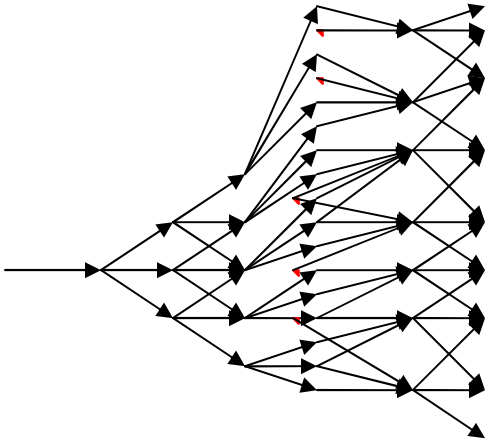
I am studying this code myself with a hope to get a professional job in this area. I have a PhD in Mechanical Engineering with computational mechanics.
I was studying Hull-White numerical solution with Ornstein Uhlenbeck Process for valuation of Bermuda swaption. There are few mandatory times where the price of the Bermuda swaption will be calculated. But the time steps in the discretization of trinomial tree could be different. I believe there is a potential bug in the construction of trinomial tree when time step is smaller compare to average time step. In the following picture, time step dt4 is very small. Because dt4 is very small, the variance is also very small and dx4 (at line 44 of trinomialtree.cpp) are also very small. The expectation of the stochastic process between time t3 and t4 are almost same, but the number of branching is getting increased by a lot (temp at line 50 of trinomialtree.cpp). Some of the nodes at time step 4 are floating i.e. (the red dots in the picture will have state price of zero). However, it increases the unnecessary computational time. Furthermore some of the active nodes at time step 4 will have probability of one going from one node at time step 4 to another time step 5. Even though, the results are not wrong, the code could be improved a lot for computational efficiency while using trinomial tree. It seems to me that there is a problem in computation of "temp" at line 50 of trinomialtree.cpp. There is an adjustment needed for the computation of "temp" so that tree does not get many unnecessary branches when time step is small. Any comment is appreciated.

Trinomial Tree structure with different time steps (dt and dx are not drawn to scale. dt4 and dx4 are much smaller compare to other time step.)



--dt1-----dt2-----dt3--dt4---dt5----dt6---