. In this case, actually, the 2-D design space is triangular, based on the upper limit conditions defined by Marcuvitz. Also, for the results, we avoid a divide by zero in Equation (21) by using minimum values of 0.05 and 0.015 for b and D respectively. However, the RSM could have been built over the full range of the sample space by implementing the conditions:

- $|S_{11}| = 1$, when $b = 0$
- $|S_{11}| = 0$, when $D = 0$

This results in the following limiting conditions:

$$0.05 < b < 0.2 \quad D < 0.3b \quad (22)$$

The reflection coefficient of waveguide is plotted in Figure 18.

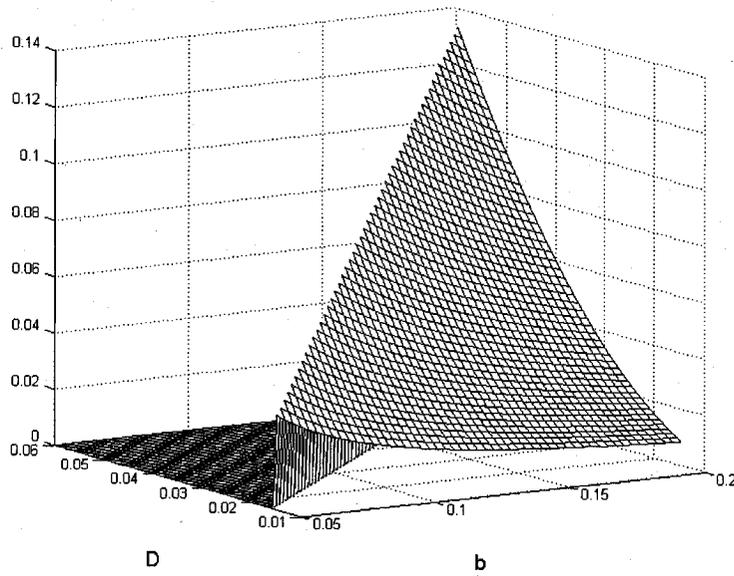


Figure 18 - $|S_{11}|$ of waveguide configuration in Fig. 17

The initial three points were placed at the corners of the design space, and the plot shows the error for up to ten adaptively added points. The results are shown in Figure 19. Again, neither the HMQ nor GIMQ are assured a monotonic convergence.

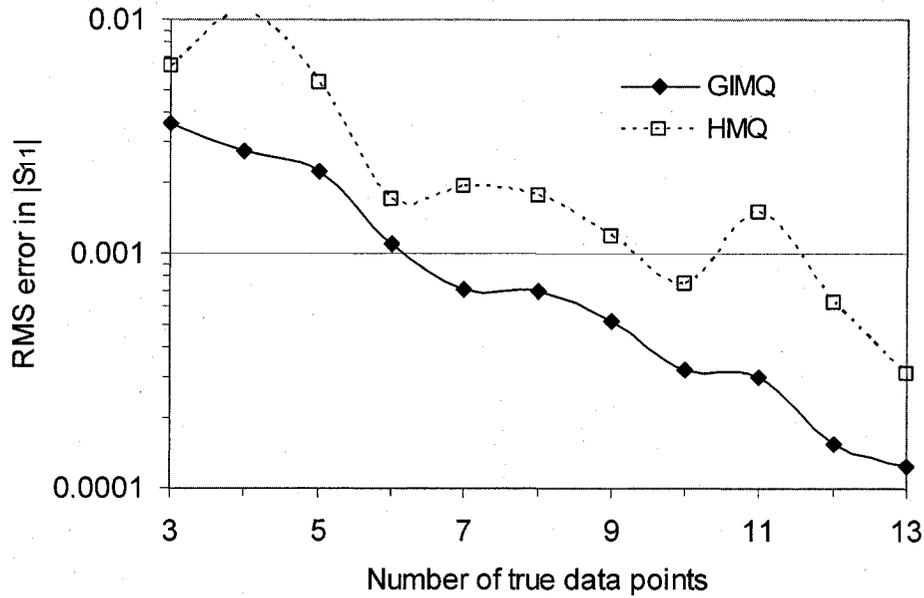


Figure 19 - Error in the interpolation of reflection coefficient for a post in a waveguide

4.2.2 Partial-Height Post in Rectangular Waveguide

Now consider a rectangular waveguide containing a post, having the configuration shown in Figure 20. Here, there are three dimensions: height (h), radius (r), position (s). There is, however, no analytical solution available for this problem. Instead, FEA can be used to solve the 3-D electromagnetic wave problem and find the reflection coefficient and its gradients [25].

The waveguide cross section has dimensions $a \times b$, and the excitation frequency is 10.3 GHz. The interpolated Response Surfaces were built within the following ranges:

$$0.05 < \frac{h}{b} < 0.95 \quad 0.05 < \frac{r}{a} < 0.22 \quad 0.16 < \frac{s}{a} < 0.84 \quad (23)$$

As well, the 3-D design space is confined to the regions where:

$$\frac{s}{a} - \frac{r}{a} > 0.05 \quad \frac{s}{a} + \frac{r}{a} < 0.95 \quad (24)$$

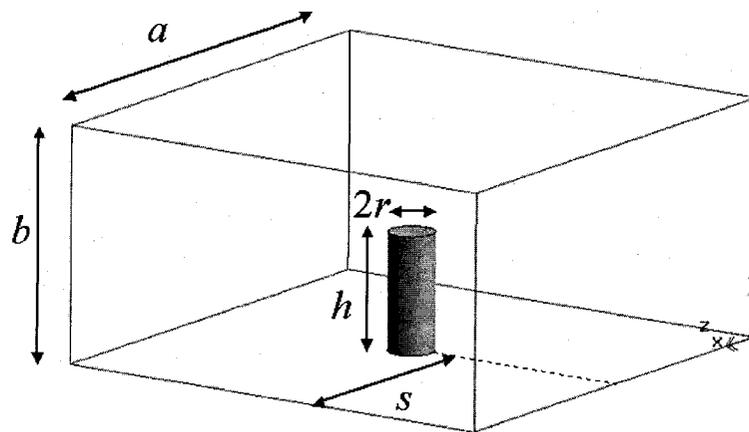


Figure 20 - Partial height round metallic post in WR-90 rectangular waveguide.

$$b = 1.016 \text{ cm}, a = 2.25b$$

To evaluate the RMS error of the interpolated surfaces, a 13x13x13 grid was used. However, it is too expensive to run the FEA at all of these points to get the true solutions as reference. Instead, the initial RSM is first built using 36 data points which

are distributed in a fairly uniform manner. The algorithm was then run until 74 points were added adaptively, for a total of 110 data points. The HMQ and GIMQ containing 110 data points was then used as the reference for calculating the RMS error of those Response Surfaces having much fewer points. The idea is that the interpolation approaches the true function as more data points are included. The results are shown in Figure 21.

The graph of results shows us that GIMQ is out-performing HMQ. In fact, the two error curves appear to be diverging, so that GIMQ is converging to its reference much more rapidly.

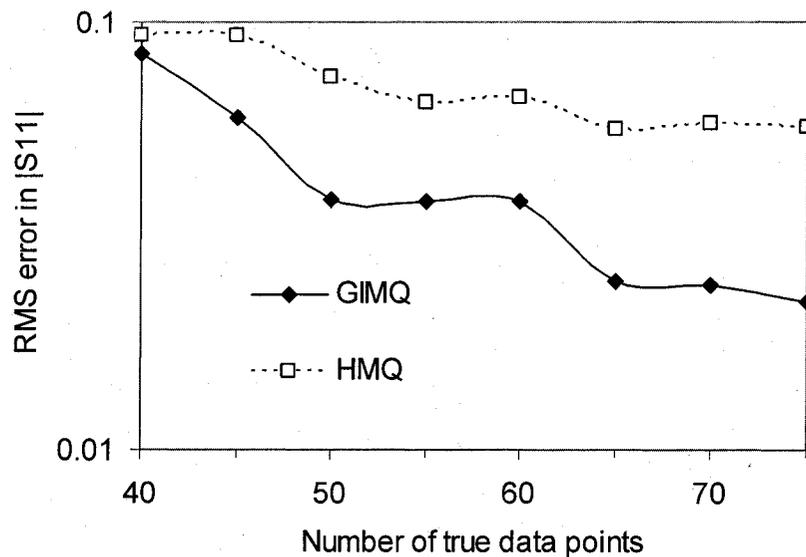


Figure 21 - Error in the interpolation of the reflection coefficient for post shown in Fig. 19

The design space considered for this problem, defined in Equation (23), was chosen so as to be a fairly smooth function. It was found that regions of resonance exist in the complete problem space, but these were excluded because interpolating these becomes much more difficult. The algorithm presented appears to be inadequate in handling these regions. To resolve this, one possibility might be to somehow include data points known to be the extremes.

To illustrate the resonance behavior, an example is presented in Figure 22. The plot shows $|S_{11}|$ as a function of the post radius, for $h = 0.8$ cm and $s = 1.125$ cm. We can see the very sharp behavior around $r = 0.665$ cm. This is equal to $r/a = 0.291$, which is out the range of our sample space, defined in Eq. (23).

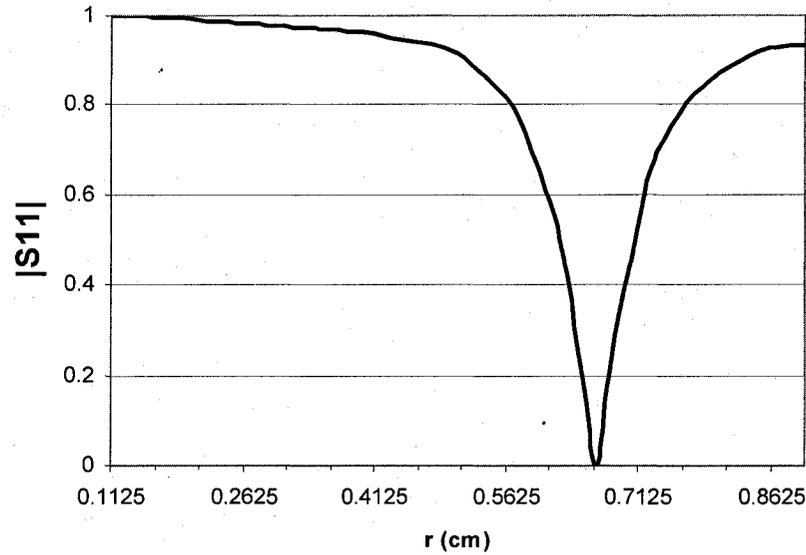


Figure 22 - $|S_{11}|$ as function of radius, showing resonance around $r = 0.665$ cm

In order to get an idea of the influence of the height and position, the reflection coefficient is plotted as a function of h and s , for $r = 0.25$ cm in Figure 23.

As a side note, considering the general behavior of the reflection coefficient as a function of the height, radius and position, this is a more difficult function to interpolate compared to the 2-D analytic waveguide problem. Instead of a simple sloping roof with a slight curvature, as in Figure 18, this waveguide problem will have some more curves, more regions of varying flatness and steepness in 3-D.

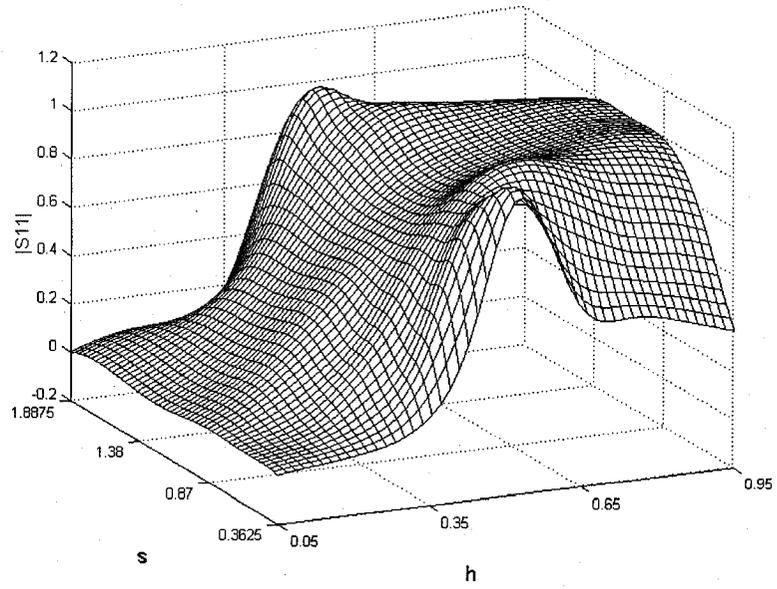


Figure 23 - $|S_{11}|$ as a function of height and position

5 Conclusion

From the results presented, it is clear that using gradients can greatly improve the quality of an interpolation. Hardy's multiquadric was already known to be one of the best multi-dimensional interpolation methods for scattered data. But when sensitivities, such as gradients, can be obtained at almost no extra cost using FEA, this information should be included in the construction of the interpolation.

Using the same basic RBF introduced by Hardy, the Gradient Influenced Multiquadric method proves to be a relatively simple yet effective way of including gradients as part of the response surface.

Additionally, the inclusion of pseudo-points in the RBFs proved to be an easy way of completing the GIMQ matrix to make it solvable. Unlike previous methods, these were attached to each sample point, with a pseudo-point for each dimension.

Another benefit of GIMQ is that combining it with HMQ leads to an easy way of determining where a new data point should be adaptively sampled. As shown in the results, this point will often appear in a region of rapid variation, so that when this point is added, knowing its slopes is that much more beneficial, since that region can be more accurately interpolated.

Because GIMQ is based on HMQ, it suffers from one of the same major difficulties: choosing an optimal shift parameter h . As explained in [22], the choice of h can greatly affect the interpolation, and the authors review several techniques for choosing it. We used an optimization technique, known as "leave out one", but it was apparent that this is not always reliable in providing the optimal h . Nevertheless, "leave out one" was fairly simple to implement, and from our experiments it appears the inclusion of gradients makes the interpolation less sensitive to h than in HMQ.

Another aspect of GIMQ that may be worth further exploration in the future is the positioning of the pseudo-points relative to the true sample points. It is not totally clear why having them closer to the data points seems to lead to a better interpolation, since

nothing is evaluated at the pseudo-points when building the matrix. But similar to h , their optimal positions will probably vary from function to function. For now, we use a fixed ratio between closest neighboring data points to position them, instead of a fixed distance.

Despite the difficulty in choosing an optimum shift parameter, GIMQ does seem to have the potential to provide a better response surface for representing an objective function than HMQ, in some cases, with many fewer sample points. It also appears that GIMQ converges to the true function much faster than HMQ when new sample points are added, presumably since more information is being included.

Finally, our examples were relatively continuous. The theoretical examples did have some regions with high gradients, but they were still smooth. We avoided problems having regions of very sharp changes in behavior, typical of resonance. As stated in the 3-D waveguide problem, GIMQ appears to have some difficulty in dealing with this: this would have been even more significant if the reflection coefficients were represented in decibels, resulting in extremely high gradients around resonance. However, the importance of the difficulty in choosing an optimum h in these problems is still unclear. On the other hand, Padé series have proven successful in modeling frequency responses of resonant structures [27]: this might be the basis for further research in improving GIMQ for resonant structures.

REFERENCES

- [1] Amidror, "Scattered data interpolation methods for electronic imaging systems: a survey", *Journal of Electronic Imaging*, vol. 11, no. 2, pp.157-176, April 2002.
- [2] D. Tsao, J. P. Webb, "Construction of device performance models using adaptive interpolation and sensitivities", p.250, Eleventh Biennial IEEE Conference on Electromagnetic Field Computation, Seoul, Korea, June 6-9, 2004.
- [3] R. Franke, "Scattered data interpolation: test of some methods", *Math. Comput.*, vol. 38, 181-200, 1982.
- [4] D. Shepard, "A two-dimensional interpolation function for irregularly-spaced data", Proc. 23th Nat. Conf. ACM, 517-523, 1968.
- [5] P. Lancaster and K. Salkauskas, *Curve and Surface Fitting*, Academic Press, London, 1986.
- [6] W.J. Gordon, J.A. Wixom, "Shepard's method of 'metric interpolation' to bivariate and multivariate interpolation", *Math. Comput.*, vol. 32, no. 141, pp. 253-264, 1978.
- [7] R.L. Hardy, "Multiquadric equations of topography and other irregular surfaces", *J. geophys. Res.*, vol. 76, 1905-1915, 1971.
- [8] P. Alotto, M. Gaggero, G. Molinari, M. Nervi, "A design of experiment and statistical approach to enhance the generalised response surface method in the optimisation of multimimima problems", *IEEE Trans. on Magnetics*, Vol. 33, No. 2, March 1997.
- [9] D.N. Dyck, D.A. Lowther, "Response surface modeling of magnetic device performance using function value and gradient", *Int. Jour. Applied Electromagnetics and Mechanics*, vol. 9, no. 3, pp. 241-248, 1998.
- [10] Z. Malik, D. Dyck, J. Nelder, R. Spence, D. Lowther, "Response surface models using function values and gradient information, with application to the design of an electromagnetic device", in *Design Reuse, Eng. Design Conf. '98*, Brunel, Uxbridge, U.K., pp. 199-209, 1998.
- [11] D. Dyck, D. Lowther, Z. Malik, R. Spence, J. Nelder, "Response surface models of electromagnetic devices and their application to design", *IEEE Trans. on Magnetics*, Vol. 35, No. 3, May 1999.
- [12] S.H. Min, S. Dalmia, M. Swaminathan, "Development of scalable models for embedded passive devices using selective sampling", *IEEE Signal Propagation on Interconnects*, 2001.
- [13] T. Dhaene, J. Ureel, N. Faché, D. De Zutter, "Adaptive frequency sampling algorithm for fast and accurate S-parameter modeling of general planar structures", in *Proc. IEEE MTT-S Symp. Dig.*, 1427-1430, 1995.

- [14] E.K. Miller, "Model-based parameter estimation in electromagnetics: Part II. Applications to EM integral equations", *IEEE Antennas Propagat. Mag.*, vol. 40, pp. 51-65, 1998.
- [15] M. Mattes, J.R. Mosig, "A novel adaptive sampling algorithm based on the survival-of-the-fittest principle of genetic algorithms", *IEEE Trans. on Microwave Theory and Techniques*, vol. 52, no. 1, 2004.
- [16] J. De Geest, T. Dhaene, N. Faché, "Adaptive CAD-model building algorithm for general planar microwave structures", *IEEE Trans. on Microwave Theory and Techniques*, vol. 47, no. 9, 1999.
- [17] A. Canova, G. Gruosso, M. Repetto, "Magnetic design optimization and objective function approximation", *IEEE Trans. on Magnetics*, vol. 39, no. 5, September 2003.
- [18] M. Farina, J.K. Sykulski, "Comparative study of evolution strategies combined with approximation techniques for practical electromagnetic optimization problems", *IEEE Trans. on Magnetics*, vol. 37, no. 5, 2001.
- [19] P. Alotto, A. Caiti, G. Molinari, M. Repetto, "A multiquadrics-based algorithm for the acceleration of simulated annealing optimization procedures", *IEEE Trans. on Magnetics*, vol. 32, no. 3, 1996.
- [20] U. Pahner, K. Hameyer, "Adaptive coupling of differential evolution and multiquadrics approximation for the tuning of the optimization process", *IEEE Trans. on Magnetics*, vol. 36, no. 4, 2000.
- [21] A. Alotto, M.A. Nervi, "An efficient hybrid algorithm for the optimization of problems with several local minima", *Int. J. Numeric. Meth. Engng*, vol. 50, no. 4, pp. 847-868, 2001.
- [22] S. Rippa, "An algorithm for selecting a good value for the parameter c in radial basis function interpolation", *Advances in Computational Math.*, vol. 11, 193-210, 1999.
- [23] W.H. Press, *Numerical recipes in C: the art of scientific computing*, Cambridge University Press, 1992.
- [24] A. Alotto, M.A. Nervi, "An efficient hybrid algorithm for the optimization of problems with several local minima", *Int. J. Numeric. Meth. Engng*, vol. 50, no. 4, pp. 847-868, 2001.
- [25] H. Akel, J. P. Webb, "Design sensitivities for scattering-matrix calculation with tetrahedral edge elements", *IEEE Trans. on Magnetics*, vol. 36, no. 4, pp.1043-1046, July 2000.
- [26] N. Marcuvitz, *Waveguide Handbook*, McGraw-Hill, 1951; section 5.13.
- [27] X.-M. Zhang, J.-F. Lee, "Application of the AWE method with the 3-D TVFEM to model spectral responses of passive microwave components", *IEEE Trans. on Microwave Theory and Techniques*, vol. 46, no. 11, pp.1735-1741, November 1998.