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

fd_gauss_downout

Input parameters:

- SpaceStepNumber N
- TimeStepNumber M
- Theta $\frac{1}{2} \leq \theta \leq 1$

Output parameters:

- Price
- Delta

To obtain accurate prices the grid points is located on the barrier, where we impose Dirichlet boundary conditions.there In the american case we use the splitting methods. It seems that it converges very slowly.

/*Memory Allocation*/

/*Time Step*/
Define the time step $k = \dfrac{T}{N}$.

/*Space localisation*/
Define the integration domain $D = [down, l]$ using the probabilistic estimate there.

/*Space Step*/
Define the space step $h = \dfrac{2l}{M}$.

/*Peclet Condition*/
If $|r - \delta| / \sigma^2$ is not small, then a more stable finite difference approximation is used. cf there.

/*Lhs factor of theta scheme*/
Initialize the matrix M^h issued from the discretization of the operator A in the case of Dirichlet Boundary conditions.cf there.

/*Rhs factor of theta scheme*/
Initialize the matrix N issued from the implicit method in the cases of Dirichlet conditions. there

/*Set up Gauss*/
This part concerns the factorization LU of the tridiagonal matrix M^h. The first loop initialize U, whereas the others initialize L.

/*Terminal value*/
Put the value of the payoff saved in *Obst* into a vector P which will be used to save the option value.

/*Dirichlet Boundary Condition*/
We set Dirichlet Boundary conditions on the barrier.

/*Finite difference Cycle*/
At any time step, described by the loop in the variable *TimeIndex*, we have to solve the system $M^h v = NP$.

/*Set rhs*/
Compute NP and save the result in the vector S.

/*Solve the system*/
We solve the system $M^h v = S$ in two steps:

1. First loop consists in solving $L\bar{v} = S$. The result is saved in S.
   there.
2. Second loop consists in solving $Uv = \bar{v} = S$. The result is saved in P.

/*Splitting for American case*/
For American options, we compare at each time step the solution of $M^h v = NP$ saved in P with the payoff function saved in *Obst*. We save the result in P there.

/*Price*/
One uses linear interpolation to find the option value corresponding to the initial stock price.

/*Delta*/
One uses linear interpolation to find the delta value corresponding to the initial stock price. If the initial stock price is close to barrier one uses one-sided second-order difference approximation

$$\frac{\delta u_h}{\delta x}(x_i) = \frac{1}{h^2}(-u_h(x_{i+2}) - 4u_h(x_{i+1}) - 3u_h(x_i))$$

/*Memory Desallocation*/