On the Precision of Certain Multidimensional Isotropic Quadrature Schemes

John Burkardt
Interdisciplinary Center for Applied Mathematics
& Information Technology Department
Virginia Tech .....
http://people.sc.fsu.edu/~jburkardt/presentations/sgmga_precision.pdf

July 27, 2010

1 Introduction

This note is intended to summarize some data about the precision of various multidimensional quadrature rules. The real aim of this investigation is to justify the claim that a sparse grid is "better" than a product rule, at least if the spatial dimension is high enough.

In this context, one rule is better than another if, for a given precision and spatial dimension, it requires fewer function evaluations, a quantity which we will symbolize by $\mathbf{N}$. We expect that a sparse grid rule will always "eventually" be better than a product rule, by which we mean that there is a crossover dimension $\mathbf{M_0}$ and precision $\mathbf{P_0}$ so that the sparse grid is better than the product rule for any pair $(\mathbf{M}, \mathbf{P})$ for which both $\mathbf{M_0} \leq \mathbf{M}$ and $\mathbf{P_0} \leq \mathbf{P}$.

Since there are many ways of constructing product grids and sparse grids, it is not surprising that this data is hard to track down. Here, we concentrate on some simple, popular cases, and work out the details in tables for moderate dimensions and precisions.

2 Multidimensional Precision

A 1D quadrature rule is said to have *precision* P if, for any polynomial $p(x)$ of degree $\mathbf{P}$ or less, the estimated integral produced by the quadrature rule is exact. By linearity of integration and quadrature, it is sufficient to verify that a quadrature rule produces the exact result for the $\mathbf{P+1}$ monomials 1, x, x^2, ..., x^P.

Similarly, an $\mathbf{M}$-dimensional quadrature rule has precision $\mathbf{P}$ if, for any polynomial $p(\vec{x})$ of total degree $\mathbf{P}$ or less, the estimated integral produced by the quadrature rule is exact. The total degree of a polynomial in the $\mathbf{M}$-dimensional variable $\vec{x}$ is the maximum of the sums of the exponents of the individual terms. Thus, the total degree of $x^3y^2z + x^5yz^4 + y^3$ is 10. Again, to verify that a multidimensional quadrature rule has precision $\mathbf{P}$, it is sufficient to check that the rule produces the exact integral of each monomial $\prod_{i=1}^{M} x_i^{e_i}$ for which $\sum_{i=1}^{M} e_i \leq P$.

3 Multidimensional Monomials of Total Degree P

To measure multidimensional precision, you have to generate multidimensional monomials. The number of monomials of total degree $\mathbf{P}$ increases rapidly with the spatial dimension $\mathbf{M}$. A formula for the number of

M-dimensional monomials of total degree exactly equal to **P** is

$$\frac{(P+M-1)!}{P!(M-1)!}$$

This formula is simply counting the number of partitions of **P** into **M** ordered parts.

To justify this formula, it is useful to turn to the theory of permutations. We can represent any **M**-dimensional monomial of total degree **P** as a sequence of **P** 1's and **M-1** divider symbols |. For instance, the monomial x^5yz^4, of total degree 10, would be represented by **11111|1|1111**. This representation creates a 1-to-1 mapping between the monomials and strings of **1**'s and |'s. But such strings are easy to count using the standard formula for the number of permutations of **P+M-1** objects with **P** objects of one indistinguishable type, and **M-1** of another.

Here is a table of the number of **M**-dimensional monomials of degree exactly **P**:

P \ M	1	2	3	4	5	6	7	8	9	10
0	1	1	1	1	1	1	1	1	1	1
1	1	2	3	4	5	6	7	8	9	10
2	1	3	6	10	15	21	28	36	45	55
3	1	4	10	20	35	56	84	120	165	220
4	1	5	15	35	70	126	210	330	495	715
5	1	6	21	56	126	252	462	792	1,287	2,002
6	1	7	28	84	210	462	924	1,716	3,003	5,005
7	1	8	36	120	330	792	1,716	3,432	6,435	11,440
8	1	9	45	165	495	1,287	3,003	6,435	12,870	24,310
9	1	10	55	220	715	2,002	5,005	11,440	24,310	48,620
10	1	11	66	286	1,001	3,003	8,008	19,448	43,758	92,378
11	1	12	78	364	1,365	4,368	12,376	31,824	75,582	167,960

Note that this table is symmetric in the sense that column **M** matches row **P-1**.

We can usefully consider a few entries of the table much further out. In particular, for M=100, the number of monomials of the first few total degrees is 1, 100, 5050, and 171700. Similarly, for degree **P**=99, the number of monomials in the first few dimensions is 1, 100, 5050 and 171700.

Finally, note the very interesting fact that the growth factor from column **M** to **M+1** is the multiplier $\frac{P+M}{M}$, so that, as you move along row **P**, the growth factor is tending to 1. This is a remarkable feature.

4 Multidimensional Monomials of Total Degree P or Less

The number of monomials of total degree less than or equal to **P** can be derived by allowing one more divider which can be thought of as storing the amount by which the monomial's degree is less than **P**. Thus, if we were considering monomials of degree 10 or less, the representation for x^3y^2z would be **111|11|1|1111**. And again applying the formula for permutations of objects, some of which are indistinguishable, we have that the number of monomials of total degree less than or equal to **P** is

$$\frac{(P+M)!}{P!M!}$$

Thus, for $M = 100$, the number of monomials up to a given total degree is 1, 101, then 5151.

Moreover, the alert observer will note that the following table, which counts the monomials of degree up to and including P, can be derived from the previous table simply by dropping the first column.

P \ M	1	2	3	4	5	6	7	8	9	10
0	1	1	1	1	1	1	1	1	1	1
1	2	3	4	5	6	7	8	9	10	11
2	3	6	10	15	21	28	36	45	55	66
3	4	10	20	35	56	84	120	165	220	286
4	5	15	35	70	126	210	330	495	715	1,001
5	6	21	56	126	252	462	924	1,716	2,002	3,003
6	7	28	84	210	462	924	1,716	3,003	5,005	8,008
7	8	36	120	330	792	1,716	3,432	6,435	11,440	19,448
8	9	45	165	495	1,287	3,003	6,435	12,870	24,310	43,758
9	10	55	220	715	2,002	5,005	11,440	24,310	48,620	92,378
10	11	66	286	1,001	3,003	8,008	19,448	43,758	92,378	184,756
11	12	78	364	1,365	4,368	12,376	31,824	75,582	167,960	352,716

Note that this table is symmetric in the sense that column **M**, after skipping the first entry, matches row **P**.

Just as for the previous table, the rate of growth as you move along row **P** actually slows down. The multiplier to go from column **M** to **M+1** is $\frac{P+M+1}{M+1}$, so that the growth factor is tending to 1.

The number of monomials of total degree **P** or less is a rough measure of the inherent difficulty involved in producing a quadrature rule that will have precision **P**.

5 Products of the Clenshaw-Curtis Rule

Our first multidimensional rule will be formed as a product of the 1D Clenshaw-Curtis rule ("CC").

For the 1D case, the CC rule of order **N** has a precision of

$$P(\text{CC}) = \begin{cases} N-1 & \text{if } N \text{ is even;} \\ N & \text{if N is odd.} \end{cases}$$

If we employ the standard product rule construction to create a CC product rule of order N^M, the precision result is the same:

$$P(\text{CC}^M) = \begin{cases} N-1 & \text{if } N \text{ is even;} \\ N & \text{if N is odd.} \end{cases}$$

We now ask the following question: *If we wish to obtain a CC product rule of precision* **P***, how many points* **N** *will be required in our quadrature rule?*

P \ M	1	2	3	4	5
1	1	1	1	1	1
3	3	9	27	81	243
5	5	25	125	625	3,125
7	7	49	343	2,401	16,807
9	9	81	729	6,561	59,049
11	11	121	1,331	14,641	161,05

P \ M	6	7	8	9	10
1	1	1	1	1	1
3	729	2187	6,561	19,683	59,049
5	15,625	78,125	390,625	1,953,125	9,765,625
7	117,649	823,543	5,764,801	40,353,608	282,475,264
9	531,441	4,782,969	43,046,720	387,420,480	3,486,784,320
11	1,771,561	19,487,172	214,358,881	2,357,947,691	25,937,424,601

Of course, the growth factor for row **P**, to get from column **M** to **M+1**, is **P**. This does not decrease with increasing **M**, and of course it gets much worse for increasing **P**.

6 Products of the Gauss-Legendre Rule

A very common 1D quadrature rule is known as the Gauss-Legendre rule ("GL"). It is often preferred for computations in which high accuracy is desired, since it achieves about twice the precision of standard interpolatory rules.

For the 1D case, the GL rule of order **N** has a precision of

$$P(\mathrm{GL}) = 2 * N - 1.$$

If we employ the standard product rule construction to create a GL product rule of order N^M, the precision result is the same:

$$P(\mathrm{GL}^M) = 2 * N - 1.$$

The precision table for Gauss-Legendre:

P \ M	1	2	3	4	5	6	7	8	9	10
1	1	1	1	1	1	1	1	1	1	1
3	2	4	8	16	32	64	128	256	512	1024
5	3	9	27	81	243	729	2,187	6,561	19,683	59,049
7	4	16	64	256	1024	4096	16,384	65,536	262,144	1,048,576
9	5	25	125	625	3,125	15,625	78,125	390,625	1,953,125	9,765,625
11	6	36	216	1,296	7,776	46,656	279,936	1,679,616	10,077,696	60,466,176
13	7	49	343	2,401	16,807	117,649	823,543	5,764,801	40,353,607	282,475,249
15	8	64	512	4,096	32,768	262,144	2,097,152	16,777,216	134,217,728	1,073,741,824

The improvement over the CC rule is stunning. However, this relates directly to the fact that we are essentially comparing P^M and $(P+1)^M/2^M$. We are still looking at exponential growth, but the base is about half as large.

7 Clenshaw-Curtis Sparse Grids

The standard method of generating a multidimensional sparse grid from the 1D Clenshaw-Curtis rule takes full advantage of the nestedness of the rule. It does so by prescribing that the sequence of 1D rules being used show grow exponentially in order. Thus, the selected 1D rules rise rapidly in order, so that, formally, a 1D sparse grid of precision 15 uses 129 points. Actually, of course, in 1D a Clenshaw Curtis rule of order 15 would be sufficient to reach precision 15.

If we look at a row of the table, we see that the excessive number of points used in dimension 1 is moderated by a very slow growth rate with increasing dimension. This guarantees that, for a particular precision level, the product rule formulas will quickly surpass the sparse grid in terms of number of points required.

P \ M	1	2	3	4	5	6	7	8	9	10
1	1	1	1	1	1	1	1	1	1	1
3	3	5	7	9	11	13	15	17	19	21
5	5	13	25	41	61	85	113	145	181	221
7	9	29	69	137	241	389	589	849	1,177	1,581
9	17	65	177	401	801	1,457	2,465	3,937	6,001	8,801
11	33	145	441	1,105	2,433	4,865	9,017	15,713	26,017	41,265
13	65	321	1,073	2,929	6,993	15,121	30,241	56,737	100,897	171,425
15	129	705	2,561	7,537	19,313	44,689	95,441	190,881	361,249	652,065
17	257	1,537	6,017	18,945	51,713	127,105	287,745	609,025	1,218,049	2,320,385

8 Delayed Clenshaw-Curtis Sparse Grids

We observed in the previous section that the Clenshaw-Curtis sparse grid tables had a peculiar behavior in column 1, where a much lower order rule would be sufficient to achieve the desired accuracy. We presume that an inflated order in column 1 influences all the values in the same row.

This observation raises the question of whether it is possible to further improve the performance of the sparse grid procedure by controlling the excessive growth in the number of points used in the 1D case. Presumably, if we can control the first column of the table, the rows will behave as well, since the rate of growth along a row seems to be very moderate.

The proposal for treating this issue is sometimes called a *delayed sparse grid*. Essentially, this approach tries to retain the advantages of nestedness, while avoiding the costs of relentless exponential growth. This is done, roughly speaking, by reusing a given 1D rule until it would no longer satisfy the precision requirement, and only then jumping to the next (exponentially larger) rule.

To see how this would work, suppose that column 1 of the above table was replaced by the values 1, 3, 5, 9, 9, 17, 17, 17, 17. This would mean that for precision 17, dimension 1, we would use 17 points rather than 257, and that, presumably, the entries in the precision 17 row would all drop, perhaps by a linear factor of more than 10.