Group F

Scott Sidoli

6/17/19

In this document we experiment with the finite difference method’s N value and J Value. Here, J values are given by the array from 0 to 5\*K, where K is the strike price of the option. X values in the excel output are the current asset price. As an example, when we run batch 1 with J = 5\*k and N = 10,000, we get an option price of 2.12915579291956. The exact method produced 2.13337. This is almost accurate to two decimal places. With N = 100000, and J = 5\*K, we get option price of 2.129132. This is just a little less accurate. If we go up an order of magnitude or to N << 1000, we notice a lot of instability. If we adjust J for batch one we get, for J = K, 2.02561874120539 for the option price at a strike price of 60. For J = 3\*K, we get 2.12157820575451. For J = 10\*K, we get 2.13231210011457.

Based on these tests, we find that for N small (<~10^3) we have instability as well as for N large (>10^7). We regards to adjusting J, we find that for large J (relative to the strike price) we are most accurate. For small J, we are not as accurate. We have excel output for Batches 2, 3, and 4.

For Batch 2, when S = 100 we have (J = 5k, N = 100000) 7.96312055511931 (exact is 7.96557)

For Batch 3, when S = 5, we get 0.203427744003268 (exact is 0.204058)

For Batch 4, when S = 100, we get little accuracy for lots of different attempts. 65,535 for the J and N as above. For smaller and N it is the same. We get some thing a little more reasonable when J = 0.1\*K, but even then, we are to far by about 90 from the exact price. We conclude that have a large expiry time makes Batch 4 subject to lots of inaccuracy for FDM.