Sparse Grid Interpolation

Fanzi Meng

A thesis submitted for the degree of
Master of Mathematical Science (Advanced) of
The Australian National University

May, 2016

Declaration

This thesis is an account of research undertaken between February 2016 and May 2016 at The Department of Mathematics, Faculty of Science, The Australian National University, Canberra, Australia.
Except where acknowledged in the customary manner, the material presented in this thesis is, to the best of my knowledge, original and has not been submitted in whole or part for a degree in any university.

Fanzi. Meng
May, 2016

Acknowledgements

Foremost, many thanks to my supervisor Stephen Roberts, who has provided invaluable feedback and guidance throughout this project.

I am very grateful to Jouke de Baar for inspiring discussions and his warm welcome and hospitality.

I am pleased to thank all the staff in the Department of Mathematics for all help that you have provided during my two years of my study at the Australian National University.

Finally, and most importantly,I would like to express my deepest gratefulness to my beloved family for all their constant support and patience throughout my study. Without them, this thesis would have hardly been possible.

Abstract

For the approximation of multidimensional functions, using classical numerical discretization schemes such as full grids suffers the curse of dimensionality which is still a roadblock for the numerical treatment of high-dimensional problems. The number of basis functions or nodes (grid points) have to be stored and processed depend exponentially on the number of dimensions, where efficient computation are challenging in the implementation. Recently, the technique of sparse grids has been introduced to significantly reduce the cost to approximate high-dimensional functions under certain regularity conditions.
In this thesis, we present the classical sparse grid where the problem is discretized and solved on a certain sequence of conventional grids with uniform mesh sizes in each coordinate direction. Furthermore, the different types of sparse grids,i.e. Clenshaw Curtis sparse grid, have been taken into consideration to compare the accuracy and complexity of these algorithms. We then describe the sparse grid combination technique to demonstrate that it is competitive to the classical sparse grid approaches with respect to quality and run time and give proof that the interpolation by using combination approach is the classical sparse grid. We give details on the basic features of sparse grids and we consider several test problems up to dimensions. The results of numerical experiments report on the quality of approximation generated by the sparse grids, and, finally, employ the sparse grid interpolation for a real-world case to reduce a computationally expensive simulation model. We aim to obtain an efficient surrogate approximation based on a small number of simulations.

Contents

Chapter 1

Introduction

In numerical analysis, the sparse grid methods are general numerical techniques for multidimensional integration, interpolation, partial differential equations and more fields of application. The sparse grid method was evolved due to the curse of dimension: the exponational dependence of conventional approaches on the dimensionality d, a term coined in Bellmann (1961) [1]. The sparse grid method was originally introduced by the Russian mathematician Sergey A. Smolyak in 1963 [2]. Computer algorithms for efficient implementations of such grids were later developed by Michael Griebel and Christoph Zenger [3]. Compared to full grids $O(2^{nd})$, sparse grid method contains only $O(2^n \cdot n^{d-1})$ grid points during the discretization process. Under a sufficiently smooth condition, the accuracy of the approximation to describe a function f is $O(h_n^2 \log(h_n^{-1})^{d-1})$ with respect to the L_2 and L_∞ norm, if the solution has bounded second mixed derivatives, in contrast to the full grids for an accuracy of $O(h_n^2)$, in which $h_n = 2^{-n}$ represents the mesh size and n is the level of discretization [4]. This way, the curse of dimensionality, is overcome to some extent. Therefore, the sparse grid needs less points in higher dimensional spaces than conventional full sparse grids to obtain a similar approximation.
In principle, in the sparse grids, we assume that the functions to live in spaces of functions with bounded mixed derivatives instead. The sparse grid approach can be genetalized from piecewise linear basis functions to higher-order polynomials. we follow this approach. Starting from an introduction of a one-dimensional multilevel basis (see Section 2.3), preferably with an H^1- and L_2- stable one, we discuss the tensor product approach, based on the 1D multilevel bases such as the classical piecewise linear hierarchical basis. Then, if we represent a 1D function as usual as a linear combination of these basis functions, the corresponding coefficients decrease from level to level with a rate which depends on the smoothness of the function and on the given set of basis functions. From this, a multilevel basis for the higher-dimensional case (see Section 2.4) is derived from a one-dimensional multilevel basis by a simple tensor product construction. Here, 1D bases living on different levels are used in the tensor product construction, the basis functions with

anisotropic support can be obtained in the higher-dimensional case. Now, we check if the function to be expressed has bounded second mixed derivatives and we could use a piecewise linear 1D basis function as a starting point, it can be seen that the corresponding coefficients decrease with a factor proportional to $2^{-2|l|_1}$ where the multi-index $l = (l_1, \cdots, l_d)$ denotes the different levels involved. Thus, these coefficients whose absolute values are smaller than a prescribed tolerance can be omitted, we obtain sparse grids [5]. It means that the number of degrees of freedom is needed for some prescribed accuracy which no longer depends on, up to logarithmic factors, d exponentially . This allows us to obtain substantially faster solution of moderate-dimensional problems and can enable the solution of higher-dimensional problems. As i mentioned before, the sparse grid approach is not restricted to the standard piecewise linear basis functions. It can be extended to general polynomial degrees p. Also,extensions of the piecewise linear hierarchical basis to interpolates, wavelets or pre-wavelets have been successfully studied as the univariate ingredient for the tensor product construction. Finally, the sparse grid is a very widely used approach. The applications of sparse grids ranges from numerical quadrature, via the discretization of partial differential equations, to more fields such as data mining.
This thesis will first provide an overview of the principles and features of the sparse grid methods and derive the interpolation properties of the resulting sparse grid spaces. As a starting point, we use the standard piecewise linear multilevel basis in one dimension to generate higher dimensions by a tensor product construction. It is the simplest example of a multilevel series expansion which involves interpolation by piecewise linear. After that, the multilevel polynomial hierarchical bases can be employed by means of a hierarchical Lagrangian interpolation scheme. We consider the different types of sparse grids such as Clenshaw Curtis grids to analyze the quality of approximation. In Chapter 3 we present numerical results of selected experiments. To show the properties of the sparse grid approximation, we discuss four function examples from two dimensions to higher-dimensions. Furthermore, we confirm the theoretical proof of the interpolation of sparse grids (in Chapter 2) by plotting the results of numerical experiments. In Chapter 4, where we apply sparse grids to the solution of a real-world tsunami problem. We use experimental data to estimate the uncertain input parameters. We construct a surrogate-based approach to provide an inexpensive approximation of the output of the computer simulation for any parameter conguration, which enables us to estimate the parameters without further solver evaluations. The concluding remarks of Chapter 5 close this discussion of sparse grid methods.

Chapter 2

Sparse grids

2.1 Introduction

In this chapter, we will discuss the problem of interpolating smooth functions with the help of piecewise d-linear hierarchical bases. Starting from the approximation properties of sparse grids, we study a tensor product-based subspace splitting and an optimized discretization scheme can be derived. We concentrate on the L_2 -and the L_∞ norm, and to the respective types of sparse grids. In section 2.2 we introduce the finite element basis functions exemplied in approximation of functions. Section 2.3 we depict classical one-dimensional sparse grid interpolant, and Sections 2.4 covers the basic concepts and theories in mutlidimensions. In Section 2.5 we introduce the interpolation by using the combination approach. Section 2.6 describes Clenshaw Curtis sparse grid. Section 2.7 gives the proof that the combined interpolant is the hierarchical sparse grid interpolant.

2.2 Finite element basis functions

Let us start with some basic concepts while describing the conventional case of a piecewise linear finite element basis. The basis functions exemplified in approximation of functions are in general nonzero on the entire domain Ω. We turn the attention to basis functions that have compact support, meaning that functions are not zero-valued on only a restricted portion of Ω. We shall restrict the functions to be piecewise polynomials. This means that the domain is split into subdomains and the function is a polynomial on each subdomain. At the boundaries between subdomains one normally forces continuity of the function only so that when connecting two polynomials from two subdomains, the derivative becomes discontinuous.
Let V be a function space spanned by a set of basis functions $\psi_0, \cdots, \psi_N$

$$V \;=\; \text{span}\,\{\psi_0, \cdots, \psi_N\} \tag{2.1}$$

Given a function f, we wish to approximate f defined by $u \in V$. Let us divide the interval Ω on which f and u are defined into non-overlapping subintervals Ω^i, $i = 0, \cdots, N$

$$\Omega \;=\; \Omega^0 \cup \cdots \cup \Omega^N \tag{2.2}$$

We shall refer to Ω^i as an element, having number i. A set of points are introduced as nodes on each element. Nodes and elements uniquely define a finite element mesh, which is our discrete representation of the domain in the computations. A common special case is that of a uniformly partitioned mesh where each element has the same length and the distance between nodes is constant.

To produce a sufficiently accurate solution we can do refinement, giving a family of nested subspaces $V_0 \subset V_1 \subset, \ldots$. Elements of V_k are created by the refinement of level $k-1$ elements.

For each space V_k, two sets of basis functions play important roles in our discussion: the nodal basis ψ_i^k, $i = 0, \ldots, N_k$ (see Fig 2.1, bottom) and the hierarchical basis φ_i, $i = 0, \ldots, N_k$ (see Fig 2.1, top). Here, each nodal basis function is a triangle (hat) function of the same extent and the hierarchical basis functions are grouped into "levels", with the functions at the higher levels having a larger extent. We can convert between the nodal basis representation and the hierarchical basis with a simple linear matrix transform. The nodal basis $\psi_i^k \in V_k$ is defined by

$$\psi_i^k(x_j) \;=\; \delta_{ij}, \quad \delta_{ij} \;=\; \begin{cases} 1 & \text{if } i = j; \\ 0 & \text{if } i \neq j. \end{cases} \tag{2.3}$$

where x_j is a node in the mesh with global node number j. On the other hand, the hierarchical basis for V_k is built from that of V_{k-1} by adding the nodal basis functions of V_k associated with the level k nodes $x_i, i = N_{k-1}+1, \ldots, N_k$.

The basis function ψ_i holds two important properties. Firstly, a convenient interpretation of coefficients c_i as the value of u at node i

$$u(x_i) \;=\; \sum_{j \in I_i} c_j \psi_j(x_i) \;=\; \sum_{j \in I_i} c_j \varphi_j(x_i) \;=\; c_i \varphi_i(x_i) \;=\; c_i. \tag{2.4}$$

Second, for $\varphi_i(x) \neq 0$, these elements contain global node i and $\varphi_i(x)\varphi_j(x) \neq 0$ if and only if the global nodes i and j are in the same element.

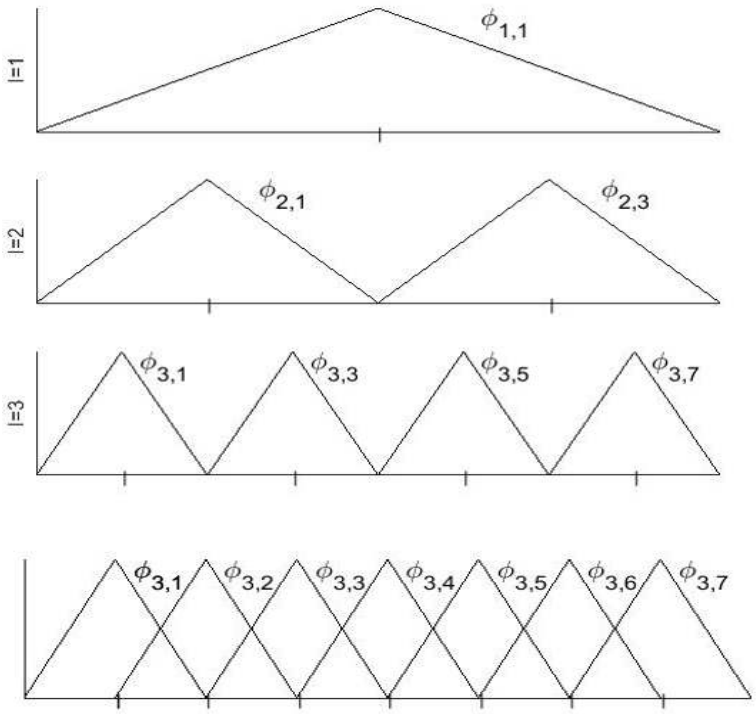

Figure 2.1: Piecewise linear hierarchical basis (top) and nodal basis (bottom) of level 3.

2.3 One-dimensional multilevel basis

In the sparse grid approach, a multidimensinal basis on the d-dimensional unit cube based on one-dimensional hierarchical basis is obtained by a tensor product construction. First, we consider a multilevel basis on one-dimensional space and introduce some notation which is necessary for a detailed discussion of sparse grids for purposes of interpolation or approximation, respectively. Let Ω_l be the equidistant grids of level l on the interval $\bar{\Omega}$ with mesh size $h_l = 2^{-l}$. This way the grid Ω_l consists of the points

$$x_{l,j} \;=\; j \cdot h_l,\; 0 \leq j \leq 2^l \tag{2.5}$$

Moreover, let V_l be the space of piesewise linear functions on grid Ω_l

$$V_l \;=\; \text{span}\left\{\phi_{l,j} : j \text{ odd},\; 1 \leq j \leq 2^l - 1\right\} \tag{2.6}$$

The basis functions $\phi_{l,j}(x)$ based on standard hat function having support $[x_{l,j} - h_l, x_{l,j} + h_l] \bigcap [0,1] = [(j-1)h_l, (j+1)h_l] \bigcap [0,1]$ are generated as

$$\begin{aligned} \phi_{l,j}(x) &= \phi\Big(\frac{x - j \cdot h_l}{h_l}\Big) \\ &= \begin{cases} 1 - \left|\frac{x - j \cdot h_l}{h_l}\right| & x \in [(j-1)h_l, (j+1)h_l] \bigcap [0,1] \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

This basis is termed nodal basis or lagrange basis. With these function spaces, the hierarchical increment spaces W_l are defined as

$$W_l = \text{span}\,\{\phi_{l,j} : j \in I_l\} \tag{2.7}$$

The index set $I_i = \big\{j \in \mathbb{N},\ j \text{ odd}, 1 \le j \le 2^l - 1\big\}$
These increment spaces allow us to write V_l as a direct sum of subspaces

$$V_l = \bigoplus_{k \le l} W_k \tag{2.8}$$

The basis corresponding to W_l is just the hierarchical basis of V_l.
such that any function $u \in V_l$ can be represented as

$$u(x) = \sum_{k \le l} \sum_{j \in I_i} \alpha_{k,j} \phi_{k,j}(x) = \sum_{k \le l} \hat{u}_k(x), \tag{2.9}$$

where $\hat{u}_k \in W_k$ and hierarchical surplus (coefficients) $\alpha_{k,j} \in \mathbb{R}$.

2.4 High-dimensional multilevel basis

A multidimensional hierarchical basis is obtained from the one-dimensional basis based on a tensor product construction. Therefore, for a multidimensional basis on the d-dimensional cube $\bar{\Omega}$, we define $\Omega_{\underline{l}}$ as anisotropic grid on $\bar{\Omega}$ with equidistant mesh size h_{l_t} in each coordinate direction t, $t = 1, \ldots, d$. Here, the $\underline{l} = (l_1, \ldots, l_d) \in \mathbb{N}^d$ is a multi-index set indicates the level of multi-dimensional sparse grids. The mesh size is denoted as $h_{\underline{l}} = (h_{l_1}, \ldots, h_{l_d}) = 2^{-\underline{l}}$. The grid points $x_{\underline{l},\underline{j}}$ of the grid $\Omega_{\underline{l}}$ are considered

$$x_{\underline{l},\underline{j}} = (x_{l_1,j_1}, \ldots, x_{l_d,j_d}) = j \cdot h_l,\ 1 \le j \le 2^l - 1 \tag{2.10}$$

Then, an associated d-dimensional piecewise d-linear basis function $\phi_{\underline{l},\underline{j}}(x)$ is constructed by the product of the resulting one-dimensional basis functions as the input

of the tensor product construction.

$$\phi_{\underline{l},\underline{j}}(x) \;=\; \prod_{t=1}^{d} \phi_{l_t,j_t}(x_t) \tag{2.11}$$

Now, each of the multidimensional basis functions $\phi_{\underline{l},\underline{j}}$ that correspond to inner grid points of Ω_l with support of the fixed size $2 \cdot h_{\underline{l}}$ are used to define an associated space $V_{\underline{l}}$

$$V_{\underline{l}} \;=\; \operatorname{span}\left\{\phi_{\underline{l},\underline{j}} \middle| j_t = 0,\ldots,2^{l_t}, t = 1,\ldots,d\right\} \;=\; \operatorname{span}\left\{\phi_{\underline{l},\underline{j}} : 1 \le j \le 2^{\underline{l}} - 1\right\}, \tag{2.12}$$

where this basis $\left\{\phi_{\underline{l},\underline{j}}\right\}$ is the standard nodal point basis of the finite dimensional space $V_{\underline{l}}$. Additionally, the hierarchical difference space $W_{\underline{l}}$ is obtained by spanning basis functions

$$W_{\underline{l}} \;=\; \operatorname{span}\left\{\phi_{\underline{l},\underline{j}} : j \in B_{\underline{l}}\right\}, \tag{2.13}$$

with the index set

$$B_{\underline{l}} \;=\; \left\{\underline{j} \in \mathbb{N}^d : 1 \le j_t \le 2^{l_t} - 1, j_t \text{ odd}, t = 1 \cdots d, \text{ if } l_t > 0\right\} \tag{2.14}$$

The hierarchical increments spaces $W_{\underline{l}}$ consist of all $\phi_{\underline{i},\underline{j}} \in V_{\underline{l}}$, which generate a new sparse grid $W_{\underline{l}'}$, any sparse grid $W_{\underline{l}}$ that meets the order relation $W_{\underline{l}} < W_{\underline{l}'}$ needs to be constructed before. Therefore, we can define a multilevel subspace decomposition and the space $V_n := V_{\underline{n}}$ can be represented as a direct sum of finite-dimensional subspaces of V

$$V_n \;=\; \bigoplus_{l_1 \le n} \cdots \bigoplus_{l_d \le n} W_{\underline{l}} \;=\; \bigoplus_{|\underline{l}|_\infty \le n} W_{\underline{l}} \tag{2.15}$$

the limit

$$\lim_{n\to\infty} V_n^{(\infty)} \;=\; \lim_{n\to\infty} \bigoplus_{|\underline{l}|_\infty \le n} W_{\underline{l}} \;=\; \bigcup_{n=1}^{\infty} V_n^{(\infty)} \;=\; V \tag{2.16}$$

where $|\underline{l}|_\infty = \max_{1\le t\le d} l_t$. V is simply the underlying Sobolev space $H_0^1(\bar{\Omega})$.
We now define a hierarchical increments space W_l via

$$W_{\underline{l}} \;=\; V_{\underline{l}} \setminus \bigoplus_{i=1}^{d} V_{\underline{l}-e_{\underline{i}}} \tag{2.17}$$

where $e_{\underline{i}}$ is the i-th unit vector. W_l consists of all $\phi_{\underline{k},\underline{j}} \in V_l$ (using the hierarchical basis) which are not included in any of the spaces V_k smaller than V_l.

Again, any function $u \in V_n$ can be uniquely represented by

$$u(\underline{x}) \;=\; \sum_{|\underline{l}|_\infty=1}^{n} \sum_{\underline{j}\in B_{\underline{l}}} \alpha_{\underline{l},\underline{j}} \phi_{\underline{l},\underline{j}}(\underline{x}) \;=\; \sum_{|\underline{l}|_\infty \le n} \hat{u}_{\underline{l}}(\underline{x}), \tag{2.18}$$

with hierarchical surplus (coefficients) $\alpha_{\underline{l},\underline{j}} \in \mathbb{R}$ in the hierarchical tensor product basis and $\hat{u}_{\underline{l}} \in W_{\underline{l}}$ is the hierarchical component functions.
We construct discrete approximation spaces that the same number of invested grid points leads to a higher order of accuracy. We deal with finite dimensional subspaces of V with the discrete spaces. First, we summarize some basic properties of the hierarchical subspaces $W_{\underline{l}}$ according to Bungartz and Griebel (1999) [6]. Concerning the subspaces W_l, we learn the dimension of $W_{\underline{l}}$ from (2.13) and (2.14), and the number of degrees of freedom (sparse grid points or basis functions) associated with $W_{\underline{l}}$:

$$|W_l| \;=\; 2^{|l-1|_1} \tag{2.19}$$

According to (2.18), the discussion of a subspace contribution to the overall interpolant could be based on the maximum norm L_∞, the L_p-norm ($p = 2$ in general) and the energy norm. Now, let us define the Sobolev-space with dominating mixed derivative H^2_{mix}. The second mixed derivatives have to be bounded

$$D^{\underline{l}}\, u \;=\; \frac{\partial^{|\underline{l}|_1} u}{\partial x_1^{l_1} \ldots \partial x_d^{l_d}}, \quad \text{where } |\underline{l}|_1 \;=\; \sum_{t=1}^{d} l_t \text{ and } |l|_\infty \;=\; \max_{1\le t\le d} l_t \tag{2.20}$$

These functions belong to a Sobolev space

$$H^2_{mix}(\Omega) \;:=\; \left\{ u : \Omega \to \mathbb{R} : D^{\underline{l}}\, u \in L_2(\Omega), |\underline{l}|_\infty \le 2, u|_{\partial\Omega} = 0 \right\} \tag{2.21}$$

Under this prerequisite, the corresponding coefficients decay rapidly $|\alpha_{\underline{l},\underline{j}}| = O(2^{-2|\underline{l}|_1})$. It follows that for the components $u_l \in W_l$ of $u \in H^2_{0,mix}(\bar{\Omega})$ from (2.18) based on L_2-norm holds [7].

$$\|u_{\underline{l}}\|_2 \le 3^{-d} \cdot 2^{-2\cdot|\underline{l}|_1} \cdot |u|_{H^2_{mix}}. \tag{2.22}$$

That means the elements are bounded and convergent because $3^{-d} \cdot 2^{-2\cdot|\underline{l}|_1}$ is less than 1 if $u \in H^2_{mix}$. It follows the hierarchical basis functions with a small support, and therefore under smoothness assumption, a small contribution to the function representation, are not included in the discrete space of level n anymore. Figure 2.2 for the 2D case shows how the supports of the basis functions of the hierarchical spaces $W_{\underline{l}}$ forming V_3.

We define the sparse grid function space $V_n^s \subset V_n$ as

$$V_n^s \;=\; \bigoplus_{|\underline{l}|_1 \le n} W_{\underline{l}} \tag{2.23}$$

Similar to (2.18), any function $u \in V_n^s$ can be uniquely written as

$$u_n^s(\underline{x}) \;=\; \sum_{|\underline{l}|_1=1}^{n} \sum_{\underline{j} \in B_{\underline{l}}} \alpha_{\underline{l},\underline{j}} \phi_{\underline{l},\underline{j}}(\underline{x}) \;=\; \sum_{|\underline{l}|_1 \le n} \hat{u}_{\underline{l}}(\underline{x}) \tag{2.24}$$

Where $u_{\underline{l}} \in W_{\underline{l}}$.
The dimension of the sparse grid space V_n^s (the number of inner grid points in the underlying grid) is given by

$$\begin{aligned} |V_n^s| &= \sum_{i=0}^{n-1} 2^i \begin{pmatrix} d-1+i \\ d-1 \end{pmatrix} \\ &= (-1)^d + 2^n \sum_{i=0}^{n-1} 2^i \begin{pmatrix} n+d-1 \\ i \end{pmatrix} (-2)^{d-1-i} \\ &= 2^n \left(\frac{n^{d-1}}{(d-1)!} + O(n^{d-2}) \right) \end{aligned} \tag{2.25}$$

This, we have

$$|V_n^s| \;=\; O(h_n^{-1} |\log_2 h_n|^{d-1}) \tag{2.26}$$

For the interpolation error of a function $f \in H_{mix}^2$ in the sparse grid space V_n^s gives

$$||f - u_n^s||_2 \;=\; O(h_n^2 \log(h_n^{-1})^{d-1}) \tag{2.27}$$

For more details and proof, we can find here Garcke (2004).

2.5 Sparse grid combination technique

A sparse grid solution obtained by a combination of anisotropic full grid solutions is often the so-called combination technique [8]. The combination technique is a multivariate extrapolation type method to achieve a function representation on a sparse grid. It exploits the approximation properties of sparse grids mentioned beforehand: the discretization of the function applies to a nodal discretization. For the solution of partial differential equations, the equations are decoupled into smaller systems on the grids and are linearly combined. Furthermore, the finite element discretization of equations asymptotic convergence properties of the combination technique can be

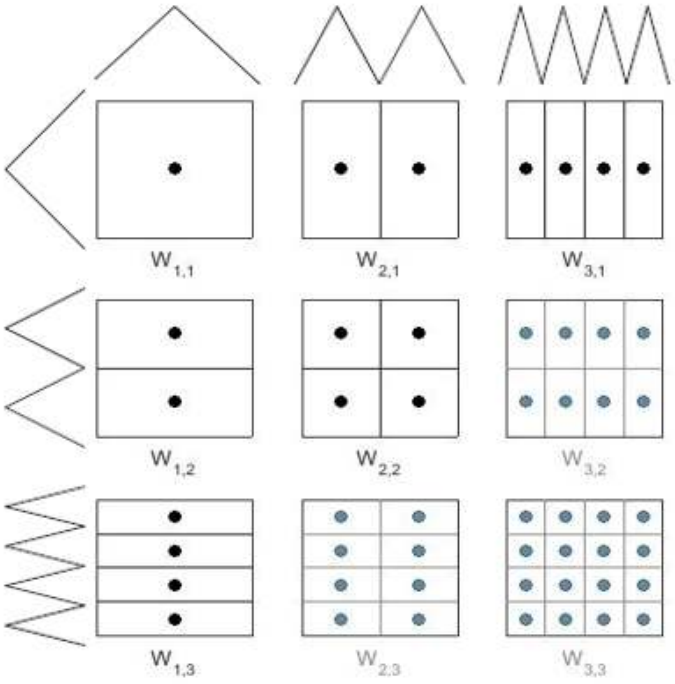

Figure 2.2: In two-dimensional space, the sparse grid space V_3^s contains the upper triangle of spaces shown in black.

preserved. The advantages of the combination technique over working directly in the hierarchical basis are that the matrix graph has considerably fewer connections and the resulting linear systems are sparse, in contrast, the stiffness matrices of sparse finite elements are not sparse and computations of the matrix-vector-product come with a high cost [9].

For the discretisation of the function space V we use a generalisation of the sparse grid combination technique. We restrict to a bounded domain $\Omega = [a, b]^d$ and consider a certain sequence of anisotropic grids $\Omega_{\underline{l}} = \Omega_{l_1,\ldots,l_d}$ which have different but uniform mesh sizes in each coordinate direction with $h_t = 2^{-l_t}$, $t = 1, \ldots, d$.

In the original combination technique considers all grids $\Omega_{\underline{l}}$ with indices

$$|\underline{l}| \;=\; l_1 + \cdots + l_d \;=\; n + (d-1) - q, \quad q \;=\; 0, \ldots, d-1, \quad l_t > 0 \tag{2.28}$$

A finite element discretisation using piecewise d-linear functions

$$\phi_{\underline{l},\underline{j}}(x) \;=\; \prod_{t=1}^{d} \phi_{l_t,j_t}(x_t) \tag{2.29}$$

on each grid $\Omega_{\underline{l}}$, where the multi-index of grid points $j_t = 0, \ldots, 2^{l_t}$.The one-dimensional basis functions $\phi_{\underline{l},\underline{j}}$ are the hat functions mentioned beforehand. In

the discrete function space $V_{\underline{l}} = \text{span}\left\{\phi_{\underline{l},\underline{j}}, j_t = 0,\ldots,2^{l_t}, t = 1,\ldots,d\right\}$ on grid $\Omega_{\underline{l}}$. A function $u_{\underline{l}} \in V_{\underline{l}}$ is represented as

$$u_{\underline{l}}(\underline{x}) \;=\; \sum_{j_1=0}^{2^{l_1}} \cdots \sum_{j_d=0}^{2^{l_d}} \alpha_{\underline{l},\underline{j}} \phi_{\underline{l},\underline{j}}(\underline{x}), \tag{2.30}$$

and uses combination coefficients to add up the partial solutions $u_{\underline{l}}$ from each grid combined to obtain the solution u_n^c on the corresponding sparse grid according to the combination formula

$$u_n^c(\underline{x}) \;=\; \sum_{j=0}^{d-1} (-1)^j \begin{pmatrix} d-1 \\ j \end{pmatrix} \sum_{|\underline{l}|=n-j} u_{\underline{l}}(\underline{x}). \tag{2.31}$$

The combination technique constructs a grid function u_n^c on a sparse grid space V_n^s, see Fig 2.3. For examples, in two dimensional space, it clearly shows that $u_n^c = \sum_{l_1+l_2=n} u_{l_1,l_2} - \sum_{l_1+l_2=n-1} u_{l_1,l_2}$. For further discussion of the combination technique see [10].

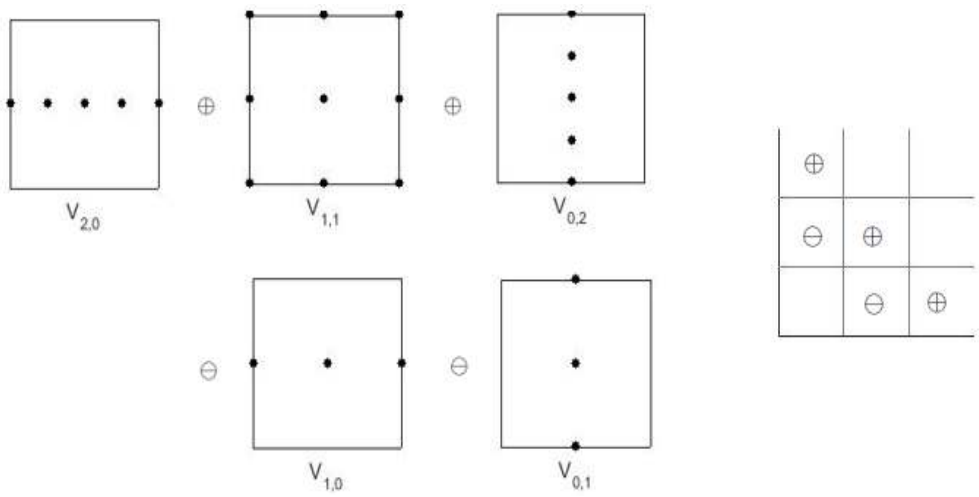

Figure 2.3: Combination technique with sparse grid level 2 in two dimension.

2.6 Clenshaw Curtis sparse grids

As we have seen, it is the hierarchical finite elements (Peano 1976) and the hierarchical bases (Yserentant 1986) to a tensor product construction with its underlying hierarchical subspace splitting that the Zenger's sparse grid concept (Zenger 1991) is based on for the numerical solution of smooth function. In this section, a very

closely related technique had been studied for purposes of approximation, or numerical integration of smooth functions. The Russian literature that has to be mentioned here is those of Smolyak (1963) studied classes of quadrature formulas of the type

$$U_n^{(d)} f \;=\; \left(\sum_{i=0}^{n} (U_i^{(1)} - U_{i-1}^{(1)}) \otimes U_{n-i}^{(d-1)} \right) f \tag{2.32}$$

where $U_n^{(d)}$ denotes a d-dimensional quadrature formula based on the 1D rule $U_n^{(1)}$ with a tensor product. Functions suitable for the Smolyak approach typically live in spaces of bounded (L_p-integrable) mixed derivatives which are closely related to our choice of u in (2.20).

For a detailed discussion of those methods, we consider a quadrature known as Clenshaw Curtis quadrature. The use of Clenshaw Curtis quadrature rule forms a sparse grid. The rules provided an indexed family have a nested set, so that all the abscissas from one rule are included in the next. The values of the abscissas and weights can be easily computed. The construction of a nested family requires that the order of the rules in the indexed family grows exponentially. We define the Clenshaw Curtis formula, denoted by CC_n as the interpolatory quadrature rule constructed on the Chebychev nodes [11].

Suppose a sparse grid constructed for a D-dimensional quadrature of function f. We consider an indexed family of underlying 1D factor quadrature rules over the interval [0,1]. The interpolating polynomial that we integrate can be expressed in a compact form as:

$$I^{(1)}(f) \;=\; \int_{\Gamma_1} f(x)dx \simeq \sum_{n=0}^{N} f(x_n) w_n \tag{2.33}$$

This is an (N + 1) points (nodes) quadrature rule having (N + 1) real values w_n called weights expressing the integral $I^{(1)}$ as a weighted sum of samples of f.

The trapezoid rule, we write the approximation of integral above as:

$$U_l^{(1)} f \;=\; \sum_{n=0}^{N_l} f(x_{n,l}) w_{n,l} \;=\; \frac{1}{2h_l} \left(f(0) + f(1) + 2 \sum_{n=1}^{N_l - 1} f(x_{n,l}) \right) \tag{2.34}$$

where $h_l = \frac{1}{2^{l-1}}$, $N_l = 2^{l-1} + 1$, $x_{n,l} = nh_l = \frac{n}{2^{l-1}}$, $w_{n,l} = \left[\frac{1}{2h_l}, \frac{1}{h_l}, \cdots, \frac{1}{2h_l}, \frac{1}{h_l} \right]$

In a Clenshaw-Curtis quadrature rule, Chebychev nodes for a given natural number N_l are: $x_{n,l} = \frac{1}{2}\left(1 - \cos \frac{\pi(n-1)}{N_l - 1}\right)$, $n = 1, \cdots, N_l$.

These quadrature rules are indexed by the variable l, which is called the 1D level. The index l begins at 0, and the nested family of Clenshaw Curtis rules is often taken to be the midpoint rule, followed by a rule of order 3 which adds the interval

endpoints, and then by rules which successively add points between each pair of points in the preceding rule [12].

A multidimensional quadrature rule is formed by the product of underlying 1D rules. Let $U_{l_i}^{(1)} : V \to \mathbb{R}$ and $f : \Omega^d \to \mathbb{R}$. It can be written as:

$$\begin{aligned}
U_l^D f(\underline{x}) &= \left(U_{l_1}^{(1)} \otimes \cdots \otimes U_{l_d}^{(1)}\right) f(x_1, \cdots, x_d) \\
&= U_{l_1}^{(1)} \otimes \cdots \otimes U_{l_{d-1}}^{(1)} \left(\sum_{n_d=1}^{N_{l_d}} f(x_1, \cdots, x_{d,n_d}) \right) \\
&= U_{l_1}^{(1)} \otimes \cdots \otimes U_{l_{d-2}}^{(1)} \left(\sum_{n_d=1}^{N_{l_d}} U_{l_{d-1}}^{(1)} f(x_1, \cdots, x_{d,n_d}) w_{l_d,n_d} \right) \\
&= U_{l_1}^{(1)} \otimes \cdots \otimes U_{l_{d-2}}^{(1)} \sum_{n_{d-1}=1}^{N_{l_{d-1}}} \sum_{n_d=1}^{N_{l_d}} f(x_1, \cdots, x_{d-1,n_{d-1}}, x_{d,n_d}) \\
&\cdots \\
&= U_{l_1}^{(1)} \left(\sum_{n_2=1}^{N_{l_2}} \cdots \sum_{n_d=1}^{N_{l_d}} f(x_1 \cdots x_{d,n_d}) w_{l_3,n_3} \cdots w_{l_d,n_d} \right) \\
&= \sum_{n_2=1}^{N_{l_2}} \cdots \sum_{n_d=1}^{N_{l_d}} U_{l_1}^{(1)} f(x_1 \cdots x_{d,n_d}) w_{l_2,n_2} \cdots w_{l_d,n_d} \\
&= \sum_{n_1=1}^{N_{l_1}} \cdots \sum_{n_d=1}^{N_{l_d}} f(x_{1,n_1} \cdots x_{d,n_d}) w_{l_1,n_1} \cdots w_{l_d,n_d}
\end{aligned} \tag{2.35}$$

where $N = \prod_{i=1}^{d} N_{l_i}$.

This product $\prod$ is a monic polynomial of degree D. The interpolation error satisfies

$$|I^D f - U^D f| \;=\; O(N_l^{-m/d}) \tag{2.36}$$

For function f in the space

$$W_{mix}^{m,\infty}([0,1]^d) = \left\{ f : [0,1]^d \to \mathbb{R};\ \max_{|\underline{l}| \le m} \| \frac{\partial^{|\underline{l}|} f}{\partial x_1^{l_1} \ldots \partial x_d^{l_d}} \|_\infty \le \infty \right\} \tag{2.37}$$

where d is the dimensions and m is the bounded value. The 1-D nodal points are $\Theta_l^{(1)} = \{x_{l,1}, \cdots, x_{l,d}\}$. The sparse grid nodal set is:

$$\Theta_l^{(D)} \;=\; \bigcup_{|\underline{l}| \le l+d-1} \Theta_{l_1}^{(1)} \times \cdots \times \Theta_{l_d}^{(1)} \tag{2.38}$$

We define the difference relations as $\Delta_l^{(1)} f = \left(U_l^{(1)} - U_{l-1}^{(1)}\right) f$. Thus, the sparse grid multidimensional quadrature rule can be expressed as:

$$U_l^{(D)} f \;=\; \sum_{|\underline{l}| \leq l+d-1} \left(\Delta_{l_1}^{(1)} \otimes \cdots \otimes \Delta_{l_d}^{(1)}\right) f \tag{2.39}$$

We say that this product rule has a product level of $|\underline{l}| = \sum_{i \leq D} l_i$, where $\underline{l} = (l_1, \cdots, l_d) \in \mathbb{N}^d$ is a multi-index.
A sparse grid can be indexed by the sparse grid level l that uses weighted combinations of those product rules. The lowest sparse grid level is taken to be 0. So the sparse grid of sparse grid level 0 is equal to the product rule of product level 0. To construct the sparse grid in a Clenshaw-Curtis quadrature rule, we have:

$$A(l, D) \;=\; \sum_{l-D+1 \leq |\underline{i}| \leq l} (-1)^{l-|\underline{i}|} \begin{pmatrix} D-1 \\ l-|\underline{i}| \end{pmatrix} \left(U_{l_1}^{(1)} \otimes \cdots \otimes U_{l_d}^{(1)}\right) \tag{2.40}$$

Formally, to describe a sparse grid using the Clenshaw Curtis rule is to substitute CC_{l_i} for each generic quadrature rule U_{l_i}. Consider the specific formulas for sparse grid level one in dimension two which are products of the 1D Clenshaw Curtis rules:

$$\begin{aligned} A(1,2) = CC_1 \otimes CC_0 \\ + CC_0 \otimes CC_1 \\ - CC_0 \otimes CC_0 \end{aligned} \tag{2.41}$$

2.7 Proof of combination formula

We discuss the relationship of hierarchical sparse grid interpolation and interpolation by using combination technique. It is shown that the combined interpolation is identical with the hierarchical sparse grid interpolation (J Garcke, 2012). The proof can be seen by rewriting each $u_{\underline{l}}$ in their hierarchical representation and some straightforward calculation using the telescoping sum. Moreover, the proof is extended from two and three dimensions to the high-dimensional cases. To demonstrate the advantages of the combination technique over working directly in the hierarchical basis, we consider certain equation problems with numerical experiments (Chapter 3). The outputs of numerical implementation are provided to compare these two interpolations and estimate computational accuracy.
For a given function u the interpolant u_n^c using the combination technique (2.31) is the hierarchical sparse grid interpolant u_n^s from (2.18).

Proof. We start in a two dimensional case. We can have

$$u_n^s = \sum_{k_1+k_2\leq n} \hat{u}_{k_1,k_2}$$

and according to the combination formula, we write

$$u_n^c = \sum_{|\underline{l}|=n} u_{\underline{l}} - \sum_{|\underline{l}|=n-1} u_{\underline{l}}$$

We learned in the equations (2.9) and (2.24), we can rewrite the function u_l in two dimensional space as

$$u_l(x) = \sum_{|\underline{k}|\leq l} \sum_{\underline{j}\in B_{\underline{k}}} \alpha_{\underline{k},\underline{j}} \phi_{\underline{k},\underline{j}}(\underline{x}) = \sum_{|k|\leq l} \hat{u}_{k_1,k_2}$$

Therefore, following the equation, we now define

$$\sum_{|l|=n} u_{\underline{l}} = \sum_{|l|=n} \sum_{|k|\leq l} \hat{u}_{k_1,k_2}, \text{ and } \sum_{|l|=n-1} u_{\underline{l}} = \sum_{|l|=n-1} \sum_{|k|\leq l} \hat{u}_{k_1,k_2}$$

For the combined interpolant we get as in

$$\begin{aligned}
u_n^c &= \sum_{|l|=n} \sum_{|k|\leq l} \hat{u}_{k_1,k_2} - \sum_{|l|=n-1} \sum_{|k|\leq l} \hat{u}_{k_1,k_2} \\
&= \sum_{l_1+l_2=n} \sum_{k_1\leq l_1} \sum_{k_2\leq l_2} \hat{u}_{k_1,k_2} - \sum_{l_1+l_2=n-1} \sum_{k_1\leq l_1} \sum_{k_2\leq l_2} \hat{u}_{k_1,k_2} \\
&= \sum_{l_1\leq n} \sum_{k_1\leq l_1} \sum_{k_2\leq n-l_1} \hat{u}_{k_1,\underline{k_2}} - \sum_{l_1\leq n-1} \sum_{k_1\leq l_1} \sum_{k_2\leq n-l_1-1} \hat{u}_{k_1,\underline{k_2}} \\
&= \sum_{l_1=n} \sum_{k_1\leq l_1} \sum_{k_2=0} \hat{u}_{k_1,\underline{k_2}} + \sum_{l_1\leq n-1} \sum_{k_1\leq l_1} \left(\sum_{k_2\leq n-l_1} \hat{u}_{k_1,\underline{k_2}} - \sum_{k_2\leq n-l_1-1} \hat{u}_{k_1,\underline{k_2}} \right) \\
&= \sum_{l_1=n} \sum_{k_1\leq l_1} \sum_{k_2=0} \hat{u}_{k_1,\underline{k_2}} + \sum_{l_1\leq n-1} \sum_{k_1\leq l_1} \sum_{k_2=n-l_1} \hat{u}_{k_1,\underline{k_2}} \\
&= \sum_{l_1\leq n} \sum_{k_1\leq l_1} \sum_{k_2=n-l_1} \hat{u}_{k_1,\underline{k_2}} \\
&= \sum_{k_1\leq n-k_2} \sum_{k_2\leq n} \hat{u}_{k_1,k_2} \\
&= \sum_{k_1+k_2\leq n} \hat{u}_{k_1,k_2}
\end{aligned} \tag{2.42}$$

where $\hat{u}_{\underline{k}} = \hat{u}_{k_1,k_2} = \hat{u}_{k_1,\underline{k}\setminus k_1} = \hat{u}_{k_1,\underline{k_2}}$ [13]. Thus, the expression of interpolant using combination technique u_n^c is exactly the same as the hierarchical sparse grid interpolant u_n^s in two dimensional case. We apply mathematical induction method

for higher dimensional spaces in terms of the statement (2.42), known as the base case. By the principle of induction, the lemma is true in any dimensions. Recall the combination technique formula

$$u_n^c(\underline{x}) = \sum_{k=0}^{d-1}(-1)^k \begin{pmatrix} d-1 \\ k \end{pmatrix} \sum_{|\underline{l}|=n-k} u_{\underline{l}}(\underline{x})$$

We note that the coefficients (the numbers in front of each $u_{\underline{l}}$ term) follow a pattern

$d=2$:					1		-1				
$d=3$:				1		-2		1			
$d=4$:			1		-3		3		-1		
$d=5$:		1		-4		6		-4		1	
$d=6$:	1		-5		10		-10		5		-1

This sequence is known as Pascal's triangle. Each of the numbers is found by adding together the two absolute numbers directly above it and put minus sign if the number k is odd.

Thus, the combination function u_n^c when $d = n$ can be expressed by the function at $d = n-1$. Any combination technique functions can be rewritten as two standard terms with respected to the previous function. Then, according to the principle of induction, we prove $u_n^c = u_n^s$ is true for $d = n$. We get the general form by using $\left\{\mathbb{Z}^d \to \mathbb{Z}^{d-1} | l_1, \ldots, l_d \to l_2, \ldots, l_d\right\}$ in the following way

$$\begin{aligned} u_n^c &= \sum_{k=0}^{d-1}(-1)^k \begin{pmatrix} d-1 \\ k \end{pmatrix} \sum_{|\underline{l}|=n-k} u_{\underline{l}} \\ &= \left(\sum_{k=0}^{d-2}(-1)^k \begin{pmatrix} d-2 \\ k \end{pmatrix} \sum_{|\underline{l}|=n-k} u_{\underline{l}}\right) - \left(\sum_{k=0}^{d-2}(-1)^k \begin{pmatrix} d-2 \\ k \end{pmatrix} \sum_{|\underline{l}|=n-k-1} u_{\underline{l}}\right) \\ &= \left(\sum_{k=0}^{d-2}(-1)^k \begin{pmatrix} d-2 \\ k \end{pmatrix} \sum_{|\underline{l}|=n-k} \sum_{|\underline{k}|\leq l} \hat{u}_{\underline{k}}\right) - \left(\sum_{k=0}^{d-2}(-1)^k \begin{pmatrix} d-2 \\ k \end{pmatrix} \sum_{|\underline{l}|=n-k-1} \sum_{|\underline{k}|\leq l} \hat{u}_{\underline{k}}\right) \end{aligned} \tag{2.43}$$

$$
\begin{aligned}
&= \left(\sum_{k=0}^{d-2} (-1)^k \begin{pmatrix} d-2 \\ k \end{pmatrix} \sum_{l_1=0}^{n} \sum_{|\underline{l_2}| \le n-k-l_1} \sum_{|\underline{k}| \le l} \hat{u}_{\underline{k}} \right) \\
&- \left(\sum_{k=0}^{d-2} (-1)^k \begin{pmatrix} d-2 \\ k \end{pmatrix} \sum_{l_1=0}^{n-1} \sum_{|\underline{l_2}| \le n-k-l_1-1} \sum_{|\underline{k}| \le l} \hat{u}_{\underline{k}} \right) \\
&= \sum_{l_1=0}^{n-1} \sum_{k_1 \le l_1} \left(\sum_{k=0}^{d-2} (-1)^k \begin{pmatrix} d-2 \\ k \end{pmatrix} \sum_{|\underline{l_2}| \le n-k-l_1} \sum_{|\underline{k}| \le l} \hat{u}_{\underline{k}} - \sum_{k=0}^{d-2} (-1)^k \begin{pmatrix} d-2 \\ k \end{pmatrix} \sum_{|\underline{l_2}| \le n-k-l_1-1} \sum_{|\underline{k}| \le l} \hat{u}_{\underline{k}} \right) \\
&+ \sum_{l_1=n} \sum_{|\underline{l_2}|=0} \hat{u}_{\underline{k}} \quad \text{(Where } u_{\underline{l}} = \hat{u}_{\underline{l}} = 0 \text{ if any if } l_i < 0,\ \underline{l} = (l_1, \cdots, l_d)\text{)} \\
&= \sum_{l_1=0}^{n-1} \sum_{k_1 \le l_1} \left(\sum_{k_2+\cdots+k_d \le n-l_1} \hat{u}_{k_1,\underline{k_2}} - \sum_{k_2+\cdots+k_d \le n-l_1-1} \hat{u}_{k_1,\underline{k_2}} \right) + \sum_{l_1=n} \sum_{k_1 \le l_1} \sum_{k_2+\cdots+k_d=0} \hat{u}_{k_1,\underline{k_2}} \\
&\qquad\qquad \text{(obtained from induction hypothesis)} \\
&= \sum_{l_1=0}^{n-1} \sum_{k_1 \le l_1} \sum_{k_2+\cdots+k_d=n-l_1} \hat{u}_{k_1,\underline{k_2}} + \sum_{l_1=n} \sum_{k_1 \le l_1} \sum_{k_2+\cdots+k_d=0} \hat{u}_{k_1,\underline{k_2}} \\
&= \sum_{l_1=0}^{n} \sum_{k_1 \le l_1} \sum_{k_2+\cdots+k_d=n-l_1} \hat{u}_{k_1,\underline{k_2}} \\
&= \sum_{k_2+\cdots+k_d \le n} \sum_{k_1 \le n-k_2-\cdots-k_d} \hat{u}_{\underline{k}} \\
&= \sum_{k_1+\cdots+k_d \le n} \hat{u}_{\underline{k}}
\end{aligned}
\tag{2.44}
$$

Chapter 3

Numerical experiments

In the preceding section, we theoretically proved that the combined interpolant is identical with the hierarchical sparse grid interpolant. However, to demonstrate the equivalency of these two interpolants and identify properties and patterns, the way is to make numerical experiments. In this section, we report a collection of numerical results and estimate the quality of the approximations for different problems solved on the hierarchical sparse grids and the combination approach. We start with the discussion of the basic interpolation properties of sparse grid methods applied to a simpler 2D model problem. Then we turn to the approximation of the Rosenbrock function, Gaussian equation and Checkboard on sparse grids in higher dimensionality. For measuring the error, we consider the errors discrete maximum norm and the discrete L_2-norm on grids. In low dimensionalities, we can compute the error terms numerically. In higher dimensionalities we can still approximate the error stochastically with Monte Carlo or quasi-Monte Carlo methods. Since the curse of a fixed grid in D dimensions requires N^D points where are too difficult to implement, the Monte Carlo method is both interesting and useful for error estimation of a higher dimensionality. Monte Carlo methods are usually presented as estimates of averages which in turn are integrals, $\frac{1}{N}\sum_{i=1}^{N}\frac{|f(X_i)-u(X_i)|^2}{u(X_i)} \approx \int |f(X_i) - u(X_i)|^2$. The error typically decreases proportionally to $\frac{1}{\sqrt{N}}$. In the context of solvers, it is important that the influence of the sparse grid level on the accuracy of the interpolants. Thus, we attempt to find out a suitable level in which we can obtain a high quality approximation and estimate the rate of convergence of prediction errors for an increasing sparse grid level. Moreover, Let us now visualize the sparse grid. Our model problems cover the classical sparse grids and the type of Clenshaw-Curtis grids. We compare the accuracy of the approximations of these two types of sparse grids and investigate the efficiency of the performance. For our numerical tests we used the MATLAB implementation of sparse grid package done in the hierarchical subspaces can be found here [14]

In the following we will consider the following four functions:

$$f(x,y) \;=\; \sin(10x) + \sin(10y), \tag{3.1}$$

with the domain $[0,1]^2$.

$$f(x_i) \;=\; \sum_{i=1}^{d-1} \left(100 \times (x_{i+1} - x_i^2)^2 + (x_i - 1)^2\right), \tag{3.2}$$

with the domain $[-2,2]^d$.

$$f(x_i) \;=\; exp\left(-\sum_{i=1}^{d} \frac{x_i - \mu}{2\sigma^2}\right), \tag{3.3}$$

with the domain $[0,1]^d$.
A discontinues function with a 2×2 checkerboard pattern with the domain $[0,1]^2$

$$f(x_i) \;=\; \begin{cases} exp\left(-\sum_{i=1}^{d} \frac{x_i-\mu}{2\sigma^2}\right), & \text{if } x_i \in [0,0.5]^2, or\ x_i \in [0.5,1]^2; \\ -exp\left(-\sum_{i=1}^{d} \frac{x_i-\mu}{2\sigma^2}\right), & \text{otherwise.} \end{cases} \tag{3.4}$$

3.1 Sine function

The first test function is a low dimensional Sine function, defined on the unit hypercube $[0,1]^2$. It is a simple and well-defined function.

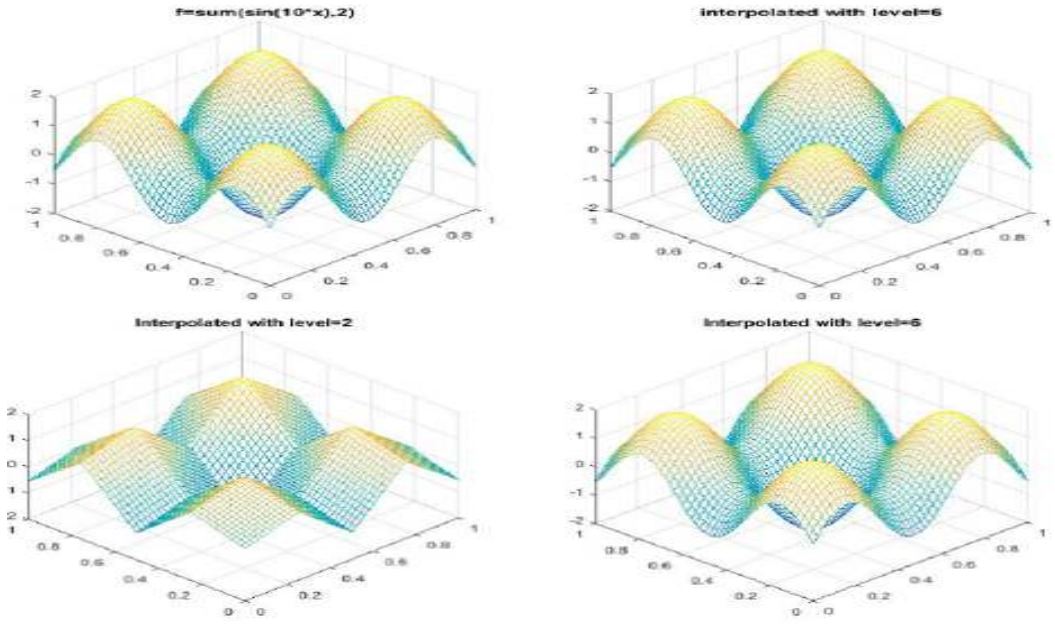

Figure 3.1: The exact solution and the hierarchical sparse grids interpolant u_n^s (on the top).The combined interpolant u_n^c with level 2 and 6 (on the bottom).

Fig 3.1 shows the approximations of the exact solution of (3.1) generated by the hierarchical sparse grid interpolant and combined interpolant. We study the accuracy of the hierarchical Lagrangian approach for the classical sparse grids and combination approach. For the same grid level 6, the hierarchical sparse grids interpolant (the upper-right corner) is equivalent to the interpolant by using combination approach (the lower-right corner). Moreover, Fig 3.1 visually illustrates the effect of the choice of sparse grid levels that the combined interpolant with $level = 2$ generates a lower accurate and rough approximation than the combined interpolant with $level = 6$. A higher accurate approximation can be obtained based on the sparse grid depth (level).

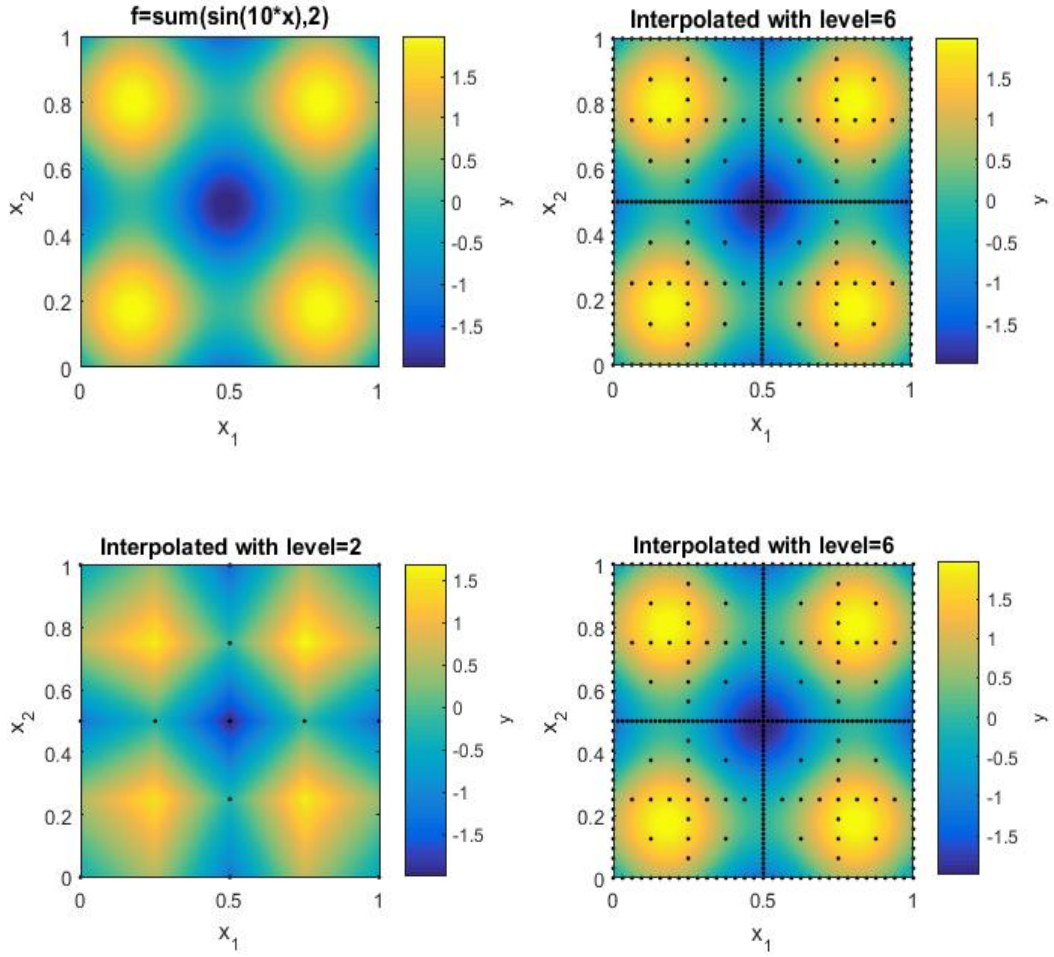

Figure 3.2: The contour plots of exact solution (in the upper-left coner) and the hierarchical sparse grids interpolant (in the upper-right coner); In the second row, the contour plots of the combined interpolants with level=2 and level=6, respectively.

In Fig 3.2, the sparse grid points visualized on the contour plots of the interpolants, we see that is the classical uniform sparse grid. For two dimensional case, the number of grid points of $level = 2$ is 13 points, and of $level = 6$ is 321 points, respectively. The contour plots of Clenshaw Curtis sparse grid have the same number

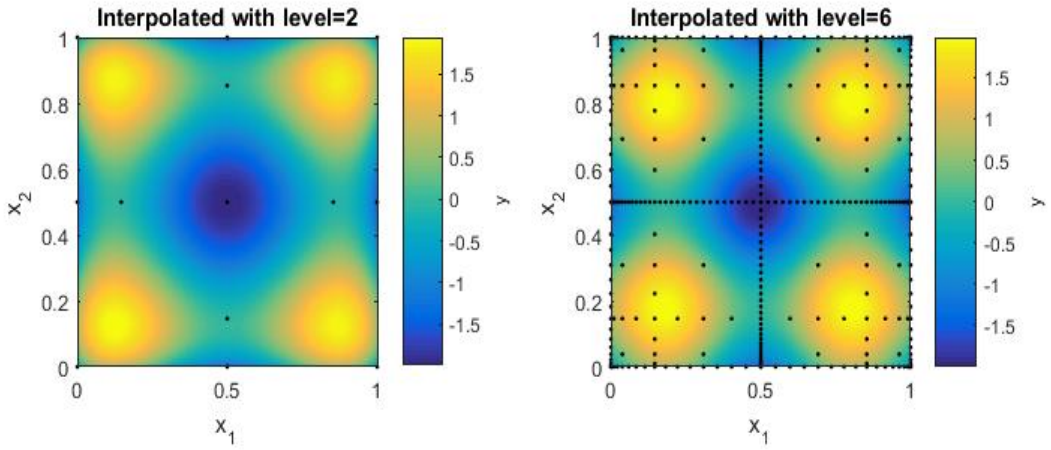

Figure 3.3: The contour plots of Clenshaw Curtis sparse grids interpolant with level=2 (the left hand side) and of Clenshaw Curtis sparse grids interpolant with level=6 (the right hand side).

of grid points compared with the classical grid, but it constructed on the Chebychev nodes, see Fig 3.3. Fig 3.4 and Fig 3.6 show the absolute error of the solution u_n^s that was computed on the hierarchical sparse grid and the error that was obtained by the combination approach u_n^c. As expected, the error on the hierarchical sparse grid and on the combined grids have the same behaviour and size. In contrast to that, the absolute error of combined solution with $level = 2$ shows large errors, see Fig 3.5. The comparison of interpolation error for $level = 6$ of classical and Clenshaw Curtis sparse grid choice (see Fig 3.4 and Fig 3.7) demonstrates that the approximation generated by Clenshaw Curtis grid are more accurate than that of the classical grid. As we can see, the Clenshaw Curtis process reduces the error equally over the whole domain. With a few terms, these are pretty accurate over the normal range that they are calculated. However, with a finite number of terms the sine function is never exactly equal to a polynomial.

In Fig 3.8, the convergence behaviour with respect to the RMSE error of the given function and for level $l \in \{0, \cdots, 6\}$ is provided. In addition to the error plots, we show the curves of expected sparse grid convergence (reference) due to the interpolation accuracy (2.27). Since the prediction error goes extremely small and the number of grid points grows exponentially as the sparse grid level increases, we create a convergence plot for an increasing sparse grid depth using a logarithmic scale for both the x-axis and the y-axis.

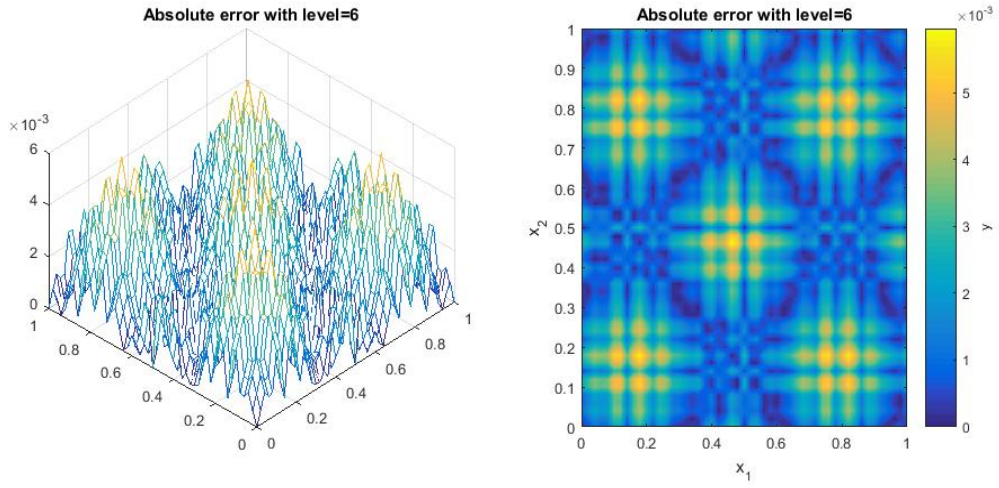

Figure 3.4: Absolute error for the hierarchical sparse grids interpolant with level=6.

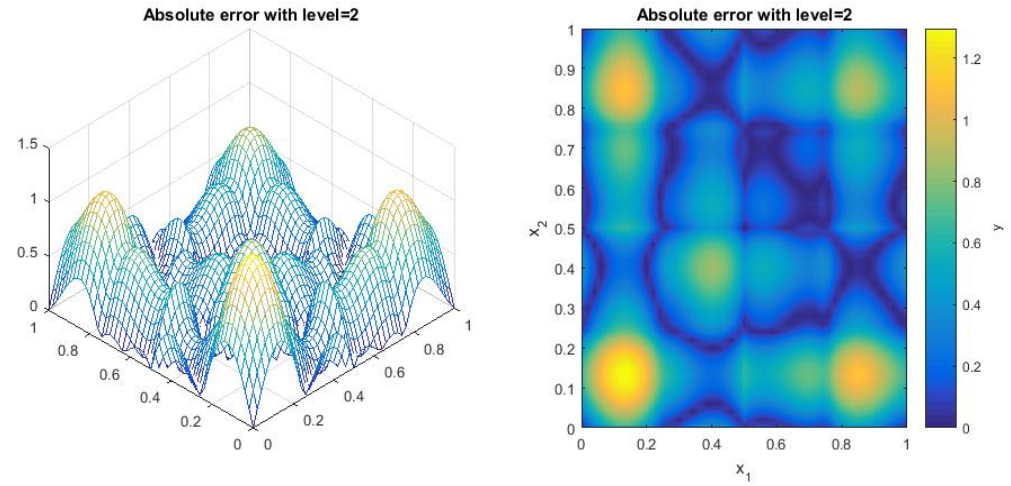

Figure 3.5: Absolute error for the combined interpolant with level=2.

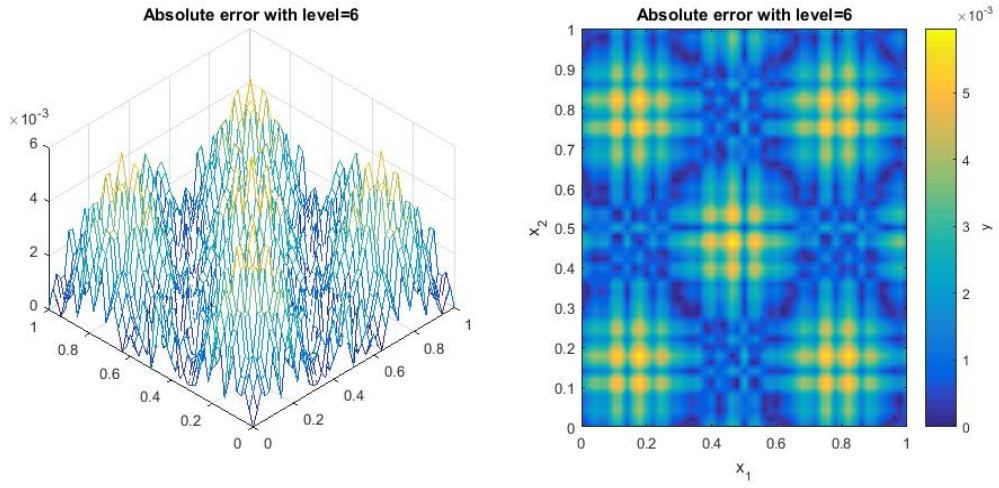

Figure 3.6: Absolute error for the combined interpolant with level=6

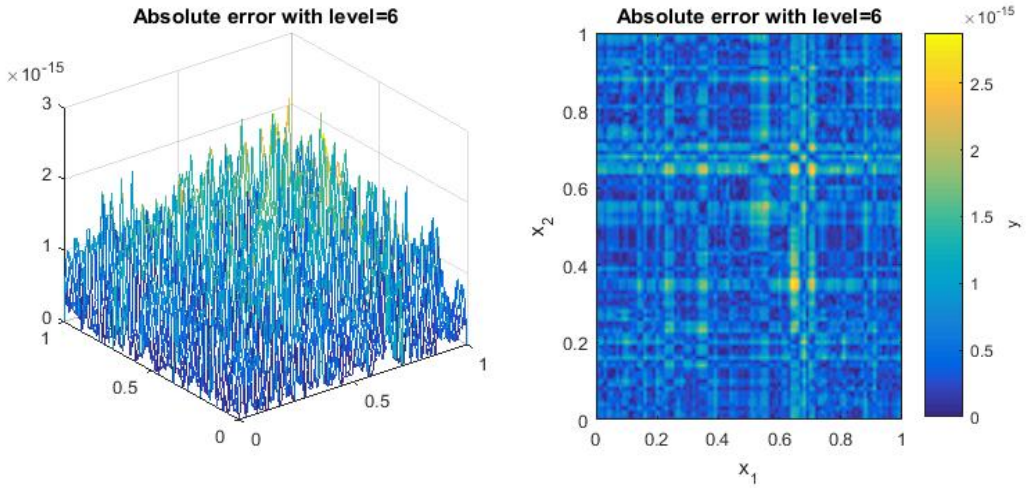

Figure 3.7: Absolute error for Clenshaw Curtis sparse grid with level=6

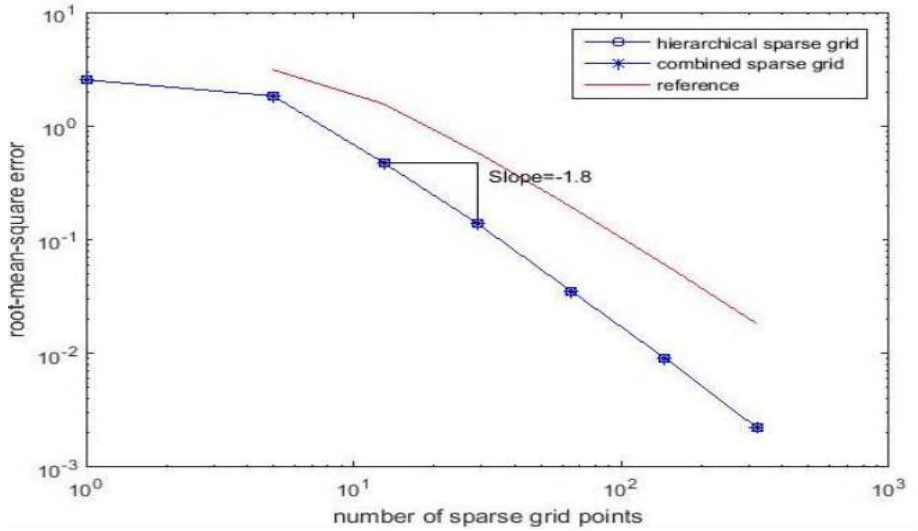

Figure 3.8: Convergence of the RMSE error of the function against an increasing level of grids. The numerically observed rate of convergence for two interpolants , compared to a reference line.

Level	Points	Max.Abs.Error	Rel.Error
1	5	3.4494	1.7880
2	13	1.3047	0.4765
3	29	0.3634	0.1377
4	65	0.0965	0.0356
5	145	0.0239	0.0089
6	321	0.0061	0.0023

Table 3.1: The maximum absolute error and relative error for the hierarchical sparse grid interpolant

From the Fig 3.8, we determine the hierarchical sparse grid and sparse grid combination technique generate the same relative prediction errors at each level of grid and have the same rate of convergence. The convergence shows almost a

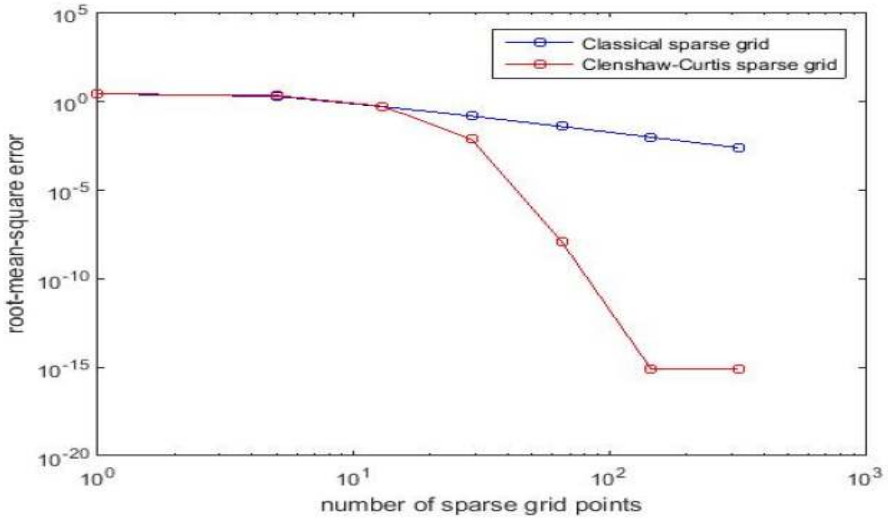

Figure 3.9: Convergence of the RMSE error of the function against an increasing level of grids. The numerically observed rate of convergence for two interpolants.

straight line for an increasing grid level. Compared with the theoretical error analysis $O(h_n^2 \log(h_n^{-1})^{d-1})$, the plot of convergence runs roughly parallel to the reference line. The rate of convergence is around -1.8. The table 3.1 displays that the maximum absolute errors and the relative errors are decreasing with a constant rate as the number of grid points increases.

Next, we turn to Clenshaw Curtis sparse grid. In Fig 3.9, we see that the RMSE error for Clenshaw Curtis grid decreases dramatically against an increasing level of grids. It fast converges to $10e^{-15}$ at grid level 5, and then be a constant. As we expect, the polynomials are never completely accurate. Fig 3.10 illustrates the cost for computing the interpolants. The combined sparse grid interpolant can be obtained at a small cost in lower grid levels, but the curves of cost of combined grid intersects the curves of cost of hierarchical grids and Clenshaw Curtis grids at sparse grid level 8 and level 9, respectively. Due to the more sophisticated algorithms required in the polynomial case, the additional cost of computing the interpolant of Clenshaw Curtis grid is considerably higher compared to the classical sparse grid interpolation of function values. However, as the grid level increases, the rate of cost decreases, as fewer function evaluation will require a computation. Thus, the performance is competitive. With these results supporting the efficiency of mesh refinement on sparse grids and the combination technique approximation, we close the discussion of the 2D Sine function and turn to higher-dimensional problems.

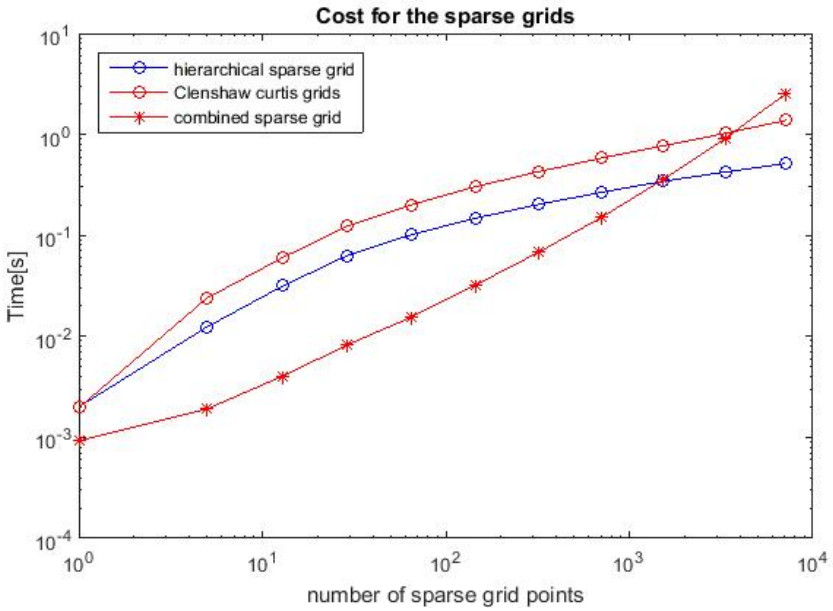

Figure 3.10: Time to compute 1000 values with these three types of sparse grids.

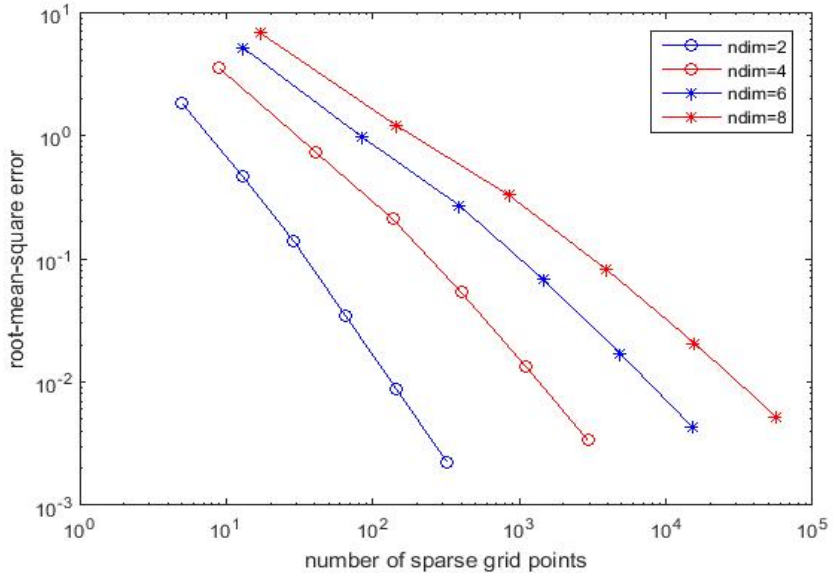

Figure 3.11: Convergence of the RMSE error of the multi-dimensional functions against an increasing level of grids on the classical grid.

In Fig 3.11, we present a summary of the convergence behaviour with respect to multi-dimensions ($d \in \{2, 4, \cdots, 8\}$). We can clearly see that it has a fast convergence rate even for a higher dimensional case. Similarly, Fig 3.20 illustrates that the RMSE error for Clenshaw Curtis grid rapidly decreases down to $10e^{-15}$ for all dimensions, and then it keeps constant as grid level increases. We compare the computational cost of the classical grid and the Clenshaw Curtis grid, see the Fig

3.13 and Fig 3.14. the computational cost increases exponentially with the number of dimensions. The computational cost of the Clenshaw Curtis grid interpolation is sightly higher than that of the uniform sparse grid interpolation.

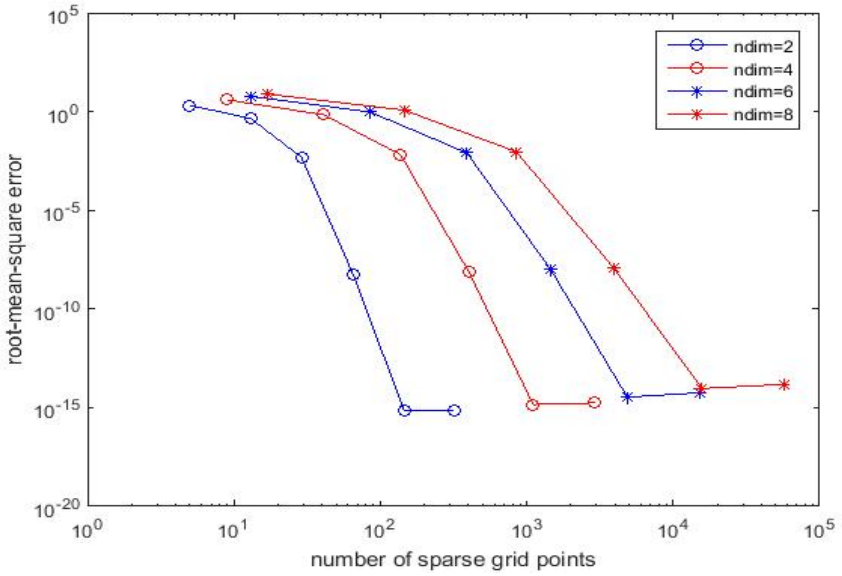

Figure 3.12: Convergence of the RMSE error of the multi-dimensional functions against an increasing level of grids on the Clenshaw Curtis grid.

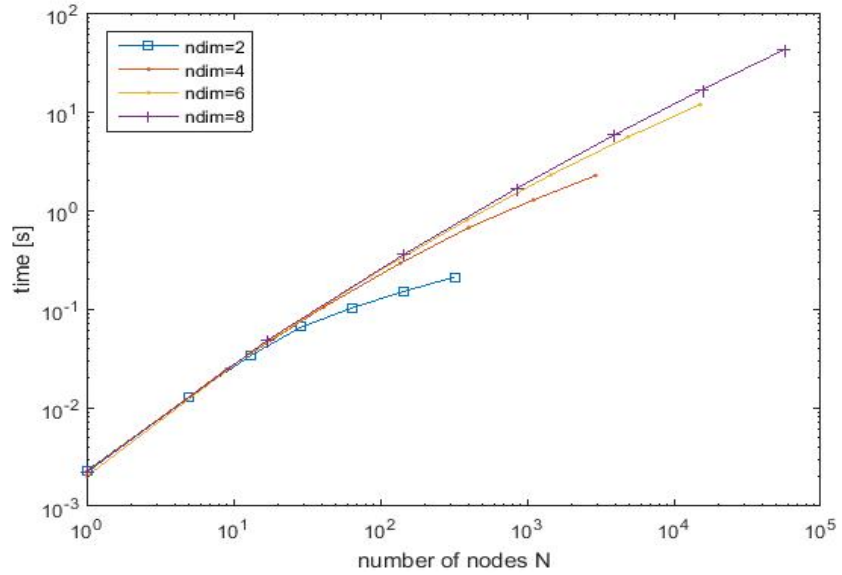

Figure 3.13: Time to compute 1000 values with these three types of sparse grids.

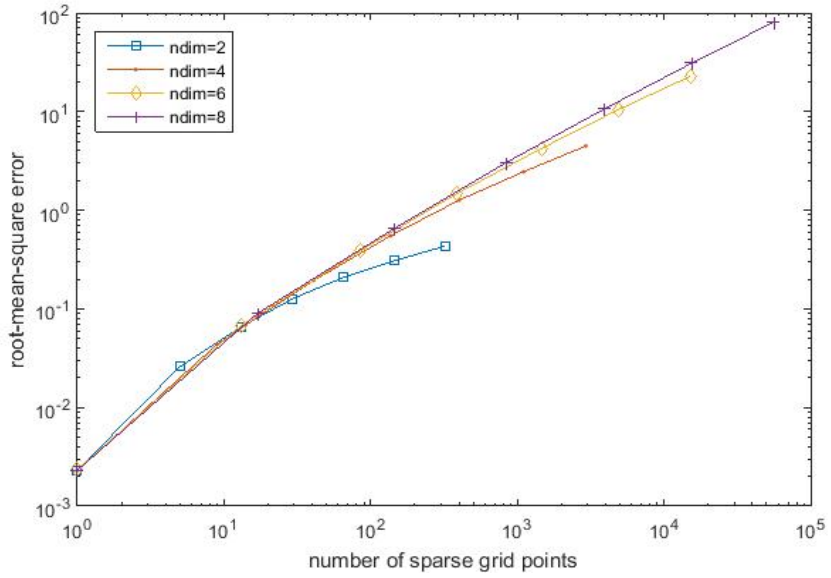

Figure 3.14: Time to compute 1000 values with these three types of sparse grids.

3.2 Rosenbrock Function

As the second test function, we consider the multi-dimensional Rosenbrock function, which is often used as a performance test problem for optimization algorithms. The Rosenbrock function, also referred to as the Valley or Banana function. The function is unimodal, and the global minimum ($f(\underline{x}) = 0,$ at $\underline{x} = (1, \cdots, 1)$) lies in a narrow, parabolic valley. However, even though this valley is easy to find, convergence to the minimum is difficult (Picheny et al., 2012) [15]. We use the following rescaled form of the Rosenbrock function (3.2) on the domain$[0, 1]^d$. Fig 3.11 is an illustration of the two-dimensional case.

$$g(x_i) \;=\; f(4x_i - 2) \tag{3.5}$$

Fig 3.15 shows that the approximations of the exact solution generated by the hierarchical sparse grid interpolant and combined interpolant. Again, the effects of the improved sparse grid levels of our interpolation are evident. It also illustrates that the hierarchical sparse grids interpolant (the upper-right corner) is identical to the interpolant by using combination approach (the lower-right corner). In Fig 3.16, the sparse grid visualized on the contour plots of the interpolants, we see that the used sparse grid is of the type classical sparse grid and the Clenshaw-Curtis. It can be shown that the interpolation of Clenshaw-Curtis grids with $level = 2$ has obtained relatively accurate approximation of the exact solution compared to the classical sparse grids. The number of grid points grows exponentially. For

two-dimensional case, the sparse grids consist of 13 points with $level = 2$ is, and of $level = 10$ is 7169 points, respectively.

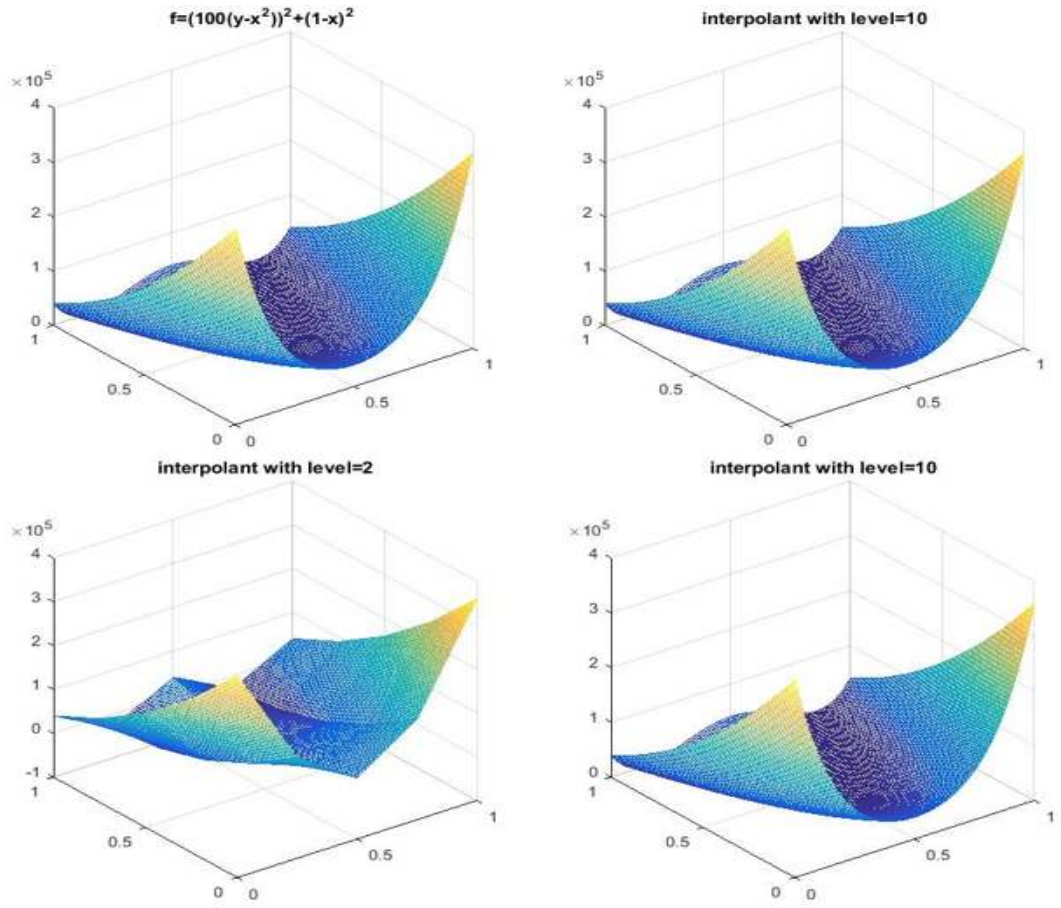

Figure 3.15: The exact solution and the hierarchical sparse grids interpolant u_n^s (on the top).The combined interpolant $u_n^c$with level 2 and 10 (on the bottom).

In Fig 3.17 and Fig 3.18, we present the absolute errors resulting from the hierarchical sparse grids and the combination approach. Obviously, the errors are the same, which confirms the theoretical proof in section 2.7. Fig 3.19 shows that the relative errors on the hierarchical sparse grids are large near the global minimum which is inside a long, narrow and parabolic shaped flat valley. It does make sense in terms of the relative error formula ($Rel = \frac{|measuredvalue - truevalue|}{truevalue+1}$). Since the relative error formula has the minimum value as a denominator, we add a small value such as 1 on the denominator. To get an impression of the Clenshaw-Curtis process, Fig 3.20 and Fig 3.21 show the absolute error and the relative error with gird $level = 10$, respectively. We can see the error is extremely small and the Clenshaw-Curtis process reduces the absolute error equally over the whole domain. For the same reason, there exist relatively large errors along the global minimum valley.

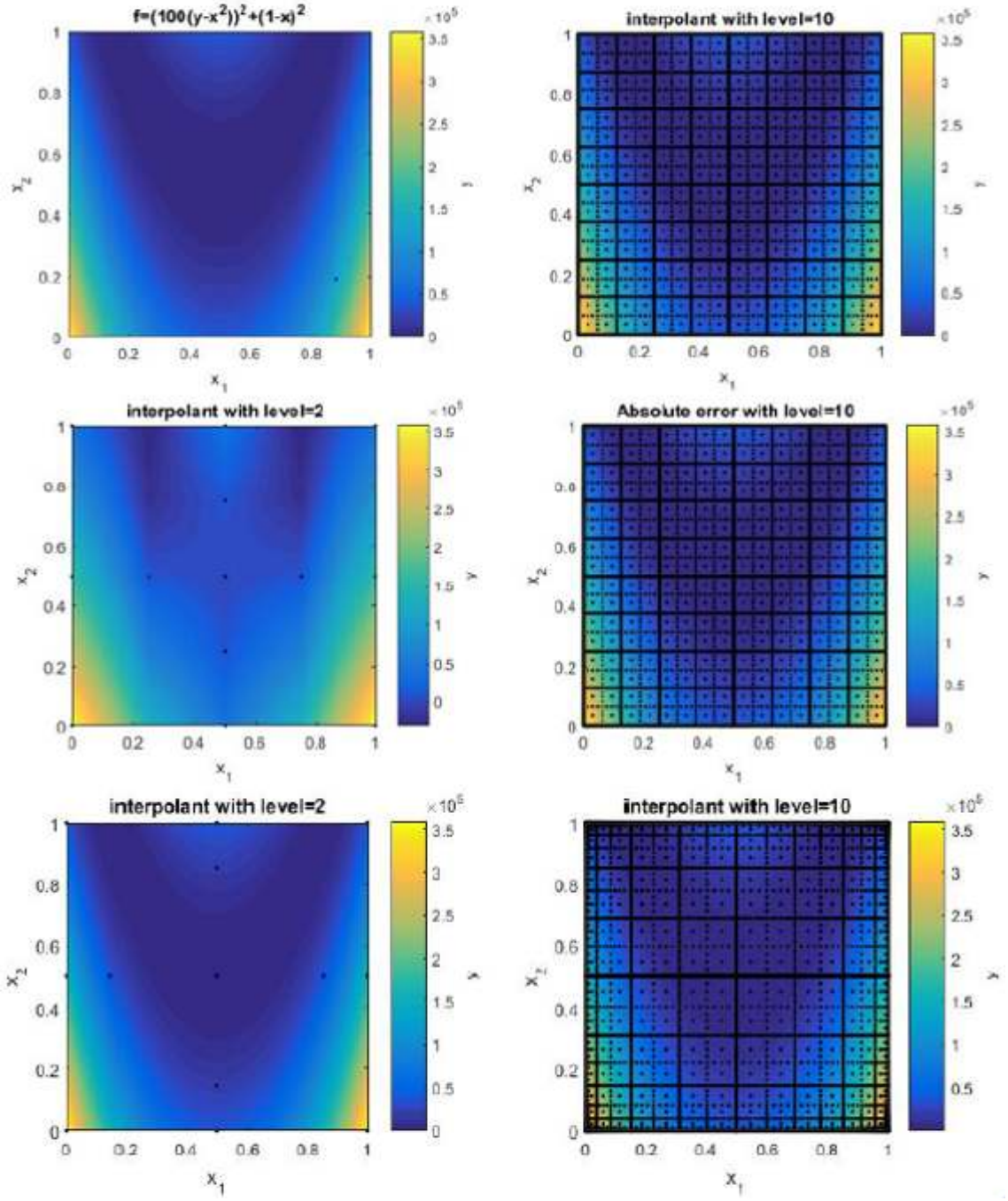

Figure 3.16: The contour plots of exact solution (in the upper-left corner) and the hierarchical sparse grids interpolant (in the upper-right corner); In the middle, the contour plots of the combined interpolants with level=2 and level=10, respectively; on the bottom, the contour plots of Clenshaw-Curtis grids.

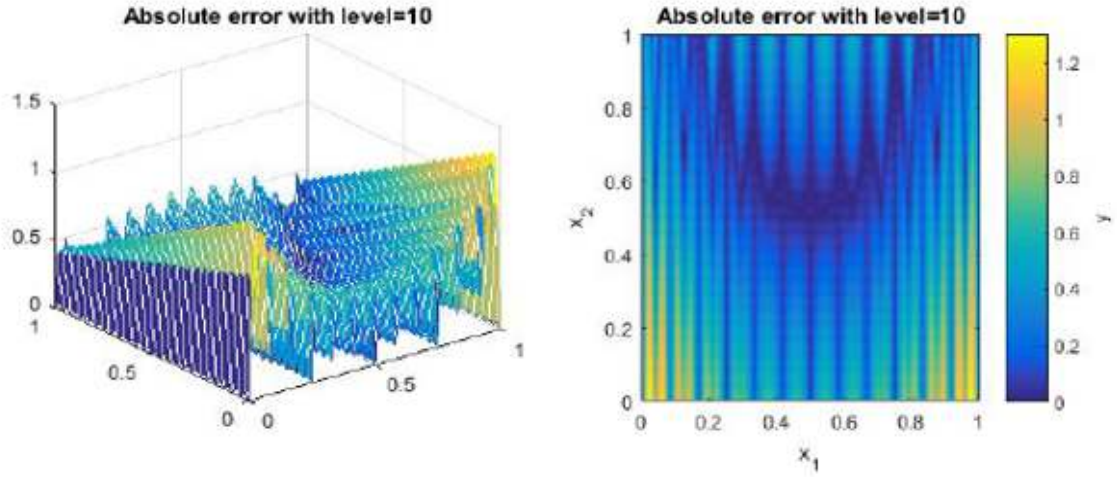

Figure 3.17: Absolute error for the hierarchical sparse grids interpolant with level=10.

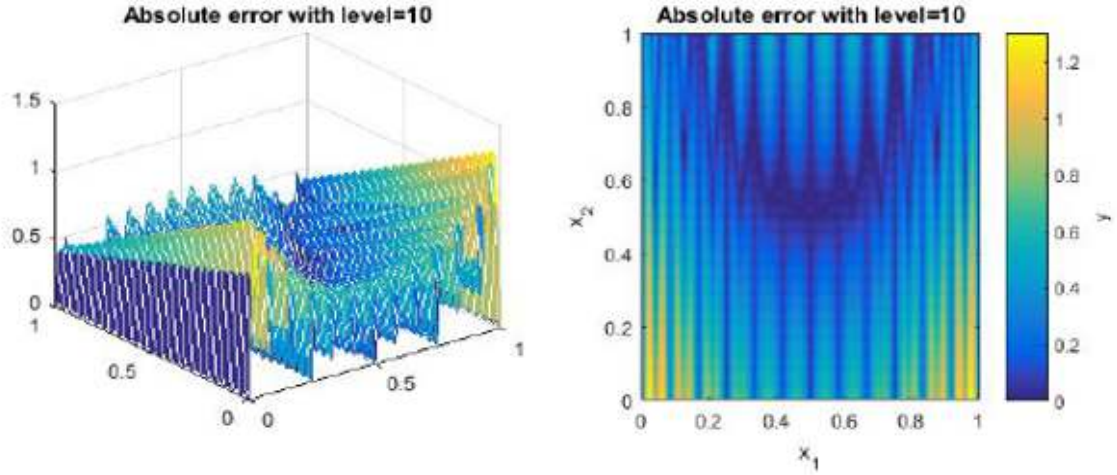

Figure 3.18: Absolute error for the combined interpolant with level=10

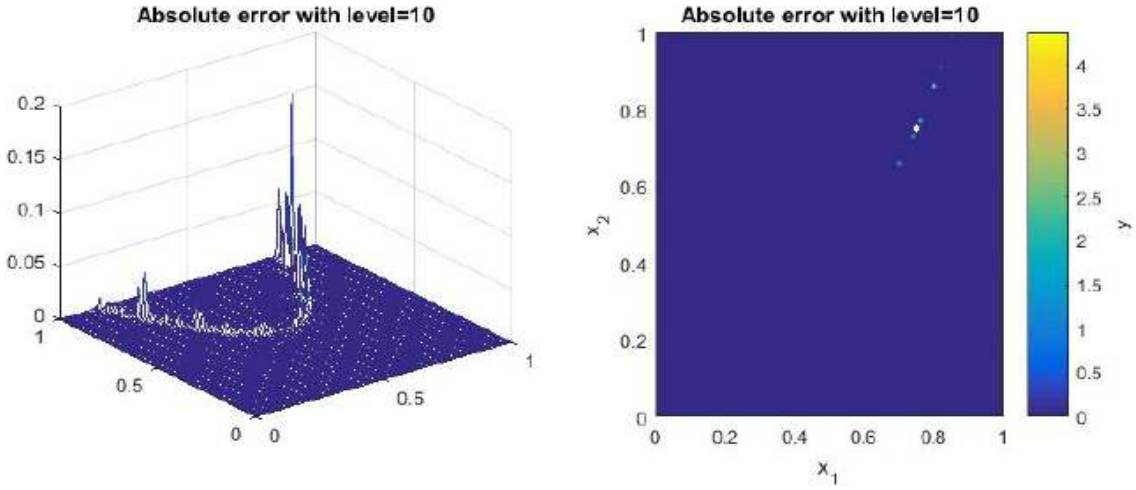

Figure 3.19: Relative error for the hierarchical sparse grids interpolant with level=10.

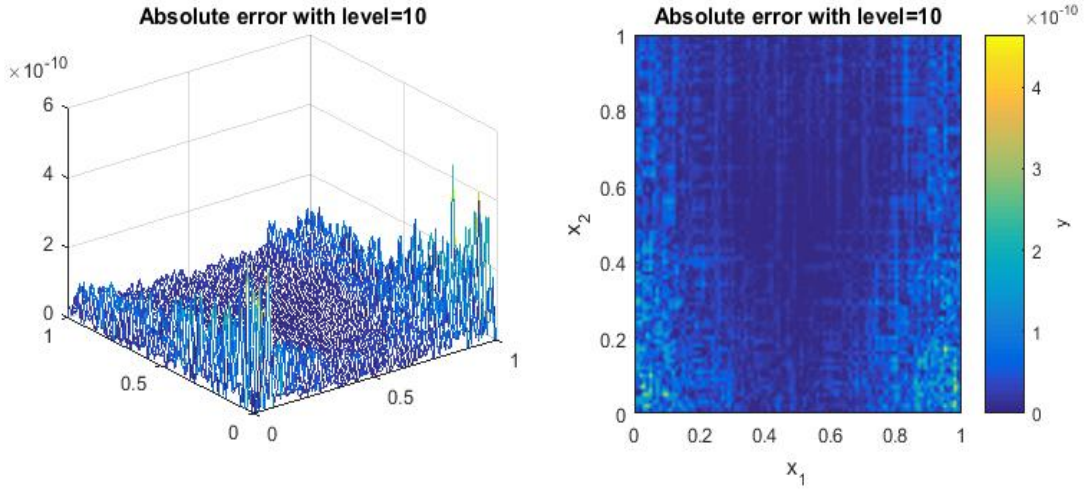

Figure 3.20: Absolute error for the Clenshaw-Curtis grids interpolant with level=10.

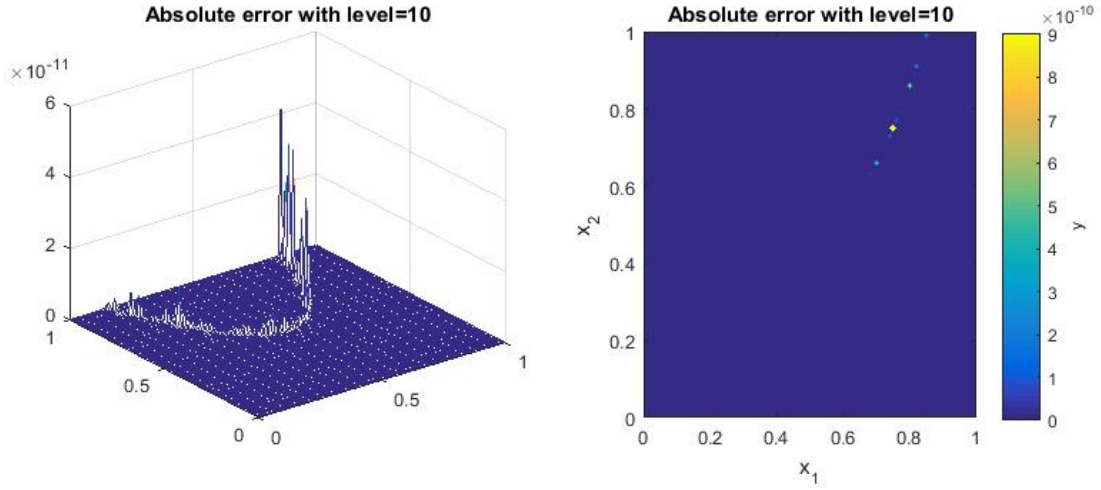

Figure 3.21: Relative error for the Clenshaw-Curtis grids interpolant with level=10.

Fig 3.22 shows the relative prediction error for 2D example for an increasing number of grid levels. From Fig 3.22, we determine the hierarchical sparse grid and sparse grid combination technique generate the same relative prediction errors for all grid levels. The solid line, the expected sparse grid convergence, indicates the behaviour of the error of sparse grids with respect to the problem of interpolating a given function. We observe an almost straight line roughly parallels to the solid line for an increasing grid level. The rate of convergence is around -1.7. In Fig 3.23, a summary of the convergence behaviour with respect to multi-dimensions ($d \in \{2, 4, \cdots, 10\}$) is provided. For all dimensions presented, we can see that the convergence rate is decreasing as the dimension increases. A strong support of our proof in section 2.7 indicates the interpolant using the combination technique is the hierarchical sparse grid for all dimensions. Next, we turn to the Clenshaw-

Curtis grids. Fig 3.24 illustrates that the RMSE error for Clenshaw Curtis grid rapidly decreases down to $10e^{-10}$ from grid level 1 to level 2, and then it keeps constant as grid level increases. The achieved accuracy will be compared to the results of interpolation on the classical sparse grids. With the Clenshaw Curtis process advancing, the higher accurate approximation of exact solution comes to fruition. Finally, to get an efficiency of the sparse grids process, Fig 3.25 and Fig 3.26 show the computational cost for the classical grid and the Clenshaw Curtis grid. The Clenshaw-Curtis sparse grid interpolant can be obtained at a very small additional cost compared to the classical sparse grid interpolant of function.

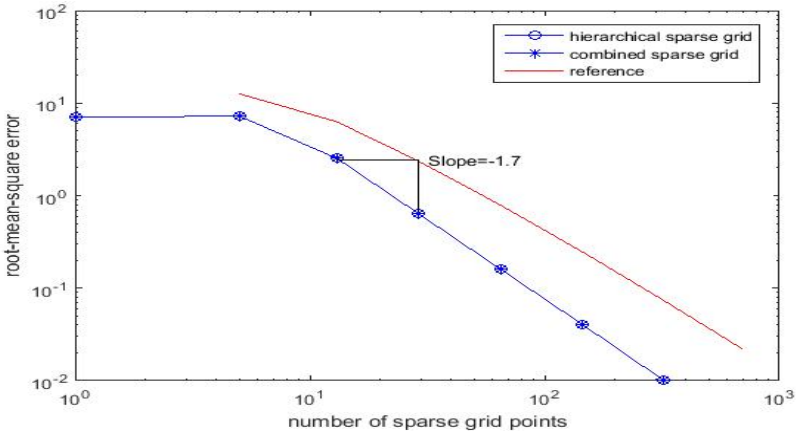

Figure 3.22: Convergence of the RMSE error of the function against an increasing level of grids. The numerically observed rate of convergence for two interpolants, compared to a reference line.

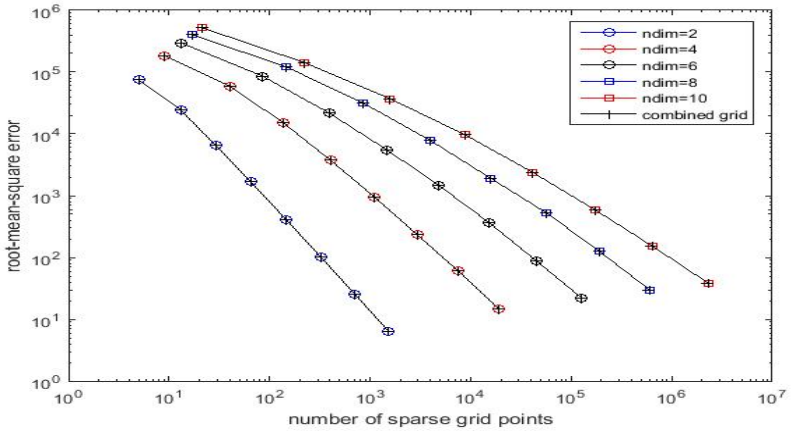

Figure 3.23: Convergence of the RMSE error of the multi-dimensional functions against an increasing level of grids. The numerically observed rate of convergence for two interpolants.

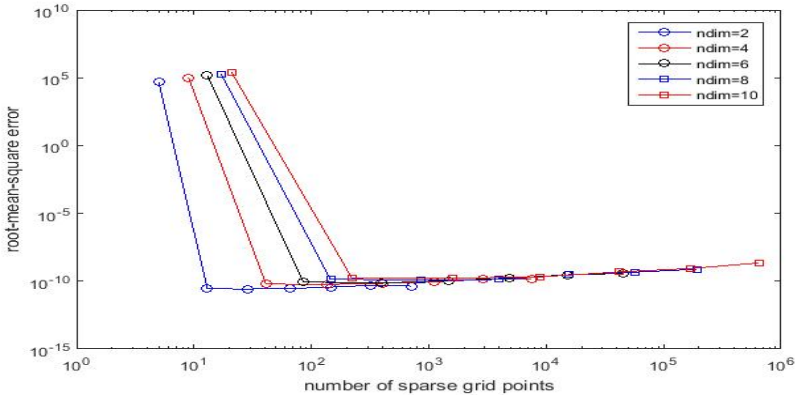

Figure 3.24: Convergence of the RMSE error of the multi-dimensional functions against an increasing level of grids. The numerically observed rate of convergence for two interpolants.

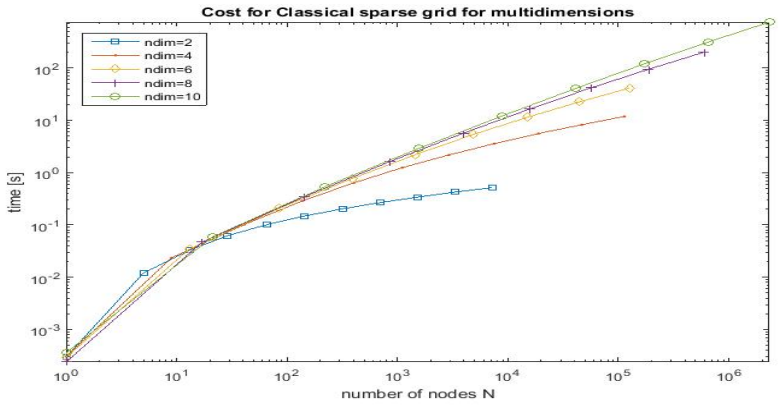

Figure 3.25: Time to compute 1000 values with the Classical sparse grids for multidimensions.

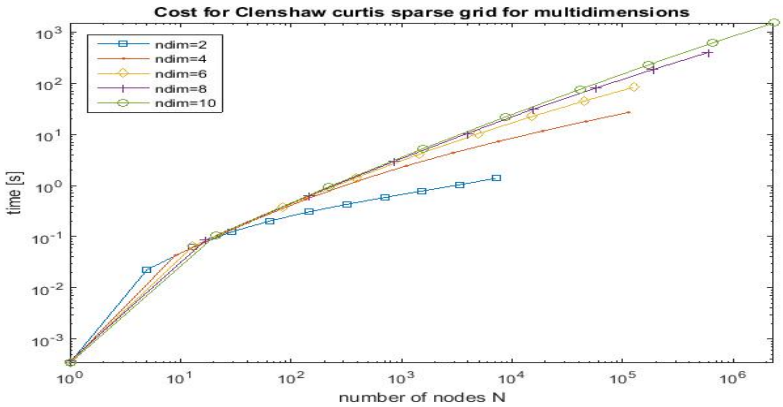

Figure 3.26: Time to compute 1000 values with the Clenshaw Curtis sparse grids for multidimensions.

3.3 Gaussian function

All examples have so for been treated with the classical sparse grid, the Clenshaw Curtis grid and the combination approach, we want to present one more result for a Gaussian function. On $\Omega = [0,1]^d$, let

$$f(x_i) \;=\; exp\left(-\sum_{i=1}^{d} \frac{x_i - \mu}{2\sigma^2}\right), \tag{3.6}$$

where μ and σ are constants.

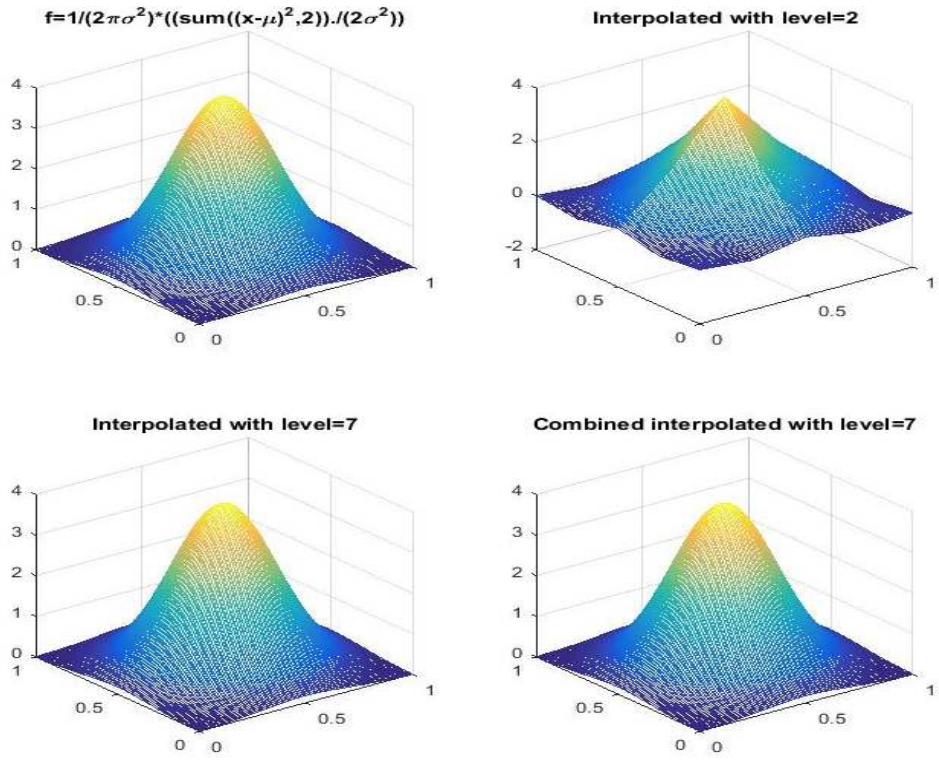

Figure 3.27: The exact solution (on the top).The hierarchical sparse grids interpolant u_n^s and the combined interpolant u_n^c with level 7 (on the bottom)..

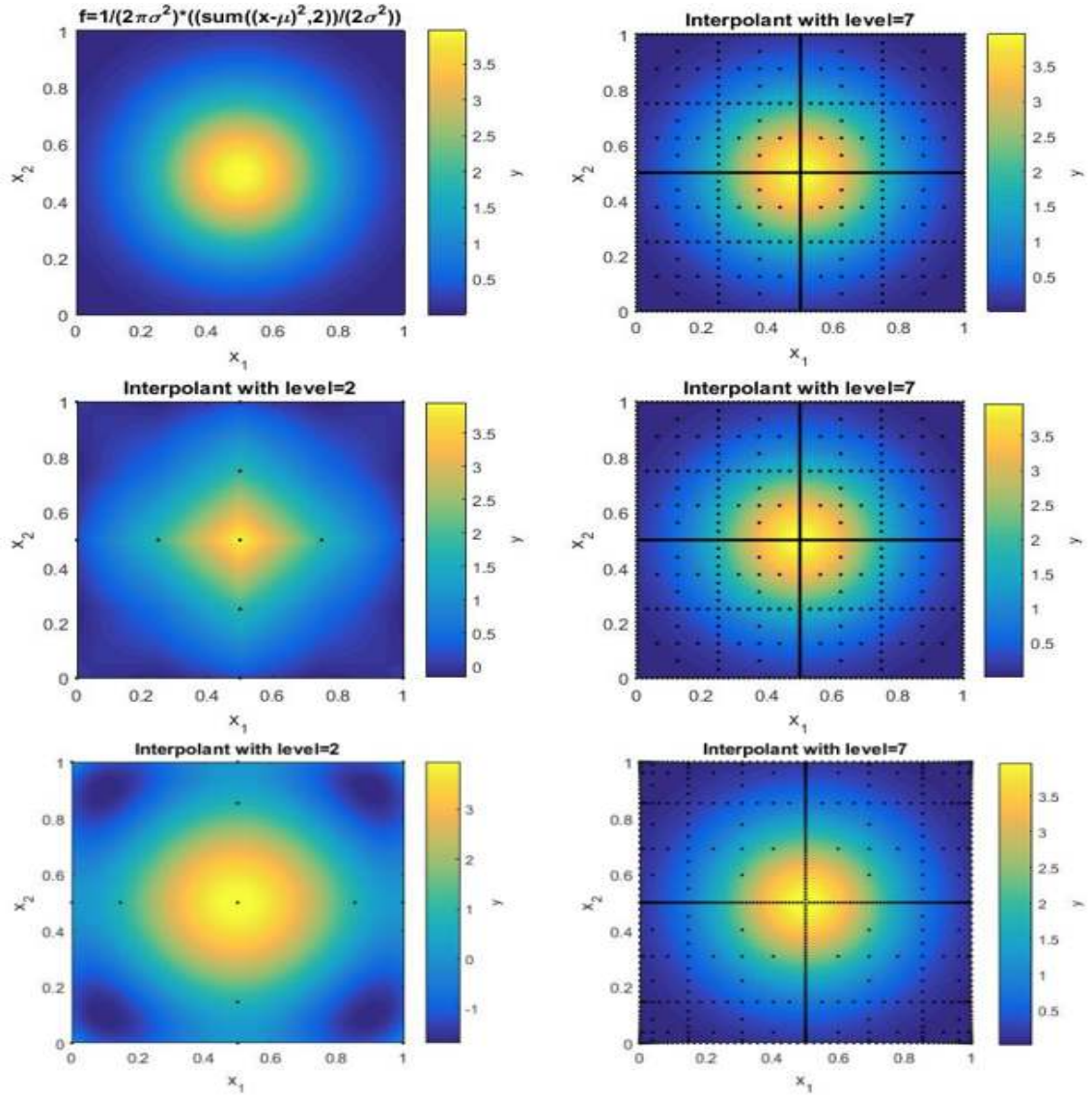

Figure 3.28: The contour plots of exact solution (in the upper-left corner) and the combined interpolants (in the upper-right corner); In the middle, the contour plots of the hierarchical sparse grids interpolant with level=2 and level=7, respectively; on the bottom, the contour plots of Clenshaw-Curtis grids.

Fig 3.27 shows the true value and the approximation of a Gaussian function in 2D. It should come as no surprise that the approximation generated by the hierarchical sparse grids interpolant is the same as the interpolant by using combination technique. The improved the sparse grid levels (depth) have effects on the quality of approximations. Fig 3.28 shows the classical sparse grid and the type of Clenshaw Curtis grids with 13 grid points ($l = 2$, left), and 705 grid points ($l = 7$, right). In Fig 3.29 and Fig 3.30, we compare the error on the classical sparse grid and on the combination approach. Fig 3.29 and Fig 3.30 show a gain in accuracy with higher grids level that is comparable to the true value of (3.5) in 2D case. However, look at

the contour plots of errors, the different types of grid nodes influence the pattern of resulting errors. The classical sparse grids show large errors in the middle, in contrast with that, the Clenshaw Curtis grids generate large error around the corners.

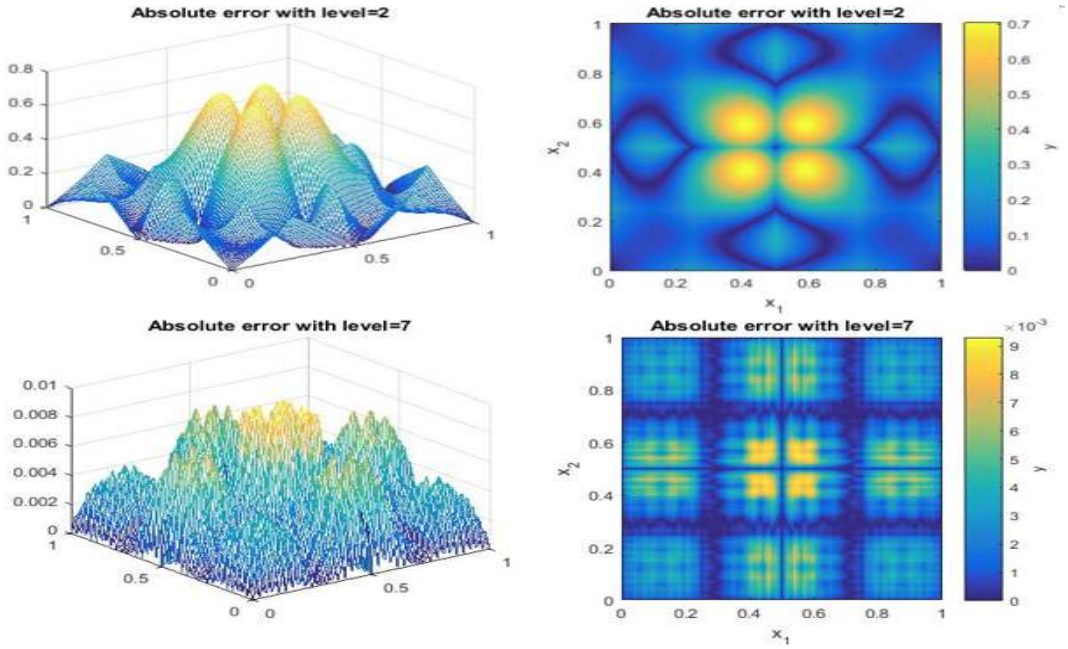

Figure 3.29: Absolute error for the hierarchical sparse grids interpolant with level=2 and level=7, respectively.

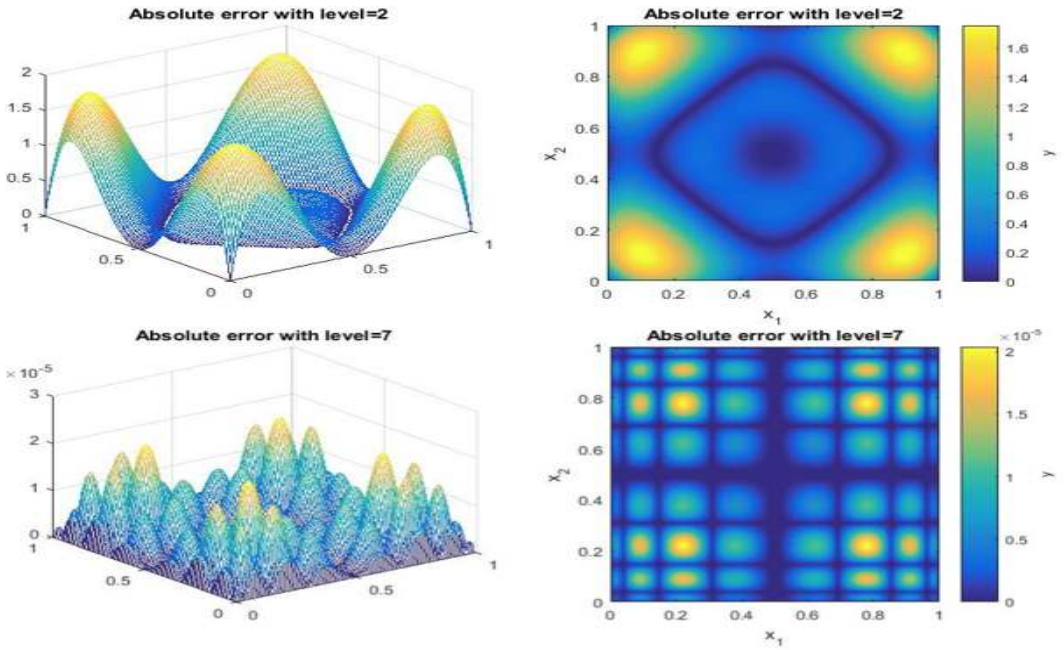

Figure 3.30: Absolute error for the Clenshaw Curtis sparse grids interpolant with level=2 and level=7, respectively.

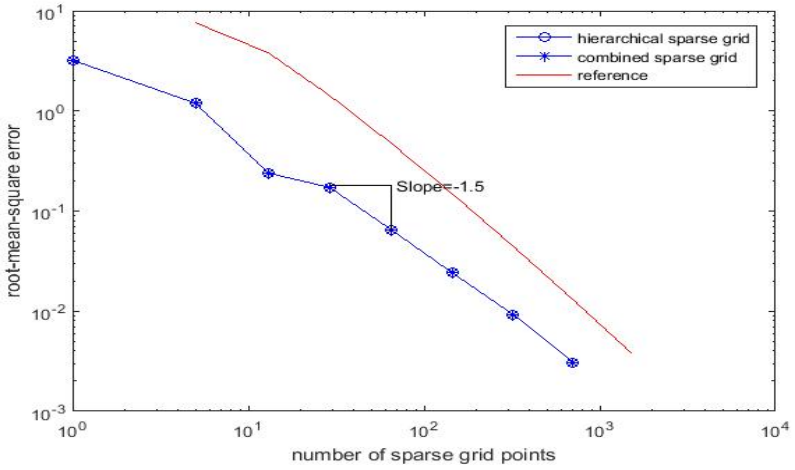

Figure 3.31: Convergence of the RMSE error of the function against an increasing level of grids. The numerically observed rate of convergence for two interpolants, compared to a reference line.

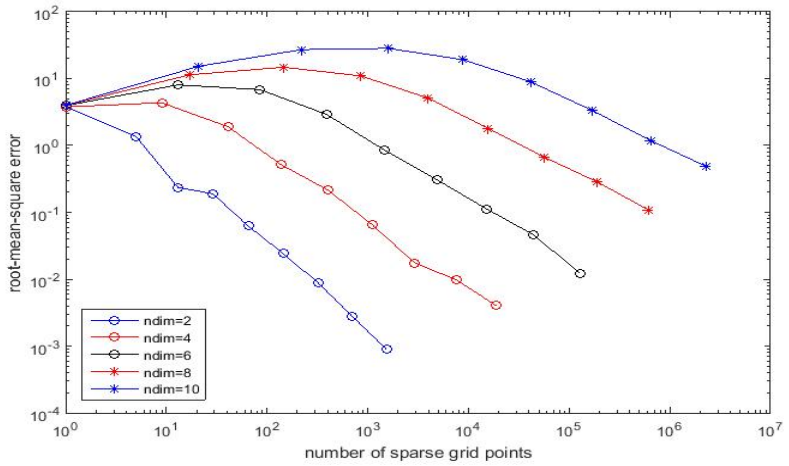

Figure 3.32: Convergence of the RMSE error of the multi-dimensional functions against an increasing level of grids. The numerically observed rate of convergence for two interpolants.

In Fig 3.31, we present the convergence of the conventional sparse grid method with the curves of expected sparse grid convergence (reference). We observe that the rate of convergence decreases against the number of sparse grid points. The curves of the classical sparse grid convergence is not parallel to the reference line, but it reasonably works well in 2D. The rate of convergence is around -1.5. Fig 3.32 illustrates the convergence behaviour on the classical sparse grid for higher dimensions. For higher dimensional case, it suggests slow convergence, convergence in the L_2 norm is not achieved in regions. It seems that the convergence behaviour needs large number of grids points to appear. This was to be expected, since this is consequence of the fact that Gaussian function distribution mostly sits in a thin shell around the mean (μ) as the dimension d goes to infinity. Fig 3.33 shows the

convergence behaviour on the Clenshaw Curtis sparse grid for higher dimensions. Again, as in our previous experimental results of the classical grids, we can see that the sparse grid method achieves a fast convergence rate in 2D but it does not work well for high-dimensional cases. Compared to the classical sparse grid, the Clenshaw Curtis has lower rate of convergence in the higher dimensions. We check if the sparse grids provide a good compromise between accuracy and computational cost. Fig 3.34 and Fig 3.35 indicate that the computational cost of Clenshaw Curtis grid is considerably higher compared to the classical sparse grids interpolation of function.

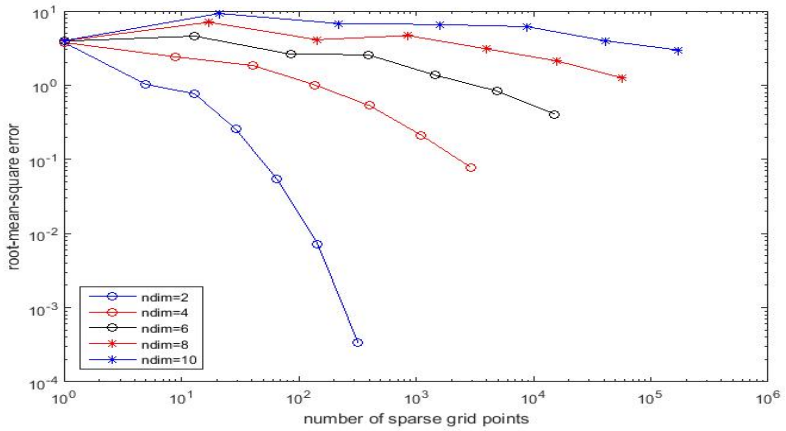

Figure 3.33: Convergence of the RMSE error of the multi-dimensional functions against an increasing level of grids. The numerically observed rate of convergence for two interpolants.

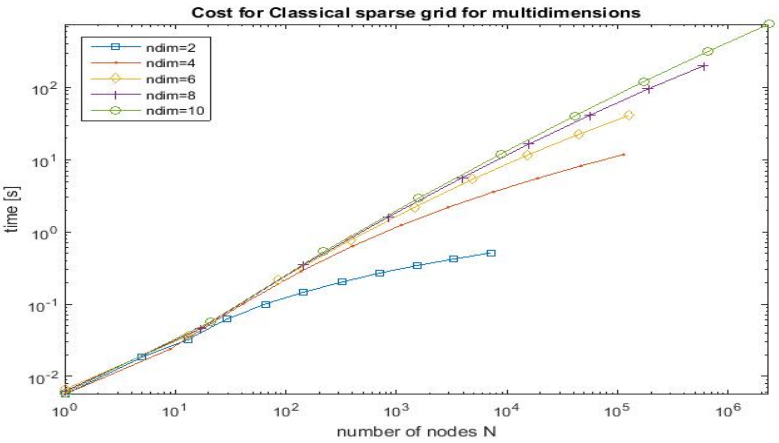

Figure 3.34: Time to compute 1000 values with the classical sparse grids for multidimensions.

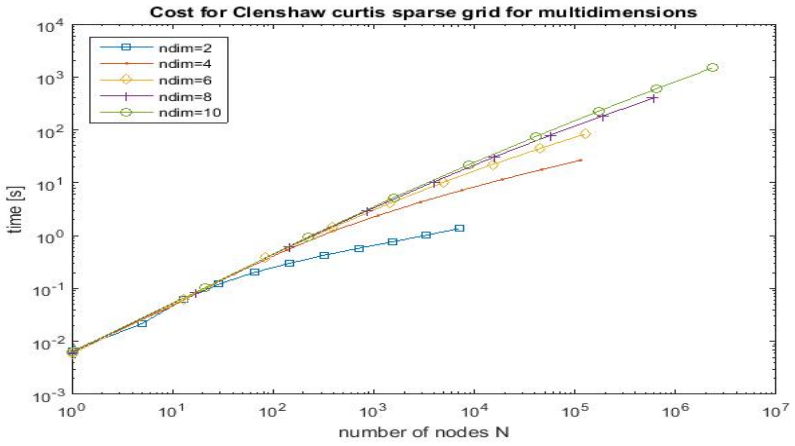

Figure 3.35: Time to compute 1000 values with the Clenshaw Curtis sparse grids for multidimensions.

3.4 Checkerboard

The last example of this section demonstrates that our approach is limited to discontinuous function such as the Checkerboard, since for the conventional sparse grid methods, a priori selection of grid points, optimal under certain smoothness conditions. Unfortunately, the discontinuous function itself cannot be successfully approximated by a continuous sparse grid interpolant in the first place.
We first consider two dimensional problems in $\Omega = [0,1]^2$. The class labels ± 1 have been assigned in a 2×2 and 3×3 checkerboard pattern. A 2×2 checkerboard pattern is provided, see Fig 3.36. Now, looking at two dimensional the discontinues function with a checkerboard pattern

$$f(x_i) \;=\; \begin{cases} exp\left(-\sum_{i=1}^{d} \frac{x_i-\mu}{2\sigma^2}\right), & \text{if } x_i \in [0,0.5]^2, or\ x_i \in [0.5,1]^2; \\ -exp\left(-\sum_{i=1}^{d} \frac{x_i-\mu}{2\sigma^2}\right), & \text{otherwise.} \end{cases} \tag{3.7}$$

In Fig 3.37, we show the convergence behaviour with respect to L_2 error for regular sparse grids. In addition to the error plots, we present the curves of expected sparse grid convergence (reference). Moreover, Table 3.2 provides the numerical values of maximum errors and relative errors on conventional sparse grid methods. Due to the violation of the smoothness requirements, the L_2 error converges much worse than $O(h_n^2 \log(h_n^{-1})^{d-1})$ as in the smooth case. As we can see for this measure, using more grid points does not necessarily lead to a better accuracy, the error can increase. Investigating the accuracy if depending on how the separation manifold is located relative to the sparse grid structure, we examine a 3×3 checkerboard pattern. Fig 3.38 indicates that again low convergence rates are encountered. We cannot expect the convergence to behave as good in the general case under smoothness.

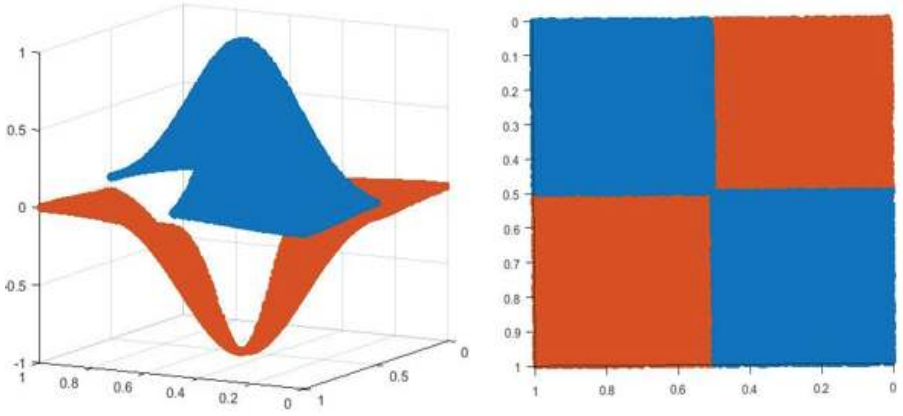

Figure 3.36: The class labels -1 (red) and +1 (blue) have been assigned in a 2×2 checkerboard pattern.

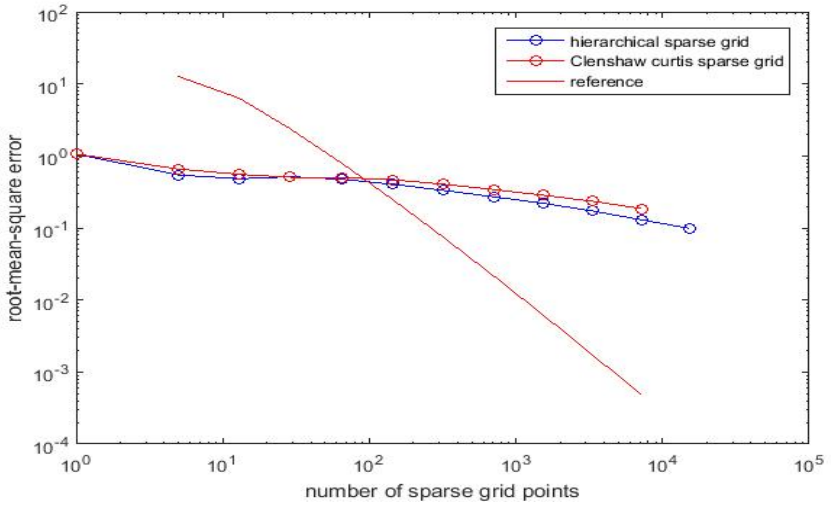

Figure 3.37: Numerically observed convergence of the RMSE error of the 2×2 checkerboard function against an increasing level of grids, compared to a reference line.

		The Classical gird		Clenshaw curtis grid	
Level	Points	Rel.Error	Max.Abs.Error	Rel.Error	Max.Abs.Error
1	5	0.5437	1.9873	0.6564	1.9997
2	13	0.4811	1.9857	0.5546	1.9996
3	29	0.5139	1.9906	0.5057	1.9995
4	65	0.4747	1.9847	0.5002	1.9995
5	145	0.4032	1.9967	0.4662	1.9993
6	321	0.3322	1.9968	0.4043	1.9989
7	705	0.2727	1.9944	0.3415	1.9978
8	1537	0.2179	1.9873	0.2839	1.9951
9	3329	0.1718	1.9698	0.2346	1.9887
10	7169	0.1296	1.9301	0.1854	1.9743

Table 3.2: The maximum absolute error and relative error for the hierarchical sparse grid interpolant

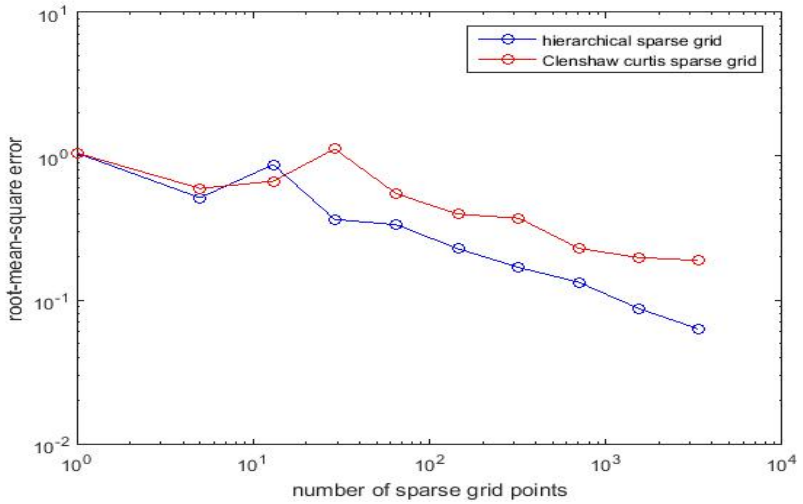

Figure 3.38: Numerically observed convergence of the RMSE error of the 3×3 checkerboard function against an increasing level of grids.

		The Classical gird		Clenshaw curtis grid	
Level	Points	Rel.Error	Max.Abs.Error	Rel.Error	Max.Abs.Error
1	5	0.5060	1.3533	0.5969	1.5880
2	13	0.8782	1.8172	0.6699	1.3145
3	29	0.3641	1.2109	1.1229	2.6406
4	65	0.3472	1.6433	0.5475	2.1250
5	145	0.2421	1.3919	0.3952	1.3413
6	321	0.1812	1.0807	0.3707	1.5299
7	705	0.1482	1.1513	0.2284	1.3083
8	1537	0.1041	0.8800	0.1974	1.2291
9	3329	0.0828	1.274	0.1886	1.7019
10	7169	0.0627	0.9037	0.1126	1.1779

Table 3.2: The maximum absolute error and relative error for the hierarchical sparse grid interpolant

Finally, and to give a checkerboard that a similar behaviour can be expected in higher-dimensional settings, we show the error for a checkerboard in 3D case, see Fig 3.35. We have shown that the effect that checkerboard functions violate the sparse grids' smoothness requirements, non-continuous functions have been studied. Whereas the classical refinement criterion does not target the error, some refinement strategies can further improve the convergence of the error quite significantly. For a detailed discussion of an extension of the classical sparse grid approach by spatially adaptive refinement, see Dirk Pflüger (2010)[18], for instance.

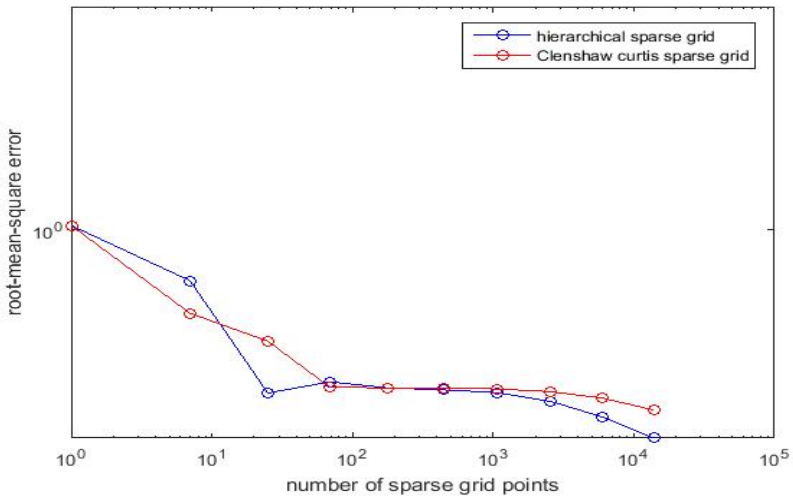

Figure 3.39: Numerically observed convergence of the RMSE error of the checkerboard in 3D against an increasing level of grids.

Chapter 4

An experimental study of Hokkaido Nansei-oki tsunami

In this chapter, we look for a surrogate method to provide an approximation of the output of an input-output relationships using as few model evaluations as possible. Many engineering design problems involve black-box functions whose values are outcomes of computationally expensive simulations, so an approximation model of the outcome is used instead. For example, an input-output system with a known multivariate input distribution $p(\underline{x})$, the Monte Carlo statistical sampling allows us to estimate the statistical moments of the output $u(\underline{x})$. However, Monte-Carlo approaches of exploring the input parameter space require a large number of L of expensive simulations. One approach of alleviating the burden is by developing surrogate models, alternatively known as response surface models, metamodels or emulators (Sacks at al., 1989) [16]. The surrogate approximation is based on a small set of $M \ll L$ simulations, known as 'samples' or 'training data'. Our objective is to build an accurate and efficient surrogate approximation by using a small set of M samples. Moreover, since we increase the number of uncertain input parameters, such studies suffer from the curse of dimensionality. In this work, we use the sparse grid interpolation to reduce the curse of dimensionality. A sparse adaptive surrogate model is constructed for the Hokkaido-Nansei-Oki tsunami, for which we give a description in Section 4.1. We study two different types of sparse grid methods: the classical sparse grids and Clenshaw Curtis grids. We start with the discussion of the simple two uncertain input parameters test-case. Then we illustrate our approach of a large number of uncertain input parameters to quantify the uncertainty in the output. We demonstrate the experimental results in Okushiri wave flume, which reproduce the maximum value of the time-dependent average tsunami height on top of the Monai zone in Okushiri Island in 1993.

4.1 The Hokkaido-Nansei-Oki tsunami

The Hokkaido-Nansei-Oki earthquake on July 12 produced one of the largest tsunamis in Japan's history. Within 2-5 minutes, extremely large waves hit the central west coast of Hokkaido and the small, offshore island of Okushiri in the Sea of Japan. The maximum run-up was measured at 32 m in a small valley north of Monai. A model consists of small curved pocket beach (205m long) of the Monai coast. The model scale is 1/400 with no-distorted. We consider the Okushiri wave tank benchmark test-case to produce the maximum of the time-dependent average tsunami height at Monai zone. The waves comes in from the west, the area of interest is the ellipse with a major axis length = 0.4m and a minor axis length = 0.2m. For more detailed description of the tank benchmark, see de Baar and Roberts (2016), for instance. The input wave data used for numerical simulation was collected from [17]. The data set consists of the value of water surface (m) depending on time (s).

4.2 Results of experiments

Concerning the efficient surrogate approximation of the Okushiri tsunami test-case based on the sparse grid interpolation has been made (de Baar and Roberts, 2016), assuming the incoming wave consists of a number of Gaussian bumps, which is uncertain. Then the parametrised incoming wave can be expressed as:

$$g_0(t,\xi) \;=\; \sum_{n=1}^{N} \xi_n \alpha_n exp\left(-\frac{(t-\tau_n)^2}{2\theta_n^2}\right) + R_N(t), \tag{4.1}$$

where α, τ and θ are the parameters of height, centre and width of Gaussian bumps, respectively, ξ the uncertain input parameter with i.i.d. $\xi \sim \mathcal{N}(1, 0.5^2)$. For the sparse grid interpolation, we transform $\xi \in [0,1]$. $R_N(t)$ is the residual. For example, setting the initial residual $R_0(t)$ is the deterministic incoming wave with a single a single Gaussian bump. The function (4.1) can be rescaled by a factor b, thus the the incoming wave has a constant energy.

We first consider a simple uniform sparse grid to create the two-dimensional input parameter space. Fig 4.1 indicates the output of the maximum of average wave level (m) with respect to the relative bump height ξ during the time (start from 0 s and end up to 22.5 s) by using a uniform grid and a Clenshaw Curtis grid, respectively. From Fig 4.1, we can see the range of maximum values of average water level (m) in the area of interest is between 0.002 m and 0.018 m. The effect of relative bump height on the maximum of wave level shows a positive relationship, which the wave increases (decreases) when we increase (decrease) the relative bump height. The

Table 4.1 shows the numerical values generated by these two sparse grids are almost the same, except for some nodes in different location. Moreover, Fig 4.2 presents more simulations by using uniform grid (left) and the Clenshaw Curtis grid (right) in two dimensional input parameter space. We expect a good compromise between accuracy and computational complexity, thus we compare the computational cost of the simulation based on two uncertain input parameters for these sparse grid interpolations. Fig 4.3 shows that the computational cost of Clenshaw Curtis grid is higher than that of the uniform sparse grid interpolation.

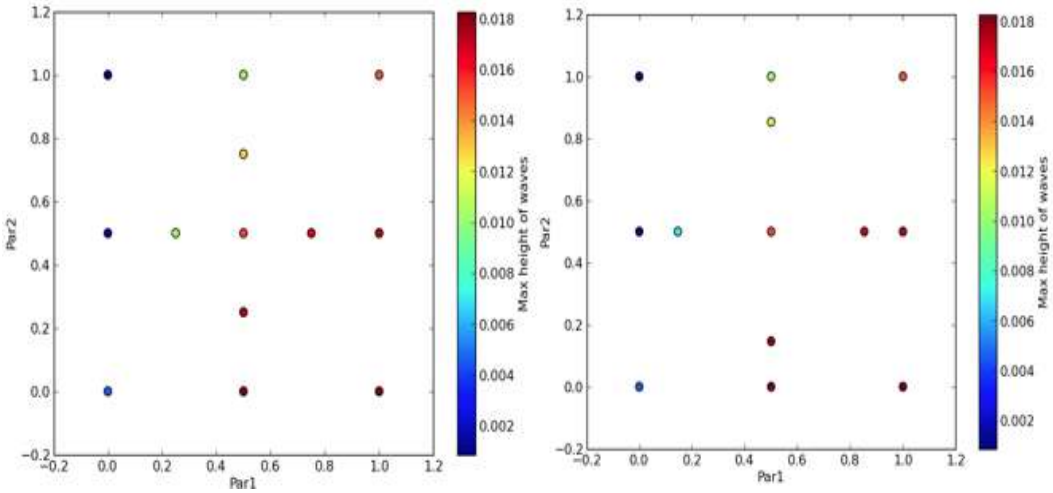

Figure 4.1: The uniform grid (left) and the Clenshaw Curtis grid (right) to sample two dimensional input parameter space. The output is the maximum of average water height (m) in the area of interst.

Simulation	The uniform grid	Clenshaw Curtis gird
1	1.5236004517611444 20e-02	1.5236004517611444 20e-02
2	1.3848605204247743 66e-03	1.3848605204247743 66e-03
3	1.7332085319910241 21e-02	1.7332085319910241 21e-02
4	1.8248149580292270 80e-02	1.8248149580292270 80e-02
5	1.0410268379780214 52e-02	1.0410268379780214 52e-02
6	1.0572359928070700 07e-02	7.0490449860333495 58e-03
7	1.6799185499548639 77e-02	1.7096123767969213 79e-02
8	4.5914156437384875 95e-03	4.5914156437384875 95e-03
9	1.8259595191141779 48e-02	1.8259595191141779 48e-02
10	8.7321368608270916 18e-04	8.7321368608270916 18e-04
11	1.5019420938577353 33e-02	1.5019420938577353 33e-02
12	1.7383091390699219 33e-02	1.7945915466785207 49e-02
13	1.2717430529331323 62e-02	1.1726243718298744 18e-02

Table 4.1: The output is the maximum of average water height (m) in the area of interst

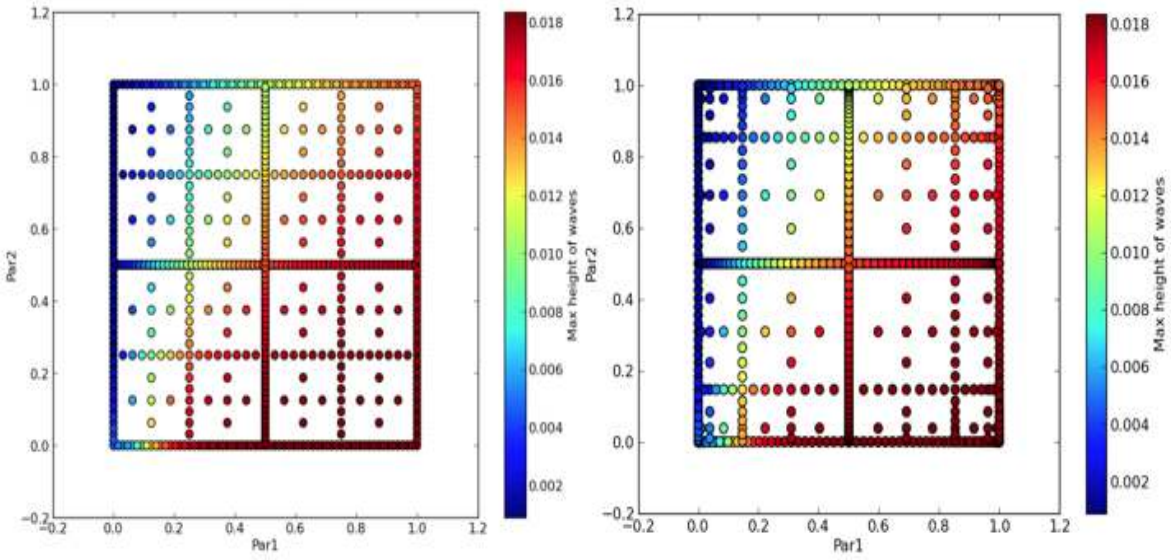

Figure 4.2: The 705 simulations of uniform grid (left) and the Clenshaw Curtis grid (right) in two dimensional input parameter space. The output is the maximum of average water height (m) in the area of interst

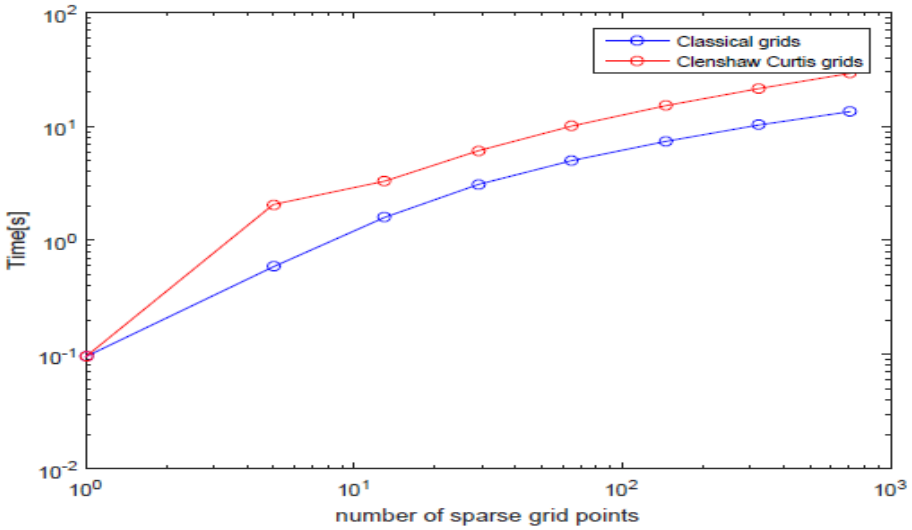

Figure 4.3: The computational cost of running a simulation with two dimensional input parameters.

To represent an increasing number of uncertain input parameters, we fit a sequence of Gaussian bumps ξ. Now, we quantify the uncertainty in the output taking into consideration three uncertain input parameters. Similarly, in Fig 4.4, we present the simulation with respect to three uncertain input parameters by using the uniform sparse grid and the Clenshaw Curtis grid, respectively. We can see that the resulting output of the maximum of the incoming wave height (m) with three uncertain input parameters is similar with the output based on two uncertain input parameters. Moreover, the simulation sampled both on the uniform and the

Clenshaw Curtis sparse grids provides no large different outputs. Fig 4.5 shows that a large number of simulation has been made. We compare the numerical values of simulations of the uniform sparse grid with the Clenshaw Curtis grid, see Table 4.2. Again, Fig 4.6 indicates that the computational cost of the simulation based on three uncertain input parameters by using Clenshaw Curtis grid interpolation is higher than that of the uniform sparse grid interpolation. Further work will include a simulation with a larger number of uncertain input parameters.

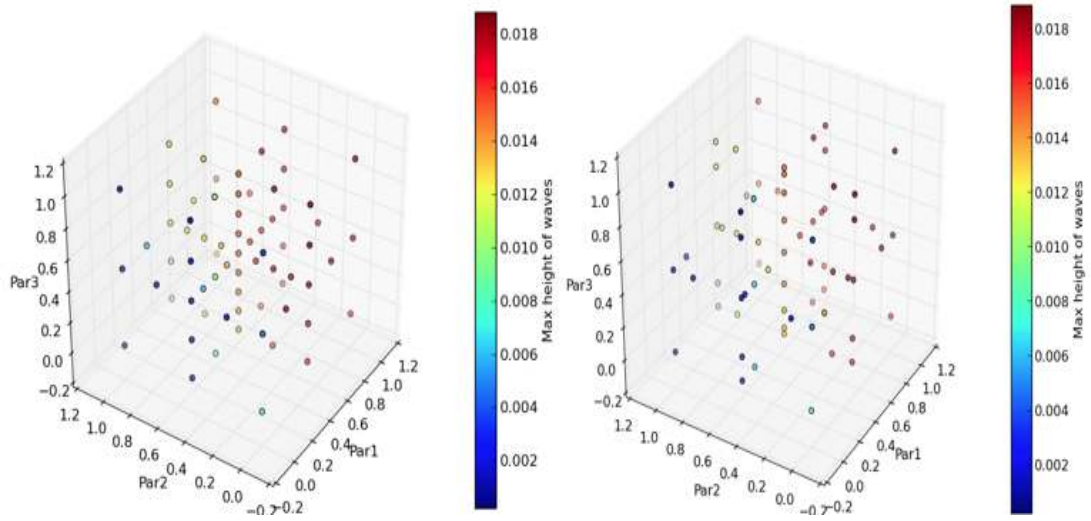

Figure 4.4: The uniform grid (left) and the Clenshaw Curtis grid (right) to sample three dimensional input parameter space. The output is the maximum of average water height (m) in the area of interst.

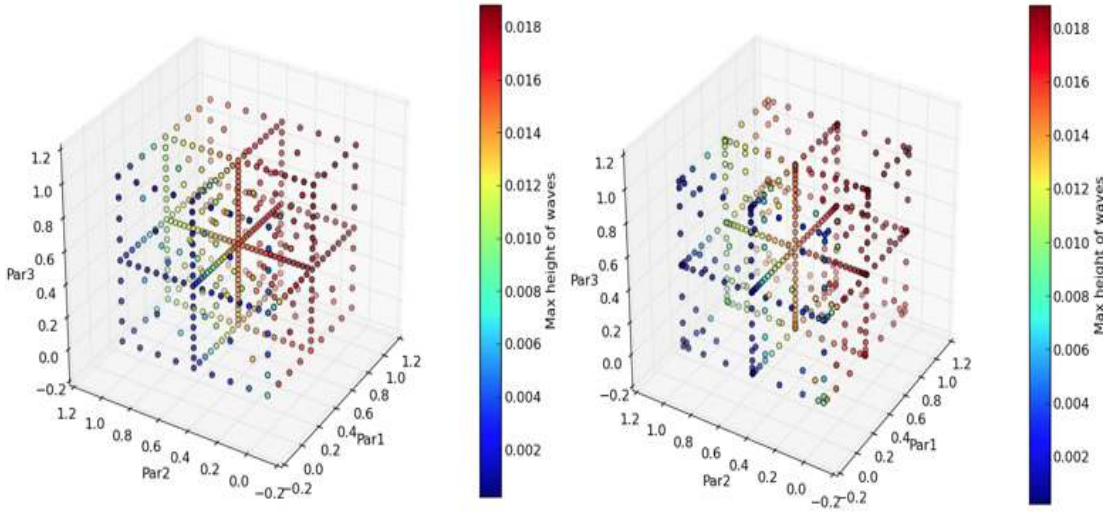

Figure 4.5: A large number of simulations of uniform grid (left) and the Clenshaw Curtis grid (right) in three dimensional input parameter space. The output is the maximum of average water height (m) in the area of interst

Simulation	The uniform grid	Clenshaw Curtis
1	1.523600451953149165e-02	1.523600451953149165e-02
2	1.384860545117223840e-03	1.384860545117223840e-03
3	1.733208531976273420e-02	1.733208531976273420e-02
4	1.824814958012884597e-02	1.824814958012884597e-02
5	1.041026836341906597e-02	1.041026836341906597e-02
6	1.371959107197964073e-02	1.371959107197964073e-02
7	1.553344011803307195e-02	1.553344011803307195e-02
8	1.057235993238774679e-02	7.049038307056894326e-03
9	1.679918551024738721e-02	1.709612375899659947e-02
10	4.591416366657601056e-03	4.591416366657601056e-03
11	1.825959519707771178e-02	1.825959519707771178e-02
12	8.732137088536054415e-04	8.732137088536054415e-04
13	1.501942094074133066e-02	1.501942094074133066e-02
14	1.738309139112857032e-02	1.794591549329294794e-02
15	1.271743052925893708e-02	1.172624371784981162e-02
16	6.219963032505354539e-04	6.219963032505354539e-04
17	1.619740534018162398e-02	1.619740534018162398e-02
18	1.311346015331782411e-03	1.311346015331782411e-03
19	1.790927349468871693e-02	1.790927349468871693e-02
20	1.667133584986756761e-02	1.667133584986756761e-02
21	9.181909615137033576e-03	9.181909615137033576e-03
22	1.825833264238390513e-02	1.825833264238390513e-02
23	1.091956545667991524e-02	1.091956545667991524e-02
24	1.470680631559035528e-02	1.428710005453914106e-02
25	1.543861734568164473e-02	1.551558042330878490e-02

Table 4.2: The output is the maximum of average water height (m) based on three uncertain input parameters in the area of interst

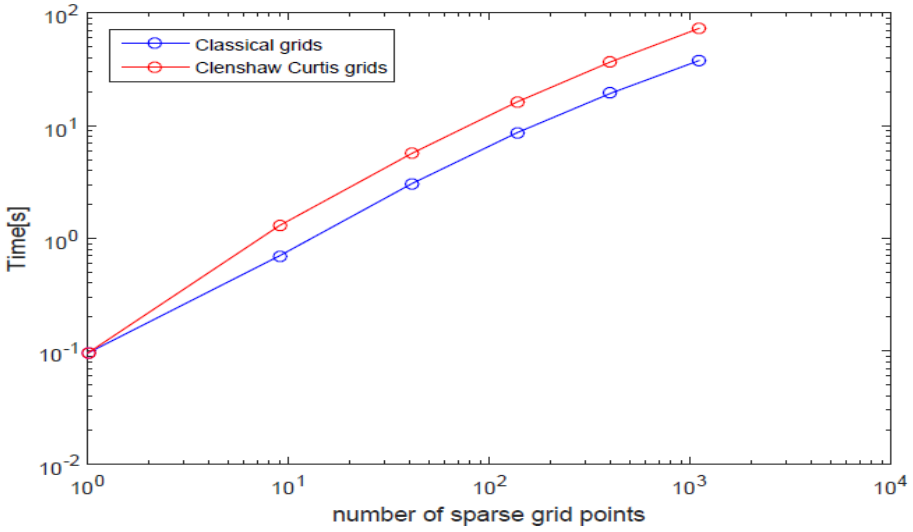

Figure 4.6: The computational cost of running a simulation with three dimensional input parameters.

Chapter 5

Conclusion

The dominant motivation for developing the sparse grids is to break the curse of dimensionality. We start from the underlying tensor product approach, based upon different 1D multilevel bases such as the classical piecewise linear hierarchical basis to higher-dimensional multilevel bases. We then presented the sparse grids of combination technique and proved that the hierarchical sparse grid interpolation is equivalent to the interpolant using combination approach. We introduced the Clenshaw Curtis quadrature grid to compare with the classical sparse grid. Moreover, we demonstrated the effectiveness of sparse grids in a series of experiments and discussed their properties with respect to computational complexity, discretization error, and smoothness requirements. The presented numerical results of these experiments include 2D and multi-dimensions model problems. Finally, we applied the uniform sparse grids and the Clenshaw Curtis quadrature grid to uncertainty quantification of the output of the Okushiri tsunami simulation. The output provides the maximum of the time-dependent average tsunami height for an increasing number of uncertain input parameters.

Future work will include an investigation of the adaptive sparse grid since ordinary sparse grids only work well under certain smoothness conditions. Further discussion of the reduction of the curse of dimensionality for different test functions, as well as possible development of a surrogate method based on the sparse grids.

Bibliography

[1] Bellman, Richard E, *Adaptive control processes: a guided tour*, Princeton university press, 1961

[2] Smolyak, Sergey A, *Quadrature and interpolation formulas for tensor products of certain classes of functions*, Dokl. Akad. Nauk SSSR, Vol. 4, No. 240-243. pp 123, 1963

[3] Zenger, Christoph, *Sparse grid tutorial*, Proceedings of the Research Workshop of the Israel Science Foundation on Multiscale Phenomenon, Modelling and Computation pp. 86, 1997

[4] Garcke, Jochen and others, *Sparse grids in a nutshell*, Mathematical Sciences Institute, Australian National University, Canberra Australia, pp 7, 2006

[5] Bungartz, Hans-Joachim and Griebel, Michael, *Sparse grids*, Acta numerica. Cambridge Univ Press, Vol. 13, No. 1. pp 147–269, 2004

[6] Bungartz, Hans-Joachim and Griebel, Michael, *A note on the complexity of solving Poisson's equation for spaces of bounded mixed derivatives*, Journal of Complexity. Elsevier, Vol. 15, No. 2. pp 167–199, 1999

[7] Pardo, Luis M, *Foundations of computational mathematics, Santander 2005*, Cambridge University Press, Vol. 13, No. 2. pp 137–140, 2006

[8] Griebel, Michael and Schneider, Michael and Zenger, Christoph, *A combination technique for the solution of sparse grid problems*, Technische Universität, 1990

[9] Kirby, Robert C and Knepley, Matthew and Logg, Anders and Scott, L Ridgway, *Optimizing the evaluation of finite element matrices*, SIAM, Vol. 27, No. 3. pp 741–758, 2005

[10] Hegland, Markus and Garcke, Jochen and Challis, Vivien, *Algebra and its Applications*, Elsevier, Vol. 420 No. 2. pp 137–140, 2007

[11] Barden, Jeffrey Michael, *A Modified Clenshaw-Curtis Quadrature Algorithm*, Worcester Polytechnic Institute, 2013

[12] Burkardt, John, *Slow exponential growth for Clenshaw Curtis sparse grids*, September, 2012

[13] Garcke, Jochenl, *Sparse grids in a nutshell*, Springer, No. 2. pp 57–80, 2012

[14] Andreas Klimke, *Sparse grid interpolation in Matlab*, $http://people.sc.fsu.edu/\ jburkardt/m_src/toms847/toms847.html$, 2005

[15] Rosenbrock, HoHo, *An automatic method for finding the greatest or least value of a function*, The Computer Journal. Br Computer Soc, Vol. 3, No. 3. pp175–184, 1960

[16] Sacks, Jerome and Welch, William J and Mitchell, Toby J and Wynn, Henry P, *Design and analysis of computer experiments*, Statistical science. JSTOR, pp 409–423, 1989

[17] Matsuyama, Masafumi and Tanaka, Hiroyoshi, *An experimental study of the highest run-up height in the 1993 Hokkaido Nansei-oki earthquake tsunami*, Vol. 21, No. 3. pp 879–889, 2001

[18] Dirk and Peherstorfer, Benjamin and Bungartz, Hans-Joachim, *Spatially adaptive sparse grids for high-dimensional data-driven problems* Journal of Complexity, vol 26, No.5, pp.249–275, 2010,