UNIVERSITY OF Southampton
Engineering and the Environment
Aeronautics, Astronautics and Computational Engineering

Application of tensor-product adaptive sparse quadrature to uncertainty quantification

UTC for Computational Engineering
A. Makrodimopoulos, A.J. Keane, Faculty of Engineering and the Environment
R. Bates, Rolls-Royce plc.

Sparse quadrature & uncertainty quantification

The calculation of the statistical moments is related to the evaluation of multiple integrals. I.e. assuming a function of d variables varying independently in $\Delta = [a_1, b_1] \times \ldots \times [a_d, b_d]$, with $\xi_i \in [a_i, b_i]$, the mean and the variance are given by

$$\mu = \int_{a_1}^{b_1} \ldots \int_{a_d}^{b_d} f(\xi_1, \ldots, \xi_d) p(\xi_1, \ldots, \xi_d)\, \mathrm{d}\xi_1 \ldots \mathrm{d}\xi_d = \int_\Delta f(\boldsymbol{\xi}) p(\boldsymbol{\xi})\, \mathrm{d}\boldsymbol{\xi}$$

$$\sigma^2 = \int_{a_1}^{b_1} \ldots \int_{a_d}^{b_d} \left(f(\xi_1, \ldots, \xi_d) - \mu\right)^2 p(\xi_1, \ldots, \xi_d)\, \mathrm{d}\xi_1 \ldots \mathrm{d}\xi_d = \int_\Delta \left(f(\boldsymbol{\xi})\right)^2 p(\boldsymbol{\xi})\, \mathrm{d}\boldsymbol{\xi} - \mu^2$$

where p is the distribution function.

Uncertainty quantification using numerical quadrature and tensor-product adaptivity

It is apparent that the calculation of the statistical moments is related to the evaluation of general multidimensional integrals. In numerical quadrature the calculation of a general integral reads:

$$I = \int_\Delta f(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, \mathrm{d}\boldsymbol{\xi} = w_1 f(\boldsymbol{\xi}^{(1)}) + \ldots + w_n f(\boldsymbol{\xi}^{(n)})$$

where $W(\boldsymbol{\xi})$ is the weight function, n is the number of quadrature points, $\boldsymbol{\xi}^{(1)}, \ldots, \boldsymbol{\xi}^{(n)}$ are their abscissas and $w_1, \ldots, w_n$ are their weights. All the above depend on the choice of the quadrature rule for the given weight function, the desired level of accuracy and the type of multidimensional quadrature, i.e. if it is full or sparse quadrature. In the first case the final grid and the weights are the tensor product of the points and the weights of the one dimensional rule. This leads to a high number of quadrature points. Sparse quadrature is based on Smolyak's algorithm where the final grid is produced by the superposition of various tensor products of parts of the one dimensional quadrature rule. For given polynomial accuracy, using sparse quadrature requires significantly less points than in full quadrature (see e.g. Makrodimopoulos et. al.; 2011).

Moreover, for moderate dimension (e.g. 15) and reasonable polynomial accuracy (e.g. 5-7) the resulting number of points can be around 500, much less than the number of evaluations needed when we apply Monte-Carlo (around 3,000-5,000). The question is whether the number of function evaluations can be reduced even further. As some variables contribute less than others, we can take into account the special features of the examined function. Indeed Gerstner and Griebel (2003) have shown that for nested quadrature rules (e.g. Clenshaw-Curtis and Gauss-Patterson), we can select only the tensor products which are more likely to contribute. Very briefly they suggested to start with the lowest level of accuracy (usually this is level zero), evaluate the function at the current quadrature points and then add only the points of the new tensor products which are the "children" of the tensor product with the highest contribution to the calculation of the integral. This tensor product will be called "parent". The function is evaluated at the new points and the integral is recalculated. The procedure is repeated and stops when the contribution of all children becomes negligible.

As the definition of "negligible" is rather vague here we considered that the ratio of the summation of the absolute values of the contribution of all children to the current absolute value of the integral should be less than 10^{-3}. Further criteria are also employed for nearly zero integrals. Moreover, as uncertainty quantification consists of the calculation of two different integrals, the whole process was adapted as shown in Fig. 1. Our code was implemented in MATLAB. As a test we considered the plane strain structure of Fig. 2 where the uncertainty parameters are the positions of the two holes with $-0.30 \leq \delta x, \delta y \leq 0.30$. The four uncertainty parameters are distributed uniformly and the maximum von Mises stress is the quantity of interest. Fig. 3 shows the advantage of using sparse quadrature in relation to Monte-Carlo and Fig. 4 shows that we can gain further benefits by using the aforementioned tensor-product adaptive sparse quadrature procedure. We see that with only 177 evaluations we get results comparable to those obtained by level 4 (769 function evaluations).

Acknowledgement

This work is funded by the Strategic Investment in Low-carbon Engine Technology (SILOET) program.

References

A. Makrodimopoulos, A.J. Keane and R. Bates (2011). Implementation of sparse quadrature in IsightTM & some further results. Poster presentation in *SILOET annual meeting of Rolls-Royce*, Derby, UK, October 2011.
T. Gerstner and M. Griebel (2003). Dimension-adaptive tensor-product quadrature. *Computing*, 71, 65-87.

```
set purpose = 'mean'
get the grid for level = 0
calculate I,J
set criteria = false
while criteria=false
    if purpose='mean'
       find ICmax(I)
    end
    if purpose = 'variance
       find ICmax(J)
    end
    generate the children of Icmax
    check the acceptance of the new children
    update the grid and the weights
    update I, J, Ichildren, Jchildren
    if purpose = 'mean' & convergence(I) = true
       set purpose = 'variance'
    end
    if purpose = 'variance' & if convergence(J) = true
       criteria = true
    end
end
mean = I
variance = J - (mean^2)
```

SOME DEFINITIONS - EXPLANATIONS

By `I,J` we mean $I = \int_\Delta f(\boldsymbol{\xi}) p(\boldsymbol{\xi})\, \mathrm{d}\boldsymbol{\xi}$, $J = \int_\Delta \left(f(\boldsymbol{\xi})\right)^2 p(\boldsymbol{\xi})\, \mathrm{d}\boldsymbol{\xi}$

`ICmax(I)` = tensor product with the highest contribution to the integral `I`
`Ichildren` = the sum of the absolute values of the contributions of the children tensor products to the integral `I`
The convergence of `I` depends on its current value and the value of `Ichildren(I)`
`ICmax(J)`,`Jchildren` are defined analogously for the integral `J`

Fig. 1 Pseudocode for the application of tensor-product adaptive sparse quadrature to uncertainty quantification.

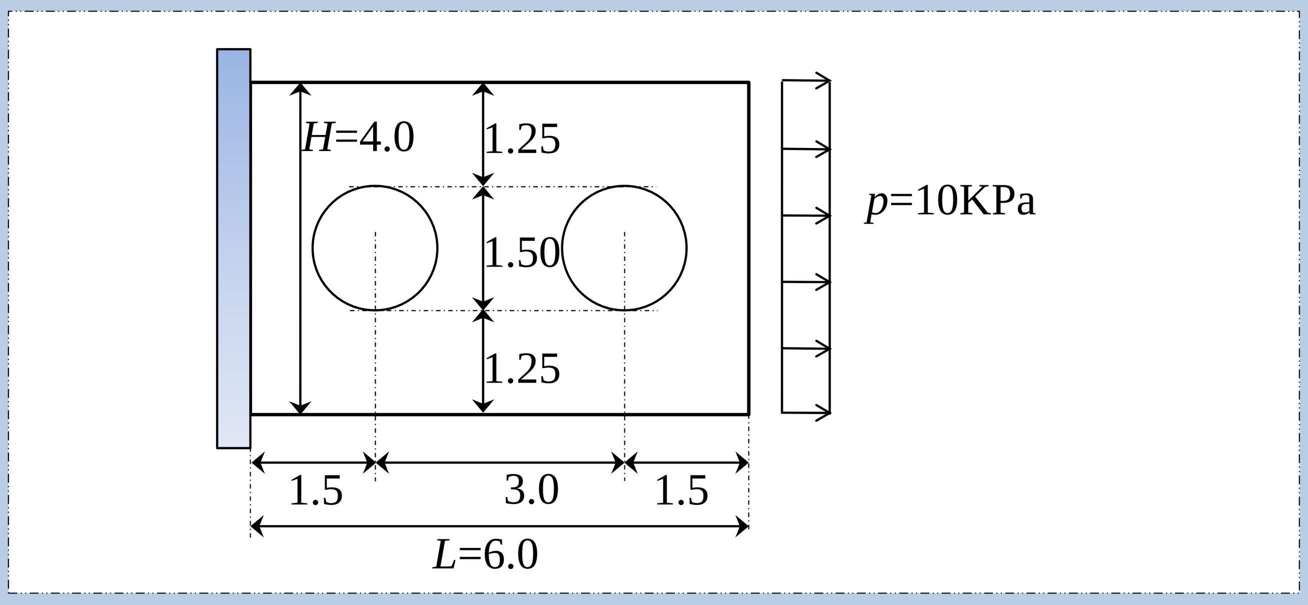

Fig. 2 Numerical application: plane strain structure with two holes. The position of the holes (x,y coordinates) are the parameters of the problem.

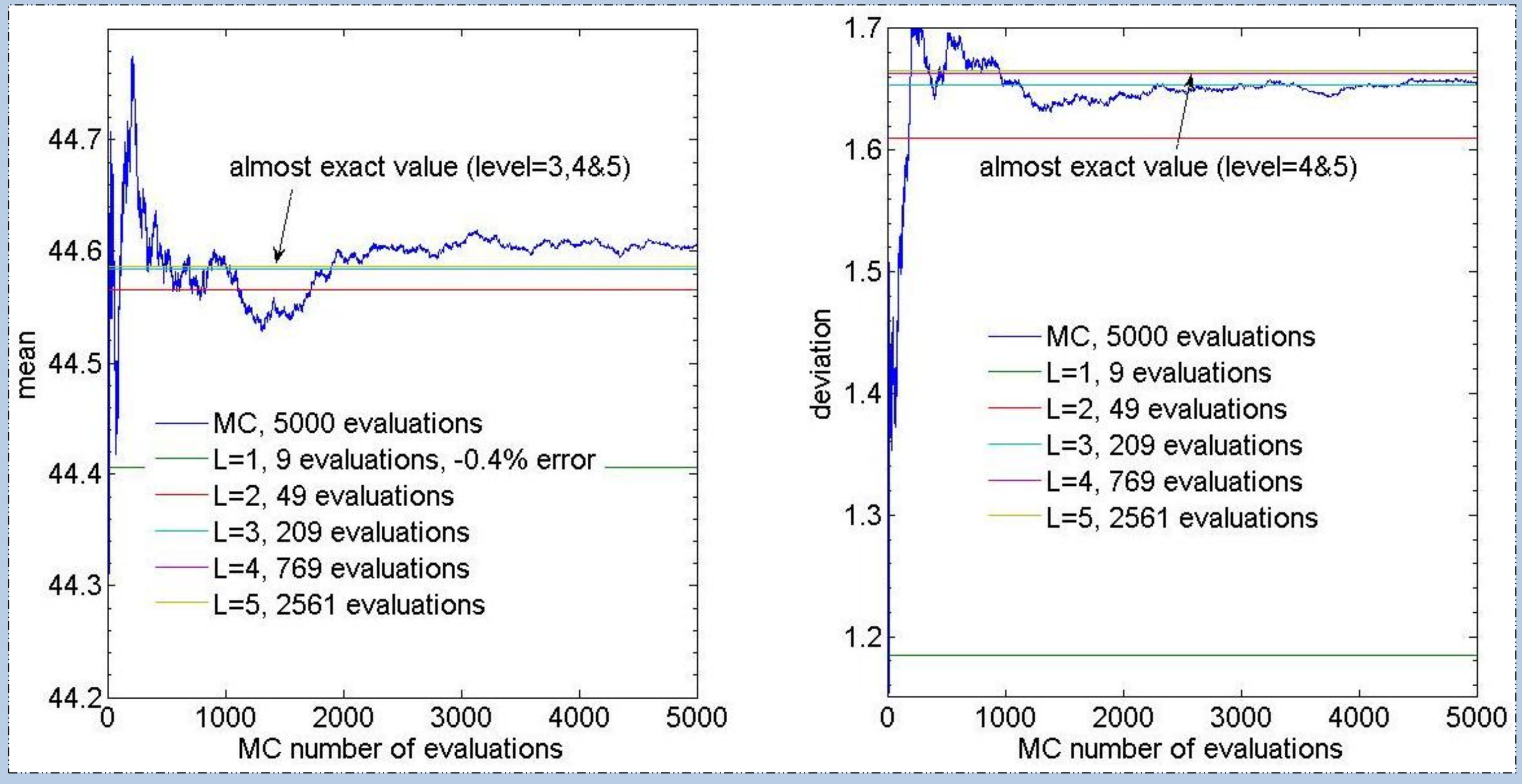

Fig. 3 Comparison of Monte-Carlo (MC) with sparse quadrature (Gauss-Patterson quadrature rule), for various levels (L) of accuracy. Sparse quadrature converges faster to the exact result.

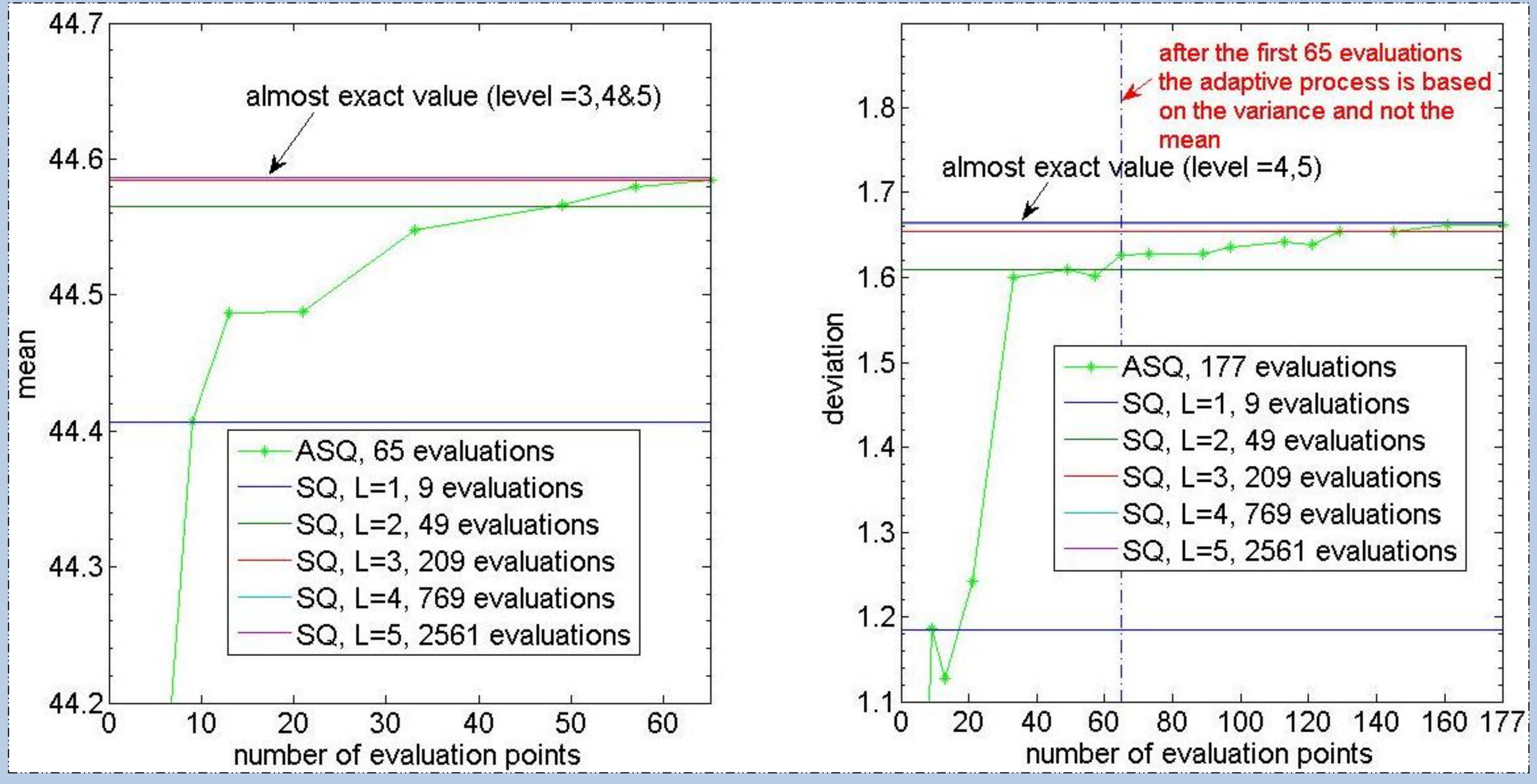

Fig. 4 Application of tensor-product adaptive sparse quadrature (ASQ) using Gauss-Patterson rule. The advantages in relation to the standard one are more clear in the calculation of the standard deviation.

http://www.soton.ac.uk/engineering/research/groups/CED/posters.page | email: am1a06@soton.ac.uk
Computational Engineering & Design Group, University of Southampton, SO17 1BJ, U.K.