A HYPER-SPHERICAL ADAPTIVE SPARSE-GRID METHOD FOR HIGH-DIMENSIONAL DISCONTINUITY DETECTION [*]

G. ZHANG[†], C. WEBSTER[‡], M. GUNZBURGER[§], AND J. BURKARDT[¶]

Abstract. This work proposes and analyzes a hyper-spherical adaptive hierarchical sparse-grid method for detecting jump discontinuities of functions in high-dimensional spaces. The method is motivated by the theoretical and computational inefficiencies of well-known adaptive sparse-grid methods for discontinuity detection. Our novel approach constructs a function representation of the discontinuity hyper-surface of an N-dimensional discontinuous quantity of interest, by virtue of a hyper-spherical transformation. Then, a sparse-grid approximation of the transformed function is built in the hyper-spherical coordinate system, whose value at each point is estimated by solving a one-dimensional discontinuity detection problem. Due to the smoothness of the hyper-surface, the new technique can identify jump discontinuities with significantly reduced computational cost, compared to existing methods. Moreover, hierarchical acceleration techniques are also incorporated to further reduce the overall complexity. Rigorous complexity analyses of the new method are provided as are several numerical examples that illustrate the effectiveness of the approach.

Key words. discontinuity detection, hyper-spherical coordinate system, adaptive sparse grid, rare events, hierarchical acceleration

1. Introduction.

Numerical approximation is an important tool used to define solution techniques for physical, biological, social, and economic systems. In simulations of such systems, the relationship between the inputs that drive the system and the outputs, i.e., the system responses, are described by a multivariate function which is usually the target of the numerical approximation. Often the target function exhibits jump discontinuities, which have motivated many research efforts devoted to discontinuity detection. Traditionally, discontinuity detection has been associated with capturing jump discontinuities of a process with respect to temporal and/or spatial variables; thus, most efforts are restricted to low-dimensional problems. However, high-dimensional discontinuity detection is of significant importance to cases where the system outputs depend on a large number of input variables. For example, this challenge arises in uncertainty quantification (UQ), where physical systems with uncertainties are described by stochastic partial differential equations (SPDEs) with random input data. It is well known that an output of interest derived from of the solution of an SPDE may depend on a large number of random variables. Outputs of interest often contain jump discontinuities, sometimes because of irregular behavior in random coefficients, or because the output of interest itself is defined in terms on non-smooth functions, e.g., indicator functions. Such challenges also arise in optimization and control problems where again, the controls are characterized using a large number

[*] This material is based upon work supported in part by the U.S. Air Force of Scientific Research under grant numbers 1854-V521-12 and FA9550-11-1-0149; by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Applied Mathematics program under contract and award numbers ERKJ259, ERKJE45, and DE-SC0010678; and by the Laboratory Directed Research and Development program at the Oak Ridge National Laboratory, which is operated by UT-Battelle, LLC., for the U.S. Department of Energy under Contract DE-AC05-00OR22725.

[†] Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831(zhangg@ornl.gov).

[‡] Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831(webstercg@ornl.gov).

[§] Department of Scientific Computing, Florida State University, Tallahassee, FL 32306 (gunzburg@fsu.edu).

[¶] Department of Scientific Computing, Florida State University, Tallahassee, FL 32306 (jburkardt@fsu.edu).

of parameters and discontinuous cost functionals typically arise. As such, the development of accurate and efficient numerical techniques for detecting high-dimensional discontinuities is important to not only UQ and control, but to other mathematics, engineering, and science research communities.

A straightforward approach to resolve the challenges faced when approximating discontinuous functions is to first subdivide the high-dimensional parameter domain into several subdomains, in each of which the target function is continuous or even smoother. Then, in each subdomain, construct a piecewise continuous polynomial approximation using well-known methods such as sparse-grid interpolants [4, 18, 19] or even orthogonal polynomial expansions [7]. Obviously, these approaches require that the boundaries of the subdomains follow the discontinuity manifolds of the target function. Although such approaches are conceptually easy to understand, they are severely challenged numerically when one requires accurate representations of the detected discontinuities in even moderate dimensions. Recently, several attempts have been made to alleviate the challenges in locating discontinuities. In [1, 2], a polynomial annihilation approach, originally developed for one and two-dimensional edge detection, was extended to solve problems in higher dimensions. However, such methods rely on the evaluation of the target function based on a set of local tensor-product grids, so that the number of function evaluations grows exponentially as the dimension increases. Improvements were made in [15] by incorporating an adaptive hierarchical sparse-grid (AHSG) approximation in order to reduce the computational cost. The AHSG methods have been demonstrated to be effective in high-dimensional function approximation [4,23,25], numerical integration [3,8,9,11,20], finite difference methods [10] and Bayesian inference [17, 24, 26], etc. Moreover, the AHSG methods also have successfully impacted a variety of important applications, including data mining [5], manifold learning [6], image processing [21] and financial engineering [14], etc. However, the effectiveness of the AHSG approximations inextricably relies on the smoothness of the target function. When approximating a discontinuous function, mesh refinement is invariably needed in the vicinity of discontinuities, resulting in a significant deterioration in the sparsity of the grid, i.e., using an AHSG method, a discontinuity of an N-dimensional function, which typically occurs across an $N-1$ dimensional hyper-surface, has to be approximated using a “dense” grid, as illustrated by Example 3.1 in §3.2. This disadvantage dramatically limits the applicability of AHSG methods for high-dimensional discontinuity detection.

To combat these challenges, in this work, we propose a hyper-spherical adaptive hierarchal sparse-grid (HS-AHSG) method for approximating N-dimensional functions containing jump discontinuities across an $N-1$ dimensional hyper-surface. Our HS-AHSG approach achieves the desired performance and retains most of the grid sparsity of AHSG methods, that are known to be effective for the approximation of smooth functions. The basic idea is to approximate the discontinuity hyper-surface directly instead of approximating the discontinuous function, motivated by observing that the hyper-surface itself is often continuous or even smoother. Therefore, the number of grid points needed to approximate the hyper-surface can be significantly reduced compared to existing AHSG methods. To achieve this, the first step is to define a function representation of the $N-1$-dimensional discontinuity hyper-surface. Under a mild assumption about its geometry, the hyper-surface is transformed to a function in a hyper-spherical coordinate system. Note that the transformed function is defined in the subspace constituted by $N-1$ angle coordinates; the function value at a certain point is the Euclidean distance between the origin of the hyper-spherical coordinate system and the discontinuity along the direction determined by the $N-1$ angles. The next step is to develop an approach to evaluate the transformed function, i.e., calculating the desired Euclidean distance at a given point. Fortunately, this is relatively easy to implement because it reduces to a one-dimensional discontinuity detection problem along each of the directions determined by the $N-1$ angles. Many existing techniques can be used to fulfill this relatively straightforward task, such as the polynomial annihilation or an existing adaptive method. In particular, if the discontinuous function has a *characteristic property* (defined in §2), e.g., a characteristic function, then root-finding methods can be applied as well. Based on the above two steps, an HS-AHSG approximation of the discontinuous hyper-surface can be constructed in the $N-1$-dimensional subspace, with the use of the hyper-spherical coordinate system.

The efficiency of our algorithm is characterized by the total number of function evaluations required by the HS-AHSG approximation. Thus, the computational complexity is not the number of sparse-grid points, but is the sum of the number of function evaluations consumed by all the one-dimensional discontinuity detection problems. Taking the bisection method as an example, the number of iterations required to achieve a prescribed accuracy is determined by the length of initial search interval. Thus, to further improve the computational efficiency, we incorporate the hierarchal acceleration technique proposed in [7, 13] into the HS-AHSG framework. Specifically, the HS-AHSG approximation on a coarse sparse grid is used to predict the value of the transformed function at the new added points on a finer sparse grid. In this way, the length of the initial search interval for each bisection simulation is significantly reduced, as well as the necessary number of search iterations. It is worth noting that the proposed method differs from existing efforts on manifold learning [6, 22], which requires a set of data points that are on/near the target manifold, and the solution procedure is to find the optimal approximation in the sparse polynomial subspace by minimizing objective functions. In our approach, we do not require a set of points on/near the target hyper-surface to run our algorithm. The sparse grid points on/near the hyper-surface are captured using one-dimensional root-finding algorithms.

The main contributions of this paper are summarized as follows.

- A comprehensive framework of the HS-AHSG method for high-dimensional discontinuity detection is constructed.
- The performance of several approaches for the evaluation of the transformed function are investigated.
- The computational efficiency of the HS-AHSG method is improved by incorporating hierarchical acceleration techniques.
- Rigorous complexity analyses are provided for the proposed algorithms.
- Numerical examples are provided to illustrate the theoretical results and demonstrate the efficiency of HS-AHSG methods.

The rest of the paper is organized as follows. Specific problem definition and preliminary notions are discussed in §2. In §3, the AHSG method is briefly reviewed and an example is given to illustrate its disadvantages when attempting to detect even moderate dimensional discontinuities. Our main results are given in §4. The function representation of the discontinuity hyper-surface and its evaluation are discussed in §4.1 and §4.2, respectively; the basic and accelerated HS-AHSG algorithms are presented in §4.3 and §4.4, respectively. Rigorous complexity analyses are conducted in §4.5. Numerical tests are provided in §5; concluding remarks, limitations of this approach as well as future directions are discussed in §6.

2. Problem setting. Let Γ denote an open bounded domain in $\mathbb{R}^N$, $N \geq 1$, and let $\partial\Gamma$ denote its boundary. We assume there exists an $N-1$ dimensional hyper-surface in Γ, denoted by γ, separating the domain Γ into disjoint open subdomains Γ_1 and Γ_2, such that $\Gamma = \Gamma_1 \cup \gamma \cup \Gamma_2$, $\overline{\Gamma}_1 \cap \overline{\Gamma}_2 = \gamma$, and $\Gamma_1 \cap \gamma = \Gamma_2 \cap \gamma = \Gamma_1 \cap \Gamma_2 = \emptyset$. We observe that the volume of γ in $\mathbb{R}^N$ is zero and Γ_1 and Γ_2 are both open along γ. We consider a generic N-dimensional discontinuous function $f(\boldsymbol{y}) : \Gamma \to \mathbb{R}$ given by

$$f(\boldsymbol{y}) = \begin{cases} f_1(\boldsymbol{y}) & \text{if } \boldsymbol{y} \in \overline{\Gamma}_1 \\ f_2(\boldsymbol{y}) & \text{if } \boldsymbol{y} \in \overline{\Gamma}_2 \backslash \gamma, \end{cases} \tag{2.1}$$

where $\boldsymbol{y} = (y_1, \ldots, y_N) \in \mathbb{R}^N$ and $f_1(\boldsymbol{y})$ and $f_2(\boldsymbol{y})$ are continuous functions in $\overline{\Gamma}_1$ and $\overline{\Gamma}_2 \backslash \gamma$, respectively. Based on the fact that $f(\boldsymbol{y}) = f_1(\boldsymbol{y})$ for $\boldsymbol{y} \in \gamma \subseteq \partial\Gamma_1$, we assume $f(\boldsymbol{y})$ has a jump discontinuity on γ such that $f_1(\boldsymbol{y}^*) = \lim_{\Gamma_1 \ni \boldsymbol{y} \to \boldsymbol{y}^* \in \gamma} f_1(\boldsymbol{y}) \neq \lim_{\Gamma_2 \ni \boldsymbol{y} \to \boldsymbol{y}^* \in \gamma} f_2(\boldsymbol{y}) < +\infty$ for $\boldsymbol{y}^* \in \gamma$, which means the discontinuity only occurs when approaching γ from the subdomain Γ_2. The goal is to accurately capture the discontinuity hyper-surface γ. We also assume that $\partial\Gamma_1$ is a *continuous* hyper-surface such that Γ_1 and Γ_2 are disjoint. As such, there exists a *continuous* function $G(\boldsymbol{y})$ defined in $\overline{\Gamma}$ such that $\gamma = \{\boldsymbol{y} \in \overline{\Gamma} \,|\, G(\boldsymbol{y}) = 0\}$, i.e., γ is implicitly defined by the equation $G(\boldsymbol{y}) = 0$, and such that $f = f_1(\boldsymbol{y})$ for $G(\boldsymbol{y}) > 0$ (i.e., $\boldsymbol{y} \in \overline{\Gamma}_1 \backslash \gamma$) and $f = f_2(\boldsymbol{y})$ for $G(\boldsymbol{y}) < 0$ (i.e., $\boldsymbol{y} \in \overline{\Gamma}_2 \backslash \gamma$). Note that $G(\boldsymbol{y}) = 0$ is only an abstract representation of γ and that its availability is not necessary for detecting the discontinuity. Moreover, for a specific γ, the function $G(\boldsymbol{y})$ is not unique.

In one dimension ($N = 1$), γ reduces to one or two points in $\Gamma \subset \mathbb{R}$, so that it is relatively easy to capture the discontinuity of $f(\boldsymbol{y})$. However, in higher dimensions ($N > 1$), detecting discontinuities becomes difficult because γ is, in general, an $N-1$ dimensional hyper-surface with measure zero in $\mathbb{R}^N$. What is worse, there is no direct information available about the location or geometry of γ, so that we can only rely on indirect information about $f(\boldsymbol{y})$ and $G(\boldsymbol{y})$ to infer the location of γ. In this work, $f(\boldsymbol{y})$ in (2.1) is treated as a black-box function, i.e., given any $\boldsymbol{y} \in \overline{\Gamma}$ as an input, the function value can be obtained as an output without any knowledge about the analytical expressions of $f(\boldsymbol{y})$ or $G(\boldsymbol{y})$. Before moving forward, we provide two examples of discontinuous functions of interest.

Example 2.1. Consider the generic function $f(\boldsymbol{y}) : \overline{\Gamma} \to \mathbb{R}$ defined by

$$f(\boldsymbol{y}) = \begin{cases} f_1(\boldsymbol{y}) & \text{if } y_1^2 + \cdots + y_N^2 \leq \mu^2, \\ f_2(\boldsymbol{y}) & \text{if } y_1^2 + \cdots + y_N^2 > \mu^2, \end{cases} \tag{2.2}$$

where f_1 and f_2 are continuous functions, μ is a positive real constant. In this case, the function $G(\boldsymbol{y})$ is defined by $G(\boldsymbol{y}) = \mu^2 - \sum_{n=1}^N y_n^2$. The discontinuity $\gamma = \{\boldsymbol{y} \in \mathbb{R}^N \,|\, G(\boldsymbol{y}) = 0\}$ is a sphere in $\mathbb{R}^N$ with radius μ and $\partial\Gamma_1 \cup \partial\Gamma = \emptyset$, $\gamma = \partial\Gamma_1$. There are three specific scenarios one must consider:

(S_1) $f(\boldsymbol{y})$ can be evaluated as a black-box function;

(S_2) $f(\boldsymbol{y})$ is the characteristic function of Γ_1, e.g., $f_1(\boldsymbol{y}) = 1$ and $f_2(\boldsymbol{y}) = 0$;

(S_3) Both $f(\boldsymbol{y})$ and $G(\boldsymbol{y})$ can be evaluated as black-box functions. □

Example 2.2. [Probability of an event depending on the solution of an SPDE] Let the function $u(\boldsymbol{y}, \mathbf{x}) : \Gamma \times D \to \mathbb{R}$ denote the solution of an SPDE given by $\mathcal{L}(a(\boldsymbol{y}, \mathbf{x}))(u(\boldsymbol{y}, \mathbf{x})) = h(\boldsymbol{y}, \mathbf{x})$ where the coefficient $a(\boldsymbol{y}, \mathbf{x})$ of the differential operator $\mathcal{L}$ and the right-hand side $h(\boldsymbol{y}, \mathbf{x})$ are random fields, and $\mathbf{x} \in D \subset \mathbb{R}^d (d = 1, 2, 3)$, $\boldsymbol{y} \in \Gamma \subset \mathbb{R}^N$ denote physical variables and random variables, respectively. By assuming that $\boldsymbol{y}$ has a joint probability density function $\rho(\boldsymbol{y})\colon \Gamma \to \mathbb{R}_+$ with $\rho(\boldsymbol{y}) \in L^\infty(\Gamma)$, in practice, we may be interested in quantifying the probability of an event about $u(\boldsymbol{y}, \mathbf{x})$. For example, such an event may be the spatial average $F(u) = \frac{1}{|D|}\int_D u(\boldsymbol{y}, \mathbf{x}) d\mathbf{x}$ exceeding a threshold value $\overline{u}$, where $|D|$ denotes the volume of the physical domain D. This probability can be expressed as

$$\mathbb{P}\big(F(u) \geq \overline{u}\big) = \int_\Gamma \mathcal{X}_{\{F(u)\geq\overline{u}\}}(\boldsymbol{y})\rho(\boldsymbol{y})d\boldsymbol{y}, \tag{2.3}$$

where $\mathcal{X}_{\{F(u)\geq\overline{u}\}}(\boldsymbol{y})$ is the characteristic function. As such, the target function $f(\boldsymbol{y})$ is $\mathcal{X}_{\{F(u)\geq\overline{u}\}}(\boldsymbol{y})\rho(\boldsymbol{y})$ and γ is determined by $G(\boldsymbol{y}) = F(u(\boldsymbol{y})) - \overline{u} = 0$. □

From the above examples, we observe that, in practice, there may be additional indirect information available about $f(\boldsymbol{y})$ and $G(\boldsymbol{y})$ that can help one capture discontinuities. For instance, in Example 2.2, when defining $\Gamma_1 = \{\boldsymbol{y} \in \Gamma \,|\, \mathcal{X}_{\{F(u)\geq\overline{u}\}}(\boldsymbol{y}) = 1\}$, the function $G(\boldsymbol{y})$ can be evaluated as well and the membership of a given $\boldsymbol{y} \in \Gamma$ in the subdomain Γ_1 can be determined by the computable value of $f(\boldsymbol{y})$. Thus, we consider discontinuity detection problems under one of the following three assumptions:

$\boldsymbol{A_1}$: Given $\boldsymbol{y} \in \Gamma$, only $f(\boldsymbol{y})$ can be evaluated;

$\boldsymbol{A_2}$: Given $\boldsymbol{y} \in \Gamma$, the value $f(\boldsymbol{y})$ can determine if $f(\boldsymbol{y}) = f_1(\boldsymbol{y})$ or $f(\boldsymbol{y}) = f_2(\boldsymbol{y})$, i.e., if $\boldsymbol{y} \in \Gamma_1$ or $\boldsymbol{y} \in \Gamma_2$;

$\boldsymbol{A_3}$: Given $\boldsymbol{y} \in \Gamma$, both $f(\boldsymbol{y})$ and $G(\boldsymbol{y})$ can be evaluated.

It is easy to see that $\boldsymbol{A_2}$ is a sufficient condition for $\boldsymbol{A_1}$ and that $\boldsymbol{A_3}$ is a sufficient condition for both $\boldsymbol{A_1}$ and $\boldsymbol{A_2}$. Under $\boldsymbol{A_1}$, it is known that there exist jump discontinuities in Γ, but no information about the location of γ can be inferred from the function values of $f(\boldsymbol{y})$. In the context of $\boldsymbol{A_2}$, function values of $f(\boldsymbol{y})$ can indicate the membership of a given point $\boldsymbol{y} \in \Gamma$ in the subdomain $\overline{\Gamma}_1 \in \Gamma$, which is referred to as the *characteristic property.* Under $\boldsymbol{A_3}$, because $G(\boldsymbol{y})$ can be evaluated directly, detecting γ is equivalent to finding all the roots of the implicit equation $G(\boldsymbol{y}) = 0$. In one dimension ($N = 1$), this is straightforward to accomplish using classic root-finding algorithms, e.g., the bisection method. In higher dimensions, classic root-finding methods might make it easy to find one root but approximately determining the whole surface γ is, in general, difficult. It is natural to look for more efficient algorithms for dealing with discontinuous functions satisfying $\boldsymbol{A_2}$ or $\boldsymbol{A_3}$. Such improved methods are discussed in detail in §4.

Because it is almost impossible to solve for the analytical expression describing the hyper-surface γ, the main goal of our effort is to efficiently construct, in N dimensions, an accurate approximate hyper-surface, denoted by $\widetilde{\gamma}$. To assess the performance of our approaches, the *accuracy* of $\widetilde{\gamma}$ as an approximation of γ is measured by the distance between γ and $\widetilde{\gamma}$ defined as

$$e_\infty = \mathrm{dist}(\gamma, \widetilde{\gamma}) = \sup_{\boldsymbol{x}\in\gamma} \inf_{\boldsymbol{x}'\in\widetilde{\gamma}} |\boldsymbol{x} - \boldsymbol{x}'|. \tag{2.4}$$

In addition, as indicated in (2.3), we are also interested in estimating the integral of $f(\boldsymbol{y})$ over a subdomain of interest, i.e., either Γ_1 or Γ_2. Without loss of generality, the *accuracy* of $\widetilde{\gamma}$ is thus also assessed by the metric

$$e_{\mathrm{int}} = \left| \int_{\Gamma_1} f(\boldsymbol{y})d\boldsymbol{y} - \int_{\widetilde{\Gamma}_1} f(\boldsymbol{y})d\boldsymbol{y} \right|, \tag{2.5}$$

where $\widetilde{\Gamma}_1$ is the approximation of Γ_1 resulting from the approximation $\widetilde{\gamma}$ of γ. On the other hand, as shown in Example 2.2, the computational cost on evaluating $f(\boldsymbol{y})$ or $G(\boldsymbol{y})$ often dominates the total cost of constructing $\widetilde{\gamma}$, e.g., because of the complexity of the PDE solver required to perform those evaluations. Thus, we use the number of function evaluations of either $f(\boldsymbol{y})$ or $G(\boldsymbol{y})$ as the metric to assess the *efficiency* of constructing $\widetilde{\gamma}$.

As discussed in §1, a straightforward way to estimate the integral $\int_{\Gamma_1} f(\boldsymbol{y})d\boldsymbol{y}$ is to use MC methods, but the computational cost may not be affordable due to the slow convergence. Alternatively, the adaptive hierarchical sparse-grid (AHSG) method has been employed in discontinuity detection [15], but its efficiency deteriorates dramatically as the dimension N increases. The new approach proposed in §4 is a variant of the AHSG method but features much improved efficiency. To set the stage, before introducing our approach, we will briefly review, in §3, the standard AHSG method and illustrate its unsatisfactory performance in discontinuity detection.

3. Adaptive hierarchical sparse-grid approximation.

In §3.1, we briefly review hierarchical sparse-grid interpolation that is the foundation of adaptive hierarchical sparse-grid (AHSG) interpolation (see [4] for details). In §3.2, the AHSG method is introduced and its shortcomings in high-dimensional discontinuity detection is illustrated via a numerical example.

3.1. Hierarchical sparse-grid interpolation.

The goal is to construct a Lagrange interpolant to a function $\eta(\boldsymbol{y}) : \Gamma \to \mathbb{R}$ using hierarchical piecewise polynomials [4]. We begin with the one-dimensional hat function having support $[-1, 1]$, given by $\psi(y) = \max\{0\,,\, 1-|y|\}$. An arbitrary hat function with support $(y_{l,i}-h_l, y_{l,i}+h_l)$ can be generated by dilation and translation, i.e., $\psi_{l,i}(y) = \psi\big(\frac{y+1-ih_l}{h_l}\big)$, where l denotes the resolution level, $h_l = 2^{-l+1}$ for $l = 0, 1, \ldots$ denotes the grid size of the level l grid, and $y_{l,i} = i\,h_l - 1$ for $i = 0, 1, \ldots, 2^l$ denotes the grid points. The basis function $\psi_{l,i}(y)$ has local support with respect to the level l grid and is centered at the grid point $y_{l,i}$; the number of grid points in the level l grid is $2^l + 1$. The space spanned by the nodal basis $\{\psi_{l,i}(y)\}_{i=0}^{2^l}$ is the standard piecewise linear finite element space. For each grid level $l > 0$, the interpolant of a continuous function $\eta(y)$ in terms of the nodal basis $\{\psi_{l,i}(y)\}_{i=0}^{2^l}$ is given by $\mathcal{U}_l(\eta) = \sum_{i=0}^{2^l} \eta(y_{l,i})\psi_{l,i}(y)$. By observing that $\mathcal{U}_{l-1}(\eta) = \mathcal{U}_l\big(\mathcal{U}_{l-1}(\eta)\big)$, we define the incremental operator

$$\Delta_l = \mathcal{U}_l - \mathcal{U}_{l-1} \quad \text{for } l \geq 0 \qquad \text{with} \quad \mathcal{U}_{-1} = 0, \tag{3.1}$$

where $\Delta_l(\eta)$ only involves the basis functions $\psi_{l,i}(y)$ for $i = 1, 3, 5, \ldots, 2^l - 1$. Then, the interpolant $\mathcal{U}_l(\eta)$ can be represented in the following *hierarchical* form

$$\mathcal{U}_l(\eta) = \mathcal{U}_{l-1}(\eta) + \Delta_l(\eta) = \cdots = \mathcal{U}_0(\eta) + \sum_{l'=1}^{l} \Delta_{l'}(\eta), \tag{3.2}$$

where the basis functions needed by $\mathcal{U}_0(\eta)$ and $\Delta_{l'}(\eta)$ for $l' = 1, \ldots, l$ constitute the one-dimensional hierarchical basis of level l.

Next, we consider the hierarchical sparse-grid interpolation of a multivariate function $\eta(\boldsymbol{y})$ defined, without loss of generality, over the unit hypercube $\Gamma = [-1, 1]^N \subset \mathbb{R}^N$. The one-dimensional hierarchical basis can be directly extended to the N-dimensional domain Γ using tensorization. Specifically, the N-variate basis function $\psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y})$ associated with the point $\boldsymbol{y}_{\mathbf{l},\mathbf{i}} = (y_{l_1,i_1}, \ldots, y_{l_N,i_N})$ is defined using tensor products, i.e., $\psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y}) := \prod_{n=1}^N \psi_{l_n,i_n}(y_n)$, where $\mathbf{l} = (l_1, \ldots, l_N) \in \mathbb{N}^N$ is a multi-index indicating the resolution level of the basis function. Note that the resolution level can be different in each of the N directions. Then, the sparse-grid the level L hierarchical sparse-grid approximation $\eta_L(\boldsymbol{y})$ of the target function $\eta(\boldsymbol{y})$ is given by

$$\begin{aligned}\eta_L(\boldsymbol{y}) &= \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \left(\Delta_{l_1} \otimes \cdots \otimes \Delta_{l_N}\right)(\eta)(\boldsymbol{y}) \\ &= \eta_{L-1}(\boldsymbol{y}) + \sum_{|\mathbf{l}|=L} \left(\Delta_{l_1} \otimes \cdots \otimes \Delta_{l_N}\right)(\eta)(\boldsymbol{y}) \\ &= \eta_{L-1}(\boldsymbol{y}) + \sum_{|\mathbf{l}|=L} \sum_{\mathbf{i} \in B_{\mathbf{l}}} [\eta(\boldsymbol{y}_{\mathbf{l},\mathbf{i}}) - \eta_{L-1}(\boldsymbol{y}_{\mathbf{l},\mathbf{i}})] \psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y}), \\ &= \eta_{L-1}(\boldsymbol{y}) + \sum_{|\mathbf{l}|=L} \sum_{\mathbf{i} \in B_{\mathbf{l}}} \omega_{\mathbf{l},\mathbf{i}} \psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y}) = \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \sum_{\mathbf{i} \in B_{\mathbf{l}}} \omega_{\mathbf{l},\mathbf{i}} \psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y}),\end{aligned} \tag{3.3}$$

where the multi-index set $B_{\mathbf{l}}$ is given by

$$B_{\mathbf{l}} := \left\{ \mathbf{i} \in \mathbb{N}^N \;\middle|\; \begin{array}{lll} i_N \in \{0\} & \text{for } n = 1, \ldots, N & \text{if } l_n = 0 \\ i_N \in \{0, 2\} & \text{for } n = 1, \ldots, N & \text{if } l_n = 1 \\ i_N \in \{1, 3, 5, \ldots, 2^{l_N} - 1\} & \text{for } n = 1, \ldots, N & \text{if } l_n > 1 \end{array} \right\}, \tag{3.4}$$

$|\mathbf{l}| \equiv l_1 + \cdots + l_N \leq l$ defines the resolution level of the incremental operator $\Delta_{l_1} \otimes \cdots \otimes \Delta_{l_N}$ and $\omega_{\mathbf{l},\mathbf{i}} = \eta(\boldsymbol{y}_{\mathbf{l},\mathbf{i}}) - \eta_{L-1}(\boldsymbol{y}_{\mathbf{l},\mathbf{i}})$ denotes the *multi-dimensional hierarchical surplus*. This interpolant is a direct extension, via the Smolyak algorithm, of the one-dimensional hierarchical interpolant. The definition of the surplus $w_{\mathbf{l},\mathbf{i}}$ is based on the facts that $\eta_l(\eta_{l-1}(\boldsymbol{y})) = \eta_{l-1}(\boldsymbol{y})$ and $\eta_{l-1}(\boldsymbol{y}_{\mathbf{l},\mathbf{i}}) - \eta(\boldsymbol{y}_{\mathbf{l},\mathbf{i}}) = 0$ for $|\mathbf{l}| = l$.

We denote by $\mathcal{H}_{\mathbf{l}}(\Gamma) = \{\boldsymbol{y}_{\mathbf{l},\mathbf{i}} | \mathbf{i} \in B_{\mathbf{l}}\}$ the set of points corresponding to the operator $\Delta_{l_1} \otimes \cdots \otimes \Delta_{l_N}$ with $\mathbf{l} = (l_1, \ldots, l_N)$. Then, the sparse grid corresponding to the interpolant η_l is given by

$$\mathcal{H}_l(\Gamma) = \bigcup_{|\mathbf{l}| \leq l} \mathcal{H}_{\mathbf{l}}(\Gamma),$$

where $\mathcal{H}_l(\Gamma)$ is nested, i.e., $\mathcal{H}_{l-1}(\Gamma) \subset \mathcal{H}_l(\Gamma)$. In addition, with $\Delta\mathcal{H}_0(\Gamma) = \mathcal{H}_0(\Gamma)$, we denote by $\Delta\mathcal{H}_l(\Gamma) = \mathcal{H}_l(\Gamma) \backslash \mathcal{H}_{l-1}(\Gamma)$ the set of newly added grid points on level l.

3.2. Adaptive hierarchical sparse-grid interpolation.

By virtue of the surpluses $\omega_{\mathbf{l},\mathbf{i}}$, the interpolant in (3.3) can be represented in a hierarchical manner, i.e.,

$$\eta_L(\boldsymbol{y}) = \eta_{L-1}(\boldsymbol{y}) + \Delta\eta_L(\boldsymbol{y}),$$

where $\eta_{L-1}(\boldsymbol{y})$ is the sparse-grid interpolant and $\Delta\eta_L(\boldsymbol{y})$ is the hierarchical surplus interpolant. According to the analyses in [4], for smooth functions, the surpluses $\omega_{\mathbf{l},\mathbf{i}}$ of the sparse-grid interpolant $\eta_L(\boldsymbol{y})$ tend to zero as L tends to infinity. For example, in the context of using piecewise-linear hierarchical bases and $\eta(\boldsymbol{y})$ having bounded mixed second-order derivatives, the surplus $\omega_{\mathbf{l},\mathbf{i}}$ can be bounded by

$$|\omega_{\mathbf{l},\mathbf{i}}| \leq C_{\text{surp}} 2^{-2 \cdot |\mathbf{l}|} \quad \text{for } \mathbf{i} \in B_{\mathbf{l}}, \tag{3.5}$$

where the constant C_{surp} is independent of the level $|\mathbf{l}|$ and the dimension N (see Lemma 3.3 in [4]). This provides a good avenue for constructing adaptive sparse grids using the magnitude of the surplus as an error indicator, especially for irregular functions having, e.g., steep slopes or jump discontinuities. An alternative adaptive sparse-grid approach based on wavelet basis is described in [12].

Specifically, the one-dimensional hierarchical grid has a tree-like structure where each grid point $y_{l,i}$ on level l has two children on level $l+1$, namely $y_{l+1,2i-1}$ and $y_{l+1,2i+1}$. At each successive interpolation level, the basic idea of adaptivity is to use the hierarchical surplus as an error indicator to refine the grid by adding two new points on the next level for each point on the current level for which the magnitude of its surplus is larger than the prescribed error tolerance. Then, it is straightforward to extend the adaptivity from the one-dimension to the multi-dimensional adaptive sparse grid. In general, a grid point in an N-dimensional space has $2N$ children which are also the neighbor points of the parent point. We start with an isotropic sparse grid of level $L_{\min} > 0$ and build an approximation $\eta_{L_{\min}}(\boldsymbol{y})$ in order to capture the main profile of $\eta(\boldsymbol{y})$. Thus, for $L \geq L_{\min}$, we only add those grid points on level L whose parent on level $L-1$ has a surplus greater than the prescribed tolerance. In this way, the N-dimensional adaptive sparse-grid interpolant of level L with the error tolerance being $\alpha > 0$ can be represented by

$$\eta_{L,\alpha}(\boldsymbol{y}) = \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \sum_{\mathbf{i} \in B_{\mathbf{l}}^{\alpha}} \omega_{\mathbf{l},\mathbf{i}} \psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y}), \tag{3.6}$$

where the multi-index set $B_{\mathbf{l}}^{\alpha} \subseteq B_{\mathbf{l}}$ is defined by $B_{\mathbf{l}}^{\alpha} = \left\{ \mathbf{i} \in B_{\mathbf{l}} \,|\, |\omega_{\mathbf{l},\mathbf{i}}| \geq \alpha \right\}$. The corresponding adaptive sparse grid can be represented by $\mathcal{H}_{L,\alpha}(\Gamma) = \cup_{l=0}^{L} \Delta\mathcal{H}_{l,\alpha}(\Gamma)$, where $\Delta\mathcal{H}_{l,\alpha}(\Gamma) = \Delta\mathcal{H}_{l}(\Gamma)$ for $l \leq L_{\min}$, and $\Delta\mathcal{H}_{l,\alpha}(\Gamma)$ for $l > L_{\min}$, only contains the sparse grid points added by the mesh refinement.

In the literature, the AHSG method has been used to approximate irregular functions [4, 16] in low dimensional spaces ($N \leq 3$). However, in these cases, the AHSG method cannot achieve the desired efficiency as in approximating smooth functions. What is worse, it will eventually converge slower than a simple Monte Carlo method, even for a moderate 4-dimensional discontinuous function, as shown in the following example.

Example 3.1. The target $f(\boldsymbol{y})$ is the characteristic function in $\mathbb{R}^N$ given by

$$f(\boldsymbol{y}) = \begin{cases} 1 & \text{if } 1 - y_1^2 - \cdots - y_N^2 \geq 0, \\ 0 & \text{otherwise,} \end{cases} \tag{3.7}$$

where the discontinuity hyper-surface γ is the unit hyper-sphere in $\mathbb{R}^N$. For $N = 1, 2, 3, 4$, $L_{\min} = 4$, and $L_{\max} = 100$, we build AHSG approximation $f_{L,\alpha}(\boldsymbol{y})$ with $\alpha = 0.01$. The error is measured by the metric e_{int} defined in (2.5). Because the surplus will not decay to zero around the hyper-sphere, mesh refinement will not stop until the level L reaches $L_{\max}$. Thus, we compute and plot, in Figure 3.1, of the error e_{int} vs. the number of function evaluations, by increasing the resolution level L up to $L_{\max}$. For comparison, the error of Monte Carlo simulations are also plotted in Figure 3.1. We observe that the AHSG approximation outperforms Monte Carlo in the one and two-dimensional cases, but performs similarly in three dimensions, and, in four dimensions, Monte Carlo outperforms the AHSG. To investigate the reason of such failures, we plot, in Figure 3.2, the resulting adaptive sparse grids in the two and three dimensions for an error $e_{\text{int}} < 0.01$. Note that mesh refinement places a dense set of grid points in the vicinity of the discontinuities, resulting in a loss of the desired grid sparsity. In fact, the N-dimensional hypershere γ, across which the function is discontinuous, is approximated by an extremely dense grid. It is the loss of the sparsity that makes the AHSG approximation fail when attempting discontinuity detection in high-dimensional space. Moreover, because the target function $f(\boldsymbol{y})$ is discontinuous, the accuracy of the AHSG approximation cannot be improved by using high-order hierarchical basis [4]. In fact, the accuracy is worse for piecewise-quadratic approximations than it is for piecewise-linear approximations; see Figure 3.1.

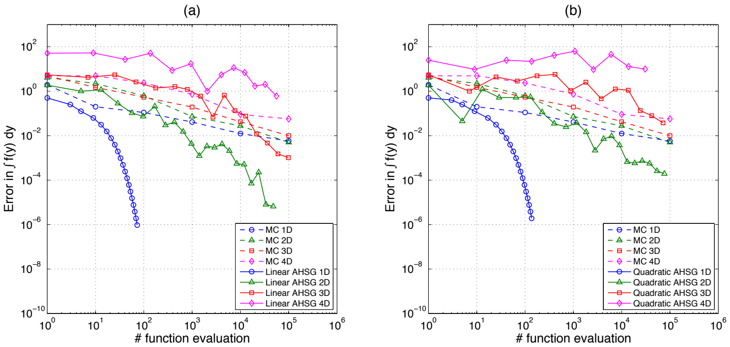

Fig. 3.1: The error in the approximations of the integral of $f(\boldsymbol{y})$ given by (3.7) vs. the number of function evaluations, using the AHSG and Monte Carlo methods with $N = 1, 2, 3, 4$ for (a) the AHSG approximations with a piecewise-linear hierarchical basis and (b) the AHSG approximations with a piecewise-quadratic hierarchical basis.

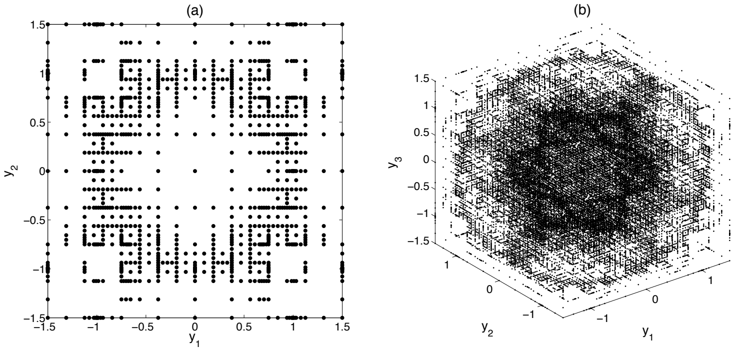

Fig. 3.2: The grids produced by the AHSG method for approximating the integral of $f(\boldsymbol{y})$ given by (3.7) with linear hierarchical basis and $e_{\text{int}} < 0.01$: (a) the two-dimensional sparse grid has 969 points; (b) the three-dimensional sparse grid has 936,093 points.

4. Hyper-spherical adaptive sparse-grid method for discontinuity detection. In this section, we propose a hyper-spherical adaptive hierarchical sparse-grid (HS-AHSG) method that overcomes the disadvantages of the AHSG method for high-dimensional discontinuity detection. The basic idea is to directly approximate the discontinuity hyper-surface γ itself, instead of refining the sparse grid in the vicinity of γ. Unlike the discontinuous function $f(\boldsymbol{y})$ in (2.1), the hyper-surface γ is continuous or even smoother, so that the drawbacks of the AHSG method mentioned above can be avoided when directly approximating γ. However, in general, γ is usually not a function in the Cartesian coordinate system in $\mathbb{R}^N$, so that the hyper-spherical transformation is introduced into our approach to convert γ into a function in the hyper-spherical coordinate system. Details about the conversion and the evaluation of the transformed function are discussed in §4.1 and §4.2, respectively. The main HS-AHSG algorithm is described in §4.3. In §4.4, the efficiency of the proposed algorithm is further improved by incorporating the hierarchal acceleration technique proposed in [13]. Rigorous error estimate and ε-complexity analyses are provided in §4.5, for the algorithms discussed in §4.3 and §4.4.

4.1. Representation of the discontinuity surface in the hyper-spherical coordinate system. A hyper-spherical coordinate system is a generalization of the three-dimensional spherical coordinate systems. It has: one radial coordinate r ranging over $[0,\infty)$; one angular coordinate θ_{N-1} ranging over $[0,2\pi]$ and; $N-2$ angular coordinates $\theta_1,\ldots,\theta_{N-2}$ ranging over $[0,\pi]$. Denoting by $\Gamma_s = [0,\pi]^{N-2}\times[0,2\pi]$, the relation between the hyper-spherical coordinates $(r,\theta_1,\ldots,\theta_{N-1})\in[0,\infty)\times\Gamma_s$ and the Cartesian coordinates $\boldsymbol{y}=(y_1,\ldots,y_N)\in\mathbb{R}^N$ is given by

$$\begin{cases} y_1 = y_{0,1} + r\cos(\theta_1) \\ y_2 = y_{0,2} + r\sin(\theta_1)\cos(\theta_2) \\ \quad\vdots \\ y_N = y_{0,N} + r\sin(\theta_1)\cdots\sin(\theta_{N-2})\sin(\theta_{N-1}), \end{cases} \tag{4.1}$$

where $\boldsymbol{y}_0=(y_{0,1},\ldots,y_{0,N})$ denotes the origin of the hyper-spherical coordinate system. Based on this transformation, we would like to transform the discontinuity hyper-surface γ defined by the *implicit* equation $G(\boldsymbol{y})=0$ into the hyper-spherical coordinate system, and represent it by an *explicit* function of $\boldsymbol{\theta}=(\theta_1,\ldots,\theta_{N-1})$. To this end, we make the following assumption on the geometry of the subdomain Γ_1 and the origin $\boldsymbol{y}_0$.

Assumption 4.1. *For the underlying domain* $\Gamma=\Gamma_1\cup\gamma\cup\Gamma_2$ *in* (2.1)*, we assume that* Γ_1 *is a star-convex domain in* $\mathbb{R}^N$ *and that a point* $\boldsymbol{y}_0$ *in* Γ_1 *is given such that, for all* $\boldsymbol{y}\in\Gamma_1$*, the line segment* $\{\boldsymbol{y}_0+t\boldsymbol{y}\,|\,t\in[0,1]\}$ *from* $\boldsymbol{y}_0$ *to* $\boldsymbol{y}$ *is in* Γ_1*.*

Remark 4.2. When Γ_1 is a convex domain, it is also star-convex and any point in Γ_1 can be used as the origin $\boldsymbol{y}_0$; the function given in Example 2.1 provides an example of this case. If $\boldsymbol{y}_0$ is not known a priori, it can be obtained by Monte Carlo sampling in Γ, as long as $f(\boldsymbol{y})$ has the characteristic property. In practice, $\boldsymbol{y}_0$ is sometimes available for the problem of interest. For instance, as discussed in Example 2.2, the interest of investigating the probability of an event usually results from the occurrence of such event in a physical experiment with a specific set of parameter values. In this case, these values can be used to define the origin $\boldsymbol{y}_0$. On the other hand, if Γ_1 is not convex, the set of points qualified to be used as $\boldsymbol{y}_0$ is only a subset of Γ_1. In this case, especially when the target function has no characteristic property, it is much more difficult to choose a qualified $\boldsymbol{y}_0$.

Based on the transformation (4.1) with origin $\boldsymbol{y}_0$ satisfying Assumption 4.1, there exists a unique $N-1$ dimensional continuous function $g(\boldsymbol{\theta}):\Gamma_s\to[0,\infty)$ such that $\partial\Gamma_1=\{(g(\boldsymbol{\theta}),\boldsymbol{\theta})\,|\,\forall\,\boldsymbol{\theta}\in\Gamma_s\}$. The value of $g(\boldsymbol{\theta})$ is the Euclidean distance between $\boldsymbol{y}_0$ and $\partial\Gamma_1$ along the direction $\boldsymbol{\theta}$. Under the definitions in §2, given $\boldsymbol{\theta}\in\Gamma_s$, there are two possibilities for the location of $(g(\boldsymbol{\theta}),\boldsymbol{\theta})$, i.e., $(g(\boldsymbol{\theta}),\boldsymbol{\theta})\in\partial\Gamma\cap\partial\Gamma_1$ or $(g(\boldsymbol{\theta}),\boldsymbol{\theta})\in\gamma$. Thus, $g(\boldsymbol{\theta})$ is the desired function representation of the discontinuity hyper-surface γ. Unlike the equation $G(\boldsymbol{y})=0$, $g(\boldsymbol{\theta})$ is an *explicit* representation of γ, so that it becomes feasible to estimate γ directly by approximating $g(\boldsymbol{\theta})$ in Γ_s. However, Assumption 4.1 only guarantees the existence of $g(\boldsymbol{\theta})$ and the value of $g(\boldsymbol{\theta})$ at $\boldsymbol{\theta}\in\Gamma_s$ is unknown a priori. Therefore, a strategy of evaluating $g(\boldsymbol{\theta})$ is provided in §4.2.

4.2. Evaluation of the function representation of γ. Before moving forward, for clarity, in Table 4.1 we list and explain the notations used in the sequel. Essentially, the evaluation of $g(\boldsymbol{\theta})$ becomes a discontinuity detection problem for the one-dimensional function $f_{\boldsymbol{\theta}}(r)$ in the interval $[S_r(\boldsymbol{y}_0),S_r(\boldsymbol{\beta}_{\boldsymbol{\theta}})]$. If $(g(\boldsymbol{\theta}),\boldsymbol{\theta})\in\partial\Gamma\cap\partial\Gamma_1$, then $f_{\boldsymbol{\theta}}(r)$ is a continuous function on the line segment $\{\boldsymbol{y}_0+t\boldsymbol{\beta}_{\boldsymbol{\theta}}\,|\,t\in[0,1]\}$, such that $g(\boldsymbol{\theta})=|\boldsymbol{\beta}_{\boldsymbol{\theta}}-\boldsymbol{y}_0|$; if $(g(\boldsymbol{\theta}),\boldsymbol{\theta})\in\gamma=\partial\Gamma_1\backslash\partial\Gamma$, then $f_{\boldsymbol{\theta}}(r)$ is discontinuous at $r=g(\boldsymbol{\theta})$

Table 4.1: Definition of notations

Notation	Explanation
$g(\boldsymbol{\theta})$	the function representation of $\partial\Gamma_1$ in $\Gamma_s = [0,\pi]^{N-2} \times [0, 2\pi]$
$\widetilde{g}(\boldsymbol{\theta})$	the approximation of $g(\boldsymbol{\theta})$
$S(\boldsymbol{y})$	the transformation from Cartesian coordinates $\boldsymbol{y}$ to hyper-spherical coordinates $(r, \boldsymbol{\theta})$
$S^{-1}(\boldsymbol{y})$	the inverse transformation of $S(\boldsymbol{y})$
$S_r(\boldsymbol{y})$	the transformation from $\boldsymbol{y}$ to the radial coordinate r
$S_\theta(\boldsymbol{y})$	the transformation from $\boldsymbol{y}$ to the angular coordinates $\boldsymbol{\theta} = (\theta_1, \dots, \theta_{N-1})$
$\boldsymbol{\beta_\theta}$	the Cartesian coordinates $(\beta_{\boldsymbol{\theta},1}, \dots, \beta_{\boldsymbol{\theta},N})$ of the intersection point of $\partial\Gamma \cap \partial\Gamma_1$ and the ray from $\boldsymbol{y}_0$ along the direction $\boldsymbol{\theta}$
$f_{\boldsymbol{\theta}}(r)$	the target function f restricted to the ray along the direction $\boldsymbol{\theta}$, i.e.,$f_{\boldsymbol{\theta}}(r) = f(S^{-1}(r, \boldsymbol{\theta}))$

so that $g(\boldsymbol{\theta})$ can be estimated by capturing the discontinuity of $f_{\boldsymbol{\theta}}(r)$. We discuss the evaluation of $f(\boldsymbol{y})$ in the absence of the characteristic property (under Assumption $\boldsymbol{A_1}$) in §4.2.1 and with the characteristic property (under Assumption $\boldsymbol{A_2}$ or $\boldsymbol{A_3}$) in §4.2.2.

4.2.1. $f(\boldsymbol{y})$ without the characteristic property. Under Assumption $\boldsymbol{A_1}$ given in §2, we cannot distinguish whether $f_{\boldsymbol{\theta}}(r)$ is continuous or discontinuous in $[S_r(\boldsymbol{y}_0), S_r(\boldsymbol{\beta_\theta})]$, so that one needs to first approximate the whole profile of $f_{\boldsymbol{\theta}}(r)$, then identify the existence and location of the discontinuity by analyzing the approximation. For each $\boldsymbol{\theta}$, we use the one-dimensional ASGH approach to construct an adaptive approximation of $f_{\boldsymbol{\theta}}(r)$ in the interval $[S_r(\boldsymbol{y}_0), S_r(\boldsymbol{\beta_\theta})]$. The adaptivity will automatically refine in the region where $f_{\boldsymbol{\theta}}(r)$ has large variations, including jump discountinuities. To find a value $\widetilde{g}(\boldsymbol{\theta})$ such that $|\widetilde{g}(\boldsymbol{\theta}) - g(\boldsymbol{\theta})| \leq \tau$, an adaptive interpolant is constructed by setting $\eta = f_{\boldsymbol{\theta}}$ in (3.6), with the maximum level of the adaptive grid being $L_{\max} = \lceil \log_2(|\boldsymbol{y}_0 - \boldsymbol{\beta_\theta}|/\tau) \rceil$. Note that the hierarchical surplus decays to zero as the level L increases in the smooth region of $f_{\boldsymbol{\theta}}(r)$, but not near the jump discontinuity. Thus, if the mesh refinement stops automatically at a level $L < L_{\max}$, it means that $f_{\boldsymbol{\theta}}(r)$ is continuous in $[S_r(\boldsymbol{y}_0), S_r(\boldsymbol{\beta_\theta})]$ so that $g(\boldsymbol{\theta}) = |\boldsymbol{\beta_\theta} - \boldsymbol{y}_0|$ and $(g(\boldsymbol{\theta}), \boldsymbol{\theta}) \in \partial\Gamma \cup \partial\Gamma_1$. Otherwise, due to Assumption 4.1, $f_{\boldsymbol{\theta}}(r)$ has only one jump discontinuity at $(g(\boldsymbol{\theta}), \boldsymbol{\theta})$ and thus $\widetilde{g}(\boldsymbol{\theta})$ can be determined by

$$(r_1, r_2) = \underset{r, r' \in \mathcal{H}_{L_{\max},\alpha}([S_r(\boldsymbol{y}_0), S_r(\boldsymbol{\beta_\theta})])}{\arg\max} \frac{|f_{\boldsymbol{\theta}}(r) - f_{\boldsymbol{\theta}}(r')|}{|r - r'|} \quad \text{and} \quad \widetilde{g}(\boldsymbol{\theta}) = \frac{1}{2}(r_1 + r_2),$$

where $[r_1, r_2]$ is the interval which contains the largest variation of $f_{\boldsymbol{\theta}}(r)$ based on the available samples in $\mathcal{H}_{L_{\max},\alpha}([S_r(\boldsymbol{y}_0), S_r(\boldsymbol{\beta_\theta})])$. If τ is sufficiently small, then we have $g(\boldsymbol{\theta}) \in [r_1, r_2]$ with $|r_1 - r_2| < \tau$.

4.2.2. $f(\boldsymbol{y})$ with characteristic property. Under Assumptions $\boldsymbol{A_2}$ or $\boldsymbol{A_3}$, relying on the characteristic property, root-finding approaches can be employed to improve the efficiency of searching. Specifically, to evaluate $g(\boldsymbol{\theta})$ at $\boldsymbol{\theta} \in \Gamma_s$, we first evaluate $f(\boldsymbol{y})$ at $\boldsymbol{y} = \boldsymbol{\beta_\theta}$. If $f(\boldsymbol{\beta_\theta}) = f_1(\boldsymbol{\beta_\theta})$, then$(g(\boldsymbol{\theta}), \boldsymbol{\theta}) \in \partial\Gamma \cap \partial\Gamma_1$ and $|\boldsymbol{\beta_\theta} - \boldsymbol{y}_0|$ is the exact value of $g(\boldsymbol{\theta})$. Otherwise, we have $(g(\boldsymbol{\theta}), \boldsymbol{\theta}) \in \gamma \subset \partial\Gamma_1$ and $g(\boldsymbol{\theta})$ is the location of the jump discontinuity of the function $f_{\boldsymbol{\theta}}(r)$ which can be represented by

$$f_{\boldsymbol{\theta}}(r) = \begin{cases} f_1\left(S^{-1}(r,\boldsymbol{\theta})\right) & \text{if } r \le g(\boldsymbol{\theta}) \\ f_2\left(S^{-1}(r,\boldsymbol{\theta})\right) & \text{if } r > g(\boldsymbol{\theta}), \end{cases}$$

where $S^{-1}(r,\boldsymbol{\theta}) \in \Gamma$. The simplest root-finding approach is the bisection method starting with $r_{-1} = 0$ and $r_0 = |\boldsymbol{\beta}_{\boldsymbol{\theta}} - \boldsymbol{y}_0|$. In the k-th iteration, we have $r_k = (r_{k-1}+r_{k-2})/2$, where $f_{\boldsymbol{\theta}}(r_{k-1}) = f_1(S^{-1}(r_{k-1},\boldsymbol{\theta}))$ and $f_{\boldsymbol{\theta}}(r_{k-2}) = f_2(S^{-1}(r_{k-2},\boldsymbol{\theta}))$. For a prescribed accuracy τ such that $|\widetilde{g}(\boldsymbol{\theta}) - g(\boldsymbol{\theta})| \le \tau$, the necessary number of iterations K is given by

$$K = \left\lceil \log_2\left(|\boldsymbol{y}_0 - \boldsymbol{\beta}_{\boldsymbol{\theta}}|/\tau\right)\right\rceil, \tag{4.2}$$

and the final approximation is defined by $\widetilde{g}(\boldsymbol{\theta}) = (r_K + r_{K-1})/2$. Note that the bisection method is preferable when $f(\boldsymbol{y})$ has the characterization property because it only adds one neighboring point at each iteration whereas the AHSG method adds two neighboring points at a time.

In case $f(\boldsymbol{y})$ satisfies Assumption $\boldsymbol{A_3}$, i.e., $G(\boldsymbol{y})$ can be evaluated as a black-box function, other root-finding methods with faster convergence rates can be used to improve the efficiency of searching. For instance, $G(\boldsymbol{y}) = F(u(\boldsymbol{y})) - \overline{u}$ in Example 2.2 can be evaluated such that the discontinuity of $f_{\boldsymbol{\theta}}(r)$ can also be detected by searching the root of $G(S^{-1}(r,\boldsymbol{\theta})) = F(u(S^{-1}(r,\boldsymbol{\theta}))) - \overline{u} = 0$. In this work, relying on the smoothness of $G(\boldsymbol{y})$, we use the Regula Falsi method [**?**], a variant of the secant method. As is the case of using the bisection method, we start with $r_{-1} = 0$ and $r_0 = |\boldsymbol{\beta}_{\boldsymbol{\theta}} - \boldsymbol{y}_0|$; in the $k+1$-th iteration, r_{k+1} is defined by

$$r_{k+1} = r_k - G(S^{-1}(r_k,\boldsymbol{\theta})) \cdot \frac{r_k - r_{k'}}{G(S^{-1}(r_k,\boldsymbol{\theta})) - G(S^{-1}(r_{k'},\boldsymbol{\theta}))}, \tag{4.3}$$

where k' is the maximum index less than k such that $G(S^{-1}(r_k,\boldsymbol{\theta}))\cdot G(S^{-1}(r_{k'},\boldsymbol{\theta})) < 0$. It is known that the Regula Falsi method converges slower than the secant method but the iterates generated by (4.3) are all contained within the initial interval $[S_r(\boldsymbol{y}_0), S_r(\boldsymbol{\beta}_{\boldsymbol{\theta}})]$. Thus, one does not need to worry about the issue of getting negative r_k from (4.3). When $|r_k - r_{k'}|$ becomes sufficiently small, one can switch to the secant method to obtain faster convergence.

4.3. The hyper-spherical adaptive hierarchical sparse-grid algorithm. We now describe a complete procedure for the HS-AHSG method. Under Assumption 4.1, we would like to build an adaptive sparse-grid interpolant of $g(\boldsymbol{\theta})$ representing γ in the $N-1$ dimensional domain Γ_s. At each grid point $\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}$, $g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})$ is estimated by $\widetilde{g}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})$ using the approaches discussed in §4.2. Thus, we actually construct an interpolant of the approximation $\widetilde{g}(\boldsymbol{\theta})$. For fixed $L_{\min} > 0$ and $\alpha > 0$, the adaptive sparse-grid interpolant at level L is defined by setting $\eta(\boldsymbol{\theta}) = \widetilde{g}(\boldsymbol{\theta})$ in (3.6), i.e.,

$$\widetilde{g}_{L,\alpha}(\boldsymbol{\theta}) = \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \sum_{\mathbf{i}\in B_{\mathbf{l}}^{\alpha}} \widetilde{\omega}_{\mathbf{l},\mathbf{l}} \cdot \psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{\theta}), \tag{4.4}$$

where the surpluses $\{\widetilde{\omega}_{\mathbf{l},\mathbf{i}} \mid |\mathbf{l}| \le L, \mathbf{i} \in B_{\mathbf{l}}^{\alpha}\}$ are computed based on the set of approximate function values $\left\{\widetilde{g}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}) \;\middle|\; \boldsymbol{\theta}_{\mathbf{l},\mathbf{i}} \in \mathcal{H}_{L,\alpha}(\Gamma_s)\right\}$. Recall that if $(g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}), \boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})$ is on the boundary $\partial\Gamma \cap \partial\Gamma_1$, $\widetilde{g}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}) = g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})$ has no numerical error; otherwise, $\widetilde{g}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})$ is computed by either the AHSG method discussed in §4.2.1 or one of the root-finding methods discussed in §4.2.2. The approximated hyper-surface $\widetilde{\gamma}$ is given by

$$\widetilde{\gamma} = \left\{ (\widetilde{g}_{L,\alpha}(\boldsymbol{\theta}), \boldsymbol{\theta}) \;\middle|\; \boldsymbol{\theta} \in \Gamma_s \right\}.$$

Algorithm 1 is the main algorithm we use to construct our HS-AHSG approximation, where the bisection method is used under the Assumption $\boldsymbol{A_2}$.

Algorithm 1: *The hyper-spherical adaptive sparse-grid approximation*

```
Initialize $N$, $L_{\min}$, $\alpha$, $\tau$, $\boldsymbol{y}_0$
$l = -1$
while $l = -1$ or $\left\{\Delta\mathcal{H}_{L,\alpha}(\Gamma_s) \neq \emptyset \text{ and } l+1 \leq L_{\max}\right\}$ do
    Generate $\Delta\mathcal{H}_{l+1,\alpha}(\Gamma_s)$
    for $\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}} \in \Delta\mathcal{H}_{l+1,\alpha}(\Gamma_s)$ do
        Search $\boldsymbol{\beta}_{\theta_{\mathbf{l},\mathbf{i}}} = (\beta_{\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}},1}, \ldots, \beta_{\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}},N}) \in \Gamma$
        if $f(\boldsymbol{\beta}_{\theta_{\mathbf{l},\mathbf{i}}}) = f_1(\boldsymbol{\beta}_{\theta_{\mathbf{l},\mathbf{i}}})$ then
            $\widetilde{g}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}) = |\boldsymbol{y}_0 - \boldsymbol{\beta}_{\theta_{\mathbf{l},\mathbf{i}}}|$
        else
            Define $K = \left\lceil \log_2\left(|\boldsymbol{y}_0 - \boldsymbol{\beta}_{\theta_{\mathbf{l},\mathbf{i}}}|/\tau\right) \right\rceil$
            Run bisection $\widetilde{g}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}) = r_K$ where $|r_K - g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})| \leq |\boldsymbol{\beta}_{\theta_{\mathbf{l},\mathbf{i}}} - \boldsymbol{y}_0|/2^K$
        end if
        $\widetilde{\omega}_{\mathbf{l},\mathbf{i}} = \widetilde{g}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}) - \widetilde{g}_{L,\alpha}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})$
    end for
    Update to $\mathcal{H}_{l+1,\alpha}(\Gamma_s)$ by adding $\Delta\mathcal{H}_{l+1,\alpha}(\Gamma_s)$
    $l = l + 1$
end while
```

By building the approximation $\widetilde{g}_{L,\alpha}(\boldsymbol{\theta})$, we decompose a high-dimensional discontinuity detection problem to a set of *one-dimensional* discontinuity detection problems which are much easier to solve than the original problem. Due to the continuity of $g(\boldsymbol{\theta})$ resulting from the closeness of $\partial\Gamma_1$, the sparsity of the resulting sparse grid can be retained such that all the disadvantages of the traditional adaptive sparse-grid approach shown in Example 3.1 can be completely avoided. As mentioned in §2, since the cost of function evaluations is usually dominant, so that the total computational cost of constructing the HS-AHSG approximation $\widetilde{g}_{L,\alpha}(\boldsymbol{\theta})$ can be measured by

$$\mathcal{C}_{\text{total}} = \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \sum_{\mathbf{i}\in B_{\mathbf{l}}^{\alpha}} M_{\mathbf{l},\mathbf{i}}^{\tau}, \tag{4.5}$$

where $M_{\mathbf{l},\mathbf{i}}^{\tau}$ is the number of function evaluations for obtaining $\widetilde{g}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})$ with the accuracy τ. Note that $M_{\mathbf{l},\mathbf{i}}^{\tau} = 1$ in the sense that $f(\boldsymbol{y})$ has the characteristic property and $(g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}), \boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})$ is on the boundary of Γ. It is well known that the convergence of either the AHSG method or of root-finding methods heavily depends on the size of the search interval. So far, the search interval for each $\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}$ is set to $[S_r(\boldsymbol{y}_0), S_r(\boldsymbol{\beta}_{\boldsymbol{\theta}})]$ which is the largest possible interval, because we assume that no knowledge about the function value $g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})$ is known a priori. In the next section, the efficiency of constructing $\widetilde{g}_{L,\alpha}(\boldsymbol{\theta})$ is improved by taking advantage of its hierarchal structure.

4.4. Accelerated approximation using sparse-grid hierarchies. In (4.5), at each grid point $\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}$, $M_{\mathbf{l},\mathbf{i}}^{\tau}$ is determined by the prescribed accuracy τ and the initial search interval. So far, the initial search interval for each $\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}$ is set to $[S_r(\boldsymbol{y}_0), S_r(\boldsymbol{\beta}_{\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}})]$ because no knowledge about the value $g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})$ is known a priori. Such an assumption is true on level $L = 0$. However, when $L \geq 1$, by the definition of surplus, we have

$$\widetilde{g}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}) = \widetilde{g}_{L-1,\alpha}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}) + \widetilde{\omega}_{\mathbf{l},\mathbf{i}},$$

for each newly added point $\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}} \in \Delta\mathcal{H}_{L,\alpha}(\Gamma_s)$ on level L. As such, the HS-AHSG approximation of level $L-1$ can provide a prediction of $\widetilde{g}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})$ with the error being the unknown surplus. Based on the upper bound given in (3.5), such a prediction will become more and more accurate as L increases. Thus, for $L \geq 1$, we utilize the HS-AHSG approximation of the previous level to reduce the size of the initial search interval in order to accelerate the evaluation of $\widetilde{g}(\boldsymbol{\theta})$.

Assuming the target function $f(\boldsymbol{y})$ has the characteristic property, we give the algorithm for the accelerated bisection method in Algorithm 2 which can be extended to other approaches with relative ease. The basic idea behind Algorithm 2 is to set one of the endpoints, e.g., r_{-1}, of the initial search interval to the predicted value given by the interpolated value $\widetilde{g}_{L,\alpha}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})$ at the new added point $\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}$. Besides that, several practical issues in terms of efficiency and robustness are considered as well. First, one needs to properly define the other endpoint r_0 such that $|r_{-1} - r_0|$ will become smaller as the level L increases and the interval $[r_{-1}, r_0]$ can cover the discontinuity location $g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})$. Theoretically, r_0 can be chosen according to the upper bound of the error $|g(\boldsymbol{\theta}) - \widetilde{g}_{L,\alpha}(\boldsymbol{\theta})|$. However, since the a priori error bound is only known up to a constant, in the computations, we use the hierarchical surplus, which acts as an a posteriori error estimate, to choose the other endpoint r_0. Specifically, for the new added grid points on level L, we initially set the length $|r_{-1} - r_0|$ to the maximum magnitude, denoted by ξ, of all surpluses on level $L-1$. Note that such surpluses actually characterize the error of the interpolant on level $L-2$ which means ξ is not the optimal choice, but in most cases, it is big enough to cover the discontinuity and it also decays to zero as L increases. However, in order to avoid the scenario that both r_{-1} and r_0 are on the same side of the discontinuity, e.g., $r_{-1}, r_0 < g(\boldsymbol{\theta})$, we add two loops in Algorithm 2 to recursively enlarge the length $|r_{-1} - r_0|$ by ξ until the interval $[r_{-1}, r_0]$ covers the value $g(\boldsymbol{\theta})$.

Algorithm 2: *The accelerated bisection method to compute* $\widetilde{g}_{\mathbf{l},\mathbf{i}} \approx g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})$ *for* $\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}} \in \Delta\mathcal{H}_{L,\alpha}(\Gamma_s)$, *given* $\widetilde{g}_{L,\alpha}(\boldsymbol{\theta})$

```
$\xi = \max\Big\{|\omega_{\mathbf{l}'\mathbf{i}'}| \;\Big|\; \boldsymbol{\theta}_{\mathbf{l}'\mathbf{i}'} \in \mathcal{H}_{L-1,\alpha}(\Gamma_s)$ and $|\mathbf{l}'| = L-1\Big\}$
Search $\boldsymbol{\beta}_{\theta_{\mathbf{l},\mathbf{i}}} = (\beta_{\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}},1}, \ldots, \beta_{\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}},N}) \in \Gamma$
$r_{-1} = \min\left\{\max\left\{\widetilde{g}_{L-1,\alpha}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}), 0\right\}, |\boldsymbol{y}_0 - \boldsymbol{\beta}_{\theta_{\mathbf{l},\mathbf{i}}}|\right\}$
if $f(S^{-1}(r_{-1}, \boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})) = f_1(S^{-1}(r_{-1}, \boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}))$ then
    $r_0 = \min\{r_{-1} + \xi, |\boldsymbol{y}_0 - \boldsymbol{\beta}_{\theta_{\mathbf{l},\mathbf{i}}}|\}$
    while $f(S^{-1}(r_0, \boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})) \neq f_2(S^{-1}(r_0, \boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}))$ do
        $r_0 = \min\{r_0 + \xi, |\boldsymbol{y}_0 - \boldsymbol{\beta}_{\theta_{\mathbf{l},\mathbf{i}}}|\}$
    end while
else
    $r_0 = \max\{r_{-1} - \xi, 0\}$
    while $f(S^{-1}(r_0, \boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})) \neq f_1(S^{-1}(r_0, \boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}))$ do
        $r_0 = \max\{r_0 - \xi, 0\}$
    end while
end if
Define $K = \lceil \log_2 (|r_0 - r_{-1}|/\tau) \rceil$
Run bisection $\widetilde{g}_{\mathbf{l},\mathbf{i}} = r_K$ where $|r_K - g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})| \leq |r_0 - r_{-1}|/2^K$
```

4.5. ε-complexity analyses. We provide error estimates and ε-complexity analyses of the proposed HS-AHSG method for approximating the discontinuity hyper-surface γ, i.e., the function $g(\boldsymbol{\theta})$. For simplicity, we assume the target function $f(\boldsymbol{y})$ satisfies Assumption $\boldsymbol{A_2}$. The analyses are carried out in the context of the isotropic sparse-grid interpolation, given in (3.3), coupled with a bisection method. The underlying domain Γ is set to $(0,1)^N$. For the sake of notational convenience, we set $\widehat{N} = N-1$ in the following derivation, and use $\|\cdot\|$ to denote the $L^\infty(\Gamma_s)$ norm.

First, we observe that the total error $e = g(\boldsymbol{\theta}) - \widetilde{g}_L(\boldsymbol{\theta})$ can be decomposed as

$$e = g(\boldsymbol{\theta}) - \widetilde{g}_L(\boldsymbol{\theta}) = \underbrace{g(\boldsymbol{\theta}) - g_L(\boldsymbol{\theta})}_{e_1} + \underbrace{g_L(\boldsymbol{\theta}) - \widetilde{g}_L(\boldsymbol{\theta})}_{e_2}, \tag{4.6}$$

where $g_L(\boldsymbol{\theta})$ is the isotropic HS-AHSG approximation of the exact target function $g(\boldsymbol{\theta})$. Here, we study a specific type of discontinuity hyper-surfaces satisfying the following assumption.

Assumption 4.3. *Under the assumption 4.1, the transformed function $g(\boldsymbol{\theta})$ characterizing γ is in the space*

$$X^q(\Gamma_s) := \left\{ g : \Gamma_s \to [0, +\infty) \mid D^{\boldsymbol{\alpha}}(g) \in L^\infty(\Gamma_s), |\boldsymbol{\alpha}|_\infty \leq q \right\},$$

where $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_{\widehat{N}})$, $|\boldsymbol{\alpha}|_\infty = \max_{1\leq n\leq \widehat{N}} \alpha_n$ and $D^{\boldsymbol{\alpha}}(g) = \partial^{|\alpha|_1} g / \partial\theta_1^{\alpha_1} \cdots \partial\theta_{\widehat{N}}^{\alpha_{\widehat{N}}}$.

An estimate for e is given in the following lemma.

Proposition 4.4. *Under Assumption 4.1 and 4.3, if the transformed function $g(\boldsymbol{\theta})$ is in the space $X^2(\Gamma_s)$, then the error $e = e_1 + e_2$ in (4.6) can be bounded by*

$$\|e\| \leq C_{\text{sg}} 2^{-2L} \sum_{k=0}^{\widehat{N}-1} \binom{L+\widehat{N}-1}{k} + 2^{\widehat{N}} \binom{L+\widehat{N}}{\widehat{N}} \tau, \tag{4.7}$$

where τ is the tolerance for the bisection method, the constant C_{sg} is independent of the level L and the dimension $\widehat{N}$.

Proof. The proof is given in Appendix A.1 □

Now, we analyze the cost of constructing $\widetilde{g}_L(\boldsymbol{\theta})$ with the prescribed error $\varepsilon > 0$. According to the error estimate in Proposition 4.4, a sufficient condition of $\|e\| = \|g(\boldsymbol{\theta}) - \widetilde{g}_L(\boldsymbol{\theta})\| \leq \varepsilon$ is that

$$\|e_1\| \leq C_{\text{sg}} 2^{-2L} \sum_{k=0}^{\widehat{N}-1} \binom{L+\widehat{N}-1}{k} \leq \frac{\varepsilon}{2} \tag{4.8}$$

and

$$\|e_2\| \leq 2^{\widehat{N}} \binom{L+\widehat{N}}{\widehat{N}} \tau \leq \frac{\varepsilon}{2}. \tag{4.9}$$

Let $\mathcal{C}_{\min}$ denote the *minimum cost*, i.e., the minimum number of function evaluations, needed to satisfy (4.8) and (4.9). The goal is to determine an upper bound for $\mathcal{C}_{\min}$. Note that, for fixed dimension N and level L, the total cost $\mathcal{C}_{\text{total}}$ is determined by solving the inequality (4.9). The larger is L, the smaller is τ which means, when using the bisection method, a greater number of function evaluations are needed to achieve the accuracy τ. Therefore, the estimation of $\mathcal{C}_{\min}$ has two steps. Given N and ε, we first determine upper bounds for the minimum L needed to achieve (4.8); then, we substitute the obtained value into (4.9) to get an upper bound for $\mathcal{C}_{\min}$.

To perform the first step, we need to estimate the numbers of degrees of freedom of $\mathcal{H}_l$ and $\Delta\mathcal{H}_l$ for $l \leq L$, denoted by $|\mathcal{H}_L|$ and $|\Delta\mathcal{H}_l|$, respectively. The estimation of $|\mathcal{H}_L|$ has been studied in [4, 18], but the estimate in [18] is not sufficiently sharp and the estimate in [4] has no results related $|\Delta\mathcal{H}_l|$. In the following lemma, we provide estimates for $|\Delta\mathcal{H}_l|$ which directly leads to an estimate of $|\mathcal{H}_L|$.

Lemma 4.5. *The dimensions of the subspaces $\Delta\mathcal{H}_l$ and $\mathcal{H}_L$ for $\widehat{N} \geq 2$, i.e., the numbers of grid points in $\Delta\mathcal{H}_l(\Gamma_s)$ and $\mathcal{H}_L(\Gamma_s)$, respectively, are bounded by*

$$|\Delta\mathcal{H}_l| \leq 2^l \binom{l+\widehat{N}-1}{\widehat{N}-1} \leq 2^l \left(\frac{l+\widehat{N}-1}{\widehat{N}-1}\right)^{\widehat{N}-1} \mathrm{e}^{\widehat{N}-1}$$

for $0 \leq l \leq L$ and, correspondingly,

$$|\mathcal{H}_L| \leq 2^{L+1} \binom{L+\widehat{N}-1}{\widehat{N}-1} \leq 2^{L+1} \left(\frac{L+\widehat{N}-1}{\widehat{N}-1}\right)^{\widehat{N}-1} \mathrm{e}^{\widehat{N}-1}.$$

Proof. The proof is given in Appendix A.2 □

Similar to the analyses in [23], we solve the inequality (4.8) to obtain an upper bound for L such that the error of the isotropic sparse-grid interpolant $g_L(\boldsymbol{\theta})$ is smaller than the prescribed accuracy $\frac{\varepsilon}{2}$.

Lemma 4.6. *For $\varepsilon < 2C_{\mathrm{sg}}$ in (4.8), the accuracy $\|e_1\| \leq \frac{\varepsilon}{2}$ can be achieved with a minimum level L such that*

$$L \leq \lceil L_k \rceil = \left\lceil \frac{t_k \widehat{N}}{2\ln 2} \right\rceil \quad \textit{with} \quad h = \frac{2e}{\ln 2}\left(\frac{2C_{\mathrm{sg}}}{\varepsilon}\right)^{\frac{1}{\widehat{N}}},$$

where $\{t_k\}_{k=0}^{\infty}$ is a monotonically decreasing sequence defined by

$$t_k = \ln(t_{k-1}h) \quad \textit{with} \quad t_0 = \frac{\mathrm{e}}{\mathrm{e}-1}\ln h.$$

Proof. The proof is given in Appendix A.3 □

Corollary 4.7. *Under Lemma 4.6, for $k \in \mathbb{N}$, we have*

$$\binom{L_k+\widehat{N}}{\widehat{N}} \leq \frac{\varepsilon}{2\widehat{N}C_{\mathrm{sg}}} \cdot 2^{2L_k}. \tag{4.10}$$

It is an immediate result by substituting (A.4) into (A.3). Now we derive an upper bound for $\mathcal{C}_{\min}$ in the context of the isotropic HS-AHSG method using linear basis without acceleration

Theorem 4.8. *Under Lemma 4.5 and 4.6, the minimum total cost $\mathcal{C}_{\min}$ for building the isotropic sparse-grid approximation to $g(\boldsymbol{\theta})$ with accuracy $\|g(\boldsymbol{\theta}) - \widetilde{g}_L(\boldsymbol{\theta})\| \leq \varepsilon$ based on Algorithm 1 satisfies the estimate*

$$\mathcal{C}_{\min} \leq \frac{\alpha_1}{\widehat{N}} \left\{ \alpha_2 + \alpha_3 \frac{\log_2\left(\frac{2C_{\mathrm{sg}}}{\varepsilon}\right)}{\widehat{N}} \right\}^{\alpha_4 \widehat{N}} \left(\frac{2C_{\mathrm{sg}}}{\varepsilon}\right)^{\alpha_5} \left\{ \alpha_6 \log_2\left(\frac{2C_{\mathrm{sg}}}{\varepsilon}\right) + \alpha_7 \widehat{N} + \alpha_8 \right\},$$

where C_{sg} is defined as in (4.8) and the constants $\alpha_1, \cdots, \alpha_8$ are defined by

$$\begin{aligned} &\alpha_1 = 2, \quad \alpha_2 = \frac{2\mathrm{e}^2}{(\mathrm{e}-1)} \log_2\left(\frac{2\mathrm{e}}{\ln 2}\right), \quad \alpha_3 = \frac{2\mathrm{e}^2}{(\mathrm{e}-1)}, \quad \alpha_4 = \frac{3}{2}, \\ &\alpha_5 = \frac{1}{2}, \quad \alpha_6 = \frac{\mathrm{e}}{\mathrm{e}-1}, \quad \alpha_7 = \frac{\mathrm{e}}{\mathrm{e}-1} \log_2\left(\frac{2\mathrm{e}}{\ln 2}\right) + 1, \quad \alpha_8 = 2 - \log_2(C_{\mathrm{sg}}). \end{aligned} \tag{4.11}$$

Proof. The proof is given in Appendix A.4 □

Next, we analyze the computational cost of the accelerated Algorithm 1 by exploiting Algorithm 2. Unlike the unaccelerated Algorithm 1 for which the length τ_0 of the initial search interval is set to be of the same scale as the domain Γ, in Algorithm 2, for each new added sparse grid point $\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}$ with $L = |\mathbf{l}| \geq 1$, the desired function value $g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})$ is firstly predicted by the level $L-1$ HS-AHSG interpolant $\widetilde{g}_{L-1}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})$, and then this prediction is used as one endpoint of the initial search interval in the bisection simulation, i.e., $r_{-1} = \widetilde{g}_{L-1}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})$. For simplicity, the other endpoint is defined by the upper bound of the error of the prediction, i.e., $|g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}) - \widetilde{g}_{L-1}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})|$. In this case, the interval $[r_{-1}, r_0]$ will include the exact function value $g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})$. This is slightly different from the strategy used in Algorithm 2 in which the local error indicator, i.e., the surplus, is used because the upper bound of $|g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}) - \widetilde{g}_{L-1}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})|$ is only known up to a constant. In the following derivation, the error bound given in (4.7) is still valid, but, at sparse grid points $\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}$ for $|\mathbf{l}| = L$, we can obtain a sharper bound for the error of $\widetilde{g}_{L-1}(\boldsymbol{\theta})$. The result is provided in the following lemma.

Lemma 4.9. *If the transformed function $g(\boldsymbol{\theta})$ is in the space $X^2(\Gamma_s)$, then, at each sparse grid point $\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}$ with $L = |\mathbf{l}| \geq 1$ and $\mathbf{i} \in B_{\mathbf{l}}$ defined in* (3.4)*, the error $g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}) - \widetilde{g}_{L-1}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})$ satisfies the estimate*

$$\left| g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}) - \widetilde{g}_{L-1}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}) \right| \leq C_{\text{surp}} 2^{-2L} + 2^{\widehat{N}} \tau,$$

where C_{surp} is independent of L and τ is the tolerance of the bisection algorithm.

Proof. The proof is given in Appendix A.5 □

Finally, the upper bound of $\mathcal{C}_{\min}$ in the context of using the HS-AHSG method with acceleration is in the following theorem.

Theorem 4.10. *Under Lemma 4.5, 4.6, and 4.9, the minimum total cost $\mathcal{C}_{\min}$ incurred of building the isotropic sparse-grid approximation to $g(\boldsymbol{\theta})$ with accuracy $\|g(\boldsymbol{\theta}) - \widetilde{g}_L(\boldsymbol{\theta})\| \leq \varepsilon$ using the accelerated HS-AHSG method satisfies the estimate*

$$\mathcal{C}_{\min} \leq \alpha_1 \left[\alpha_2 + \alpha_3 \frac{\log_2\left(\frac{2C_{\text{sg}}}{\varepsilon}\right)}{\widehat{N}} \right]^{\alpha_4 \widehat{N}} \left(\frac{2C_{\text{sg}}}{\varepsilon} \right)^{\alpha_5} \left[2\widehat{N} - \log_2(\widehat{N}) + \alpha_9 \right],$$

where C_{sg} is the constant in (4.8)*, the constants $\alpha_1, \cdots, \alpha_5$ are defined as in Theorem 4.8 and α_9 is defined by*

$$\alpha_9 = \log_2\left(\frac{C_{\text{surp}}}{C_{\text{sg}}} \right) + 2.$$

Proof. The proof is given in Appendix A.6 □

Remark 4.11. Theorem 4.8 and 4.10 tell us that the total cost of the HS-AHSG method is mainly determined by the number of sparse-grid points. Asymptotically, the growth rate of $|\mathcal{H}_L|$ is characterized by the constants α_4 and α_5, and the cost due to inaccurate initial searching interval is of order $\log_2(1/\varepsilon)$. According to the analyses in [4, 23], the growth rate can be reduced when using high-order hierarchical polynomial bases. In general, with a p-th order hierarchical basis, we have $\alpha_4 = (p+2)/(p+1)$ and $\alpha_5 = 1/(p+1)$. Note that the use of acceleration technique with accurate initial guesses will reduce the total cost by a factor of $\log_2(1/\varepsilon)$ asymptotically, which will be demonstrated in the following section.

5. Numerical examples.

In this section, we use three discontinuity detection problems to illustrate the performance of the proposed method. The first example is used to test the HS-AHSG method in approximating discontinuities of functions with the characteristic property. In the second example, a generic discontinuous function without the characteristic property is considered and the importance of the choice of the origin $\boldsymbol{y}_0$ is demonstrated. The third example is an application of the HS-AHSG method in computing the probability of an event that depends on the solution of a partial differential equation with random inputs.

Example 5.1. Consider the two characteristic functions in $\mathbb{R}^N$

$$F_1(\boldsymbol{y}) = \begin{cases} 1 & \text{if } \displaystyle\sum_{n=1}^{N} y_n^2 \geq 1 \\ 0 & \text{otherwise} \end{cases} \tag{5.1}$$

$$F_2(\boldsymbol{y}) = \begin{cases} 1 & \text{if } |y_3 - y_1| \leq 0.5 \text{ for } \boldsymbol{y} \in [0,1]^N \\ 0 & \text{otherwise} \end{cases} \tag{5.2}$$

where the characteristic domains of $F_1(\boldsymbol{y})$ and $F_2(\boldsymbol{y})$ are a unit hyper-sphere and a layer in the unit hyper-cube, respectively. The linear hierarchal basis is used for building the HS-AHSG approximation and the bisection method is used to estimate the value of transformed function $g(\boldsymbol{\theta})$ at the sparse-grid points.

First, to illustrate the distribution of the sparse grid points generated by the HS-AHSG method, we set $N = 3$ and plot, in Figures 5.1 and 5.2, the discontinuity surface γ, the surface of $g(\boldsymbol{\theta})$, and the sparse-grid points for$F_1(\boldsymbol{y})$ and $F_2(\boldsymbol{y})$, respectively. By comparing Figure 3.2(b) and Figure 5.1, we can see the advantage of the HS-AHSG method. The resulting sparse grid contains only 160 points to achieve the desired accuracy whereas the classic AHSG method requires 36,093 grid points. Instead of directly approximating the discontinuous function $F_1(\boldsymbol{y})$, we approximate the transformed surface shown in Figure 5.1(b), where its smoothness retains the sparsity of the resulting grid. In Figure 5.2, we can see that the surface γ is only a part of the boundary $\partial\Gamma_1$ but $\overline{\Gamma}_1$ is a closed subdomain in $\overline{\Gamma}$. There are a total of 1120 sparse grid points on $\partial\Gamma_1$ with only 349 points on γ and 771 points on $\partial\Gamma_1$. According to the discussion in 4.2, at the sparse-grid points placed on $\partial\Gamma_1\backslash\gamma$, there is no need to run the bisection algorithm to evaluate $g(\boldsymbol{\theta})$ at the 771 points on $\partial\Gamma_1$ so that a significant amount of computational effort is saved. In Figures 5.2, we observe that $g(\boldsymbol{\theta})$ is not differentiable at the edges and vertices of the characteristic domain Γ_1 so that the HS-AHSG approximation does mesh refinement around these regions. Although the lack of a derivative is not as bad as a jump discontinuity, it may result in a failure of the HS-AHSG method if the volume of such a singularity grows fast as the dimension N increases. This issue will be considered in future work.

Next, we test the convergence of the HS-AHSG method in detecting the discontinuity of $F_1(\boldsymbol{y})$ in three cases: the HS-ASHG method with isotropic sparse grids and no acceleration, the HS-AHSG method with isotropic sparse grids and acceleration, and the HS-AHSG method with adaptive sparse grids and acceleration. The origin of the hyper-spherical coordinate system is set to $\boldsymbol{y}_0 = (0.2/\sqrt{N}, \ldots, 0.2/\sqrt{N})$ which is 0.2 away from the origin $(0, \ldots, 0)$. In Table 5.1, we list the computational costs of the three tested cases for $N = 2, 3, 4, 5, 7, 9$ where the interpolation error is tested is measured by the metric e_∞ in (2.4). As expected, for high-dimensional discontinuity detection, our approach achieves faster convergence rates than the well known AHSG method. Moreover, for the same accuracy, acceleration and adaptivity can provide a significant saving in computational cost.

Example 5.2. Consider the two-dimensional discontinuous function on $[-1.5, 1.5] \times [-1.5, 1.5]$ given by

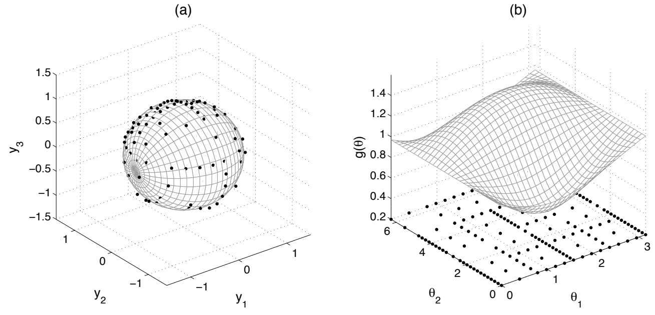

Fig. 5.1: (a) The discontinuity surface γ with sparse-grid points; (b) the transformed surface $g(\boldsymbol{\theta})$ in the hyper-spherical coordinate system. The parameters for the HS-AHSG approximation are $L_{\min} = 4$, $L_{\max} = 12$, $\alpha = 0.01$, and $\boldsymbol{y}_0 = (0.1, 0.2, 0.3)$; the total number of sparse-grid points is 160.

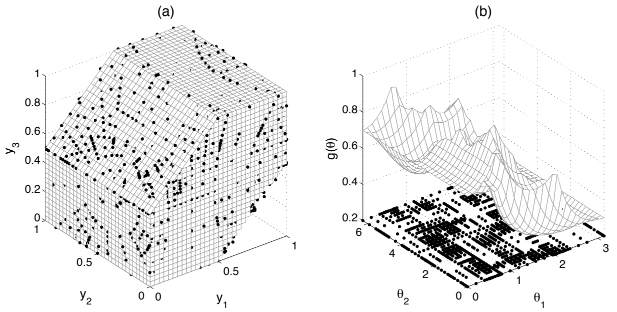

Fig. 5.2: (a) The discontinuity surface γ with sparse-grid points; (b) the transformed surface $g(\boldsymbol{\theta})$ in the hyper-spherical coordinate system. The parameters for the HS-AHSG approximation are $L_{\min} = 4$, $L_{\max} = 12$, $\alpha = 0.01$, and $\boldsymbol{y}_0 = (0.3, 0.4, 0.5)$; the total number of sparse-grid points is 1120 of which only 349 are off the boundary.

$$f(\boldsymbol{y}) = \begin{cases} y_1^2 + y_2^2 & \text{if } \sqrt{y_1^2 + y_2^2} \leq 1 + \dfrac{1}{4}\cos\left(4\arctan\left(\dfrac{y_2}{y_1}\right)\right) \\[2ex] y_1^2 + y_2^2 + \dfrac{1}{2} & \text{otherwise} \end{cases} \tag{5.3}$$

which is plotted in Figure 5.3(a), where the jump discontinuity is along the curve $\sqrt{y_1^2 + y_2^2} = 1 + \frac{1}{4}\cos\left(4 \cdot \arctan\left(y_2/y_1\right)\right)$. The domain Γ_1 is defined as the interior of this curve. Note that Γ_1 is star-convex but not convex so that only a subset of points in Γ_1 can be used as the origin of the hyper-spherical coordinate system. We test our approach with the origin being $(0, 0)$ and $(-1.1, 0)$ and the bisection method is used to approximate the transformed function $g(\boldsymbol{\theta})$. The captured discontinuity curves are plotted in Figure 5.3(b). It is easy to see that the point $(0, 0)$ is a qualified point to be the origin of the polar system so that the approximate curve captures the discontinuity very well. In contrast, $(-1.1, 0)$ does not satisfy the Assumption 4.1 so that the function $f_{\boldsymbol{\theta}}(r)$ has multiple roots along some directions wheras the bisection algorithm can only find one root. Thus, the discontinuity curve is not captured correctly in a subdomain of Γ_s, as shown in Figure 5.3(b).

Table 5.1: Computational costs (total number of function evaluations) and savings of the HS-AHSG method with acceleration and adaptivity in Example 5.1

Dim	Error	IsoSG	IsoSG+acceleration		AHSG+acceleration	
		cost	cost	saving	cost	saving
2D	1.0e-3	384	274	28.7%	251	34.7%
	1.0e-4	1,603	955	40.4%	832	48.1%
	1.0e-5	7,230	2,968	58.9%	2,737	62.1%
3D	1.0e-3	7,046	4,461	36.7%	3,022	57.1%
	1.0e-4	42,541	18,021	57.6%	12,817	69.9%
	1.0e-5	224,978	67,721	69.9%	54,439	75.8%
4D	5.0e-2	880	682	22.5%	584	33.6%
	1.0e-3	66,207	38,165	42.4%	26,329	60.2%
	1.0e-4	542,632	241,337	55.5%	161,354	70.3%
5D	5.0e-2	5,135	3,645	29.0%	2,082	59.5%
	1.0e-2	23,782	16,558	30.4%	14,694	38.2%
	1.0e-3	383,884	207,862	45.9%	94,148	75.5%
7D	1.0e-1	24,757	11,770	52.5%	6,327	74.4%
	5.0e-2	67,671	40,221	40.6%	25,111	62.9%
	5.0e-3	773,113	479,984	37.9%	354,040	54.2%
9D	1.0e-1	26,593	14,843	44.2%	6,426	75.8%
	5.0e-2	157,851	80,507	49.0%	58,849	62.7%
	1.0e-2	1,472,441	983,101	33.2%	513,163	65.1%

In addition, if we assume the function $f(\boldsymbol{y})$ has no characteristic property, the one-dimensional AHSG approach discussed in §4.2.1 has to be used to estimate the value of the transformed function $g(\boldsymbol{\theta})$. For comparison, the decays of the interpolation error measured by the metric e_∞ in (2.4) are plotted in Figure 5.3(c). When setting the origin $\boldsymbol{y}_0 = (-1.1, 0)$, the error does not decay toward zero because of the violation of Assumption 4.1. When $\boldsymbol{y}_0 = (0, 0)$, we can see that the use of the bisection method based on the characteristic property and the hierarchical acceleration can significantly reduce the number of function evaluations to achieve the prescribed accuracy.

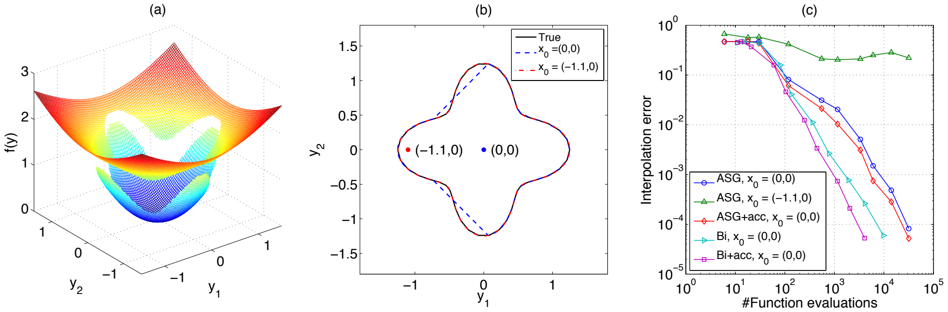

Fig. 5.3: (a) The surface of the target function $f(\boldsymbol{y})$ in Example 5.2; (b) the true discontinuity curve and the approximated curves with the origin of the polar coordinate system being $(0, 0)$ and (-1.1, 0); (c) the decay of the interpolation error measured by the metric in (2.4).

Example 5.3. Consider a two-dimensional steady heat equation with stochastic diffusivity

$$\begin{cases} -\nabla\cdot(\kappa(x,\omega)\nabla u(x,\omega)) = h(x) & \text{in } [0,1]^2\times\Omega, \\ u(x,\omega) = 0 & \text{on } \partial D\times\Omega \end{cases} \tag{5.4}$$

with $h(x) = 2000 + \exp\left(-\frac{(x_1-0.6)^2+(x_2-0.8)^2}{0.05^2}\right)$ and

$$\begin{aligned} \kappa(x,\omega) = y_1(\omega)^2 + \exp\big[&y_2(\omega)^4\sin(\pi x_1) + y_3(\omega)^2\sin(\pi x_2) \\ &+ \cos(\pi x_1) + \cos(\pi x_2)\big], \end{aligned} \tag{5.5}$$

where $\boldsymbol{y}(\omega) = (y_1(\omega),\, y_2(\omega),\, y_3(\omega))$ are i.i.d. random variables following the uniform distribution $U([-1,1])$. The quantity of interest is the probability of the event that the integral of the solution $u(x,\omega)$ over D is larger than the threshold value 1.2, i.e.,

$$\text{QoI} = \int_\Omega \mathcal{X}_{\Gamma_1}(\boldsymbol{y})\rho(\boldsymbol{y})d\boldsymbol{y} \;\text{ where }\; \Gamma_1 = \left\{\boldsymbol{y}\in\mathbb{R}^3 \;\middle|\; \int_D u(x,\boldsymbol{y})dx > 1.2\right\}. \tag{5.6}$$

where Γ_1 is the domain of interest described by the characteristic function $\mathcal{X}_{\Gamma_1}(\boldsymbol{y})$. Note that this example satisfies the Assumption $\boldsymbol{A_3}$ given in §2 and the implicit function is defined by $G(\boldsymbol{y}) = 1.2 - \int_D u(x,\boldsymbol{y})dx = 0$ which is smooth due to the regularity of the solution u. In this case, we can use more advanced root-finding methods discussed in §4.2.2, such as the Regula Falsi method. Here we use the HS-AHSG method with both acceleration and adaptivity and only compare the performances of different root-finding methods. At each point $\boldsymbol{y}\in\Gamma$, the PDE in (5.4) is solved using the finite element (FE) method on a 50×50 mesh in the physical domain. Because we are focusing on improving the accuracy of capturing discontinuities in the stochastic domain, the error from the spatial discretization is ignored, and the FE projection is treated as the exact solution. An approximation to the exact value of the QoI in (5.6) is computed using Monte Carlo sampling. The origin $\boldsymbol{y}_0$ is set to $(0.01, 0.2, 0.05)$. The surface γ and its transformed representation are plotted in Figure 5.4(a) and 5.4(b), respectively. For $L_{\min} = 3$ and $\alpha = 0.01$, we end up with a total of 344 sparse grid points in the hyper-spherical domain Γ_s; those points are also plotted in Figure 5.4(a) and 5.4(b). In Figure 5.4(c), we show the error decay of the HS-AHSG approximations using the bisection and Regula Falsi methods, respectively, where the error is measured by the metric e_{int} in (2.5). We observe that the Regula Falsi method can provide additional savings in computational costs over the bisection method by taking advantage of the availability and smoothness of $G(\boldsymbol{y})$. Further evidence can been seen in Table 5.2.

Table 5.2: The computational cost (number of function evaluations) of the HS-AHSG method using the bisection method and the Regula Falsi method in Example 5.3

Error	Bisection	Regula Falsi	Saving
1.0e-3	4,116	3,381	17.8%
1.0e-4	17,464	13,047	25.3%
1.0e-5	68,299	48,555	28.9%

6. Conclusions. In this paper, we proposed a comprehensive methodology, for detecting an $N-1$ discontinuity hyper-surface of function defined in an N-dimensional bounded domain. This approach removes the continuity assumption required by adaptive sparse-grid methods, thus enabling high-dimensional approximations. Both the-

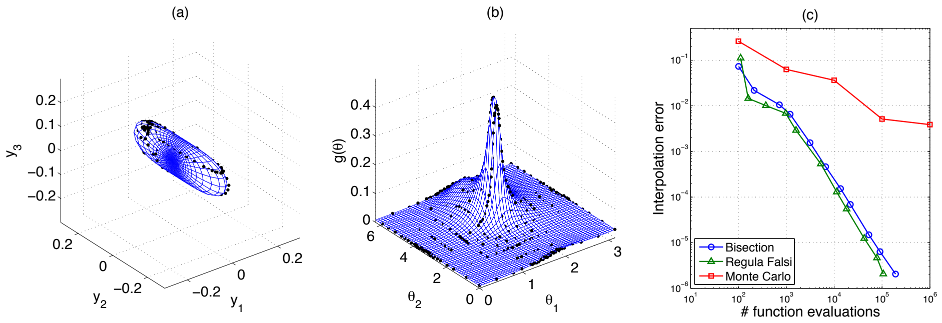

Fig. 5.4: (a) The discontinuity surface γ with 344 sparse grid points; (b) the transformed surface $g(\boldsymbol{\theta})$ in the hyper-spherical coordinate system. (c) The error decays of the Monte Carlo method and the HS-AHSG approximation using the bisection and the Regula Falsi methods.

oretical and numerical results demonstrate the significant improvements yielded by our approach, when compared to existing techniques. Moreover, this technique is not restricted to extensions of sparse grid methods; it can be combined with any other numerical method for high-dimensional approximation such as radial basis approaches and mesh-free interpolation schemes. However, since our approach requires the hypersurface in which the discontinuity is located at an $N-1$-dimensional manifold, we are only able to detect a single hyper-surface at a time, determined by the origin of the hyper-spherical system. More importantly, this choice of origin is also critical to the performance of our approach, as it dictates the shape, and therefore the steepness of the gradient, of the transformed hyper-surface. Finally, our technique relies on the fact the hyper-surface satisfies the star convexity assumption, and thus, for general non-convex hyper-surfaces the proposed approach may fail to converge. Our future efforts will focus on relaxing this assumption to include general geometries of the domain of interest, so that problems with more complicated hyper-surfaces can be investigated.

Appendix A.

A.1. The proof of Proposition 4.4.

Proof. According to the analyses in [4], for $g(\boldsymbol{\theta}) \in X^2(\Gamma_s)$, e_1 is the error arising from the linear isotropic sparse-grid interpolation which is bounded by

$$\|e_1\| \leq C_{\mathrm{sg}} 2^{-2L} \sum_{k=0}^{\widehat{N}-1} \binom{L+\widehat{N}-1}{k}, \tag{A.1}$$

where the constant C_{sg} is independent of the level L and the dimension $\widehat{N}$ (see Theorem 3.8 in [4]). According to the definition in (3.3), e_2 can be written as

$$e_2 = g_L(\boldsymbol{\theta}) - \widetilde{g}_L(\boldsymbol{\theta}) = \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \left(\Delta_{l_1} \otimes \cdots \otimes \Delta_{l_{\widehat{N}}}\right)(g - \widetilde{g})(\boldsymbol{\theta}), \tag{A.2}$$

where $\|g(\boldsymbol{\theta}) - \widetilde{g}(\boldsymbol{\theta})\| \leq \tau$. Thus, it is seen that estimating e_2 is equivalent to estimating the Lebesgue constant, denoted by $\Lambda_{\widehat{N},L}$, of the interpolation operator involved. From the representation in (3.3), $\Lambda_{\widehat{N},L}$ can be estimated using triangle inequality, i.e.,

$$\Lambda_{\widehat{N},L} \leq \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \Lambda_{\mathbf{l}} \leq \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \prod_{n=1}^{\widehat{N}} \Lambda_{l_n},$$

where $\Lambda_{\mathbf{l}} = \Pi_{n=1}^{\widehat{N}} \Lambda_{l_n}$ is the Lebesgue constant of $\Delta_{l_1} \otimes \cdots \otimes \Delta_{l_{\widehat{N}}}$ and Λ_{l_n} is the Lebesgue constant of Δ_{l_n}. By the definition in (3.1), it is easy to show that

$$\Lambda_{l_n} = \sup \left\{ \frac{\|\Delta_{l_n}(g)\|}{\|g\|} \;\middle|\; g \text{ is continuous and } g \neq 0 \right\} \leq \lambda_{l_n} + \lambda_{l_n-1},$$

where λ_{l_n} and λ_{l_n-1} are the Lebesgue constants of $\mathcal{U}_{l_n}$ and $\mathcal{U}_{l_n-1}$, respectively. In the context of linear hierarchical polynomials, we have $\lambda_{l_n} = 1$. Thus, the Lebesgue constant $\Lambda_{\widehat{N},L}$ can be bounded by

$$\begin{aligned}\Lambda_{\widehat{N},L} &\leq \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \prod_{n=1}^{\widehat{N}} (\lambda_{l_n} + \lambda_{l_n-1}) \leq \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} 2^{\widehat{N}} \\ &= 2^{\widehat{N}} \sum_{l=0}^{L} \binom{l+\widehat{N}-1}{\widehat{N}-1} = 2^{\widehat{N}} \sum_{l=0}^{L} \binom{l+\widehat{N}-1}{l} = 2^{\widehat{N}} \binom{L+\widehat{N}}{\widehat{N}}.\end{aligned}$$

Thus, the error e_2 in (A.2) can be estimated by

$$\|e_2\| \leq \Lambda_{\widehat{N},L} \|g(\boldsymbol{\theta}) - \widetilde{g}(\boldsymbol{\theta})\| \leq 2^{\widehat{N}} \binom{L+\widehat{N}}{\widehat{N}} \tau,$$

so that, along with (A.1), (4.7) is obtained. □

A.2. The proof of Lemma 4.5.

Proof. Using the (3.3) and exploiting the nesting structure of the sparse grid, the dimension of $\mathcal{H}_L$ can be represented by

$$|\mathcal{H}_L| = \sum_{l=0}^{L} |\Delta \mathcal{H}_l| = \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \prod_{n=1}^{\widehat{N}} (m_{l_n} - m_{l_n-1}),$$

where $m_{l_n} = 2^{l_n} + 1$ is the number of grid points involved in the one-dimensional interpolant $\mathcal{U}_{l_n}(\cdot)$ and $m_{-1} = 0$. For the linear hierarchical basis, $m_{l_n} - m_{l_n-1} = 2^{l_n} - 1$ for $l_n \geq 1$. We now derive an upper bound for $|\Delta \mathcal{H}_l|$ for $l \geq 1$. Note that there are $\binom{\widehat{N}-1+l}{\widehat{N}-1}$ ways to form the sum l with $\widehat{N}$ nonnegative integers, so we have

$$|\Delta \mathcal{H}_l| = \prod_{n=1}^{\widehat{N}} (m_{l_n} - m_{l_n-1}) \binom{\widehat{N}-1+l}{\widehat{N}-1} \leq 2^l \frac{\left(\widehat{N}-1+l\right)!}{\left(\widehat{N}-1\right)! \cdot l!}.$$

By an inequality from Stirling's approximation of a factorial, i.e.,

$$d_n \leq n! \leq d_n \left(1 + \frac{1}{4n}\right) \quad \text{with} \quad d_n = \sqrt{2\pi n} \left(\frac{n}{\mathrm{e}}\right)^n, \;\; n \in \mathbb{N}^+,$$

we obtain that

$$
\begin{aligned}
|\Delta\mathcal{H}_l| &\le 2^l \left(1+\frac{1}{4(\widehat{N}-1+l)}\right)\frac{d_{\widehat{N}-1+l}}{d_{\widehat{N}-1}\cdot d_l}\\
&= 2^l \frac{\left(1+\frac{1}{4(\widehat{N}-1+l)}\right)\sqrt{\widehat{N}-1+l}}{\sqrt{2\pi l(\widehat{N}-1)}}\left(\frac{\widehat{N}-1+l}{\widehat{N}-1}\right)^{\widehat{N}-1}\left(\frac{\widehat{N}-1+l}{l}\right)^{l}\\
&\le 2^l\left(\frac{l+\widehat{N}-1}{\widehat{N}-1}\right)^{\widehat{N}-1}\left(1+\frac{\widehat{N}-1}{l}\right)^{l} \le 2^l\left(\frac{l+\widehat{N}-1}{\widehat{N}-1}\right)^{\widehat{N}-1}\mathrm{e}^{\widehat{N}-1}.
\end{aligned}
$$

It is easy to see that $|\Delta\mathcal{H}_0|$ satisfies the above inequality as well. This concludes the proof about $|\Delta\mathcal{H}^l|$. The estimate for $|\mathcal{H}_L|$ can be obtained immediately based on the estimate of $|\Delta\mathcal{H}_l|$. □

A.3. The proof of Lemma 4.6.

Proof. We observe that the value of the minimal solution of the inequality (4.8) has two possibilities, i.e., $L<\widehat{N}$ and $L\ge\widehat{N}$. In the former case, all values larger than $\widehat{N}$ are also solutions of (4.8). Hence, we assume the solution of (4.8) is larger than $\widehat{N}$. It is also observed that if $L\ge\widehat{N}$, we have

$$
\sum_{k=0}^{\widehat{N}-1}\binom{L+\widehat{N}-1}{k}\le\widehat{N}\binom{L+\widehat{N}-1}{\widehat{N}-1}\le\widehat{N}\binom{L+\widehat{N}}{\widehat{N}}\le\widehat{N}\left(\frac{2L}{\widehat{N}}\right)^{\widehat{N}}\mathrm{e}^{\widehat{N}}. \qquad \text{(A.3)}
$$

Thus, instead of solving (4.8) directly, it is sufficient to solve

$$
C_{\mathrm{sg}}2^{-2L}\widehat{N}\left(\frac{2L}{\widehat{N}}\right)^{\widehat{N}}\mathrm{e}^{\widehat{N}}\le\frac{\varepsilon}{2}\quad\text{and}\quad L\ge\widehat{N}. \qquad \text{(A.4)}
$$

Now, we temporarily treat L as a positive real number for convenience and the desired iteration number is $\lceil L\rceil$. Let $L=t\widehat{N}/\ln 4$ in (A.4). Then, we have

$$
\begin{aligned}
&\left(\frac{2L}{\widehat{N}}\right)^{\widehat{N}}\mathrm{e}^{\widehat{N}}\left(\frac{2\widehat{N}C_{\mathrm{sg}}}{\varepsilon}\right)\le 2^{2L} \iff \left(\frac{t}{\ln 2}\right)^{\widehat{N}}\mathrm{e}^{\widehat{N}}\left(\frac{2\widehat{N}C_{\mathrm{sg}}}{\varepsilon}\right)\le 4^{\frac{t}{\ln 4}\widehat{N}}\\
\iff &\left(\frac{t\mathrm{e}}{\ln 2}\right)\left(\frac{2\widehat{N}C_{\mathrm{sg}}}{\varepsilon}\right)^{\frac{1}{\widehat{N}}}\le 4^{\frac{t}{\ln 4}} \iff \ln t+\ln\left[\frac{\mathrm{e}}{\ln 2}\left(\frac{2C_{\mathrm{sg}}}{\varepsilon}\right)^{\frac{1}{\widehat{N}}}\widehat{N}^{\frac{1}{\widehat{N}}}\right]\le t\\
\Longleftarrow\ &\ln t+\ln\left[\frac{2\mathrm{e}}{\ln 2}\left(\frac{2C_{\mathrm{sg}}}{\varepsilon}\right)^{\frac{1}{\widehat{N}}}\right]\le t,
\end{aligned}
$$

so that (A.4) is satisfied with minimum L given by $L=t\widehat{N}/\ln 4$ if t satisfies

$$
t\ge\ln t+\ln h\quad\text{with}\quad h=\frac{2\mathrm{e}}{\ln 2}\left(\frac{2C_{\mathrm{sg}}}{\varepsilon}\right)^{\frac{1}{\widehat{N}}},
$$

where $h>1$ by hypothesis. Letting $t_0=\frac{\mathrm{e}}{\mathrm{e}-1}\ln h$, it is easy to verify that

$$
t_0-\ln h=\frac{1}{\mathrm{e}-1}\ln h\ge 1+\ln\left(\frac{1}{\mathrm{e}-1}\ln h\right)=\ln\left(\frac{\mathrm{e}}{\mathrm{e}-1}\ln h\right)=\ln t_0,
$$

and that the inequality (A.4) is satisfied. Furthermore, for $k\ge 0$, $t_k=\ln(t_{k-1}h)\le t_{k-1}$ is also the solution of (A.4) due to the fact that

$$
\ln t_k+\ln h=\ln(\ln t_{k-1}+\ln h)+\ln h\le\ln t_{k-1}+\ln h=\ln(t_{k-1}h)=t_k. \qquad \text{(A.5)}
$$

Thus, the sequence $\{t_k\}_{k=0}^{\infty}$ monotonically converges to a unique solution t^* such that $t^* = \ln t^* + \ln h$. Based on the sequence $\{t_k\}_{k=0}^{\infty}$, we can easily find a sequence of upper bounds $\{L_k\}_{k=0}^{\infty}$ for the minimum L satisfying the inequality (4.8). □

A.4. The proof of Theorem 4.8.

Proof. According to the definition in (4.5), the cost $\mathcal{C}_{\min}$ can be bounded by

$$\mathcal{C}_{\min} \leq |\mathcal{H}_{L_k}|\, K(\tau_0, \varepsilon, L_k, \widehat{N}), \tag{A.6}$$

where L_k for $k \in \mathbb{N}$ is determined from Lemma 4.6 and $K(\tau_0, \varepsilon, L, \widehat{N})$ is the necessary number of iterations for the bisection method to achieve the accuracy $\varepsilon/2$ in (4.9) at each grid point for fixed $\widehat{N}$, L, ε, and initial search interval length τ_0. We can see that the necessary tolerance τ of the bisection method is determined by (4.9), i.e.,

$$\tau(\widehat{N}, L, \varepsilon) = 2^{-\widehat{N}-1}\varepsilon \Bigg/ \binom{L+\widehat{N}}{\widehat{N}};$$

$K(\tau_0, \varepsilon, L, \widehat{N})$ can be represented by

$$K(\tau_0, \varepsilon, L, \widehat{N}) = \log_2\left[\frac{2^{\widehat{N}+1}\tau_0}{\varepsilon}\binom{L+\widehat{N}}{\widehat{N}}\right], \tag{A.7}$$

where we temporarily treat K as a positive real number for convenience and the desired iteration number is $\lceil K \rceil$. According to the discussion in §4.2.2, τ_0 is set to $|\boldsymbol{y}_0 - \boldsymbol{\beta}_{\boldsymbol{\theta}}|$ without any prior knowledge, thus $\tau_0 \leq (\widehat{N}+1)^{\frac{1}{2}}$ which is the length of the diagonal of $[0,1]^N$. Substituting L_0 into (A.7), we have

$$\begin{aligned}
&K(\tau_0, \varepsilon, L_0, \widehat{N})\\
\leq&\log_2\left(\frac{2^{\widehat{N}+1}\tau_0}{\varepsilon}\right) + \log_2\left(\frac{\varepsilon}{2\widehat{N}C_{\rm sg}}2^{2L_0}\right) = \log_2\left(\frac{2^{\widehat{N}+1}\tau_0}{C_{\rm sg}\widehat{N}}\right) + 2L_0\\
\leq&\log_2\left(\frac{2^{\widehat{N}+1}(\widehat{N}+1)^{\frac{1}{2}}}{C_{\rm sg}\widehat{N}}\right) + \frac{{\rm e}\widehat{N}}{{\rm e}-1}\log_2\left[\frac{2{\rm e}}{\ln 2}\left(\frac{2C_{\rm sg}}{\varepsilon}\right)^{\frac{1}{\widehat{N}}}\right]\\
\leq&\widehat{N} + \frac{{\rm e}\widehat{N}}{{\rm e}-1}\log_2\left[\frac{2{\rm e}}{\ln 2}\left(\frac{2C_{\rm sg}}{\varepsilon}\right)^{\frac{1}{\widehat{N}}}\right] + 2 - \log_2(C_{\rm sg})\\
=&\frac{{\rm e}}{{\rm e}-1}\log_2\left(\frac{2C_{\rm sg}}{\varepsilon}\right) + \widehat{N}\left\{\frac{{\rm e}}{{\rm e}-1}\log_2\left(\frac{2{\rm e}}{\ln 2}\right) + 1\right\} + 2 - \log_2(C_{\rm sg})\\
=&\alpha_6\log_2\left(\frac{2C_{\rm sg}}{\varepsilon}\right) + \alpha_7\widehat{N} + \alpha_8.
\end{aligned} \tag{A.8}$$

On the other hand, substituting L_1 into the upper bound of $\mathcal{H}_{L_1}$, we have

$$
\begin{aligned}
|\mathcal{H}_{L_1}| &\leq 2^{L_1+1} \begin{pmatrix} L_1 + \widehat{N} - 1 \\ \widehat{N} - 1 \end{pmatrix} \leq 2^{L_1+1} \begin{pmatrix} L_1 + \widehat{N} \\ \widehat{N} \end{pmatrix} \\
&\leq 2^{L_1+1} \left(\frac{\varepsilon}{2\widehat{N} C_{\mathrm{sg}}} \right) 2^{2L_1} \leq \left(\frac{\varepsilon}{\widehat{N} C_{\mathrm{sg}}} \right) 2^{\frac{3 t_1 \widehat{N}}{2 \ln 2}} \\
&= \left(\frac{\varepsilon}{\widehat{N} C_{\mathrm{sg}}} \right) 2^{\frac{3 \ln(t_0 h) \widehat{N}}{2 \ln 2}} = \left(\frac{\varepsilon}{\widehat{N} C_{\mathrm{sg}}} \right) t_0^{\frac{3}{2}\widehat{N}} \left[\frac{2\mathrm{e}}{\ln 2} \left(\frac{2C_{\mathrm{sg}}}{\varepsilon} \right)^{\frac{1}{\widehat{N}}} \right]^{\frac{3}{2}\widehat{N}} \\
&= \left(\frac{\varepsilon}{\widehat{N} C_{\mathrm{sg}}} \right) \left(\frac{\mathrm{e}}{\mathrm{e}-1} \ln h \right)^{\frac{3}{2}\widehat{N}} \left(\frac{2\mathrm{e}}{\ln 2} \right)^{\frac{3}{2}\widehat{N}} \left(\frac{2C_{\mathrm{sg}}}{\varepsilon} \right)^{\frac{3}{2}} \\
&= \frac{2}{\widehat{N}} \left(\frac{2C_{\mathrm{sg}}}{\varepsilon} \right)^{\frac{1}{2}} \left\{ \frac{2\mathrm{e}^2}{\mathrm{e}-1} \log_2 \left[\frac{2\mathrm{e}}{\ln 2} \left(\frac{2C_{\mathrm{sg}}}{\varepsilon} \right)^{\frac{1}{\widehat{N}}} \right] \right\}^{\frac{3}{2}\widehat{N}} \\
&= \frac{2}{\widehat{N}} \left\{ \frac{2\mathrm{e}^2}{\mathrm{e}-1} \log_2 \left(\frac{2\mathrm{e}}{\ln 2} \right) + \frac{2\mathrm{e}^2}{\mathrm{e}-1} \frac{\log_2 \left(\frac{2C_{\mathrm{sg}}}{\varepsilon} \right)}{\widehat{N}} \right\}^{\frac{3}{2}\widehat{N}} \left(\frac{2C_{\mathrm{sg}}}{\varepsilon} \right)^{\frac{1}{2}} \\
&= \alpha_1 \left\{ \alpha_2 + \alpha_3 \frac{\log_2 \left(\frac{2C_{\mathrm{sg}}}{\varepsilon} \right)}{\widehat{N}} \right\}^{\alpha_4 \widehat{N}} \left(\frac{2C_{\mathrm{sg}}}{\varepsilon} \right)^{\alpha_5}.
\end{aligned}
\tag{A.9}
$$

Hence, by substituting (A.8) and (A.9) into (A.6), the proof is finished. □

A.5. The proof of Lemma 4.9.

Proof. As in (4.7), we split the error into two parts, i.e.,

$$
g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}) - \widetilde{g}_{L-1}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}) = \underbrace{g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}) - g_{L-1}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})}_{e_1} + \underbrace{g_{L-1}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}) - \widetilde{g}_{L-1}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}})}_{e_2},
$$

where e_1 is the definition of the hierarchical surplus $\omega_{\mathbf{l},\mathbf{i}}$ whose upper bound is given in [4], i.e., $|e_1| \leq C_{\mathrm{surp}} \cdot 2^{-2L}$ with C_{surp} is independent of L and e_2 measures the error between the exact prediction of the surplus and the perturbed one. To estimate e_2, we need to extend the formula for calculating surpluses given in [4] by including the sparse grid points on the boundary. Based on [4, Lemma 3.2], we can see that for each grid point $\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}$ with $|\mathbf{l}| \geq 1$, its exact surplus $\omega_{\mathbf{l},\mathbf{i}}$ can be computed from the function values of $g(\boldsymbol{\theta})$ as follows:

$$
\omega_{\mathbf{l},\mathbf{i}} = \mathcal{A}_{\mathbf{l},\mathbf{i}}(g) = \left(\prod_{n=1}^{\widehat{N}} \mathcal{A}_{l_n, i_n} \right)(g),
$$

where $\mathcal{A}_{\mathbf{l},\mathbf{i}}(\cdot)$ is an $\widehat{N}$-dimensional stencil that provides the coefficients for a linear combination of the nodal values of the function g to compute $\omega_{\mathbf{l},\mathbf{i}}$. Specifically, $\mathcal{A}_{\mathbf{l},\mathbf{i}}$ is product of $\widehat{N}$ one-dimensional stencils $\mathcal{A}_{l_n,i_n}$ for $l_n > 1, n = 1, \ldots, \widehat{N}$, defined by

$$
\begin{aligned}
\mathcal{A}_{l_n,i_n}(g) &= \left[-\frac{1}{2} \quad 1 \quad -\frac{1}{2} \right]_{l_n,i_n} (g) \\
&= -\frac{1}{2} g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}} - h_{l_n} \boldsymbol{e}_n) + g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}) - \frac{1}{2} g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}} + h_{l_n} \boldsymbol{e}_n),
\end{aligned}
\tag{A.10}
$$

where $\boldsymbol{e}_n$ is a vector of zeros except for the n-th entry that is one, and h_{l_n} is a scalar equal to a half of the length of the support of the basis function $\psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{\theta})$ in the n-th direction. Note that we have $\mathcal{A}_{0,0}(g) = [0,1,0]_{0,0}(g)$, for $l_n = 0, i_n = 0$, $\mathcal{A}_{1,0}(g) = [0,1,-1]_{1,0}(g)$ for $l_n = 1, i_n = 0$ and $\mathcal{A}_{1,2}(g) = [-1,1,0]_{1,2}(g)$ for $l_n = 1, i_n = 2$. It is easy to see that the sum of the absolute values of the coefficients of $\mathcal{A}_{\mathbf{l},\mathbf{i}}(\cdot)$ is equal to $2^{\widehat{N}}$. Note that all the involved grid points in (A.10) belong to $g_{L-1}(\boldsymbol{\theta})$ except for $\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}$. Thus, due to the fact that $|g(\boldsymbol{\theta}) - \widetilde{g}(\boldsymbol{\theta})| \le \tau$, the error e_2 can be estimated by

$$|e_2| = |\mathcal{A}_{\mathbf{l},\mathbf{i}}(g - \widetilde{g}) - (g(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}) - \widetilde{g}(\boldsymbol{\theta}_{\mathbf{l},\mathbf{i}}))| \le 2^{\widehat{N}}\tau,$$

which completes the proof. □

A.6. The proof of Theorem 4.10.

Proof. For $L = L_1$, according to the definition in (4.5), $\mathcal{C}_{\min}$ can be bounded by

$$\begin{aligned}
\mathcal{C}_{\min} &\le \sum_{l=0}^{L_1} |\Delta\mathcal{H}_l|\, K(\tau_0^l, \varepsilon, L_1, \widehat{N}) \\
&\le \sum_{l=0}^{L_1} 2^l \binom{l+\widehat{N}-1}{\widehat{N}-1} \log_2\left[\frac{2^{\widehat{N}+1}\tau_0^l}{\varepsilon}\binom{L_1+\widehat{N}}{\widehat{N}}\right],
\end{aligned} \tag{A.11}$$

where we temporarily treat K as a positive real number for convenience and the desired iteration number is $\lceil K \rceil$. Based on Lemma 4.9, we define the initial search interval τ_0^l on level l by $\tau_0^l = C_{\text{surp}}2^{-2l} + 2^{\widehat{N}}\tau$, where τ is the tolerance of the bisection method. For sufficiently small ε, the logarithmic function in (A.11) is positive. Substituting such τ_0^l into (A.11), we obtain

$$\begin{aligned}
\mathcal{C}_{\min} &\le \sum_{l=0}^{L_1} 2^l \binom{l+\widehat{N}-1}{\widehat{N}-1} \log_2\left[\frac{2^{\widehat{N}+1}}{\varepsilon}\binom{L_1+\widehat{N}}{\widehat{N}}\left(C_{\text{surp}}2^{-2l} + 2^{\widehat{N}}\tau\right)\right] \\
&= \sum_{l=0}^{L_1} 2^l \binom{l+\widehat{N}-1}{\widehat{N}-1} \log_2\left[\frac{2^{\widehat{N}+1}}{\varepsilon}\binom{L_1+\widehat{N}}{\widehat{N}}\left(C_{\text{surp}}2^{-2l} + \frac{\varepsilon}{2\binom{L_1+\widehat{N}}{\widehat{N}}}\right)\right] \\
&= \sum_{l=0}^{L_1} 2^l \binom{l+\widehat{N}-1}{\widehat{N}-1} \left\{\log_2\left[\frac{2^{\widehat{N}+1}C_{\text{surp}}2^{-2l}}{\varepsilon}\binom{L_1+\widehat{N}}{\widehat{N}}\right] + \widehat{N}\right\} \\
&\le \sum_{l=0}^{L_1} 2^l \binom{l+\widehat{N}-1}{\widehat{N}-1} \left\{\log_2\left[\frac{2^{\widehat{N}+1}C_{\text{surp}}2^{2(L_1-l)}}{\varepsilon}\frac{\varepsilon}{2\widehat{N}C_{\text{sg}}}\right] + \widehat{N}\right\} \\
&= \sum_{l=0}^{L_1} 2^l \binom{l+\widehat{N}-1}{\widehat{N}-1} \left[2(L_1-l) + \log_2\left(\frac{C_{\text{surp}}}{C_{\text{sg}}}\right) + 2\widehat{N} - \log_2(\widehat{N})\right] \\
&\le \binom{L_1+\widehat{N}}{\widehat{N}} \sum_{l=0}^{L_1}(L_1-l)2^l + 2^{L_1+1}\binom{L_1+\widehat{N}}{\widehat{N}}\left[\log_2\left(\frac{C_{\text{surp}}}{C_{\text{sg}}}\right) + 2\widehat{N} - \log_2(\widehat{N})\right] \\
&\le 2^{L_1+1}\binom{L_1+\widehat{N}}{\widehat{N}}\left[\log_2\left(\frac{C_{\text{surp}}}{C_{\text{sg}}}\right) + 2\widehat{N} + 2 - \log_2(\widehat{N})\right] \\
&\le \alpha_1\left[\alpha_2 + \alpha_3\frac{\log_2\left(\frac{2C_{\text{sg}}}{\varepsilon}\right)}{\widehat{N}}\right]^{\alpha_4\widehat{N}}\left(\frac{2C_{\text{sg}}}{\varepsilon}\right)^{\alpha_5}\left[2\widehat{N} - \log_2(\widehat{N}) + \alpha_9\right],
\end{aligned}$$

which completes the proof. □

REFERENCES

[1] R. Archibald, A. Gelb, R. Saxena, and D. Xiu, *Discontinuity detection in multivariate space for stochastic simulations*, Journal of Computational Physics, 228 (2009), pp. 2676–2689.

[2] R. Archibald, A. Gelb, and J. Yoon, *Polynomial fitting for edge detection in irregularly sampled signals and images*, SIAM Journal on Numerical Analysis, 43 (2005), pp. 259–279 (electronic).

[3] H. J. Bungartz and S. Dirnstorfer, *Multivariate Quadrature on Adaptive Sparse Grids*, Computing, 71 (2003), pp. 89–114.

[4] H.-J. Bungartz and M. Griebel, *Sparse grids*, Acta Numerica, 13 (2004), pp. 1–123.

[5] H. J. Bungartz, D. Pflüger, and S. Zimmer, *Adaptive Sparse Grid Techniques for Data Mining*, in Modeling, Simulation and Optimization of Complex Processes, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 121–130.

[6] C. Feuersänger and M. Griebel, *Principal manifold learning by sparse grids*, Computing, 85 (2009), pp. 267–299.

[7] D. Galido, P. Jantsch, C. Webster, and G. Zhang, *Accelerating Hierarchical Stochastic Collocation Methods for Partial Differential Equations with Random Input Data*, ORNL Technical Report, (2014).

[8] T. Gerstner and M. Griebel, *Numerical integration using sparse grids*, Numerical Algorithms, 18 (1998), pp. 209–232.

[9] T. Gerstner and M. Griebel, *Dimension–Adaptive Tensor–Product Quadrature*, Computing, 71 (2003), pp. 65–87.

[10] M. Griebel, *Adaptive sparse grid multilevel methods for elliptic PDEs based on finite differences*, Computing, 61 (1998), pp. 151–179.

[11] M. Griebel and M. Holtz, *Dimension-wise integration of high-dimensional functions with applications to finance*, Journal of Complexity, 26 (2010), pp. 455–489.

[12] M. Gunzburger, C. Webster, and G. Zhang, *An Adaptive Wavelet Stochastic Collocation Method for Irregular Solutions of Partial Differential Equations with Random Input Data*, Springer Lecture Notes on Computational Science and Engineering, 97 (2014), pp. 137–170.

[13] ———, *Stochastic Finite Element Methods for Partial Differential Equations with Random Input Data*, Acta Numerica, 23 (2014), pp. 521–650.

[14] M. Holtz, *Sparse Grid Quadrature in High Dimensions with Applications in Finance and Insurance*, Springer Science & Business Media, Oct. 2010.

[15] J. D. Jakeman, R. Archibald, and D. Xiu, *Characterization of discontinuities in high-dimensional stochastic problems on adaptive sparse grids*, Journal of Computational Physics, 230 (2011), pp. 3977–3997.

[16] X. Ma and N. Zabaras, *An adaptive hierarchical sparse grid collocation algorithm for the solution of stochastic differential equations*, Journal of Computational Physics, 228 (2009), pp. 3084–3113.

[17] ———, *An efficient Bayesian inference approach to inverse problems based on an adaptive sparse grid collocation method*, Inverse Problems, 25 (2009), p. 035013.

[18] F. Nobile, R. Tempone, and C. G. Webster, *A sparse grid stochastic collocation method for partial differential equations with random input data*, SIAM Journal on Numerical Analysis, 46 (2008), pp. 2309–2345.

[19] ———, *An anisotropic sparse grid stochastic collocation method for partial differential equations with random input data*, SIAM Journal on Numerical Analysis, 46 (2008), pp. 2411–2442.

[20] E. Novak and K. Ritter, *High-dimensional integration of smooth functions over cubes*, Numerische Mathematik, 75 (1996), pp. 79–97.

[21] B. Peherstorfer, J. Adorf, D. Pflüger, and H.-J. Bungartz, *Image Segmentation with Adaptive Sparse Grids*, in AI 2013: Advances in Artificial Intelligence, Springer International Publishing, Cham, Jan. 2013, pp. 160–165.

[22] D. Pflüger, I. L. Muntean, and H.-J. Bungartz, *Adaptive Sparse Grid Classification Using Grid Environments.*, International Conference on Computational Science, 4487 (2007), pp. 708–715.

[23] G. W. Wasilkowski and H. Woźniakowski, *Explicit cost bounds of algorithms for multivariate tensor product problems*, Journal of Complexity, 11 (1995), pp. 1–56.

[24] C. G. Webster, G. Zhang, and M. Gunzburger, *An adaptive sparse-grid iterative ensemble Kalman filter approach for parameter field estimation*, International Journal of Computer

Mathematics, 91 (2013), pp. 798–817.

[25] G. Zhang, M. Gunzburger, and W. Zhao, *A sparse-grid method for multi-dimensional backward stochastic differential equations*, Journal of Computational Mathematics, 31 (2013), pp. 221–248.

[26] G. Zhang, D. Lu, M. Ye, M. Gunzburger, and C. Webster, *An adaptive sparse-grid high-order stochastic collocation method for bayesian inference in groundwater reactive transport modeling*, Water Resources Research, 49 (2013), pp. 6871–6892.