

[Source](#) | [Model](#) | [Option](#)
[Model_Option](#) | [Help on fd methods](#) | [Archived Tests](#)

fd_gauss

Input parameters:

- SpaceStepNumber N
- TimeStepNumber M
- Theta $\frac{1}{2} \leq \theta \leq 1$
- Boundary Condition: Dirichlet(0) or Neumann(1) *bound*

Output parameters:

- Price
- Delta

/*Natural Dirichlet Boundary Conditions*/

/*Natural Neumann Boundary Conditions*/

/*Memory Allocation*/

/*Time Step*/

Define the time step $k = \frac{T}{N}$.

/*Space localisation*/

Define the integration domain $D = [-l, l]$ using the probabilistic estimate [there](#).

/*Space Step*/

Define the space step $h = \frac{2l}{M}$.

/*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](#).

/*Neumann Boundary Condition*/

The value of variable *bound* indicates the user's choice of boundary condition. If *bound* = 1 then we modify M^h into account the Neumann boundary condition.cf [there](#).

/*Rhs factor of theta scheme*/

Initialize the matrix N issued from the implicit method in the cases of Dirichlet and Neumann Boundary 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.

/*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*/

/*Delta*/

/*Memory Desallocation*/

/*Memory Allocation*/