GPU Nested Monte Carlo Techniques for XVA Computations

Stéphane Crépey[1], Lokmane Abbas-Turki[2], Babacar Diallo[2,3]

[1] LaMME, Univ Evry, CNRS, Université Paris-Saclay

[2] Laboratoire de probabilités et modèles aléatoires, UMR 7599, case 188, 4, place Jussieu, 75252 Paris cedex 05, France.

[3] Crédit Agricole Corporate & Investment Bank, 12 Place des États Unis, 92120 Montrouge

We present a nested Monte Carlo (NMC) approach implemented on graphics processing units (GPU) to X-valuation adjustments (XVA), where X ranges over C for credit, F for funding, M for margin, and K for capital.

A pure Monte Carlo simulation of all these XVAs would require five layers of NMC, i.e. six nested layers of Monte Carlo. However, if the user is only interested in some of the XVA components, then only the sub-tree corresponding to the most outer XVA needs be processed computationally; If one or several layers can be computed by explicit (exact or approximate) formulas instead of Monte Carlo simulation, then the corresponding layers drop from the picture. Moreover, inner layers only need a square root number of simulation with respect to the most outer layer. In addition, some of the layers exhibit a small variance, so that fewer paths are enough for these.

As a result, with GPU at least, NMC XVA computations are doable. But, although NMC is naively suited to parallelization, their GPU implementation requires various optimizations related to forward and backward SDE simulation, sorting for risk measure computations, and an efficient simulation of expressions involving indicator functions of default times. These optimizations are important for effectively benefiting from a potential ~ 100 speedup factor when compared with an optimized CPU implementation.

This is illustrated on XVA computations involving equities, interest rate, and credit derivatives, for both bilateral and central clearing XVA metrics.