

The application of sparse grid quadrature in solving stochastic optimisation problems

Yuancheng Zhou¹

Markus Hegland²

(Received 27 February 2019; revised 3 June 2019)

Abstract

Stochastic optimisation problems minimise expectations of random cost functions. Thus they require accurate quadrature methods in order to evaluate the objective. Promising methods based on sparse grids were shown to display high quadrature accuracy for smooth integrands. But they have negative quadrature weights which potentially destroy the convexity of the objective and thus may lead to totally wrong results. We prove here that, due to their high accuracy, sparse grids maintain the convexity of the objective for sufficiently fine grids. An application to optimal control demonstrates the superiority of sparse grids over Monte Carlo and product rule based approaches.

[doi:10.21914/anziamj.v60i0.14060](https://doi.org/10.21914/anziamj.v60i0.14060) gives this article, © Austral. Mathematical Soc. 2019. Published June 17, 2019, as part of the Proceedings of the 18th Biennial Computational Techniques and Applications Conference . ISSN 1445-8810. (Print two pages per sheet of paper.) Copies of this article must not be made otherwise available on the internet; instead link directly to the DOI for this article.

Contents

1	Introduction	C17
2	Sparse grid quadrature rules	C18
3	Convexity of the sparse grid surrogate	C23
4	Application to stochastic control	C26
5	Conclusions	C31

1 Introduction

A stochastic optimisation problem is an optimisation problem of the form

$$\min_{\mathbf{u} \in \mathbf{U}} \mathbb{E} [h(\mathbf{u}, \mathbf{W})], \quad (1)$$

where \mathbf{W} is a high-dimensional real random vector, the cost function $h(\cdot, \cdot)$ is smooth and convex, and the domain $\mathbf{U} \subset \mathbb{R}^d$, with positive integer d , is convex and compact. Problems of this form occur in many applications related to complex decision making in finance, engineering and statistics [10].

If the random vector \mathbf{W} admits a probability density $p(\mathbf{w})$ on \mathbb{R}^d , then the objective is of the form

$$F(\mathbf{u}) := \mathbb{E} [h(\mathbf{u}, \mathbf{W})] = \int_{\mathbb{R}^d} h(\mathbf{u}, \mathbf{w}) p(\mathbf{w}) d\mathbf{w} = \int_{\mathbb{R}^d} f(\mathbf{u}, \mathbf{w}) d\mathbf{w}, \quad (2)$$

where $f(\mathbf{u}, \mathbf{w}) = h(\mathbf{u}, \mathbf{w}) p(\mathbf{w})$.

One approach to solving stochastic optimisation problems minimises a numerical approximation of the integral (2) by a suitable quadrature method and minimises this surrogate of the objective. In the stochastic programming

literature this is called the scenario generation method [10]. Because of the high-dimensionality of the integral, the Monte Carlo method is frequently used in stochastic optimisation. However, in many applications involving high-dimensional integrals it was shown that deterministic methods like the Quasi Monte Carlo method and sparse grid quadrature produce more accurate results [3, 7].

In Section 2 we discuss sparse grid quadrature for high-dimensional integrals. We consider methods based on the trapezoidal rule, the Clenshaw–Curtis method and the Gauss–Patterson rule. These methods are used widely and give good approximations for smooth integrands [2, 6, 7] which result in good approximations of the objective function of stochastic optimisation. A transformation of the integration domain to $[-1, 1]^d$ is required but this is unproblematic for most applications (it uses the cumulative distribution function). Sparse grids are typically better than Monte Carlo methods and preferred over Quasi Monte Carlo methods. However, sparse grids have the drawback that a convex objective may be approximated by a non-convex function. We consider this problem in Section 3 and prove that convexity is maintained for quadrature rules of sufficient accuracy. We also provide an example which suggests that convexity is maintained even for moderately sized sparse grids. In Section 4 we illustrate how our approach to stochastic optimisation is used for optimal control problems. We also provide a simple example which demonstrates the computational efficiency compared to Monte Carlo and product rules.

2 Sparse grid quadrature rules

Here we review numerical methods which approximate integrals of the form

$$I(f) = \int_{[-1,1]^d} f(\mathbf{w}) \, d\mathbf{w}, \quad (3)$$

for given functions f when the dimension \mathbf{d} is larger than three. These quadrature rules are based on methods for the case $\mathbf{d} = 1$. A challenge facing the product rule, which is commonly used for lower dimensions \mathbf{d} , is the curse of dimensionality, but this issue is addressed by sparse grids. A thorough discussion of quadrature rules, including those based on sparse grids, can be found in the literature [2, 6, 7].

Sparse grid quadrature is built upon a sequence of one-dimensional rules. The rules considered here are based either on the trapezoidal rule, the Clenshaw–Curtis rule [4] or the the Gauss–Patterson rule [9]. The j th rule in this sequence is denoted as

$$Q_j(f) = \sum_{i=1}^{N_j} c_{j,i} f(w_{j,i}), \quad j = 1, 2, \dots, \quad (4)$$

where $c_{j,i}$ and $w_{j,i}$ are the i th weight and i th quadrature point of the j th rule, respectively. The equidistant trapezoidal rule with $N_j = 2^{j-1} + 1$ quadrature points has $O(4^{-j})$ accuracy for C^2 (twice continuously differentiable) functions f (the method is substantially more accurate for periodic functions). The Clenshaw–Curtis rule with $N_j = 2^{j-1} + 1$ quadrature points integrates polynomials of degree N_j exactly and has $O(2^{-jr})$ accuracy for C^r functions f [5, 6, 7] where $r \leq N_j$. The quadrature points of Clenshaw–Curtis are extremal points of a Chebyshev polynomial. The Gauss–Patterson rule is an extension of the Gauss–Kronrod scheme [9]. It has $N_j = 2^j - 1$ quadrature points and is exact for polynomials of degree $(3N_j - 1)/2$ [5, 6]. Its accuracy is $O(2^{-jr})$ for C^r functions where r is at most equal to the polynomial accuracy.

Multidimensional quadrature rules are constructed using the differences

$$\Delta_j(f) = Q_j(f) - Q_{j-1}(f), \quad j = 1, 2, \dots, \quad (5)$$

where $Q_0(f) = 0$. One then has, for level l ,

$$Q_l(f) = \sum_{j=1}^l \Delta_j(f). \quad (6)$$

The sets $\mathcal{G}_j = \{\mathbf{w}_{j,1}, \dots, \mathbf{w}_{j,N_j}\}$ of quadrature points are nested for the three families of methods considered, that is, $\mathcal{G}_j \subset \mathcal{G}_{j+1}$. Quadrature rules with this property are called hierarchical. For hierarchical quadrature rules the $\Delta_j(\mathbf{f})$ are linear combinations of function values $f(\mathbf{w})$ for $\mathbf{w} \in \mathcal{G}_j$. The absolute value of Δ_{l+1} is of the same order as the error of Q_l .

One approach to quadrature on the unit cube uses product rules $Q_{l_1} \otimes \dots \otimes Q_{l_d}$. For example, for $\mathbf{d} = 2$ one has

$$Q_{l_1} \otimes Q_{l_2}(\mathbf{f}) = \sum_{i_1=1}^{N_{l_1}} \sum_{i_2=1}^{N_{l_2}} c_{l_1, i_1} c_{l_2, i_2} f(\mathbf{w}_{l_1, i_1}, \mathbf{w}_{l_2, i_2}), \quad (7)$$

and similarly for $\mathbf{d} > 2$. The number of quadrature points required is then the product of the points used for the one-dimensional rules. Thus the computational complexity of the product rule is $O(N_l^{\mathbf{d}})$ for $\mathbf{l}_i = \mathbf{l}$. However, the accuracy of the product rules is of the same order $O(2^{-l\mathbf{r}})$ independent of the dimension. This is related to the curse of dimensionality.

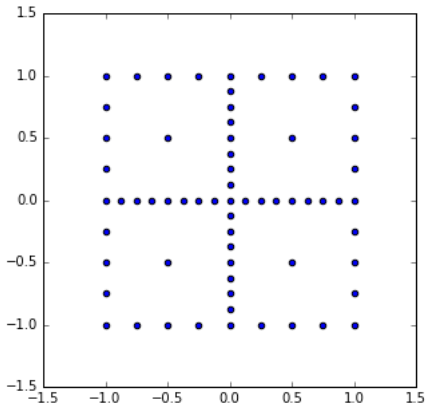
The product rules are recast in terms of the differences Δ_j as

$$Q_{l_1} \otimes \dots \otimes Q_{l_d}(\mathbf{f}) = \sum_{\mathbf{j} \leq \mathbf{l}} \Delta_{j_1} \otimes \dots \otimes \Delta_{j_d}(\mathbf{f}), \quad (8)$$

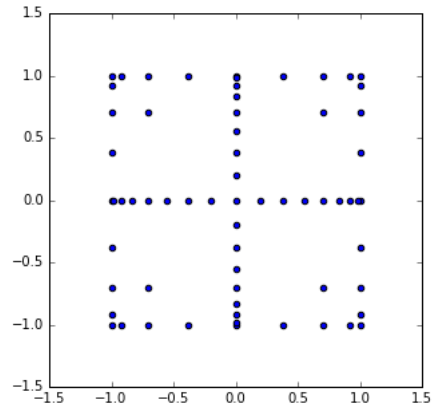
where vector $\mathbf{j} \leq \mathbf{l}$ means each of its element $j_k \leq l_k$ for $k = 1, \dots, \mathbf{d}$. Instead of using all the difference terms with index $\mathbf{j} \leq \mathbf{l}$, the sparse grid quadrature only sums up a subset of terms. The level \mathbf{l} classical sparse grid quadrature based on the hierarchical 1D rules is defined by

$$Q_{\mathbf{l}}^{\mathbf{d}}(\mathbf{f}) = \sum_{|\mathbf{j}|_1 \leq \mathbf{l} + \mathbf{d} - 1} \Delta_{j_1} \otimes \dots \otimes \Delta_{j_d}(\mathbf{f}), \quad (9)$$

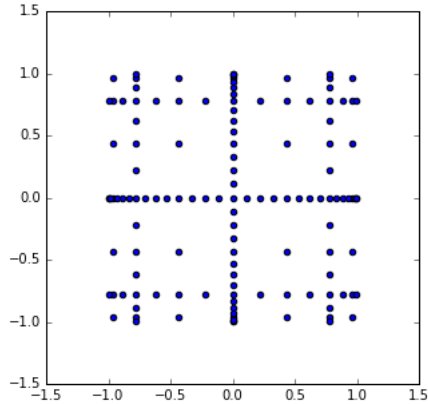
where $|\mathbf{j}|_1 = j_1 + \dots + j_d$. Although some difference terms have been omitted in the sparse grid quadrature formula (9) compared to the product rule (8), sparse grid methods can still achieve a satisfactory approximation since these dropped terms are usually very small when the integrand is smooth. If we



(a) trapezoidal rule



(b) Clenshaw–Curtis



(c) Gauss–Patterson

Figure 1: 2D sparse grids with respect to (a) the trapezoidal rule, (b) the Clenshaw–Curtis and (c) the Gauss–Patterson rules for level $l = 5$. More quadrature points are close to the boundary in the Clenshaw–Curtis and Gauss–Patterson rules.

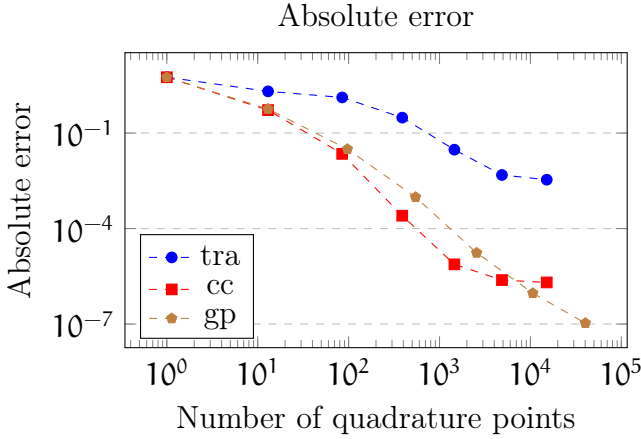


Figure 2: We compute the integral $\int_{[0,1]^d} \prod_{i=1}^d \exp(x_i) \, dx$ with $d = 6$ using sparse grids generated by the three different univariate quadrature rules. The Gauss–Patterson (gp) and Clenshaw–Curtis (cc) rules perform much better than the trapezoidal rule (tra).

assume the integrand f has bounded mixed derivatives up to order r , that is, $f \in H^r([-1, 1]^d)$ where

$$H^r([-1, 1]^d) = \left\{ f : [-1, 1]^d \rightarrow \mathbb{R} : \max_{|\alpha|_\infty \leq r} \left\| \frac{\partial^{|\alpha|_1} f}{\partial \alpha \mathbf{w}} \right\| < \infty \right\}, \quad (10)$$

where $\|\cdot\|$ denotes the L_2 norm, $|\alpha|_1 = \alpha_1 + \dots + \alpha_d$ and $|\alpha|_\infty = \max_j \alpha_j$, then the error of the sparse grid quadrature is $O(N^{-r}(\log N)^{(d-1)(r-1)})$ [2, 6, 7] where N is the number of the sparse grid quadrature points. We show three different types of sparse grid in Figure 1 and test their performances in Figure 2.

3 Convexity of the sparse grid surrogate

Since the function $h(\mathbf{u}, \mathbf{w})$ is convex, so is the objective $F(\mathbf{u}) = \mathbb{E}[h(\mathbf{u}, \mathbf{W})]$. Here we approximate this objective using sparse grid quadrature. Unlike Monte Carlo (MC) or Quasi Monte Carlo (QMC) methods, the quadrature weights of sparse grids can be negative. As a consequence, the surrogate $F_l(\mathbf{u})$ may no longer be convex. The surrogate is defined as

$$F_l(\mathbf{u}) = \sum_{i=1}^N c_i f(\mathbf{u}, \mathbf{w}_i), \quad (11)$$

where c_i and \mathbf{w}_i are the level l sparse grid weights and quadrature points, respectively. We choose sparse grids as they have superior approximation properties for smooth integrands compared to MC and QMC. We are currently working on how to address the above convexity change problem. Theorem 1 proves that for sufficiently large l , $F_l(\mathbf{u})$ is convex under certain conditions on h and the probability density $p(\mathbf{w})$ of the random vector \mathbf{W} .

In the following theorem we do not assume that $h(\cdot, \mathbf{w})$ or $f(\cdot, \mathbf{w})$ are convex for all \mathbf{w} . We use the following norm in $C^2[\mathbf{U}]$:

$$\|F\| = \sup_{\mathbf{u} \in \mathbf{U}} \left[|F(\mathbf{u})|^2 + DF(\mathbf{u})^T DF(\mathbf{u}) + \text{trace} (D^2F(\mathbf{u})^T D^2F(\mathbf{u})) \right]^{1/2}, \quad (12)$$

for $F \in C^2[\mathbf{U}]$, where DF is the gradient and D^2F the Hessian of F , respectively.

Theorem 1. *Let $\mathbf{U} \subset \mathbb{R}^n$ be convex and compact and $\Omega = [-1, 1]^n$, and let*

- (i) $F_l \in C^2(\mathbf{U})$ and $F \in C^2(\mathbf{U})$,
- (ii) $F_l \rightarrow F$ in the $C^2(\mathbf{U})$ norm for $l \rightarrow \infty$ and
- (iii) $D^2F(\mathbf{u}) > \gamma I$ for all $\mathbf{u} \in \mathbf{U}$ for some $\gamma > 0$ independent of \mathbf{u} .

Then there exists $l_0 > 0$ such that $F_l(\mathbf{u})$ is strictly convex for $\mathbf{u} \in \mathbf{U}$ and all $l \geq l_0$.

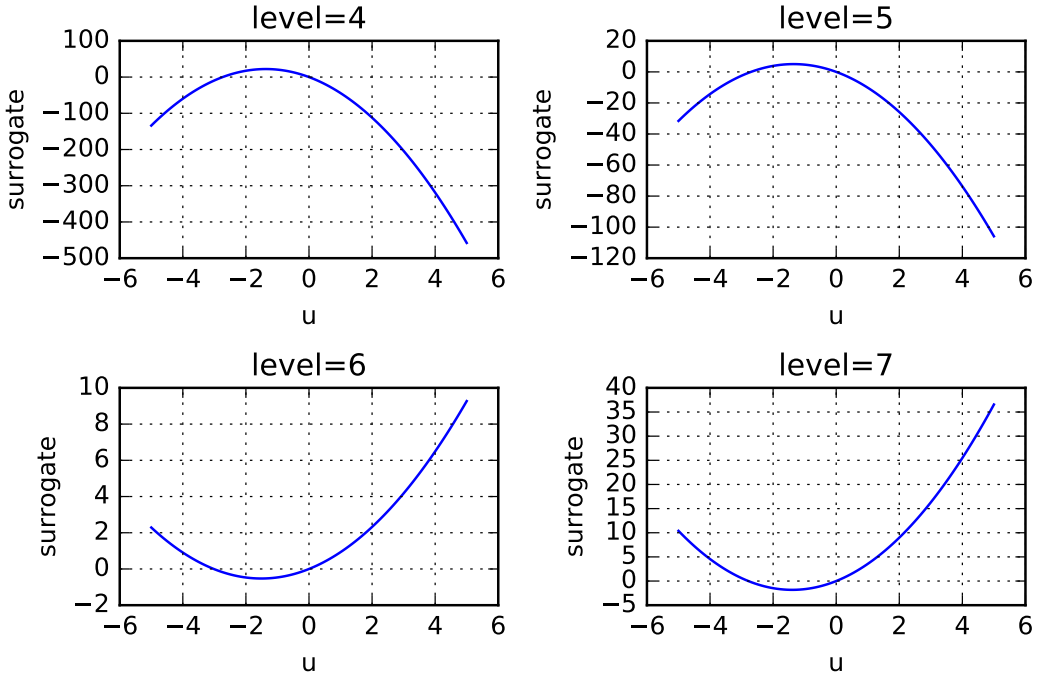


Figure 3: The dependence of the convexity of the sparse grid surrogate on the level when $\alpha = \beta = 50$.

Proof: From the assumptions (i) and (iii), one gets

$$D^2F_l(\mathbf{u}) \geq (\gamma - \|F - F_l\|) I.$$

The result then follows directly from (ii). A detailed proof of the theorem is available in the [supplementary Appendix](#)¹. ♠

In Figures 3 and 4 we compute a simple stochastic optimisation problem with cost function

$$h(\mathbf{u}, \mathbf{W}) = \mathbf{u}^2 + (W_0^2 + 10W_1^2) \mathbf{u}, \tag{13}$$

¹<http://anziamj.austms.org.au/ojs/index.php/ANZIAMJ/article/downloadSuppFile/14060/37294>

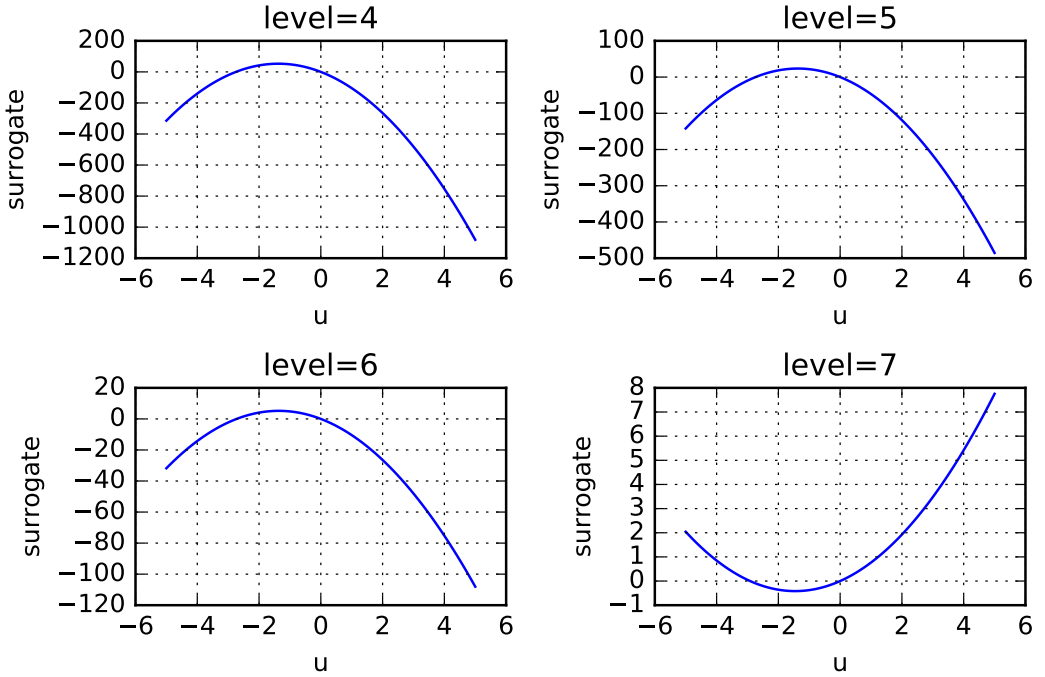


Figure 4: The dependence of the convexity of the sparse grid surrogate on the level when $\alpha = \beta = 100$.

where $\mathbf{u} \in \mathcal{U} = [-5, 5]$ and W_i with $i = 1, 2$ are i.i.d. random variables satisfying

$$W_i \sim \text{Beta}(\alpha, \beta). \tag{14}$$

Rewriting the cost function (13) explicitly in integral form, the problem becomes

$$\min_{\mathbf{u} \in \mathcal{U}} \int_0^1 \int_0^1 [\mathbf{u}^2 + (w_0^2 + 10w_1^2)\mathbf{u}] p(w_0) p(w_1) dw_0 dw_1, \tag{15}$$

where p is the probability density function of W_i . The objective function is strictly convex over \mathcal{U} and $\partial^2 F / \partial \mathbf{u}^2 = 2$ at any point of \mathcal{U} . The exact

solution \mathbf{u}^* is

$$\mathbf{u}^* = -\frac{1}{2} (\mathbb{E}[W_0^2] + 10 \mathbb{E}[W_1^2]). \quad (16)$$

Figure 3 shows the sparse grid surrogates of the example with levels $\mathfrak{l} = 4, 5, 6, 7$ for $\alpha = \beta = 50$ where the exact solution of (15) is $\mathbf{u}^* \approx -1.38861$. Similarly, Figure 4 shows the sparse grid surrogates of the example with levels $\mathfrak{l} = 4, 5, 6, 7$ for $\alpha = \beta = 100$ where the exact solution of (15) is $\mathbf{u}^* \approx -1.38184$. The univariate quadrature rule used in this example is the Gauss–Patterson rule. We see from Figure 3 that the surrogate function is concave when $\mathfrak{l} = 4, 5$ and becomes convex when the level increases to $\mathfrak{l} = 6, 7$. The minimizer of the surrogate function with $\mathfrak{l} = 4, 5$ is $F_1(\mathbf{u}) = 5$ on the boundary of \mathbf{U} , which is far from the exact solution \mathbf{u}^* which is an interior point. However, the computed minimizers are -1.50965 and -1.39318 for $\mathfrak{l} = 6, 7$, respectively, which are much better approximations for \mathbf{u}^* . Figure 4 presents a more extreme example, although it shows the same pattern of the change of convexity as that in Figure 3. The computational results in Figures 3 and 4 agree with Theorem 1 since the sparse grid surrogate becomes strictly convex when the level \mathfrak{l} is large enough under given assumptions.

4 Application to stochastic control

In this section we illustrate our sparse grid approach with an instance of a discrete time open-loop stochastic control problem. Bertsekas [1] discusses the general class of such control problems. General stochastic discrete time control problems are described by a dynamic system with states \mathbf{x}_i for times $i = 0, 1, 2, \dots$. Typically, the states are elements of some linear space. Here we consider only finite dimensional spaces and in our example we consider the case of one dimensional spaces. The dynamics are influenced by control variables \mathbf{u}_i and noise \mathbf{w}_i . In particular, the initial state \mathbf{x}_0 is given and the later states are obtained recursively by

$$\mathbf{x}_{i+1} = \psi_i(\mathbf{x}_i, \mathbf{u}_i, \mathbf{w}_i), \quad i = 0, \dots, d-1, \quad (17)$$

for some rules ψ_i . It follows that the vectors of states, control and noise, respectively,

$$\mathbf{x} = (x_0, \dots, x_{d-1}), \quad \mathbf{u} = (u_0, \dots, u_{d-1}), \quad \mathbf{w} = (w_0, \dots, w_{d-1}), \quad (18)$$

satisfy some system of equations

$$\mathbf{x} = \Psi(\mathbf{x}, \mathbf{u}, \mathbf{w}). \quad (19)$$

The state vector (computed by evaluating the functions ψ_i in turn) is then of the form

$$\mathbf{x} = \xi(\mathbf{u}, \mathbf{w}). \quad (20)$$

Control theory is concerned with the determination of the control variable \mathbf{u} . The control is optimal if it minimises some cost function $\Phi(\mathbf{u}, \mathbf{x})$. Inserting the state equations $\mathbf{x} = \xi(\mathbf{u}, \mathbf{w})$ the cost is then

$$h(\mathbf{u}, \mathbf{w}) = \Phi(\mathbf{u}, \xi(\mathbf{u}, \mathbf{w})).$$

One difficulty is that the actual noise vector \mathbf{w} is unknown. In this case, one minimises the expected costs. For this, one requires a stochastic model for the dynamical system. In particular, the noise vector \mathbf{w} is modelled as a random vector \mathbf{W} with i.i.d. components W_i with some given probability density. As we cannot assume knowledge of the sample values of W_i , we consider open-loop-control [8] and in particular the controls u_i are not random. However, the states are random vectors

$$\mathbf{X} = \Psi(\mathbf{X}, \mathbf{u}, \mathbf{W}), \quad (21)$$

and thus

$$\mathbf{X} = \xi(\mathbf{u}, \mathbf{W}). \quad (22)$$

Instead of minimising the actual cost $h(\mathbf{u}, \mathbf{w})$ we then minimise the expected cost and thus have the minimisation problem

$$\min_{\mathbf{u} \in \mathcal{U}} \mathbb{E} [h(\mathbf{u}, \mathbf{W})]. \quad (23)$$

Written in terms of the cost function Φ and the equation for the states of the dynamical system (22), the expected cost and its surrogate with N quadrature points is

$$\int_{\mathbb{R}^d} \Phi(\mathbf{u}, \xi(\mathbf{u}, \mathbf{w})) p(\mathbf{w}) d\mathbf{w} \approx \sum_{j=1}^N c_j \Phi(\mathbf{u}, \xi(\mathbf{u}, \mathbf{w}_j)) p(\mathbf{w}_j). \quad (24)$$

To illustrate our sparse grid approach and get some insights into its computational performance we consider a linear dynamical system with quadratic costs. In particular, we have

$$\mathbf{x} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{C}\mathbf{w} + \mathbf{x}_0\mathbf{e}_0 \quad (25)$$

and

$$\Phi(\mathbf{u}, \mathbf{x}) = \mathbf{u}^T \mathbf{P}\mathbf{u} + \mathbf{x}^T \mathbf{Q}\mathbf{x}, \quad (26)$$

where \mathbf{A} , \mathbf{B} , \mathbf{C} , \mathbf{P} and \mathbf{Q} are given $d \times d$ matrices and \mathbf{x}_0 is the given initial value. We then get $\xi(\mathbf{u}, \mathbf{w}) = (\mathbf{I} - \mathbf{A})^{-1}(\mathbf{B}\mathbf{u} + \mathbf{C}\mathbf{w} + \mathbf{x}_0\mathbf{e}_0)$ and from this we obtain $\mathbf{h}(\mathbf{u}, \mathbf{w})$ which is also a quadratic function.

We use the certainty equivalence principle [1] to get an explicit expression for the optimal \mathbf{u} . According to the principle, the solution of the stochastic control problem is the same as that of a corresponding deterministic problem when the objective function is quadratic and the constraints are linear. This means we get the reference solution by numerically solving the deterministic problem, as described in the [supplementary Appendix 2](#).

To test our numerical solver we use the Broyden–Fletcher–Goldfarb–Shanno algorithm to solve the surrogate problem. In the test we set $d = 7$, \mathbf{P} and \mathbf{Q} diagonal matrices with $p_i = q_i = 1/d$, $i = 0, \dots, d - 1$ on their diagonals, \mathbf{A} , \mathbf{B} and \mathbf{C} subdiagonal matrices with $a_i = (1 + 1/d)$, $b_i = 1$, $c_i = 1$, $i = 0, \dots, d - 1$ on their subdiagonal, respectively, and $W_i \sim \text{Beta}(2, 3)$,

²<http://anziamj.austms.org/ojs/index.php/ANZIAMJ/article/downloadSuppFile/14060/37295>

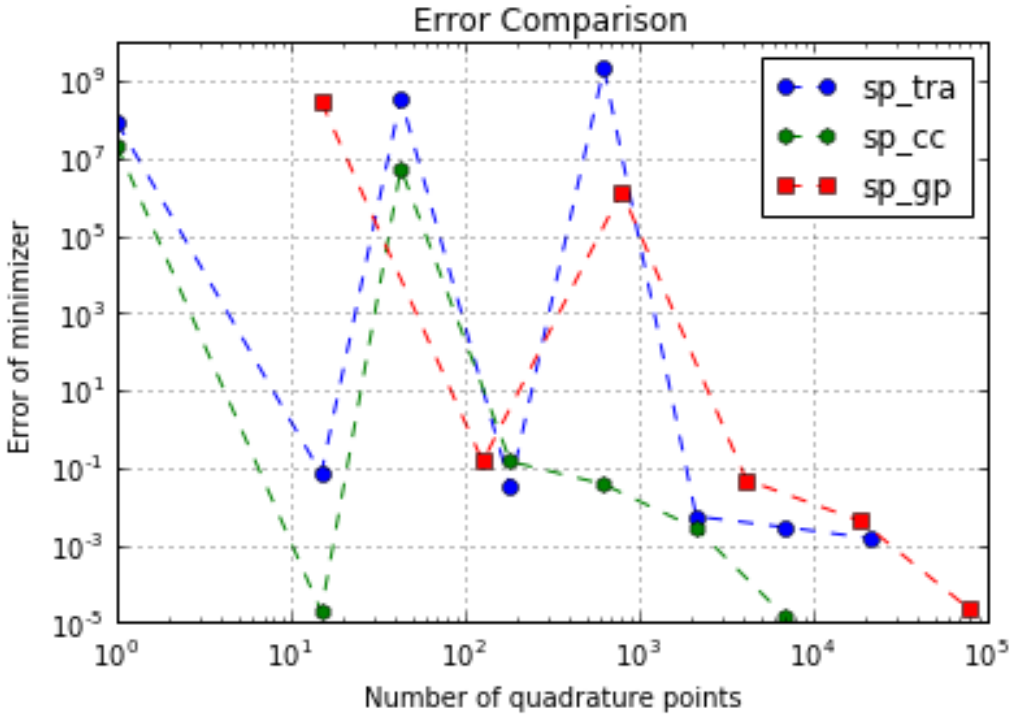


Figure 5: Computational results for the open-loop stochastic control problem with low level sparse grids for the trapezoidal (**sp_tra**), Clenshaw–Curtis (**sp_cc**), and Gauss–Patterson (**sp_gp**) methods.

$i = 0, \dots, d - 1$. Since the integrand is zero on the boundary of the domain of the integral, the quadrature points on the boundary are not counted for the sparse grid based on the trapezoidal and Clenshaw–Curtis rules. Figure 5 shows the convergence performance of the sparse grid generation method with three different univariate quadratures. As discussed in Section 3, we may obtain wrong solutions when we use low level sparse grids. However, when we increase the level a certain amount, the numerical solution of the sparse grid generation method starts to converge to the reference solution. Figure 6

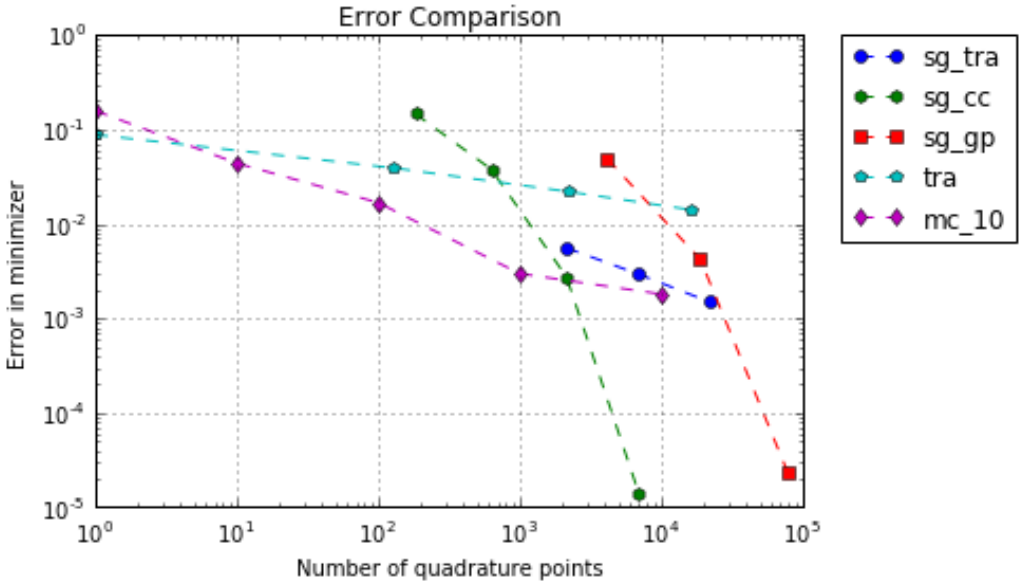


Figure 6: Computational results for the open-loop stochastic control problem for the trapezoidal (sg_tra), Clenshaw–Curtis (sg_cc), and Gauss–Patterson (sg_gp) methods without low level approximations, as well as the product trapezoidal rule (tra) and the Monte Carlo method averaged over ten runs (mc_10).

presents the errors of three different quadrature rules applied to this problem. They are the product trapezoidal rule, the Monte Carlo method averaged over ten runs and the three sparse grid methods without low level approximations. All three sparse grid methods converge faster than the product trapezoidal rule and Monte Carlo method for this example. In particular, the convergence rates of Clenshaw–Curtis and Gauss–Patterson are much better than other methods.

5 Conclusions

The numerical results show that the sparse grid generation method can be used to solve stochastic optimisation problems. Its performance is usually better than the product rule or Monte Carlo method when the integrand in the objective function is sufficiently smooth. However, we need to be very careful when choosing the level of the sparse grid since low level sparse grids cannot keep the convexity of the original problem and therefore lead to an inaccurate solution. In this article we only study convex problems. For non-convex problems Chen et al. [3] proved the epi-convergence of the sparse grid surrogate. However, how to choose a suitable level for the sparse grid prior to computation is still an open problem.

References

- [1] D. P. Bertsekas. *Dynamic programming and optimal control. Vol. I.* Athena Scientific, 2005. URL: <http://athenasc.com/dpbook.html> (cit. on pp. C26, C28).
- [2] H.-J. Bungartz and M. Griebel. “Sparse grids”. In: *Acta Numer.* 13 (2004), pp. 147–269. DOI: [10.1017/S0962492904000182](https://doi.org/10.1017/S0962492904000182) (cit. on pp. C18, C19, C22).
- [3] M. Chen, S. Mehrotra, and D. Papp. “Scenario generation for stochastic optimization problems via the sparse grid method”. In: *Comput. Optim. Appl.* 62.3 (2015), pp. 669–692. DOI: [10.1007/s10589-015-9751-7](https://doi.org/10.1007/s10589-015-9751-7) (cit. on pp. C18, C31).
- [4] C. W. Clenshaw and A. R. Curtis. “A method for numerical integration on an automatic computer”. In: *Numer. Math.* 2 (1960), pp. 197–205. DOI: [10.1007/BF01386223](https://doi.org/10.1007/BF01386223) (cit. on p. C19).

- [5] P. J. Davis and P. Rabinowitz. *Methods of numerical integration*. Computer Science and Applied Mathematics. Academic Press, 1984. DOI: [10.1016/C2013-0-10566-1](https://doi.org/10.1016/C2013-0-10566-1) (cit. on p. [C19](#)).
- [6] T. Gerstner and M. Griebel. “Numerical integration using sparse grids”. In: *Numer. Algorithms* 18 (1998), pp. 209–232. DOI: [10.1023/A:1019129717644](https://doi.org/10.1023/A:1019129717644) (cit. on pp. [C18](#), [C19](#), [C22](#)).
- [7] M. Holtz. *Sparse grid quadrature in high dimensions with applications in finance and insurance*. Vol. 77. Lecture Notes in Computational Science and Engineering. Springer-Verlag, 2011. DOI: [10.1007/978-3-642-16004-2](https://doi.org/10.1007/978-3-642-16004-2) (cit. on pp. [C18](#), [C19](#), [C22](#)).
- [8] B. Øksendal. *Stochastic differential equations*. Springer-Verlag, 1998. DOI: [10.1007/978-3-662-03620-4](https://doi.org/10.1007/978-3-662-03620-4) (cit. on p. [C27](#)).
- [9] T. N. L. Patterson. “The optimum addition of points to quadrature formulae”. In: *Math. Comp.* 22 (1968), pp. 847–856. DOI: [10.2307/2004583](https://doi.org/10.2307/2004583) (cit. on p. [C19](#)).
- [10] S. W. Wallace and W. T. Ziemba. *Applications of stochastic programming*. MOS-SIAM Series on Optimization. SIAM, 2005. DOI: [10.1137/1.9780898718799](https://doi.org/10.1137/1.9780898718799) (cit. on pp. [C17](#), [C18](#)).

Author addresses

1. **Yuancheng Zhou**, Mathematical Sciences Institute, The Australian National University, Canberra ACT 0200, Australia.
<mailto:yuancheng.zhou@anu.edu.au>
2. **Markus Hegland**, Mathematical Sciences Institute, The Australian National University, Canberra ACT 0200, Australia.
<mailto:markus.hegland@anu.edu.au>