**Answers of Group D**

Hanlin WANG

**a) Create generic functions to compute the standard deviation and standard error**

I added these codes in TesMC.cpp which compiles and runs well.

// to calculate sd and se

void SDSE(const vector<double>& vector, const double& r, const double& T)

{

long M = vector.size();

double sum = 0;

double square\_sum = 0;

std::vector<double>::const\_iterator it;

for (it = vector.begin(); it != vector.end(); it++)

{

sum += (\*it);

square\_sum += (\*it) \* (\*it);

}

double SD = sqrt((square\_sum - pow(sum, 2) / M) / (M - 1)) \* exp(-(r) \* (T));

double SE = SD / sqrt(M);

cout << "SD=" << SD << ", SE=" << SE << endl;

}

**b) Batch 1 and 2 experiment**

1) batch 1: call option

|  |  |  |  |
| --- | --- | --- | --- |
| **NT** | **NSIM** | **SD** | **SE** |
| 20 | 50000 | 4.50436 | 0.00637 |
| 100 | 50000 | 4.50473 | 0.02015 |
| 300 | 50000 | 4.55354 | 0.02036 |
| 500 | 50000 | 4.57536 | 0.02046 |
| 800 | 50000 | 4.54918 | 0.02034 |
| 500 | 10000 | 4.60808 | 0.04608 |
| 500 | 30000 | 4.58993 | 0.02650 |
| 500 | 50000 | 4.57536 | 0.02046 |
| 500 | 70000 | 4.55608 | 0.01722 |
| 500 | 90000 | 4.54850 | 0.01516 |

2) batch 1: put option

|  |  |  |  |
| --- | --- | --- | --- |
| **NT** | **NSIM** | **SD** | **SE** |
| 20 | 50000 | 6.06490 | 0.02712 |
| 100 | 50000 | 6.06167 | 0.02711 |
| 300 | 50000 | 6.06452 | 0.02712 |
| 500 | 50000 | 6.04866 | 0.02705 |
| 800 | 50000 | 6.04279 | 0.02702 |
| 500 | 10000 | 6.09454 | 0.06095 |
| 500 | 30000 | 6.05099 | 0.03494 |
| 500 | 50000 | 6.04866 | 0.02705 |
| 500 | 70000 | 6.04451 | 0.02285 |
| 500 | 90000 | 6.05454 | 0.02018 |

3) batch 2: call and put option has the same price

|  |  |  |  |
| --- | --- | --- | --- |
| **NT** | **NSIM** | **SD** | **SE** |
| 20 | 50000 | 13.08450 | 0.05852 |
| 100 | 50000 | 13.12390 | 0.05869 |
| 300 | 50000 | 13.24500 | 0.05923 |
| 500 | 50000 | 13.28840 | 0.05943 |
| 800 | 50000 | 13.22950 | 0.05916 |
| 500 | 10000 | 13.34340 | 0.13343 |
| 500 | 30000 | 13.32090 | 0.07691 |
| 500 | 50000 | 13.28840 | 0.05943 |
| 500 | 70000 | 13.23900 | 0.05004 |
| 500 | 90000 | 13.22240 | 0.04407 |

Fix NSIM and increase NT, SD and SE changes little. While fix NT and increasing NSIM, SD and SE both decrease especially SE.

When SD and SE are lower, the accuracy is higher. SD means the standard deviation of all sample paths. According to the property of Brownian Motion, the option prives of different paths can be normally distributed. When we use a biiger NSIM, thesample distribution is closer to the actual distribution, so the sample SD is closer to the theoretical SD. Since we already have many sample paths, so the SD we already get just change a little bit. When we increase NT, the condition change in the same way.

SE equals to SD/sqrt(M). Since SD barely move, SE is a decrease function according to NSIM. Therefore, when NSIM increases, SE decreases. NT has nothing to do with M so it barely influence SE.