tr_asian_fsg

Input parameters:

- StepNumber N
- Rho ρ (should be greater than 0)

Output parameters:

- Price
- Delta

This is taken from Hull and White [1] and also Barraquand and Pudet [3]. Here we also prove the convergence of the algorithm, even in the case of American options. It seems that the proof of the algorithm of [3] is a bit fallacious. This is not a strictly mathematical issue since our proof in

On the convergence of the FSG method

which rely on Kushner's local consistency condition, ie Kushner's theorem in

Convergence result for Tree methods in finance

shows that the convergence holds for a discretization of the pathdep direction which is more raffined than that of Barraquand and Pudet. This has also been noticed recently with a more analytic point of view by Forsyth, Vetzal and Zvan in [2].
Let us first describe the idea of forward shooting, which can be applied to numerous pathdep options, not only asian options: consider, in a discrete-time setting to simplify the matter, an option with a payoff of the form $\varphi(S_N, A_N)$ where A is a functional of the trajectory of the underlying S such that the pair (S, A) is Markov and also:

$$A_{n+1} = f(n, A_n, S_{n+1})$$

for some function f. Then the dynamic of the pair (S, A), if S follows the CRR dynamic, for instance, will lie in a 2D space with a one-node two-sons tree (ie there is a single source of randomness, S). In particular, if we build the exact tree of this dynamic it is obvious by the same argument as in CRR that the option may be duplicated, therefore the pricing algorithm in this discrete setting is obvious. This is shown in

The generalized Cox Ross Rubinstein model

Now the tricky point is that, in general (except for cases like the minimum or the maximum), the obtained tree is not recombining so that the straight algorithm is of exponential complexity. A very natural way to overcome this difficulty is to design a new state space in the following way: for every node (n, S_n) of the classical CRR tree (ie in the S direction), first compute the exact minimum and maximum possible values at time n of A along the paths which end at this node (forward step of the building of the tree). Very often these two values are easy to compute explicitly. Next sample the obtained range with a thin enough step so that the obtained algorithm (see below) converge. It is at this very point that our analysis diverges from that of Barraquand and Pudet: they divide the pathdep range in n pieces whereas we prove the convergence (in the asian case) only if $n\sqrt{n}$ are considered. Whatever the number chosen, consider now the classical backward recurrence scheme, at a given node (n, S_n, A_n) of the tree. The two sons will lie somewhere in the $(n + 1, dS_n, .)$ and $(n + 1, uS_n, .)$ axes, but presumably in between two points in the A direction. Nevermind: just perform the corresponding linear interpolation.
In fact in the following tree we take a fixed range in the pathdep direction at every node (n, S_n). The proof of the convergence should work alike.

/*Parameter for the pathdep discretization*/

The input parameter `unsurrho` is multiplied by $\sqrt{n}$ so that the range in the pathdep direction is sampled at each (n, S_n) node with $n\sqrt{n}$ points.

/*Memory Allocation*/

We work simultaneously with the forward prices in `C_n` and the current prices in `C_n_minus_one`

/* Up and Down factors */

Exaclty those of CRR. `expdy` is the multiplicative step in the pathdep direction, `expdz` in the spot direction.

/* Discrete risk-neutral probability */

CRR risk-neutral probability.

/*Ratio for the delta*/

In case of a fixed-strike option, we perform in fact a change of variable with a new initial spot which is `y_1`. It is easily checked that the right delta at the end is given by the formula $\frac{C_u - C_d}{(u-d)*y_1}$. In the floating-strike case, the delta is the usual one $\frac{C_u - C_d}{(u-d)*x}$.

/*Intrinsic value initialisation and terminal values*/

Fixed range in the pathdep direction at every node (n, S_n)

/*Backward resolution*/

Here p is for the upper son, m for the lower one. The `epsilon_` are the interpolation weights.

/* First Step*/

Special handling of the delta, as usual, with `delta_factor` which values `y_1` or `x`, cf above.

/* Memory Desallocation */

References

[1] J.HULL A.WHITE. Efficient procedures for valuing european and american path-dependent options. *The Journal of Derivatives*, 1:21–31, 1993. 1

[2] R.ZVAN P.A.FORSYTH K.R.VETZAL. Convergence of lattice and PDE methods for pricing asian options. *Working Paper Department of Computer Science, University of Waterloo*, 1998. 1

[3] J.BARRAQUAND T.PUDET. The pricing of american path-dependent contingent claims. *Mathematical Finance*, 6(1):17–51, 1996. 1