Excellent work cher Chiheb!
Please see my questions below

we should mention that almost similar results were obtained for the case of the second example in Table 7.1.

8 Numerical smoothing with MLMC

In this section, we want to motivate the advantage of the numerical smoothing idea in the context of MLMC method. For this aim we consider two examples: i) the first one is for computing the price of a digital option under GBM and Heston models (see Section 8.2) , and ii) the second example is for approximating a density function for dynamics following GBM and Heston models (see Section 8.3). The idea and results of Section 8.2 can be generalized to i) any kind of option having low regularity in the payoff function, or ii) for computing distribution functions, since it involves the indicator function. On the other hand, examples in Section 8.3 have two important applications: i) computing density functions which involves the use of Dirac delta functions and which is hard to approximate its expectation using MLMC due to the infinite variance, and ii) computing Greeks for an option with a non smooth payoff function.

8.1 Our contributions to the literature

Usually, for such tasks, standard MLMC will fail or not have the optimal performance, due to the singularity present in the delta or the indicator functions, implying either high variance or kurtosis for the MLMC estimator. In the literature, few works tried to address this issue. For instance,

1. Avikainen in [5], and Giles, Higham, and Mao in [11] used MLMC for such a task without smoothing.

2. On the other hand, a second approach was suggested in [13, 10], that used implicit smoothing based on the use of conditional expectations. There are two potential issues with this second approach: i) In general cases, one may have dynamics where it is not easy to derive an analytic expression for the conditional expectation and ii) This approach used a higher order scheme, that is the Milstein scheme, to improve the strong order of convergence, and consequently the complexity of the MLMC estimator. Such a scheme becomes very computationally expensive for higher dimensional dynamics. Different non smooth payoff functions were considered in [13, 10] (Asian, barrier, digital) but the only considered dynamics were under the GBM model.

3. In [12], the authors suggested a different approach based on parametric smoothing. In fact, they carefully constructed a regularized version of the QoI, based on a regularization parameter that depends on the weak and strong convergence rates and also the tolerance requirement. This approach, despite offering better performance for the MLMC estimator and a better setting for theoretical analysis, it has the practical disadvantage consisting in the difficulty of generalizing it to cases where there is no prior knowledge of the the convergence rates (that is they need to be estimated numerically), and also for each error tolerance, a new regularization parameter needs to be computed. Note also that all the numerical examples in [12] were based on the GBM dynamics.

In this work, we address a similar problem and propose an alternative approach that is based on numerical smoothing. Our approach compared to previous mentioned works, has the following advantages

1. It can be easily applied to cases where one can not apply analytic smoothing.
2. We obtain similar rates of strong convergence and MLMC complexity as in [13, 10], without the need to use higher order schemes such as Milstein scheme.
3. Our approach is parameter free compared to that of [12]. Therefore, in practice it is much easier to apply for any dynamics and QoI.
4. Compared to [13, 10, 12], we add numerical results for the Heston model were discretization is needed unlike the GBM dynamics which is considered here as a toy example.

8.2 MLMC for digital options

In this section, we motivate the idea of using the numerical smoothing idea to compute option prices for non smooth payoff function. For illustration, we consider the price of the digital option, that is we want to approximate

$$\mathrm{E}\left[g(X)\right] = \mathrm{E}\left[\mathbf{1}_{X>K}\right], \tag{8.1}$$

where K is the strike.

In this kind of problems, we have a very well known issue related to observing high kurtosis problem for very fine discretization levels of the MLMC method. We try in the following Section to explain this issue.

8.2.1 High kurtosis issue

Let g denote a random variable, and let g_ℓ denote the corresponding level ℓ numerical approximation. We also define Y_ℓ to be

$$Y_\ell = \begin{cases} g_0, & \ell = 0 \\ g_\ell - g_{\ell-1}, & \ell > 1. \end{cases}$$

and the multilevel estimator

$$Y = \sum_{\ell=0}^{L} \mathrm{E}\left[Y_\ell\right] \tag{8.2}$$

The MLMC approach needs a good estimate for $V_\ell = \mathrm{Var}[Y_\ell]$. When the number of samples N is large, the standard deviation of the sample variance for a random variable X with zero mean is approximately $\sqrt{(\kappa-1)/N}\mathrm{E}\left[X^2\right]$ where the kurtosis κ is defined as $\kappa = \frac{\mathrm{E}[X^4]}{(\mathrm{E}[X^2])^2}$. Hence $\mathcal{O}\left(\kappa\right)$ samples are required to obtain a reasonable estimate for the variance. An extreme, but important, example is when g always takes the value 0 or 1. In this case we have

$$X \equiv g_\ell - g_{\ell-1} = \begin{cases} 1, & \text{probability} \quad p \\ -1, & \text{probability} \quad q \\ 0, & \text{probability} \quad 1-p-q \end{cases}$$

if $p, q \ll 1$, then $\mathrm{E}\left[X\right] \approx 0$ and $\kappa = (p+q)^{-1} \gg 1$. Therefore, many samples are required for a good estimate of V_ℓ; otherwise, we may get all $X(n) = 0$, which will give an estimated variance of zero. Furthermore, the kurtosis will become worse as $\ell \to \infty$ since $p, q \to 0$ due to weak convergence.

8.2.2 Summary of results for the digital option

In this Section, we try to summarize the obtained results for approximating the price of a digital option using MLMC with and without numerical smoothing for two models: the GBM model (see Table 8.1 and detailed results in Section 8.2.3) and the Heston model (see Table 8.2 and detailed results in Section 8.2.4). From these tables, we can see two main results

1. The significant reduction of the kurtosis at the finest levels, κ_L, of MLMC algorithm when using numerical smoothing. In fact, the kurtosis is reduced by a factor of 200 for the GBM dynamics and by a factor of 28 for the Heston model.

2. Numerical smoothing has significantly improved the strong rate β for the example under GBM model from $\beta = 1/2$ to $\beta = 1$ resulting in reducing the order of MLMC complexity from $\mathcal{O}\left(TOL^{-2.5}\right)$ to $\mathcal{O}\left(TOL^{-2}\left(\log(TOL)\right)^2\right)$. On the other hand, numerical smoothing also improved slightly the strong rate for the example under Heston model from $\beta = 2/3$ to $\beta = 3/4$ resulting in a slight improvement of the MLMC complexity.

Method	κ_L	α	β	γ	Complexity
Without smoothing	621	1	1/2	1	$\mathcal{O}\left(TOL^{-2.5}\right)$
With numerical smoothing	3	1	1	1	$\mathcal{O}\left(TOL^{-2}\left(\log(TOL)\right)^2\right)$

Table 8.1: Summary of the MLMC numerical results observed for computing the price of the digital option under GBM model. κ_L is the kurtosis at the finest levels of MLMC with $\Delta t_L = T.2^{-10}$, (α, β, γ) are weak, strong and work rates respectively. TOL is the user selected MLMC tolerance. These results correspond to Figures 8.1 and 8.2.

I do not understand why the smoothing makes the truncation work worse

Method	κ_L	α	β	γ	Complexity
Without smoothing using the full truncation scheme	166	1.2	2/3	1	$\mathcal{O}\left(TOL^{-2.3}\right)$
With numerical smoothing using smooth scheme	6	1.7	3/4	1	$\mathcal{O}\left(TOL^{-2.1}\right)$
With numerical smoothing the full truncation scheme	6	1/2	3/4	1	$\mathcal{O}\left(TOL^{-2.5}\right)$

why do you get this Beta=3/4 and not one?

Table 8.2: Summary of the MLMC numerical results observed for computing the price of the digital option under Heston model. κ_L is the kurtosis at the finest levels of MLMC with $\Delta t_L = T.2^{-6}$, (α, β, γ) are weak, strong and work rates respectively. TOL is the user selected MLMC tolerance. These results correspond to Figures 8.3, 8.4 and 8.5.

8.2.3 Digital options under the GBM model

Using similar idea introduced in Section 6.1, we can show that

$$\begin{aligned} \mathrm{E}\left[g(X)\right] &= \mathrm{E}\left[\mathbf{1}_{X>K}\right] \\ &= \mathrm{E}\left[1 - \Phi(y^*(K))\right], \end{aligned} \tag{8.3}$$

where $y^*(K)$ is the kink location obtained by solving numerically

$$X(T; y^*(K), \mathbf{z}_{-1}) = K,$$

where $\mathbf{z}$ is $N-1$ Gaussian random vector (N is the number of time steps) used for Brownian bridge construction, and Φ is the cumulative Gaussian distribution function.

As an illustration, we choose the digital option of GBM asset with parameters: $S_0 = K = 100$, $T = 1$, $r = 0$, and $\sigma = 0.2$, and we compare MLMC results with the original payoff function (without numerical smoothing), given by Figure 8.1, and MLMC results after applying our numerical smoothing idea, given by Figure 8.2. From these two Figures we have the following conclusions:

1. The numerical smoothing has improved the rate of strong convergence from 1/2 (without smoothing) to 1 after doing the numerical smoothing (compare both top left plots in Figures 8.1 and Figure 8.2).

2. The numerical smoothing has also dropped significantly the kurtosis at the finest level, κ_L, by a factor 200, this is can be seen clearly by comparing the middle right plots in both Figures 8.1 and Figure 8.2. We stress that this is an important improvement, since one needs $\mathcal{O}(\kappa)$ samples to obtain a reasonable estimate for the variance (see Section 8.2.1).

3. Finally, improving the strong error rate after using the numerical smoothing, resulted in an improvement of the complexity rate going from $TOL^{-2.5}$ for the case without smoothing to TOL^{-2}, where TOL is a prescribed tolerance, for the case where MLMC is coupled with numerical smoothing (compare the bottom right plots in Figures 8.1 and 8.2).

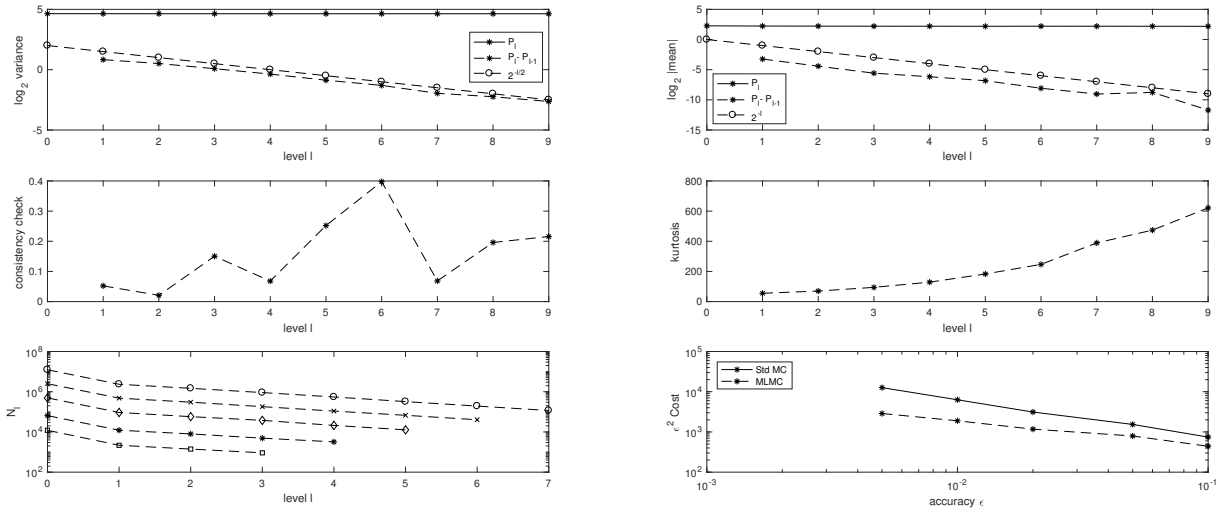

Figure 8.1: Numerical results for a digital call option using the MLMC method coupled with Euler-Maruyama discretisation of the GBM SDE, and without smoothing of the payoff.

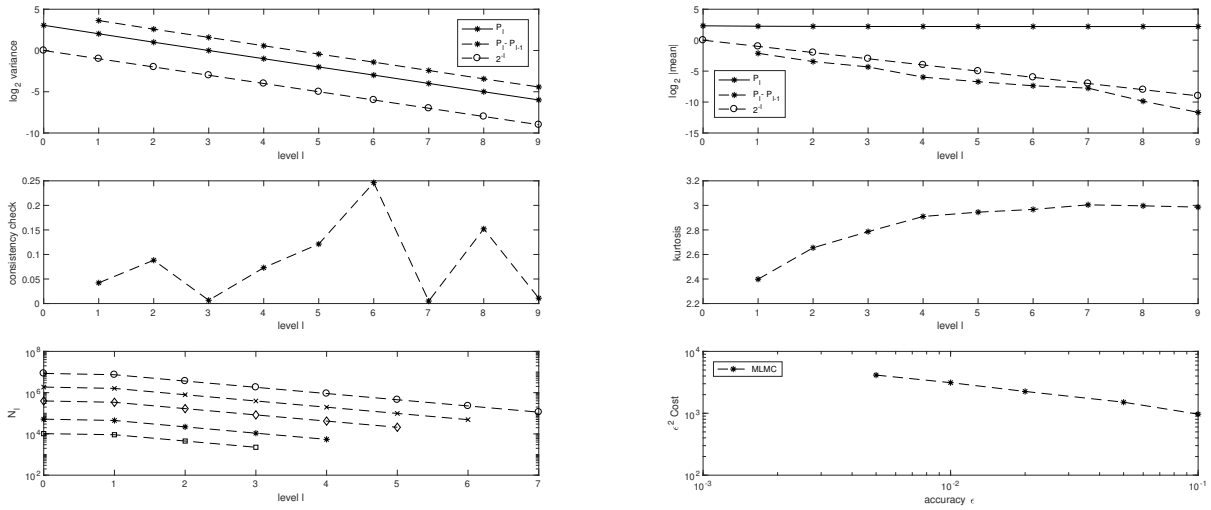

Figure 8.2: Numerical results for a digital call option using the MLMC method coupled with Euler-Maruyama discretisation of the GBM SDE, after applying the numerical smoothing to the payoff. We mention that observing a decaying variance of P_ℓ here is expected since we used Brownian bridge for path construction, and the QoI depends only on the terminal value of the Brownian bridge which has a variance scaled of order Δt.

We emphasize that our approach can be extended in a straightforward manner to any kind of dynamics, since it is based on numerical smoothing based on solving a root finding problem. In the following Section, we show the advantage of our approach for the Heston model where discretization of the asset price is indeed needed.

8.2.4 Digital option under the Heston model

Since the asset price in the Heston model (see (7.1)) has similar structure of dynamics compared to the GBM with the difference of the volatility being non constant, we can use similar idea introduced in Section 6.1 and show that

$$\begin{aligned} \mathrm{E}\left[g(X)\right] &= \mathrm{E}\left[\mathbf{1}_{X>K}\right] \\ &= \mathrm{E}\left[1-\Phi(y^*(K))\right], \end{aligned} \tag{8.4}$$

where $y^*(K)$ is the kink location obtained by solving numerically

$$X(T; y^*(K), \mathbf{z}_{-1}) = K,$$

where $\mathbf{z}$ is $2N-1$ Gaussian random vector (N is the number of time steps) used for Brownian bridge construction, and Φ is the cumulative Gaussian distribution function.

As an illustration, we choose the digital option of Heston asset with parameters: $S_0 = K = 100$, $v_0 = 0.04$, $\mu = 0$, $\rho = -0.9$, $\kappa = 1$, $\xi = 0.1$, $\theta = 0.0025$, and we compare MLMC results with the original payoff function (without numerical smoothing), given by Figure 8.3, and MLMC results

after applying our numerical smoothing idea, when using the smooth scheme (see Figure 8.4), and the Full truncation scheme (see Figure 8.5). From these two Figures we have the following conclusions:

1. The numerical smoothing has improved slightly the rate of strong convergence from 1/2 (without smoothing) to 3/4 after doing the numerical smoothing (compare top left plots in Figures 8.3, Figure 8.4 and Figure 8.5).

2. The numerical smoothing has also dropped significantly the kurtosis at the finest level, κ_L, by a factor 28, this is can be seen clearly by comparing the middle right plots in Figures 8.3, Figure 8.4 and Figure 8.5. We stress that this is an important improvement, since one needs $\mathcal{O}(\kappa)$ samples to obtain a reasonable estimate for the variance (see Section 8.2.1). Note also that although the initial level of MLMC is different for case without smoothing compared to the case we do smoothing, the final levels have the same time mesh that is $\Delta t_L = 2^{-6}.T$.

3. Improving the strong error rate after using the numerical smoothing, resulted in a slight improvement of the complexity (compare the bottom right plots Figures 8.3, Figure 8.4 and Figure 8.5).

4. Finally, observe that we observed slightly better behavior for the case of numerical smoothing using the smooth scheme (scheme in Section 7.4) compared to the numerical smoothing using the Full truncation scheme (scheme in Section 7.1) (compare Figures 8.4 and 8.5).

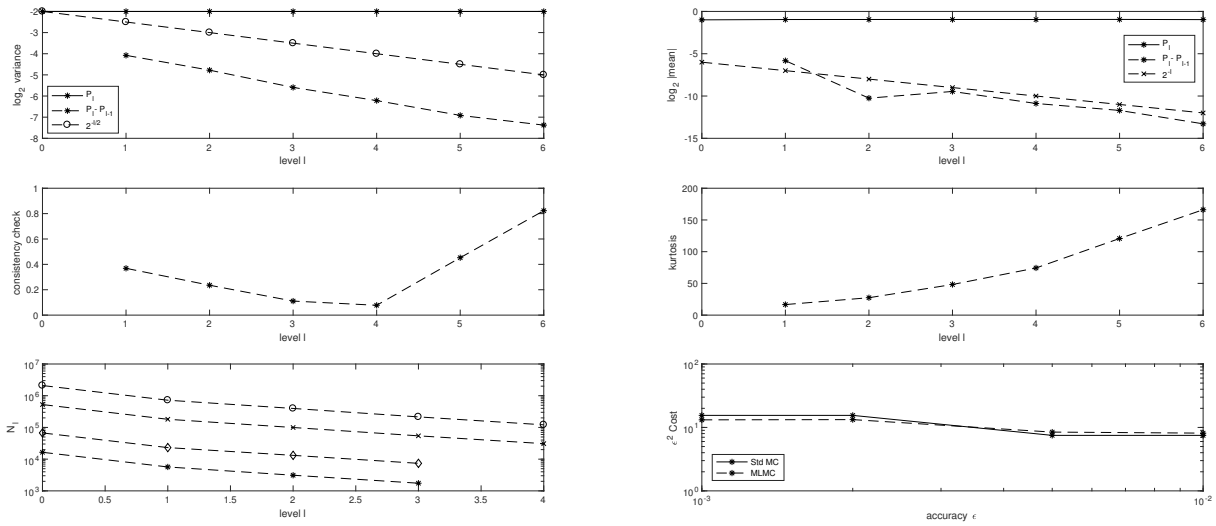

Figure 8.3: Numerical results for a digital call option under the Heston model using the MLMC method coupled with the Full truncation scheme in Section 7.1, and without smoothing of the payoff.

mention that you could also couple with weak extrapolation (talay-tubaro, G. Pages)

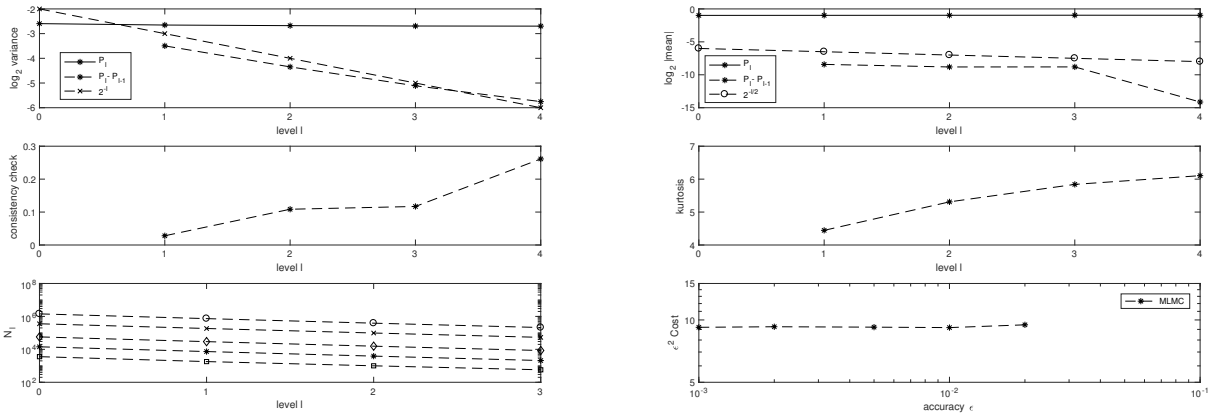

Figure 8.4: Numerical results for a digital call option under the Heston model using the MLMC method coupled with the smooth scheme in Section 7.4, and with smoothing of the payoff.

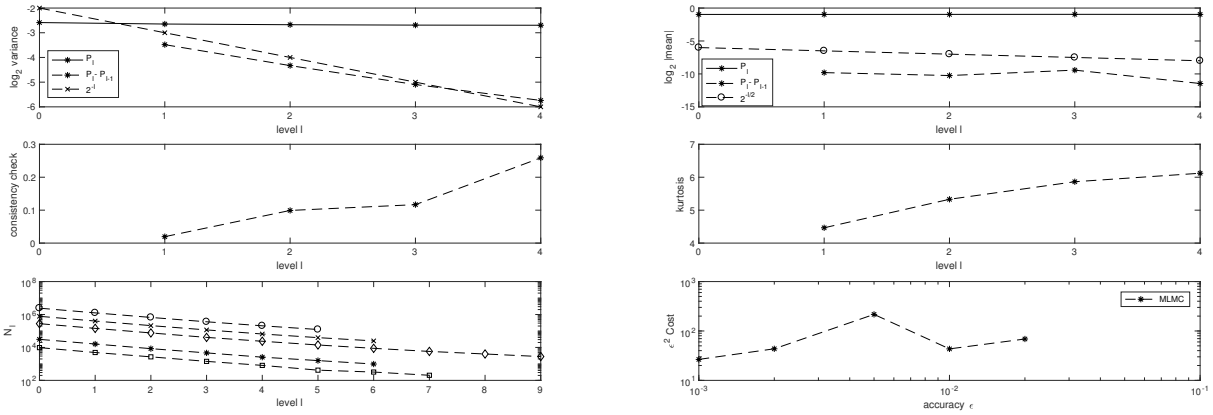

Figure 8.5: Numerical results for a digital call option under the Heston model using the MLMC method coupled with the Full truncation scheme in Section 7.1, and with smoothing of the payoff.

Remark 8.1. Although we just illustrated the benefit of our approach when coupled with MLMC for computing the digital option price under the GBM (Section 8.2.3) and Heston dynamics (Section 8.2.4), we emphasize that it can be easily extended to any kind of model dynamics and to any low regular observable, g. For instance, this idea can be applied for approximating distribution functions involving the indicator function as the observable g. Finally, in all these possible extensions, we expect similar gains to be observed compared to the case of using MLMC without smoothing.

8.3 MLMC for approximating densities and Greeks

In this section, we motivate the idea of coupling the numerical smoothing with MLMC method to compute density functions and Greeks for non smooth payoff functions. We remind that MLMC without any smoothing will fail due to the infinite variance caused by the singularity of the delta

function. The aim in this case is to approximate the density, $\rho_X(u)$, for a given stochastic process X, at point u, and which is given by

$$\rho_X(u) = \mathrm{E}\left[\delta(X-u)\right], \tag{8.5}$$

where δ is the Dirac delta function.

In Table 8.3, we summarize the obtained results for estimating the density, using MLMC coupled with numerical smoothing, under GBM and Heston dynamics.

Method	Kurtosis	α	β	γ	Complexity
GBM + numerical smoothing	5	1	3/2	1	TOL^{-2}
Heston + numerical smoothing using smooth scheme	5	1	1/2	1	$TOL^{-2.5}$
Heston + numerical smoothing the full truncation scheme	6	1	1/2	1	$TOL^{-2.5}$

Table 8.3: Summary of the MLMC numerical results observed for computing the density of asset price under GBM and Heston models. κ_L is the kurtosis at the finest levels of MLMC with $\Delta t_L = T.2^{-8}$, (α, β, γ) are weak, strong and work rates respectively. TOL is the user selected MLMC tolerance. These results correspond to Figures 8.6, 8.7 and 8.8.

Why do we get Beta=1/2 only here for Heston?

8.3.1 Approximating density under the GBM model

We can show that

$$\begin{aligned}\rho_X(u) &= \mathrm{E}\left[\delta(X-y)\right] \\ &= \exp\left(-(y^*(u))^2/2\right)\frac{dy^*}{dx}(u),\end{aligned} \tag{8.6}$$

where $y^*(K)$ is the kink location obtained by solving numerically

$$X(T; y^*(K), \mathbf{z}_{-1}) = K,$$

where $\mathbf{z}$ is $N-1$ Gaussian random vector (N is the number of time steps) used for Brownian bridge construction.

As an illustration, we choose to compute the density ρ_X such that X is a GBM with parameters: $S_0 = K = 1$, $T = 1$, $r = 0$, and $\sigma = 0.2$. In this case, as a reference solution, we know that $X(T)$ is lognormally distributed with parameters $r - \sigma^2$ and σ^2. In Figure 8.6, we show the numerical results for computing the ρ_X at $u = 1$, using MLMC coupled with the numerical smoothing idea. From Figure 8.6, we can check that we obtain a strong convergence rate of order 3/2(see top left plot in Figure 8.6), which results in a complexity of the MLMC estimator to be of order TOL^{-2}, where TOL is a prescribed tolerance.

please justify why we get Beta>1 in this case …

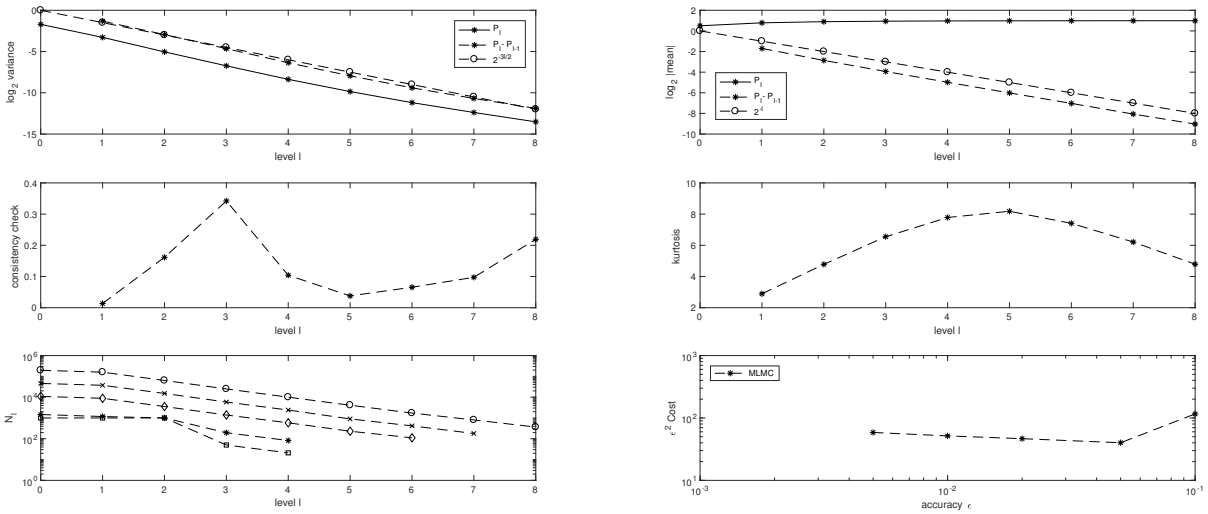

Figure 8.6: Numerical results for density estimation using the MLMC method coupled with Euler-Maruyama discretisation of the GBM SDE, after applying the numerical smoothing. We mention that observing a decaying variance of P_ℓ here is expected since we used Brownian bridge for path construction, and the QoI depends only on the terminal value of the Brownian bridge which has a variance scaled of order Δt.

We emphasize that our approach can be extended in a straightforward manner to any kind of dynamics, since it is based on numerical smoothing based on solving a root finding problem. In the following Section, we show the advantage of our approach for the Heston model where discretization of the asset price is indeed needed.

8.3.2 Approximating density under the Heston model

Since the asset price in the Heston model (see (7.1)) has similar structure of dynamics compared to the GBM with the difference of the volatility being non constant, we can show also that

$$\begin{aligned} \rho_X(u) &= \mathrm{E}\left[\delta(X-y)\right] \\ &= \exp\left(-(y^*(u))^2/2\right)\frac{dy^*}{dx}(u), \end{aligned} \tag{8.7}$$

where $y^*(K)$ is the kink location obtained by solving numerically

$$X(T; y^*(K), \mathbf{z}_{-1}) = K,$$

where $\mathbf{z}$ is $2N-1$ Gaussian random vector (N is the number of time steps) used for Brownian bridge construction.

As an illustration, we choose to compute the density ρ_X such that X is a Heston asset with parameters: $S_0 = K = 1$, $v_0 = 0.04$, $\mu = 0$, $\rho = -0.9$, $\kappa = 1$, $\xi = 0.1$, $\theta = 0.0025$. In Figures 8.7 and 8.8, we show the numerical results for computing the ρ_X at $u = 1$, using MLMC coupled with

the numerical smoothing idea. From Figures 8.7 and 8.8, we can check that we obtain a strong convergence rate of order 1/2 (see top left plot in Figure 8.7 and 8.8), which results in a complexity of the MLMC estimator to be of order $TOL^{-2.5}$, where TOL is a prescribed tolerance.

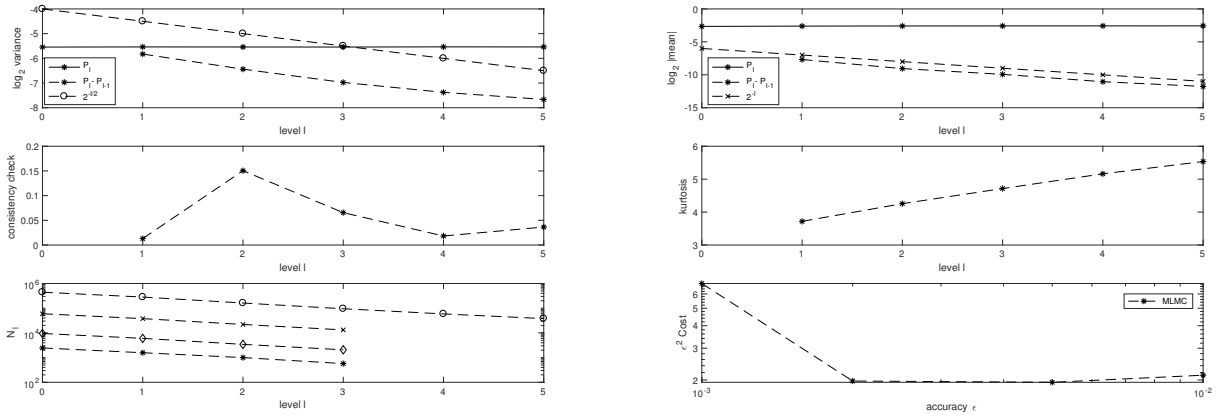

Figure 8.7: Numerical results for density estimation for Heston model using the MLMC method coupled with the smooth scheme in Section 7.4, and with smoothing of the payoff..

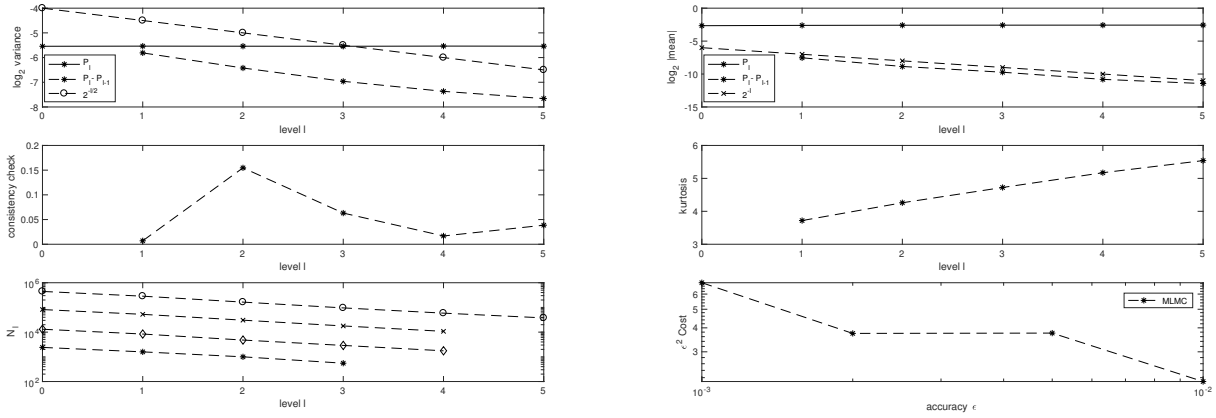

Figure 8.8: Numerical results for density estimation for Heston model using the MLMC method coupled with the Full truncation scheme in Section 7.1 , and with smoothing of the payoff.

Remark 8.2. Although we just illustrated the benefit of our approach when coupled with MLMC for computing the density of the asset price under the GBM dynamics (Section 8.3.1) and Heston dynamics (Section 8.3.2), we emphasize that it can be easily extended to any kind of model dynamics. Furthermore, our approach can be easily extended to computing Greeks of a digital options, involving the delta functions.

References Cited

[1] Peter A Acworth, Mark Broadie, and Paul Glasserman. A comparison of some Monte Carlo and quasi Monte Carlo techniques for option pricing. In *Monte Carlo and Quasi-Monte Carlo Methods 1996*, pages 1–18. Springer, 1998.

[2] Aurélien Alfonsi. High order discretization schemes for the cir process: application to affine term structure and heston models. *Mathematics of Computation*, 79(269):209–237, 2010.

[3] Leif BG Andersen. Efficient simulation of the heston stochastic volatility model. *Available at SSRN 946405*, 2007.

[4] Leif BG Andersen and Rupert Brotherton-Ratcliffe. Extended libor market models with stochastic volatility. *Journal of Computational Finance*, 9(1), 2005.

[5] Rainer Avikainen. On irregular functionals of sdes and the euler scheme. *Finance and Stochastics*, 13(3):381–401, 2009.

[6] Christian Bayer, Markus Siebenmorgen, and Rául Tempone. Smoothing the payoff for efficient computation of basket option pricing. *Quantitative Finance*, 18(3):491–505, 2018.

[7] Mark Broadie and Özgür Kaya. Exact simulation of stochastic volatility and other affine jump diffusion processes. *Operations research*, 54(2):217–231, 2006.

[8] Hans-Joachim Bungartz and Michael Griebel. Sparse grids. *Acta numerica*, 13:147–269, 2004.

[9] Russel E Caflisch, William J Morokoff, and Art B Owen. *Valuation of mortgage backed securities using Brownian bridges to reduce effective dimension*. 1997.

[10] Michael Giles, Kristian Debrabant, and Andreas Rößler. Numerical analysis of multilevel monte carlo path simulation using the milstein discretisation. *arXiv preprint arXiv:1302.4676*, 2013.

[11] Michael B Giles, Desmond J Higham, and Xuerong Mao. Analysing multi-level monte carlo for options with non-globally lipschitz payoff. *Finance and Stochastics*, 13(3):403–413, 2009.

[12] Michael B Giles, Tigran Nagapetyan, and Klaus Ritter. Multilevel monte carlo approximation of distribution functions and densities. *SIAM/ASA Journal on Uncertainty Quantification*, 3(1):267–295, 2015.

[13] Mike Giles. Improved multilevel monte carlo convergence using the milstein scheme. In *Monte Carlo and Quasi-Monte Carlo Methods 2006*, pages 343–358. Springer, 2008.

[14] Paul Glasserman. *Monte Carlo methods in financial engineering*. Springer, New York, 2004.

[15] Michael Griebel, Frances Kuo, and Ian Sloan. The smoothing effect of integration in R^{d} and the ANOVA decomposition. *Mathematics of Computation*, 82(281):383–400, 2013.

[16] Michael Griebel, Frances Kuo, and Ian Sloan. Note on the smoothing effect of integration in R^{d} and the ANOVA decomposition. *Mathematics of Computation*, 86(306):1847–1854, 2017.

[17] Andreas Griewank, Frances Y Kuo, Hernan Leövey, and Ian H Sloan. High dimensional integration of kinks and jumps–smoothing by preintegration. *arXiv preprint arXiv:1712.00920*, 2017.

[18] Abdul-Lateef Haji-Ali, Fabio Nobile, Lorenzo Tamellini, and Raul Tempone. Multi-index stochastic collocation for random PDEs. *Computer Methods in Applied Mechanics and Engineering*, 306:95–122, 2016.

[19] Steven L Heston. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *The review of financial studies*, 6(2):327–343, 1993.

[20] Junichi Imai and Ken Seng Tan. Minimizing effective dimension using linear transformation. In *Monte Carlo and Quasi-Monte Carlo Methods 2002*, pages 275–292. Springer, 2004.

[21] Monique Jeanblanc, Marc Yor, and Marc Chesney. *Mathematical methods for financial markets*. Springer Science & Business Media, 2009.

[22] Christian Kahl and Peter Jäckel. Fast strong approximation monte carlo schemes for stochastic volatility models. *Quantitative Finance*, 6(6):513–536, 2006.

[23] Roger Lord, Remmert Koekkoek, and Dick Van Dijk. A comparison of biased simulation schemes for stochastic volatility models. *Quantitative Finance*, 10(2):177–194, 2010.

[24] William J Morokoff and Russel E Caflisch. Quasi-random sequences and their discrepancies. *SIAM Journal on Scientific Computing*, 15(6):1251–1279, 1994.

[25] Bradley Moskowitz and Russel E Caflisch. Smoothness and dimension reduction in quasi-Monte Carlo methods. *Mathematical and Computer Modelling*, 23(8):37–54, 1996.

[26] Denis Talay and Luciano Tubaro. Expansion of the global error for numerical schemes solving stochastic differential equations. *Stochastic analysis and applications*, 8(4):483–509, 1990.

[27] Ye Xiao and Xiaoqun Wang. Conditional quasi-Monte Carlo methods and dimension reduction for option pricing and hedging with discontinuous functions. *Journal of Computational and Applied Mathematics*, 343:289–308, 2018.