Slow Exponential Growth for Clenshaw Curtis Sparse Grids http://people.sc.fsu.edu/∼jburkardt/presentations/... ...slow_growth_paper.pdf

John Burkardt, Clayton Webster

April 30, 2014

Abstract

A widely used example of Smolyak's sparse grid construction is based on the Clenshaw Curtis rules. A prominent feature of these rules is their nestedness. Nestedness is a crucial factor for controlling point growth of sparse grids in high dimension, but in lower dimensions, it can result in significant inefficiencies. A simple change to the construction is proposed which can greatly improve low dimensional versions of the Clenshaw Curtis sparse grid. Similar benefits can be observed when other families of rules are used.

1 Introduction

Sparse grids, as defined by Smolyak [9], are powerful and efficient tools for interpolation, quadrature, and optimization of functions with dependence on a multidimensional argument. By strictly controlling the number of function evaluations required, sparse grids can handle problems in dimensions far beyond the reach of any other methods except Monte Carlo. One of the techniques that can be used to extend the power of a sparse grid is nestedness; that is, the reuse of data from lower order calculations. A very common example of a sparse grid is based on the nested Clenshaw Curtis rules. Nestedness comes at a price, which for the Clenshaw Curtis rules means that each successive element of the family uses about twice as many points as the previous one, an example of exponential growth. For various reasons, this exponential growth does not dominate the behavior of sparse grids in high dimension and low to moderate level. However, in relatively low dimensions, say $d \leq 5$, a Clenshaw Curtis sparse grid incurs obvious, and as it turns out, *easily avoidable*, expense because of its simple-minded reliance on nested rules.

In an analysis of the properties of sparse grids based on Clenshaw Curtis rules, Novak and Ritter [6] showed that the exactness of the sparse grid is related in a simple way to the exactness of the 1D quadrature rules used to construct it; this result shows that there is actually considerable freedom available when specifying how a sparse grid is to be constructed. Using this guideline, a simple modification allows the creation of Clenshaw Curtis sparse grids that are smaller, but of the same exactness as the classic grids.

Sparse grids can also be constructed from Legendre quadrature rules. Since these rules are not nested, the construction pattern associated with Clenshaw Curtis rules is surely inappropriate as a model. Again, the Novak and Ritter guideline can be used in order to arrive at an efficient sparse grid with known exactness.

The Patterson family of quadrature rules represents an interesting mix of the features of the Clenshaw Curtis and Legendre families of rules. The family is nested (although in a different way from the Clenshaw Curtis family) and of increased accuracy (although less than the Legendre family). Again, Novak and Ritter can be used to specify a construction plan of guaranteed exactness and demonstrable efficiency.

As suggested earlier, the efficiency to be gained by this approach is most prominent when the sparse grid is generated in relatively low dimensions. However, many sparse grid techniques that are applied to high dimensions actually select a small subspace for preferential treatment, or apply anisotropic weights that have a similar effect; such cases may also be be regarded as low dimensional, and hence possibly affected by the improvements considered here.

In order to focus on the main point of the argument, a number of simplifying assumptions will be made:

- sparse grids are being constructed in order to estimate the integral $I(f)$ of a function of a multidimensional argument;
- the multidimensional integration region is the unit hypercube $[-1,+1]^m$;
- $\mathbb{Q}$ represents an indexed family of quadrature rules for the interval $[-1,+1]$, with typical element Q^i;
- the same family $\mathbb{Q}$ will be used to select each component of the sparse grid;

The outline of the remainder of this paper is as follows: Section 2 presents some background material on the Clenshaw Curtis family of 1D quadrature rules, the Smolyak construction procedure, the notion of exactness for quadrature, and the Novak and Ritter exactness constraint. Section 3 presents the classic construction of sparse grids from the Clenshaw Curtis family, and then reconstructs such sparse grids using Novak and Ritter. Sections 4 and 5 carry out similar operations for sparse grids based on Legendre and Patterson families. Section 6 presents some simple numerical tests indicating that the modified approach produces sparse grids that outperform the classic variety.

2 Construction of Sparse Grids for Quadrature

A version of the univariate quadrature problem seeks to estimate the integral $I(f) = \int_{-1}^{+1} f(x)\,dx$. A quadrature rule $Q()$ for this problem is a set of n points x and weights w which produce the integral estimate:

$$I(f) \approx Q(f) = \sum_{i=1}^{n} w_i f(x_i).$$

Such a quadrature rule is said to have *exactness of degree d* if the integral estimate is exact whenever the integrand f is a polynomial of degree d or less. A common strategy for quadrature involves assembling an indexed family $\mathbb{Q}$ of quadrature rules of increasing exactness. By applying rules of increasing index to a given problem, a reasonable estimate of the quadrature error may be made.

A version of the multivariate quadrature problem may be posed in the same way for a function of a variable $x \in [-1,+1]^m$. Quadrature rules for this problem may be constructed by making m selections from $\mathbb{Q}$ and forming the product rule. A significant drawback of this approach arises because, if the 1D rule of exactness d requires n points, then the product rule of corresponding exactness requires n^m points, a fact which rules out the product approach except for low dimensions or degrees of exactness.

The sparse grid construction of Smolyak [9] showed how simpler product rules could be combined in a way that achieved the exactness of a given product rule. Of course, if enough simple product rules are involved, the total number of points can grow arbitrarily. Thus, a key idea in the implementation of Smolyak's procedure was to prefer quadrature families $\mathbb{Q}$ that were nested, which greatly reduces the point count of the sparse grid.

Smolyak's formula produces a sequence of m-dimensional sparse grids with an index $\ell = 0, 1, 2, ...$ often called the *level*:

$$\mathcal{A}(\ell, m) = \sum_{0 \le \ell - |\mathbf{i}| \le m-1} (-1)^{\ell - |\mathbf{i}|} \left(\begin{array}{c} m-1 \\ \ell - |\mathbf{i}| \end{array} \right) (Q^{i_1} \otimes \cdots \otimes Q^{i_m})$$

It is natural to expect that, for a given spatial dimension m, the sequence of sparse grids $\mathcal{A}(\ell, m), \ell = 0, 1, 2, ...$ produce integral estimates of increasing exactness. Novak and Ritter were able to show that, if based on the classic Clenshaw-Curtis family, the sparse grid of level ℓ would have exactness $2\ell + 1$. It will be seen shortly that this theorem suggests a more efficient way to employ the Clenshaw Curtis family; it can also be extended to other families of 1D rules. At the moment, it is enough to note that this theorem demonstrates that, at least for smooth integrands, a sequence of sparse grids can be used to produce integral estimates of rapidly improving accuracy.

Table 1: Exactness and CCE/CCL/CCS 1D rule sizes

i = index	0	1	2	3	4	5	6	7	8	9	10
e_i (required)	1	3	5	7	9	11	13	15	17	19	21
n_i (CCE)	1	3	5	9	17	33	65	125	257	513	1025
n_i (CCL)	1	3	5	7	9	11	13	15	17	19	21
n_i (CCS)	1	3	5	9	9	17	17	17	17	33	33

3 Using the Clenshaw-Curtis Quadrature Family

Clenshaw and Curtis [1] presented a family of easily-computed 1D quadrature rules with positive weight and nested abscissas. Their purposes required the evaluation of a sequence of estimates, and so nesting was a valuable way of reducing the number of times the integrand was evaluated. The rules were defined on the interval $[-1, +1]$. The first rule ($i = 0$) has size $n_0 = 1$, but all subsequent rules have size $n_i = 2^i + 1$, thus exhibiting exponential growth. This 1D quadrature family will be denoted by **CCE**. Because of symmetry and the fact that the rules are all of odd size, the exactness e of each rule is the same as its size, that is $e = n$.

Novak and Ritter showed that the exactness of sparse grids constructed from a family $\mathbb{Q}$ was related to the exactness of the members of that family. In particular, they showed that, if every 1D rule Q^i had exactness at least $2i + 1$, then the sparse grid of level ℓ is guaranteed to have exactness at least $2\ell + 1$.

This condition is easily verified for the CCE family; Novak and Ritter require that the exactness sequence for the 1D family be at least $2i + 1$ but it has already been observed that $e_i = n_i = 2^i + 1 \geq 2i + 1$ (the case $i = 0$ is treated separately), and hence the desired exactness of sparse grids constructed from the CCE family follows.

Since efficiency is a vital concern for high dimensional problems, it is natural to wonder whether there is an unnecessary cost associated with the fact that the CCE family exponentially exceeds the linear Novak and Ritter requirement. This excess is most obvious if one imagines using the Smolyak procedure to construct a sequence of 1D "sparse grids", which simply produces the elements of the quadrature family.

The nesting of the CCE family, which is so beneficial in higher dimensions, is forcing the doubling in size that is so troubling in lower dimensions. It is worth asking if there are simple alternatives to the CCE family that will moderate the doubling effect.

An obvious choice would be to abandon nesting. The CCE family selected a particular sequence of sizes to guarantee nesting. Instead, it is possible to choose a sequence of sizes n that guarantee the Novak and Ritter condition, namely $n_i = 2i+1, i = 0, 1, 2, ...$ This family will be denoted by **CCL**. Because these rules have odd size, the exactness condition is still met, although the perfect nesting of the CCE family has been abandoned.

A second choice is to preserve nesting, but to change the way the individual quadrature rules are indexed. This rule will be denoted **CCS**, for the "slow growth" option. The informal definition of this family is that the i-th member of the family is the smallest member of the CCE family that has exactness at least $2i + 1$. Another way of viewing the CCS family is to imagine that the elements in the sequence are the same as those in the CCE sequence, but that any element, once it occurs in the sequence at position i, may be repeated at one or more subsequent positions, if it satisfies the corresponding exactness requirements.

Table 1 compares the sizes of the 1D rules in the CCE, CCL and CCS families:

It is clear that the three families differ little for low index i, but that the two newly-defined families avoid the exponential explosion afflicting the higher-index elements of the CCE family.

The differences are suggested by Table 2, which counts the number of points in the corresponding sparse grids for several moderate dimensions. The improvement offered by the new families is strongest in the sparse grids of lowest dimension.

Table 2: CCE/CCL/CCS sparse grid point counts in 2D, 6D, 10D

ℓ = level	0	1	2	3	4	5	6	7	8	9	10
Dimension 2											
n_ℓ (CCE)	1	5	13	29	65	145	321	705	1,537	3,329	7,169
n_ℓ (CCL)	1	5	13	29	57	105	177	281	425	611	855
n_ℓ (CCS)	1	5	13	29	49	81	129	161	225	257	385
Dimension 6											
n_ℓ (CCE)	1	13	85	389	1,457	4,865	15,121	44,689	127,105	350,657	943,553
n_ℓ (CCL)	1	13	85	389	1,433	4,533	12,961	33,817	82,153	188,039	408,995
n_ℓ (CCS)	1	13	85	389	1,409	4,289	11,473	27,697	61,345	126,401	244,289
Dimension 10											
n_ℓ (CCE)	1	21	221	1,581	8,801	41,265	171,425	652,065	2,320,385	7,836,545	25,370,753
n_ℓ (CCL)	1	21	221	1,581	8,761	40,425	162,385	584,665	?	?	?
n_ℓ (CCS)	1	21	221	1,581	8,721	39,665	155,105	536,705	1,677,665	4,810,625	12,803,073

4 Using the Legendre Quadrature Family

Gaussian quadrature rules offer increased exactness, at the same time constraining the choice of abscissa location. An n-point Gauss-Legendre rule, for instance, defined on the interval $[-1, 1]$, will have exactness $e_i = 2n_i - 1$. A quadrature family built from the Legendre rule will benefit from the increased accuracy, but has the disadvantage that there is almost no opportunity for nesting, aside from the occurrence of the abscissa $x = 0$ in every rule of odd size.

If it is desired to construct sparse grids using the Gauss-Legendre family, it is reasonable to imitate the procedure used for the classic Clenshaw-Curtis example, which can be understood as choosing the next rule by adding a new abscissa within each subinterval of the current rule. This will again amount to exponential growth, so the family will be denoted by **GLE**. Since the rules are open, the growth pattern differs from the CCE family, producing rules of size $n_i = 1, 3, 7, 15, 31, 63, ..., 2^{i+1} - 1$.

Since there is little nesting advantage available in this case, it makes more sense to be guided by the Novak and Ritter criterion. Taking it strictly, the **GLL** family can be constructed, with a linearly growing size sequence of $n_i = 1, 2, 3, 4, ..., i + 1$, which just satisfies the exactness constraint.

However, it is tempting to consider forming a family only using odd rules; this takes whatever advantage is to be had from the repeated abscissa $x = 0$, at the expense of often using slightly more powerful rules. This family will be denote **GLO**.

Table 3 compares the point counts for the three Gauss-Legendre-based quadrature families in moderate dimensions. It is hardly surprising that the GLE family grows so rapidly, but it is truly disconcerting to see that GLO sparse grids use far fewer points than GLL, and do so by using bigger 1D rules! It is a testament to the somewhat unpredictable power of nesting, in this case applied to a single repeated abscissa!

5 Using the Patterson Quadrature Family

Patterson [7] produced a family of quadrature rules that are completely nested like the CCE family, with an exponential growth in size like the GLE family, and an exactness that may be estimated as $e_i \approx \frac{3}{2} n_i$, about halfway between the exactnesses of Clenshaw-Curtis and Gauss-Legendre rules.

Let **GPE** denote the quadrature family formed by the Patterson rules, with sizes $1, 3, 7, 15, 31, ...$. Let **GPS** denote the "slow growth" quadrature family formed by applying the Novak and Ritter constraint to the Patterson rules.

Table 4 compares the point counts for the two Gauss-Patterson-based quadrature families. In this case, the advantages of using the GPS rule are already evident at level 2, and strongly persist through dimension 10.

Table 3: GLE/GLL/GLO sparse grid point counts in 2D, 6D, 10D

ℓ = level	0	1	2	3	4	5	6	7	8	9	10
Dimension 2											
n_ℓ (GLE)	1	5	21	73	221	609	1,573	3,881	9,261	21,553	49,205
n_ℓ (GLL)	1	5	13	29	53	89	137	201	281	381	501
n_ℓ (GLO)	1	5	9	17	33	45	81	97	161	181	281
Dimension 6											
n_ℓ (GLE)	1	13	109	713	3,953	19,397	86,517	357,153	1,382,361	5,065,693	17,709,469
n_ℓ (GLL)	1	13	85	389	1,433	4,541	12,841	33,193	79,729	180,077	385,901
n_ℓ (GLO)	1	13	73	257	737	1,925	4,509	9,837	20,445	40,025	75,917
Dimension 10											
n_ℓ (GLE)	1	21	261	2,441	18,881	126,925	764,365	4,208,385	21,493,065	102,935,845	466,201,781
n_ℓ (GLL)	1	21	221	1,581	8,761	40,405	162,025	581,385	1,904,465	5,778,965	?
n_ℓ (GLO)	1	21	201	1,201	5,281	19,165	61,285	177,525	474,885	1,192,425	2,835,589

Table 4: GPE/GPS sparse grid point counts in 2D, 6D, 10D

ℓ = level	0	1	2	3	4	5	6	7	8	9	10
Dimension 2											
n_ℓ (GPE)	1	5	17	49	129	321	769	1,793	4,097	9,217	20,481
n_ℓ (GPS)	1	5	9	17	33	33	65	97	97	161	161
Dimension 6											
n_ℓ (GPE)	1	13	97	545	2,561	10,625	40,193	141,569	471,041	1,496,065	4,571,137
n_ℓ (GPS)	1	13	73	257	737	1,889	4,161	8,481	16,929	30,689	53,729
Dimension 10											
n_ℓ (GPE)	1	21	241	2,001	13,441	77,505	397,825	1,862,145	8,085,505	32,978,945	127,574,017
n_ℓ (GPS)	1	21	201	1,201	5,281	19,105	60,225	169,185	434,145	1,041,185	2,347,809

Table 5: CCS/GLO/GPS point counts in 2D, 6D, 10D

ℓ = level	0	1	2	3	4	5	6	7	8	9	10
Dimension 2											
n_ℓ (CCS)	1	5	13	29	49	81	129	161	225	257	385
n_ℓ (GLO)	1	5	9	17	29	41	65	81	121	141	201
n_ℓ (GPS)	1	5	9	17	33	33	65	97	97	161	161
Dimension 6											
n_ℓ (CCS)	1	13	85	389	1,409	4,289	11,473	27,697	61,345	126,401	244,289
n_ℓ (GLO)	1	13	73	257	737	1,925	4,509	9,837	20,445	40,025	75,917
n_ℓ (GPS)	1	13	73	257	737	1,889	4,161	8,481	16,929	30,689	53,729
Dimension 10											
n_ℓ (CCS)	1	21	221	1,581	8,721	39,665	155,105	536,705	1,677,665	4,810,625	12,803,073
n_ℓ (GLO)	1	21	201	1,201	5,281	19,165	61,285	177,525	474,885	1,192,425	2,835,589
n_ℓ (GPS)	1	21	201	1,201	5,281	19,105	60,225	169,185	434,145	1,041,185	2,347,809

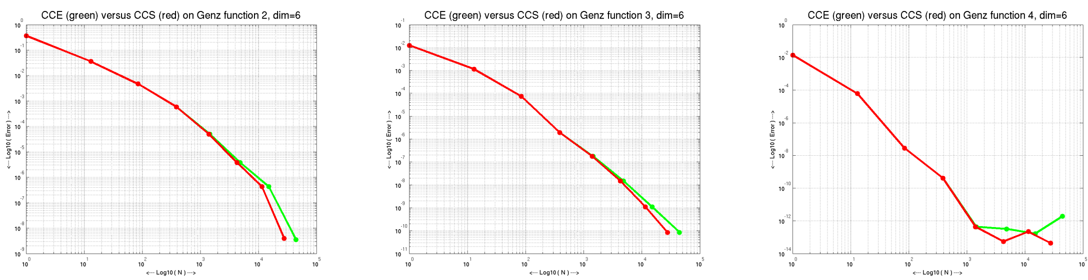

Figure 1: **CCE (green) vs CCS (red) on Genz Functions**
Error decay for Clenshaw-Curtis Exponential (CCE) and Clenshaw-Curtis Slow (CCS) sparse grids on Genz product peak, corner peak, and gaussian integrands, Dimension = 6.

Thus, the Novak and Ritter guidelines have suggested improved versions of the CCE, GLE and GPE families for use in sparse grid construction. Since these families all represent approaches to the quadrature problem, table 5 compares the point counts.

6 Numerical Examples

Genz[2] prescribed a number of test integrands for multidimensional quadrature. Given two sparse grid families, a comparison can be made by applying successive elements of each family to the test integrand, and observing the relationship between the number of function evaluations required and the quadrature error.

Figures [1], [2] and [3] display the results of such comparisons between the CCE and CCS, GLO and GLS, and GPE and GPS families, for quadrature in a space of dimension 6. For the GLO versus GLS comparison, there is a consistent advantage for the slow growth family. The advantage is not so clear for the GPE versus GPS comparison.

Figure [4] comparse the CCS, GLO and GPS families.

7 Conclusion

Smolyak's sparse grid procedure is widely used for computations in moderate and high dimension, and was devised to avoid the exponential growth in point count incurred by a straight-forward product rule approach.

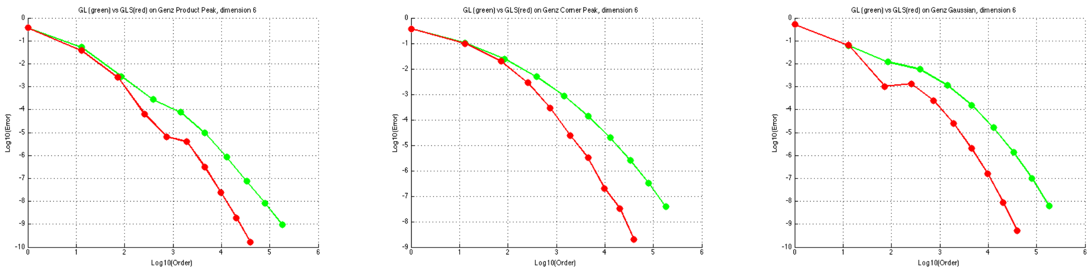

Figure 2: **GLO (green) vs GLS (red) on Genz Functions**
Error decay for Gauss-Legendre Odd (GLO) and Gauss-Legendre Slow (GLS) sparse grids on Genz product peak, corner peak, and gaussian integrands, Dimension = 6.

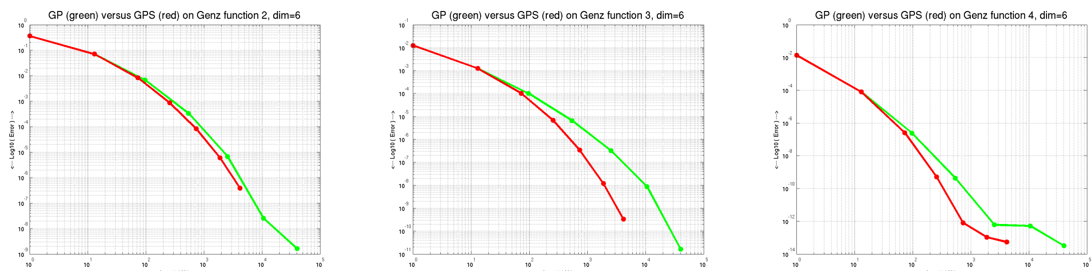

Figure 3: **GPE (green) vs GPS (red) on Genz Functions**
Error decay for Gauss-Patterson Exponential (GPE) and Gauss-Patterson Slow (GPS) sparse grids on Genz product peak, corner peak, and gaussian integrands, Dimension = 6.

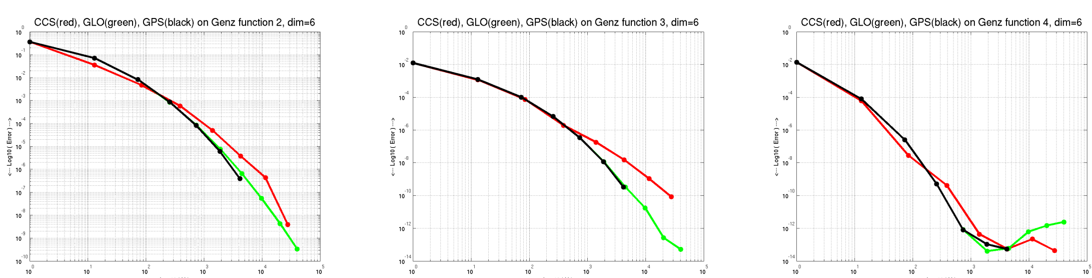

Figure 4: **CCS (red) vs GLO (green) vs GPS (black) on Genz Functions**
Error decay comparison for CCS/GLO/GPS families, Dimension 6

It is thus somewhat surprising that, in the most common example of a sparse grid, the 1D quadrature family is designed to exhibit exponential point growth. It turns out that this peculiarity is rarely an issue, since the 1D rules of very large order play very little role in practical high dimensional problems.

However, the bigger 1D rules do come into play in lower dimensional sparse grids. Since such sparse grids are also of importance, the methods discussed here offer an approach that can detect unnecessary growth in the 1D quadrature rules, and indicate how to adjust the sequence of rules according to the Novak and Ritter criterion. No new quadrature rules need to be devised; rather, some rules can be re-used for several steps of the sparse grid construction process.

The tables suggest that the benefit of this approach varies widely depending on the particular family, the spatial dimension, and the range of sparse grid levels of interest. The numerical results confirm that the adjusted rules perform with the desired accuracy.

Some publicly available software for sparse grid computations includes versions of the slow-growth strategy described here. In particular, Petras [8] presents a C program called *smolpak*, which implements slow-growth sparse grid construction for the nested Kronrod-Patterson family. Heiss and Winschel [4] describe a Matlab package called *nwspgr* that includes slow-growth sparse grid construction for quadrature over the unit hypercube, using either Gauss-Legendre or nested Kronrod-Patterson families; the package also includes slow-growth options for the Hermite weight over $\mathbb{R}^n$, using both the standard Gauss-Hermite or a nested Patterson-type family developed by Genz and Keister [3]. Recently, a C++ package known as *tasmanian* [10] has been developed at Oak Ridge National Laboratory, supporting the construction of sparse grids using the slow-growth option for several quadrature families.

The original Smolyak construction was a fundamental breakthrough in efficient analysis of high-dimensional problems. This discussion shows that, particularly in lower dimensions, the standard procedure can be usefully modified to produce sparse grids of equivalent exactness at a much reduced cost in terms of function evaluations. These results, discussed only in the isotropic case, also have implications for computations in higher dimensions, since the anisotropic approach to such problems essentially transforms them to a kind of lower dimensional problem in which the same efficiency improvements could be expected.

References

[1] Charles Clenshaw, Alan Curtis, A Method for Numerical Integration on an Automatic Computer, Numerische Mathematik, Volume 2, Number 1, December 1960, pages 197-205.

[2] Alan Genz, A package for testing multiple integration subroutines, in Numerical Integration: Recent Developments, Software and Applications, edited by Patrick Keast, Graeme Fairweather, Reidel, 1987, pages 337-340.

[3] Alan Genz, Bradley Keister, Fully symmetric interpolatory rules for multiple integrals over infinite regions with Gaussian weight, Journal of Computational and Applied Mathematics, Volume 71, 1996, pages 299-309.

[4] Florian Heiss, Viktor Winschel, Likelihood approximation by numerical integration on sparse grids, Journal of Econometrics, Volume 144, Number 1, May 2008, pages 62-80.

[5] Erich Novak, Klaus Ritter, High dimensional integration of smooth functions over cubes, Numerische Mathematik, Volume 75, Number 1, November 1996, pages 79-97.

[6] Erich Novak, Klaus Ritter, Richard Schmitt, Achim Steinbauer, Simple cubature formulas with high polynomial exactness, Constructive Approximation, Volume 15, Number 4, December 1999, pages 499-522.

[7] Thomas Patterson, The optimal addition of points to quadrature formulae, Mathematics of Computation, Volume 22, Number 104, October 1968, pages 847-856.

[8] Knut Petras, Smolyak cubature of given polynomial degree with few nodes for increasing dimension, Numerische Mathematik, Volume 93, Number 4, February 2003, pages 729-753.

[9] Sergey Smolyak, Quadrature and interpolation formulas for tensor products of certain classes of functions, Doklady Akademii Nauk SSSR, Volume 4, 1963, pages 240-243.

[10] Miroslav Stoyanov, User Manual: TASMANIAN Sparse Grids, ORNL Report, August 2013, Oak Ridge National Laboratory.