Source | Model | Option
| Model_Option | Help on tr methods | Archived Tests

tr_patry

Input parameters:

- StepNumber N
- HedgeNumber N_Hedge
- CurrentDelta alphacourant
- HedgeSampling N_Sample

Output parameters:

- Delta
- Price
- Variance
- HedgeNow

Among the N possible dates of trading, the hedger will decide to hedge only N_hedge times. We have shown (hat the optimal variance is the solution to a sequence of optimal stopping problems. We design a lattice algorithm to solve the problem in the Cox-Ross-Rubinstein setting. The algorithm is the following: we first calculate the variance of the tracking error for $n = 0$ at every node at every time step working backwards throughout the tree. Using $V_N^{1,\alpha} = 0$, we can compute the error at every node at time N. We can then apply the Dynamic Programming equation to compute the error at time $N - 1$ which is the minimunm of the immediate trading value and the present value of continuing without trading. We can then reapply this procedure at every node at every time step as the step N_Hedge.

/*Memory Allocation*/

/*Up and Down factors*/

/*Risk-Neutral Probability*/

/*FirstStep: Spot, Price, VStarZero (PrevVStar) */

/*Price initialisation*/

/*Backward Resolution*/

/*SecondStep: Vstarn computation*/
We calculate the variance of the tracking error (for n=1 as N_hedge) working backward throughout the tree.

/*CurrentV Initialisation*/

/*We start the computation at time N-n-1*/

/*Compute the new current minimum of Vn(alpha)*/

/*Price*/

/*Delta*/

/*Variance*/

/*HedgeNow*/