K_α-Shifting, Rannacher time stepping and mesh grading in Crank-Nicolson FDM for Black-Scholes option pricing

Sima Mashayekhi[*] and Jens Hugger[*]

Abstract

Non-smooth conditions in partial differential equations cause discretization error in numerical schemes and lead to decay in the convergence rate. Here the K_α-shifting method is introduced for easy handling of uniform and nonuniform meshes and for one or more singularities in the terminal condition. Combining this method with Rannacher time stepping and mesh grading for the Crank-Nicolson Finite Difference Method on some examples including call options, bet options and a butterfly spread is shown to lead to higher accuracy and better convergence rate for the numerical solution.

Keywords: Black-Scholes model; Rannacher time stepping; finite difference schemes; Crank-Nicolson scheme; European options

2010 AMS Subject Classification: 65M06; 65M12; 65N06; 65N12

1 Introduction

We consider the well established Black-Scholes model for the pricing of a few standard European vanilla options on a bounded domain:

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r-\gamma) S \frac{\partial V}{\partial S} - rV = 0 \qquad \forall (S,t) \in (0, S_{\max}) \times (0,T) \tag{1}$$

with the terminal and boundary conditions

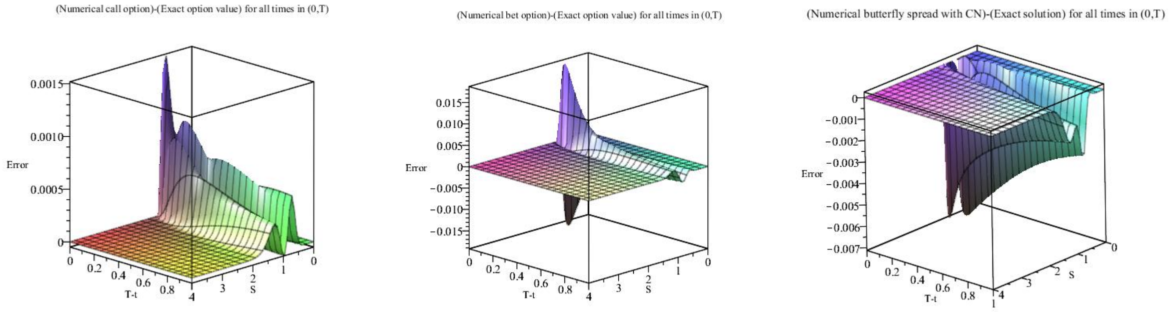
$$V(S,T) = \kappa(S,T), \quad V(0,t) = \kappa(0,t), \quad V(S_{\max},t) \simeq \kappa(S_{\max},t) \tag{2}$$

where we are using the utility function

$$\kappa(S,t) = \begin{cases} \max\{Se^{-\gamma(T-t)} - Ke^{-r(T-t)}, 0\} & \textit{call option} \\ \max\{Ke^{-r(T-t)} - Se^{-\gamma(T-t)}, 0\} & \textit{put option} \\ Be^{-r(T-t)}\mathcal{H}(S-K) & \textit{bet option} \\ \max\{(K+a)e^{-r(T-t)} - Se^{-\gamma(T-t)}, 0\}\mathcal{H}(S-K) & \\ +\max\{Se^{-\gamma(T-t)} - (K-a)e^{-r(T-t)}, 0\}\mathcal{H}(K-S) & \textit{butterfly spread} \end{cases} . \tag{3}$$

$V(S,t)$ is the (fair) option price for a value S of the risky asset at time t. r, σ and γ are the *market interest rate* (on a risk free asset), the *volatility* (of the underlying risky asset) and the *dividend yield* (on the risky asset) respectively. $S_{\max} >> K$ is the upper bound for the computational domain in the S variable and the terminal time T is the upper bound in the t variable. K is the *Strike Price* for the call and bet, B the value of the *Bet* and a is the distance from the strike prices $K \pm a$ of the long options to the strike price K of the two short options in the butterfly spread. $\mathcal{H}$ is the Heaviside function.

[*] Department of Mathematical Sciences, University of Copenhagen, Denmark



(a) $e(S,t)$ for call.

(b) $e(S,t)$ for bet.

(c) $e(S,t)$ for butterfly spread.

Figure 1: Plot of the error $e(S,t)$ as function of $S \in (0, S_{\max})$ and $t \in (0,T)$ for (a) call option, (b) bet option and (c) butterfly spread with $S_{\max} \simeq 4K$ in the *standard case* ($T = 1$, $K = 1$, $a = 0.2$, $B = 0.3$, $r = 0.04$, $\gamma = 0$, $\sigma = 0.2$ and $S_{\max} = 4K$) with the Crank-Nicolson (CN) method for a mesh with $h = 0.08$ and $k = 0.01$.

In this article we provide numerical solutions using the standard Crank-Nicolson (CN) Finite Difference Method (FDM) with a few simple adaptations. Further we compute the Greeks Delta ($\Delta(S,t) = \frac{\partial V}{\partial S}(S,t)$) and Gamma ($\Gamma(S,t) = \frac{\partial^2 V}{\partial S^2}(S,t)$) using second order finite differences, centered in the interior points and one sided at the boundaries. These methods are easy to program and account for the majority of the PDE-methods in use today. The main underlying concept is that we would like to consider simple (if possible a priori) modifications to the in practice most commonly used methods in order to show how to improve results of these methods by simple adjustments without abandoning the methods.

FDM's only provide results in grid points of the finite difference subdivisions. Results in other points are obtained by simple interpolation, typically linear but also higher order interpolations may be used if higher degree of precision is required. This is an issue if a value (or Greek) at a discontinuity is requested. If the discontinuity is a nodal point, derivatives must be defined with care and if not the interpolation in the point must be defined with care. We shall not require singularities to be nodal points since this as we shall show may result in increased error. Instead we refer to interpolation for such values. Using the Finite Element Method (including in the term all projection based methods finding solutions in a finite dimensional subspace of a sufficiently smooth function space) interpolation issues do not exist, but we shall not consider such methods here, as they are still not very common in practice, in particular not methods with enough smoothness to recover for example the Gamma ($\frac{\partial^2 V}{\partial S^2}$) since the Black-Scholes equation naturally leads to weak solutions in H^1 only offering a continuous solution V and one weak derivative ($\frac{\partial V}{\partial S}$).

The discontinuities in the terminal condition or its first derivative seen in (2) lead to decay in the convergence rate of most finite difference numerical schemes for "computable" stepsizes h in the S-variable and k in the t-variable, see for example [1, 2]. This happens also for the Crank-Nicolson (CN) method which is the one that we shall focus on in this article. Typical plots of the error e (CN solution minus exact solution in the nodal points) for call, bet and butterfly spread are shown in figure 1, where the dominating error concentrated around the singularity $S = K$ or singularities $S = K, K \pm a$ respectively is notable. The goal of this work is to investigate how the "size of the bump(s)" can be reduced without abandoning the CN method.

Rannacher [3] introduced a start-up procedure for Crank-Nicolson in which every one or more initial time steps are replaced by two half-timesteps or four quarter-timesteps of implicit Euler scheme in order to achieve the expected second order convergence in the follow up Crank-

Nicolson method since the order of the standard Crank-Nicolson scheme may be reduced all the way down to zero in the case of rough terminal data. This approach, commonly known as Rannacher time stepping, is widely adopted in financial engineering practice and hence will be considered among the simple modifications allowed in this article.

Another approach, see [2, 4], addresses the decay in convergence order by considering the position of the strike price K with respect to the grid points used in the method. It is shown that having K in the middle between two nodal points in a finite difference scheme decreases the oscillations around the strike price when compared to having K located in a nodal point and consequently increases the accuracy of the finite difference method. Pooley et al [4] consider another alternative for reducing error from nonsmooth terminal conditions, namely smoothening of the terminal data either by a simple averaging over half of the cells to the left and right of the nodal point or by a projection (an L^2 projection is suggested) onto a set of continuous piecewise linear Finite Element basis functions. While the repositioning of a singular point can be performed a priori and hence can be implemented in any existing code at very low cost, the smoothening methods require reconstructing a code and thus falls outside the goal of this article to consider only simple adjustments easily applicable to existing code. Instead they are highly relevant when we in the future extend our work to finite element methods (see section 5).

It is also well known (see for example [5, 6]), that an alternative to Rannacher timestepping is nonuniform (exponentially increasing) time steps (or equivalently a square root of time variable change). Such methods show good promises even for singularities as strong as the Dirac delta function and hence can be used also for at least some Greeks. The method requires either a transformation of the problem or schemes accepting nonuniform time steps and hence falls outside the scope of this article and is relegated to future work (see section 5).

In this article we introduce a shifting grid points method (K_α-shifting) which puts the strike price at any preselected position between nodal points. In section 2 we explain in more details the K_α-shifting method for uniform and nonuniforn meshes with one or more singularities in the terminal value and show its effect for some numerical examples. Moreover we consider stability of the optimal choice of K_α with respect to different parameters in the Black-Scholes equation.

In section 3 we compare Crank-Nicolson with and without the K_α-shifting method and with and without Rannacher time stepping. We give results for uniform as well as nonuniform graded meshes.

In section 4 we compare the orders of convergence of these four methods for option prices and the Greeks Δ and Γ.

Finally some concluding remarks and possible future work is discussed in section 5.

2 K_α-shifting

The K_α-shifting method addresses the significance of the location of singular points in the terminal condition in relation to the end points of the S-elements. We consider "reasonable" parameter values $T = 1$, $K = 1$, $a = 0.2$, $B = 0.3$, $r = 0.04$, $\gamma = 0$, $\sigma = 0.2$ and $S_{\max} = 4K$ (denoted the *standard case*) and solve the call, bet and butterfly spread. (The put option is omitted since the put-call-parity makes it somewhat superfluous). For the put, call and bet options the single singularity occurs in $S = K$ whereas for the butterfly spread there are 3 singularities in $K - a$, K and $K + a$.

Consider first the case of uniform meshes with step sizes h in the S-variable and k in the t-variable and the case of one singularity in $S = K$. First the mesh interval containing K (controlled by $\tilde{\imath}_K$) and the relative position of K in this interval (controlled by α) are found from

$$\text{Find } \tilde{\imath}_K, \alpha: \ K - S_{\min} = (\tilde{\imath}_K + \alpha)\tilde{h} \text{ for some } \tilde{\imath}_K \in \mathcal{N} \text{ and } 0 \leq \alpha < 1, \tag{4}$$

where $S_{\min}$ denotes the left endpoint of the computational S-domain which is 0 in our case, but may be $\neq 0$ in the generalizations of the K_α-shifting method below. Then $\tilde{h}$ is adjusted to h using

$$\text{Find } i_K, h: \; K - S_{\min} = (i_K + K_\alpha)h \text{ for some } i_K \in \mathcal{N}: \; \tilde{\imath}_K \leq i_K \leq \tilde{\imath}_K + 1. \tag{5}$$

i_K is given by

$$\begin{aligned} i_K &= \left\lceil \frac{K - S_{\min}}{\tilde{h}} - K_\alpha \right\rceil = \left\lceil \frac{K - S_{\min}}{\tilde{h}} - \alpha + (\alpha - K_\alpha) \right\rceil = \lceil \tilde{\imath}_K + (\alpha - K_\alpha) \rceil \\ &= \tilde{\imath}_K + \lceil \alpha - K_\alpha \rceil \in [\tilde{\imath}_K, \tilde{\imath}_K + 1], \end{aligned} \tag{6}$$

and hence

$$K - S_{\min} = (\left\lceil \frac{K - S_{\min}}{\tilde{h}} - K_\alpha \right\rceil + K_\alpha)h \Leftrightarrow h = \frac{K - S_{\min}}{\left\lceil \frac{K - S_{\min}}{\tilde{h}} - K_\alpha \right\rceil + K_\alpha}. \tag{7}$$

Note that the new S step size h is given by a simple updating formula from the known input parameters $\tilde{h}$ and K_α without actually ever computing $\tilde{\imath}_K$ and α. Also h is close to $\tilde{h}$ since

$$\begin{aligned} & \tilde{\imath}_K h \leq i_K h \leq K - S_{\min} \leq (\tilde{\imath}_K + 1)\tilde{h} \\ & \text{and } (\tilde{\imath}_K + 2)h \geq (i_K + 1)h \geq K - S_{\min} \geq \tilde{\imath}_K \tilde{h} \\ \Downarrow \; & \\ & \frac{\tilde{\imath}_K}{\tilde{\imath}_K + 2}\tilde{h} \leq h \leq \frac{\tilde{\imath}_K + 1}{\tilde{\imath}_K}\tilde{h}. \end{aligned} \tag{8}$$

For very coarse meshes the adjustment of the S step size may be substantial, like $\frac{h}{\tilde{h}} \in [0.83, 1.1]$ for K situated in the 10'th interval ($\tilde{\imath}_K = 10$) but for more realistic meshes, the adjustment is minimal, like $\frac{h}{\tilde{h}} \in [0.98, 1.01]$ for K situated in the 100'th interval ($\tilde{\imath}_K = 100$).

Two further adjustment must be made, that are not part of the K_α-shifting method, but are necessary in order to adjust $S = \tilde{S}_{\max}$ (the user requested maximal S value in the computational domain) and $t = 0$ to be nodal points. First $\tilde{S}_{\max}$ is adjusted (increased) to $S_{\max}$ lying in the nodal point (in the S-variable) closest to but at least as big as $\tilde{S}_{\max}$ using

$$S_{\max} - S_{\min} = \left\lceil \frac{\tilde{S}_{\max} - S_{\min}}{h} \right\rceil h \geq \tilde{S}_{\max} - S_{\min}. \tag{9}$$

Finally $\tilde{k}$ is adjusted (reduced) to k so that $t = 0$ is a nodal point (in the t-variable) using

$$k = \frac{T}{\left\lceil \frac{T}{\tilde{k}} \right\rceil} \leq \tilde{k} \text{ and } T - \left\lceil \frac{T}{\tilde{k}} \right\rceil k = 0 \text{ where } \left\lceil \frac{T}{\tilde{k}} \right\rceil \in \mathcal{N}. \tag{10}$$

These adjustments ($\tilde{h} \to h$, $\tilde{S}_{\max} \to S_{\max}$ and $\tilde{k} \to k$) are simple update formulas and hence cheap ($\mathcal{O}(1)$) that do not deteriorate the performance of the solution process and can be performed a priory and hence used with any existing code. They may result in slightly fluctuating errors when the requested step sizes are large and hence also the adjustments are potentially large. For "reasonable" step sizes however the results of the adjustments are negligible. Instead with the K_α-shifting method there are no fluctuation in the error caused by K "moving around" inside the i_K'th interval when adjusting the mesh interval size. This turns out to be a significant advantage in practical use, since the error from K moving around is significant (up to a factor of more than 10 for the maximal error).

The K_α-shifting method for uniform meshes easily generalizes to more than one singularity. Just divide the S domain into patches each containing one of the singularities. For each patch

— starting from the left with the patch containing $S = 0$ — compute the adjusted step size and adjust the right endpoint of the patch to be a nodal point with the adjusted step size. In (4)–(9) just use the left patch endpoint as $S_{\min}$, the right endpoint of the patch as $\tilde{S}_{\max}$ and the adjusted right endpoint of the patch as $S_{\max}$. For small requested stepsize $\tilde{h}$ all the actual stepsizes will be very close in size, so that even uniform finite difference approximations will give good results in particular because patch boundaries are situated in areas where the computed solution is almost linear, but otherwise nonuniform finite differences across the patch boundaries may be used. The only issue is that for more than one singularity the method cannot be performed entirely a priory since it requires the ability to work with slightly different stepsizes in different parts of the domain and preferably also with nonuniform finite difference approximations across the patch boundaries, which a standard uniform mesh code will not be able to handle.

For nonuniform meshes constructed by a grading function the idea would be the following for a single singularity in $S = K$: If K is contained in the element number $[S_{i_K}, S_{i_K+1}[$ then simply relocate this element without resizing it to say $[S_0, S_1[$ so that K moves into K_α-position in the element. This relocation is then followed by a uniform scaling of the rest of the elements. The global scaling factors s_{K-} and s_{K+} for the elements before and after K respectively are given by

$$s_{K-} = \frac{S_0 - S_{\min}}{S_{i_K} - S_{\min}}, \quad s_{K+} = \frac{S_{\max} - S_1}{S_{\max} - S_{i_K+1}}, \tag{11}$$

so that the size of all elements before K are multiplied by s_{K-} and the size of all elements after K are multiplied by s_{K+}. A simpler alternative would be simply to use the K_α-shifting method for uniform meshes on the uniform mesh being graded. For small elements the grading function will be sufficiently close to linear to put the singularity close enough to the K_α-position.

For adaptively constructed nonuniform meshes with one singularity in $S = K$ the idea would be very similar to the first one for the grading function approach: If the element $[S_{\hat{i}_K}, S_{\hat{i}_K+1}[$ containing K in K_α-position is up for subdivision — let us for simplicity say uniform splitting into two equal elements — then the new elements are constructed, and the new element $[S_{i_K}, S_{i_K+1}[$ containing K (either $[S_{\hat{i}_K}, \frac{S_{\hat{i}_K}+S_{\hat{i}_K+1}}{2}[$ or $[\frac{S_{\hat{i}_K}+S_{\hat{i}_K+1}}{2}, S_{\hat{i}_K+1}[$) is relocated (but not resized), say to $[S_0, S_1[$ so that K is again in K_α-position in this element. This relocation is followed by global scalings of the elements to the left and right as for the grading function approach, using the scaling factors s_{K-} and s_{K+} defined in (11).

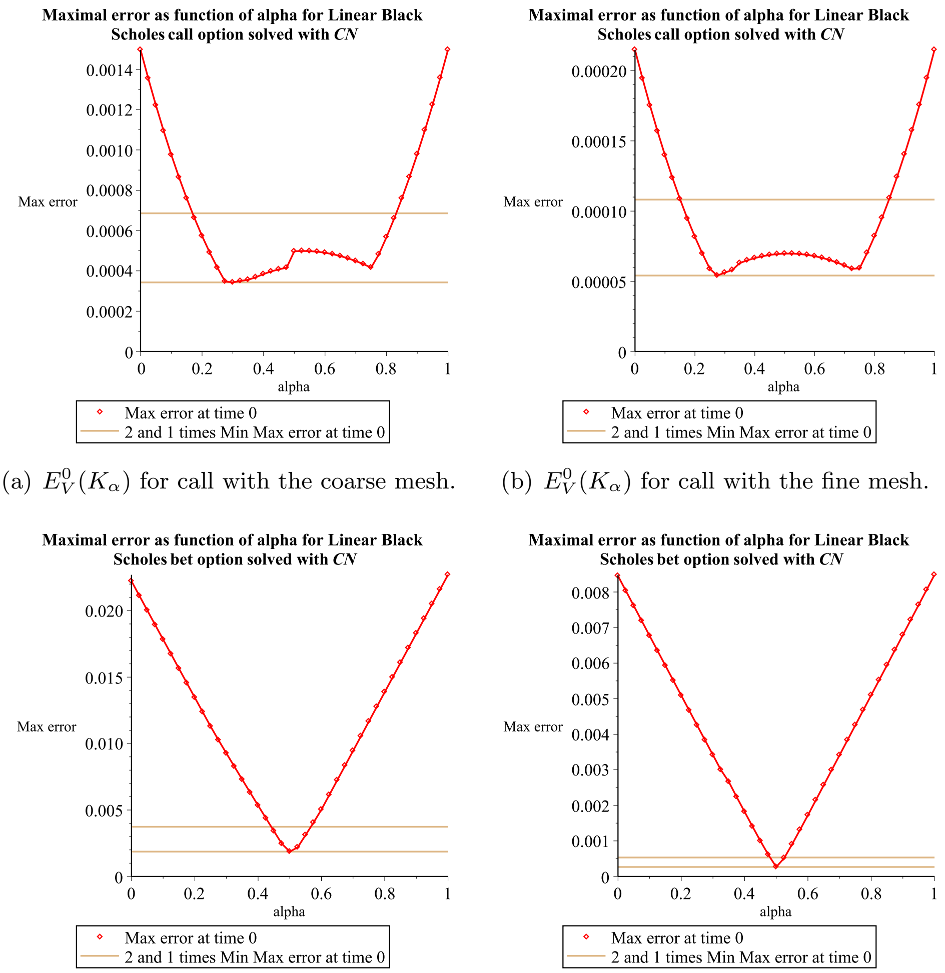
If finally $N > 1$ singularities are present with nonuniform meshes then N patches each containing exactly one singularity are constructed and each patch is scaled with individual scaling factors moving from the left to the right. If the nonuniform meshes are created with a grading function, the simple approach also generalizes. Just use the K_α-shifting method for uniform meshes with several singularities explained above on the uniform mesh being graded.

Turning to the computational examples, instead of using $\tilde{h}$, $\tilde{k}$ and $\tilde{S}_{max}$ we shall use the notation $h \simeq \ldots$, $k \simeq \ldots$ and $S_{\max} \simeq \ldots$ to account for the adjustments. For given values of all parameters we compute maximal absolute solution errors at time $t = 0$ over all S nodal points $S_1, \ldots, S_M$ as

$$E_V^0 = \max_{i=1\ldots,M} |V_{FDM}(S_i, 0) - V^{BS}(S_i, 0)| \tag{12}$$

where $V_{FDM}(S_i, 0)$ is the computed finite difference solution in the nodal point $S = S_i$ and $t = 0$ and $V^{BS}(S_i, 0)$ is the exact (Black-Scholes) solution in the same point. Similarly we define the maximal absolute errors E_Δ^0 and E_Γ^0 for the Greeks Δ and Γ.

In Figure 2 we show the maximal absolute solution errors $E_V^0(K_\alpha)$ at time $t = 0$ for two different sets of step sizes $(h, k) \simeq (0.08, 0.01)$ and $(h, k) \simeq (0.03, 0.001)$ and as a function of 41 different K_α-values uniformly distributed from 0 to 1 for the call and the bet option solution values. When $K_\alpha = 0$ or 1 (or whenever $S = K$ is a nodal point) it becomes a numerical issue how to define $V(K, T)$ for the bet option. After some experimentation we have decided



(a) $E^0_V(K_\alpha)$ for call with the coarse mesh.

(b) $E^0_V(K_\alpha)$ for call with the fine mesh.

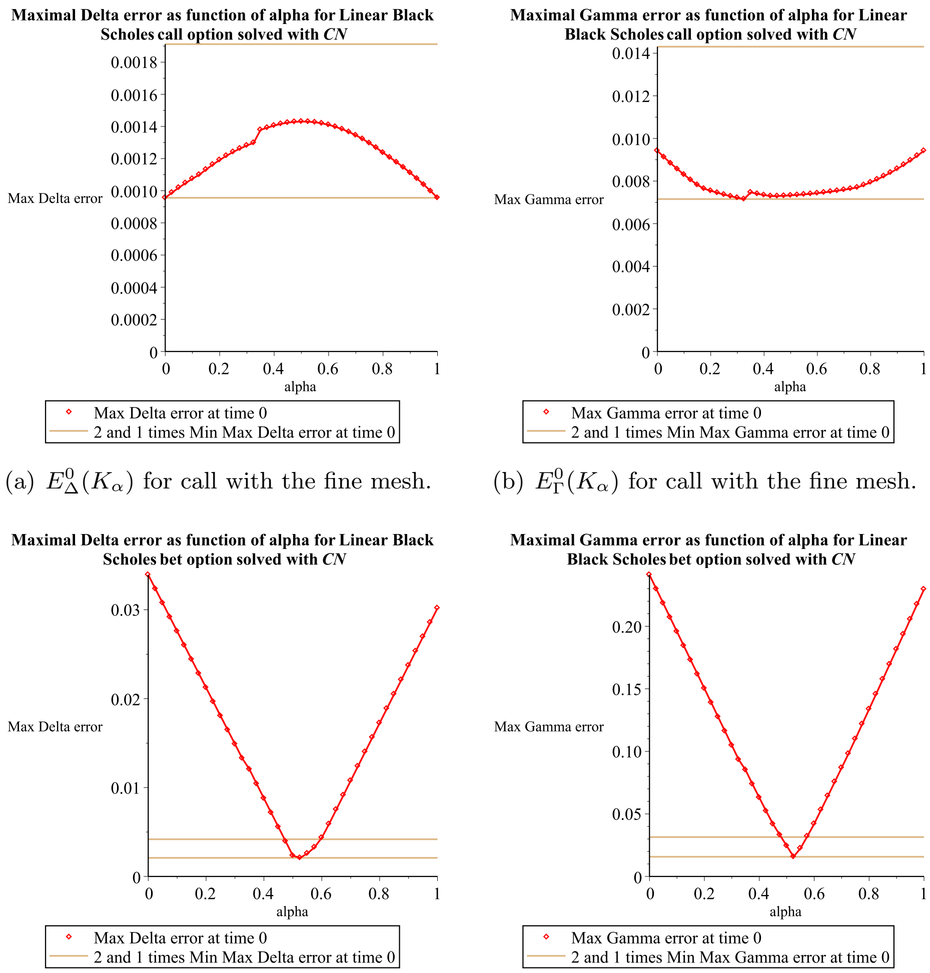
(c) $E^0_V(K_\alpha)$ for bet with the coarse mesh.

(d) $E^0_V(K_\alpha)$ for bet with the fine mesh.

Figure 2: Maximal error $E^0_V(K_\alpha)$ at time $t = 0$ as function of $K_\alpha \in [0, 1]$ for call and bet options in the standard case solved with CN using the coarse mesh $(h, k) \simeq (0.08, 0.01)$ and the fine mesh $(h, k) \simeq (0.03, 0.001)$.

to use the convention $V(K, T) = 0$ for $K_\alpha < 0.5$ and $V(K, T) = B$ for $K_\alpha \geq 0.5$ giving the smoothest graphs. We are interested in $\hat{K}_\alpha$, the optimal K_α, minimizing E^0_V over all values of $K_\alpha \in [0, 1[$. $\hat{K}_\alpha = 0.27$ turns out to be the optimal choice for the call option at time $t = 0$ (see Figures 2(a) and 2(b)) varying from 0.280 for the coarse mesh to 0.264 for the fine mesh when computed with 1001 uniformly distributed K_α-values from 0 to 1. Also we observe that the symmetric position $K_\alpha = 1 - 0.27$ is quite good, and actually the entire interval $(0.2, 0.8)$ gives good results (at most the double maximal absolute solution error compared to the optimal location). The general conclusion is that for the call option (and similarly for the put) the strike price K should under no circumstances be located close to a nodal point.

Figures 2(c) and 2(d) show that $\hat{K}_\alpha = 0.50$ is the optimal choice for the bet option at time $t = 0$, varying from 0.508 for the coarse mesh to 0.504 for the fine mesh when computed with 1001 K_α-values. Unsurprisingly $\hat{K}_\alpha = 0.50$ is optimal also when solving with graded meshes. Hence the best location for the strike price is in the middle between two consecutive nodal points. The interval where the maximal error is at most the double of the optimal error is $(0.4, 0.6)$ and hence significantly smaller for the bet option than for the call. Also the price for locating the strike price closer to a nodal point is significantly bigger for the bet than for the



(a) $E^0_\Delta(K_\alpha)$ for call with the fine mesh.

(b) $E^0_\Gamma(K_\alpha)$ for call with the fine mesh.

(c) $E^0_\Delta(K_\alpha)$ for bet with the fine mesh.

(d) $E^0_\Gamma(K_\alpha)$ for bet with the fine mesh.

Figure 3: Maximal error $E^0_\Delta(K_\alpha)$ and $E^0_\Gamma(K_\alpha)$ at time $t = 0$ for the Greeks Δ and Γ for the call and bet options as function of $K_\alpha \in [0, 1]$ in the standard case solved with CN using the fine mesh $(h, k) \simeq (0.03, 0.001)$.

call option. The general conclusion is that for the bet option the strike price K should under no circumstances be located close to a nodal point.

Figure 3 shows the maximal absolute errors at time $t = 0$ for the Greeks Δ and Γ for the call and bet options with the fine mesh $(h, k) \simeq (0.03, 0.001)$ and for 41 different K_α-values uniformly distributed from 0 to 1. Computing with 1001 uniformly distributed K_α-values from 0 to 1 we get the following optimal $\hat{K}_\alpha$-values: For the call option $\hat{K}_\alpha = 0.00$ and $\hat{K}_\alpha = 0.33$ for the Delta and Gamma respectively. Note however, that the value of K_α is of little importance for the Greeks of the call option, all minimal errors lying within a factor significantly below 2 from the smallest value. For the bet option $\hat{K}_\alpha = 0.51$ and $\hat{K}_\alpha = 0.53$ for the Delta and Gamma respectively. For the bet option the value of K_α is important also for the Greeks, the factor two interval being as small as $[0.48, 0.58]$. Concluding, for the bet option $K_\alpha = 0.5$ is the sensible choice for both the value, the Delta and the Gamma, whereas for the call K_α should be picked in the interval $[0.2, 0.8]$ and might be picked at $K_\alpha = 0.5$ without an increase of more than a factor 2 in the maximal error for the value, the Delta and the Gamma. The small irregularities visible in Figures 2–3 (for the coarse mesh solution of the call at $K_\alpha \simeq 0.5$ and for the fine mesh solutions of the call Greeks at $K_\alpha \simeq 0.35$) originate from various numerical "issues" related to the computation of either numerical or exact values. Since the irregularities do not influence the conclusions we have not investigated the exact causes in each case.

The butterfly spread requires 3 optimal K_α-values denoted $\hat{K}_\alpha^1$, $\hat{K}_\alpha^2$ and $\hat{K}_\alpha^3$ for the singularities $K-a$, K and $K+a$ respectively. The corresponding 3 patches are chosen a priori as $[0, K-\frac{a}{2}]$, $[K-\frac{a}{2}, K+\frac{a}{2}]$ and $[K+\frac{a}{2}, \tilde{S}_{\max}]$ and then adjusted by the K_α-shifting method. For different values of N we have computed with N each of K_α^1-, K_α^2- and K_α^3-values uniformly distributed in $[0,1]$ for a total of N^3 cases. Also we have computed for the coarse as well as for the fine mesh as defined above. Given the 3-dimensional parameter space $(K_\alpha^1, K_\alpha^2, K_\alpha^3) \in [0,1]^3$ visualization of the results is somewhat challenging, so here we only show the results in tabular form in Table 1. For comparison we have also in Table 1 given the errors for the two most likely cases $K_\alpha = [0,0,0]$ which would likely occur if no thought is given to the location of the singularities (typically integer multiple of decimal steplengths) and $K_\alpha = [0.5, 0.5, 0.5]$ which would likely occur if it was decided to put the singularities in a fixed position different from nodal points without considering optimality of the position. First of all the results show that K_α-optimization gives a significant reduction with a factor from 5 to 30 in the error in the solution, Δ and Γ when compared to selecting $K_\alpha = [0,0,0]$. When compared to $K_\alpha = [0.5, 0.5, 0.5]$ we still record a significant reduction in the error with an improvement of more than a factor 10 for the solution, less but still with a factor of about 5 for the Δ and least but still with a factor of around 2 for the Γ. This confirms our previous results that the K_α-optimization is less important for the Greeks than for the solution. For Γ basically any selection of K_α apart from putting the singularities in (or close to) nodal points is good.

All the results of this section indicate that positioning the strike price near the middle of a mesh interval might be a good although conservative approach giving reasonable results for many options. If looking for the very best the positioning of the strike price must be taking into consideration also the type of option.

For the K_α-optimization to be useful in practice it would need to be fairly stable against variations in the parameters. So next we investigate whether the conclusions depend on the particular selection of model parameters above. Here interest rate r and volatility σ are deemed the most important parameters, whereas T, B and K basically can be considered scaling parameters without much significance and the dividend yield γ is expected to behave like some sort of additional interest rate, a constant γ not creating new features by itself. Hence in the following two subsections we shall consider variations of the optimal K_α with interest rate r and volatility σ respectively.

2.1 Stability of K_α with respect to the interest rate

We redo the computations from Figure 2 only adding a third axis with the interest rate $r \in [-0.1, 0.1]$. Negative interest rates are considered since interest rates in Europe has fallen very close to zero after the financial crisis in 2010 and there has been discussions of whether negative interest rates were necessary in order to spawn investment in "growth" i.e. in risky assets.

Two typical results for the fine mesh are shown in Figure 4 where we have computed with equidistant r-values with the same difference 0.025 as is used for the K_α-values. For the call option the optimal K_α is situated in $(0.2, 0.3) \cup (0.7, 0.8)$. Further a K_α in the extended interval $(0.2, 0.8)$ only changes the minimal error for the call option by a factor of up to 2 whereas a K_α outside this interval may change the minimal error for the call option by a factor of up to 4.

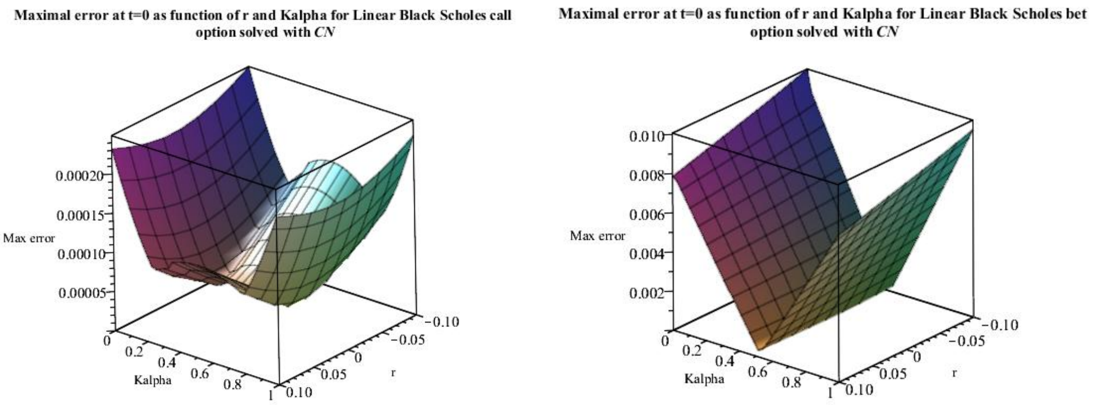
For the bet option the optimal K_α is situated in $(0.45, 0.55)$ and any K_α in this interval only changes the minimal error for the bet option by a factor of up to 2 whereas a K_α outside this interval may change the minimal error for the bet option by a factor of up to 10.

Summing up, the conclusions from section 2 hold for all values of r and t. A reasonable

Table 1: Optimal K_α-values for solution, Δ and Γ for the butterfly spread in the standard case solved with CN using the coarse mesh $(h,k) \simeq (0.08, 0.01)$ and the fine mesh $(h,k) \simeq (0.03, 0.001)$.

Mesh	No. K_α's	$\hat{K}_\alpha^1$	$\hat{K}_\alpha^2$	$\hat{K}_\alpha^3$	$\min_{K_\alpha} E_V^0(K_\alpha)$	$\frac{\max_{K_\alpha} E_V^0(K_\alpha)}{\min_{K_\alpha} E_V^0(K_\alpha)}$
Coarse V	1	0.00	0.00	0.00	0.016762	1
	1	0.50	0.50	0.50	0.009338	1
	11^3	0.50	0.33	0.33	0.000765	34.928762
	21^3	0.55	0.30	0.20	0.000622	46.712522
	41^3	0.53	0.28	0.18	0.000595	49.446807
Fine V	1	0.00	0.00	0.00	0.004325	1
	1	0.50	0.50	0.50	0.001495	1
	11^3	0.60	0.50	0.40	0.000105	93.523044
	21^3	0.60	0.50	0.40	0.000105	93.523044
	41^3	0.50	0.38	0.25	0.000091	–[a]
Coarse Δ	1	0.00	0.00	0.00	0.114845	1
	1	0.50	0.50	0.50	0.039337	1
	11^3	0.40	0.00	0.60	0.008357	29.840102
	21^3	0.45	0.05	0.65	0.008320	30.770937
	41^3	0.38	0.00	0.65	0.007997	34.449918
Fine Δ	1	0.00	0.00	0.00	0.020239	1
	1	0.50	0.50	0.50	0.006166	1
	11^3	0.20	0.00	0.80	0.001466	24.887289
	21^3	0.30	0.10	0.85	0.001354	26.942537
	41^3	0.28	0.08	0.85	0.001333	–[a]
Coarse Γ	1	0.00	0.00	0.00	0.385213	1
	1	0.50	0.50	0.50	0.186919	1
	11^3	0.40	0.10	0.90	0.116449	6.817861
	21^3	0.40	0.10	0.85	0.108821	7.503789
	41^3	0.40	0.10	0.85	0.108821	7.848675
Fine Γ	1	0.00	0.00	0.00	0.086126	1
	1	0.50	0.50	0.50	0.030656	1
	11^3	0.50	0.30	0.00	0.017846	6.569529
	21^3	0.55	0.35	0.05	0.017247	7.348245
	41^3	0.53	0.35	0.10	0.016665	–[a]

[a] Only selected subintervals of K_α are computed.



(a) $E_V^0(K_\alpha, r)$ for call with the fine mesh.

(b) $E_V^0(K_\alpha, r)$ for bet with the fine mesh.

Figure 4: Maximal error $E_V^0(K_\alpha, r)$ at time $t = 0$ as function of $K_\alpha \in [0, 1]$ and $r \in [-0.1, 0.1]$ for (a) the call and (b) the bet option in the standard case except for r solved with CN using the fine mesh $(h, k) \simeq (0.03, 0.001)$.

conservative choice is to pick $K_\alpha = 0.5$, but for the call option a more refined choice would be to select $K_\alpha = 0.725$ for negative interest rates and 0.275 for positive interest rates. Especially for the bet option a selection of $K_\alpha = 0$ is somewhat disastrous and should be avoided whether by choice or accident.

2.2 Stability of K_α with respect to the volatility

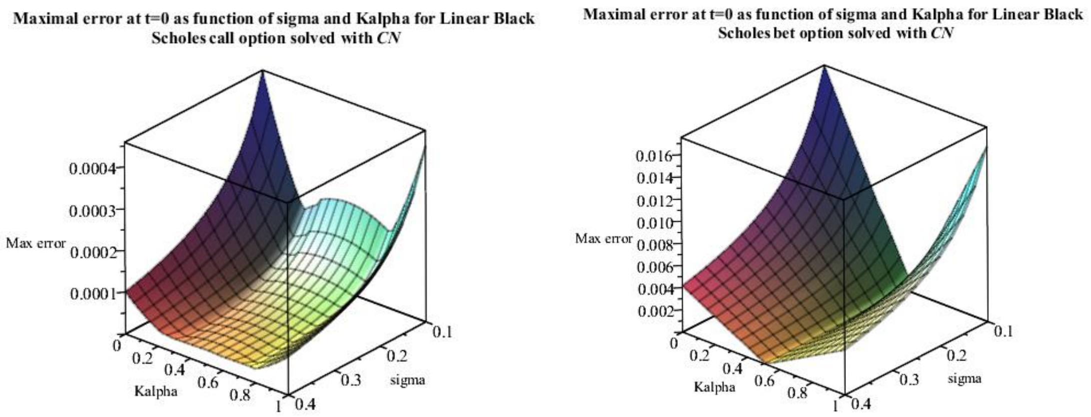
We redo the computations from Figure 2 this time adding a third axis with the volatility $\sigma \in (0.1, 0.4)$. Two results are shown in Figure 5. The conclusions are the same as before: For the call option the optimal K_α shift for some σ-values from a "lower" value close to the 0.275 observed typically for small values of σ to an "upper" value close to the symmetric value $0.725 = 1 - 0.275$ observed typically for large values of σ. The optimal K_α is situated in $(0.1, 0.3) \cup (0.7, 0.9)$. Further a K_α in the extended interval $(0.1, 0.9)$ only changes the minimal error for the call option by a factor of up to 2 whereas a K_α outside this interval may change the minimal error for the call option by a factor of up to 4.

The optimal K_α for the bet option is still located solidly in 0.5 except for a few cases with small σ, and for $K_\alpha \in (0.4, 0.6)$ the maximal error is at most the double of the minimal value of the maximal error, whereas a K_α outside this interval may change the minimal error for the bet option by a factor of up to 10.

Summing up, the conclusions from section 2 hold for all values of σ and r. A reasonable conservative choice is to pick $K_\alpha = 0.5$, but for the call option a more refined choice would be to select $K_\alpha = 0.725$ for large volatilities and 0.275 for small volatilities. Especially for the bet option a selection of $K_\alpha = 0$ is somewhat disastrous and should be avoided whether by choice or accident.

3 K_α-shifting, Rannacher time stepping and mesh grading

Recall Figure 1 showing the CN error as a function of $S \in (0, S_{\max})$ and $t \in (0, T)$. In this section we focus on reducing the size of the "bump" in the error close to $S = K$ for all values of $t \in (0, T)$. As in section 2 we consider the 2D-slice $t = 0$ but limit to the bet option with its larger error and hence bigger room for improvement than the call option. We shall investigate the error in the Greeks Δ and Γ as well as the error in the solution and compare the base results



(a) $E_V^0(K_\alpha, \sigma)$ for call with the fine mesh.

(b) $E_V^0(K_\alpha, \sigma)$ for bet with the fine mesh.

Figure 5: Maximal error $E_V^0(K_\alpha, \sigma)$ at time $t = 0$ as function of $K_\alpha \in [0, 1]$ and $\sigma \in [0.1, 0.4]$ for (a) the call and (b) the bet option in the standard case except for σ solved with CN using the fine mesh $(h, k) \simeq (0.03, 0.001)$.

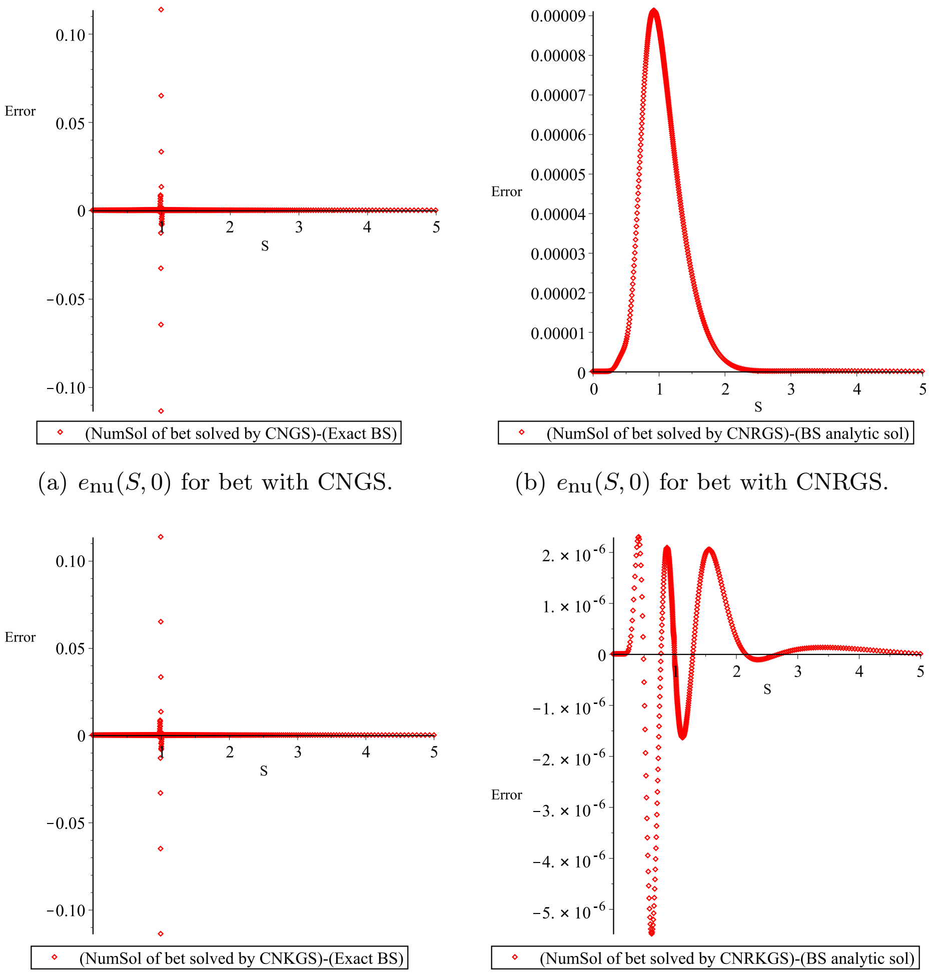
for the standard CN method to results obtained with CN with a Rannacher startup phase [CNR], CN with the optimal K_α [CNK_α] and a new combination of CN with both a Rannacher startup phase as well as the optimal K_α [CNRK_α]. We compute with a uniform mesh but for the solution errors we also show results computed with a nonuniform mesh (method suffix GS for Grid Stretching, eg. CNGS) created with the mesh grading transformation

$$S(x) = K + \frac{1}{b}\sinh(c_1(1-x) + c_2 x) \text{ with } \begin{cases} c_1 = \operatorname{arc}\sinh(-bK) \\ c_2 = \operatorname{arc}\sinh(b(S_{\max} - K)) \end{cases} \tag{13}$$

changing a uniform mesh in $x \in [0, 1]$ with stepsize dx into a nonuniform mesh in $S \in [0, S_{\max}]$. The grading of the S-mesh depends on the grading parameter b which we take to $b = 15$ (see [2, 7, 8]). The maximal absolute solution error (see (12)) with nonuniform meshes is denoted $E^0_{V,\mathrm{nu}}$.

We consider the Rannacher method in the form where the first iteration of the Crank-Nicolson method is replaced by four quarter-timesteps of the implicit Euler scheme. Giles et al (2006) have shown that four quarter-timesteps of the implicit Euler method replacing the first CN step is more accurate than replacing the first two CN steps by four half-timesteps of implicit Euler due to a reduction of the low wavenumber error introduced by the Rannacher startup. Giles et al do so using an $x = \log S$ transformation of the S-variable and no transformation of the time variable giving a reasonable expectation that the conclusion will hold also without the transformation of the S-variable which we shall not apply here. Moreover Giles et al have shown that choosing $\lambda^\star = \frac{k\sigma}{h\sqrt{2T}} \in [0.5, 1]$ causes maximum accuracy for a given computational cost. This result is not expected to carry over to our case but keeping λ^* the same as in the Giles et al paper gives a good basis for comparison since $\log S$ is almost linear in the most interesting region around $S = K = 1$.

For comparison we consider the same parameter values as chosen by Giles et al: $T = 2$, $K = 1$, $B = 0.3$, $r = 0.05$, $\gamma = 0$, $\sigma = 0.2$ and $S_{max} \simeq 5K$ (denoted the *Giles case*). Also we take $\lambda^\star = 0.5$ corresponding to $k = 5h$ and $h = 0.01$. (For the nonuniform meshes $dx = 0.01/S_{\max}$ to get the same number of elements in S). While h and k are not disclosed in [9], it is evident from [9, Fig. 1-2] that also they do consider the worst possible case ($K_\alpha = 0$). In Figures 6–9 we show the solution, Delta and Gamma errors at time $t = 0$ for the bet option with CN, CNR, CNK_α and CNRK_α. In Table 2 the maximal errors of the various cases considered in Figures 6–9 are listed.



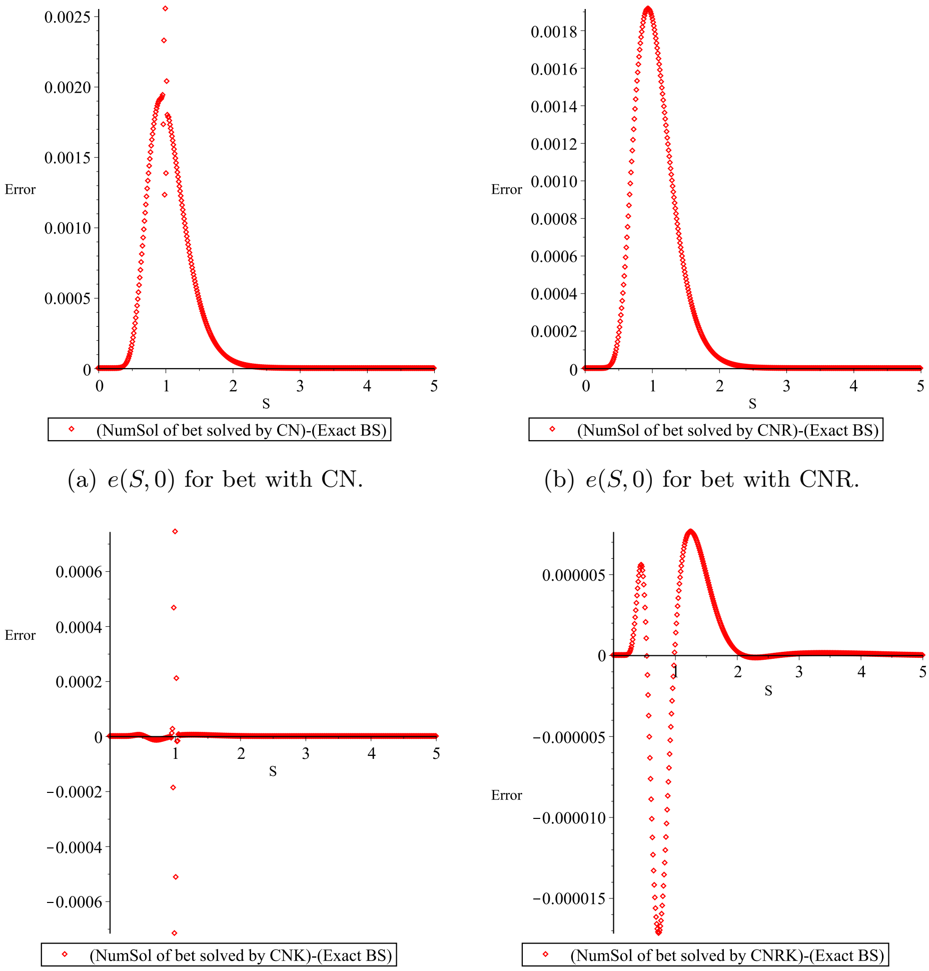
(a) $e_{\mathrm{nu}}(S, 0)$ for bet with CNGS.

(b) $e_{\mathrm{nu}}(S, 0)$ for bet with CNRGS.

(c) $e_{\mathrm{nu}}(S, 0)$ for bet with CNK_αGS.

(d) $e_{\mathrm{nu}}(S, 0)$ for bet with CNRK_αGS.

Figure 6: Solution error $e_{\mathrm{nu}}(S, 0)$ at time $t = 0$ as function of $S \in (0, S_{\max})$ with a mesh graded by (13) with $b = 15$ for the bet option in the Giles case with $dx = 0.01/S_{\max}$ and $k = 0.05$ solved with (a) CNGS with $K_\alpha = 0$, (b) CNRGS with $K_\alpha = 0$, (c) CNK_αGS with $K_\alpha = 0.5$ and (d) CNRK_αGS with $K_\alpha = 0.5$.



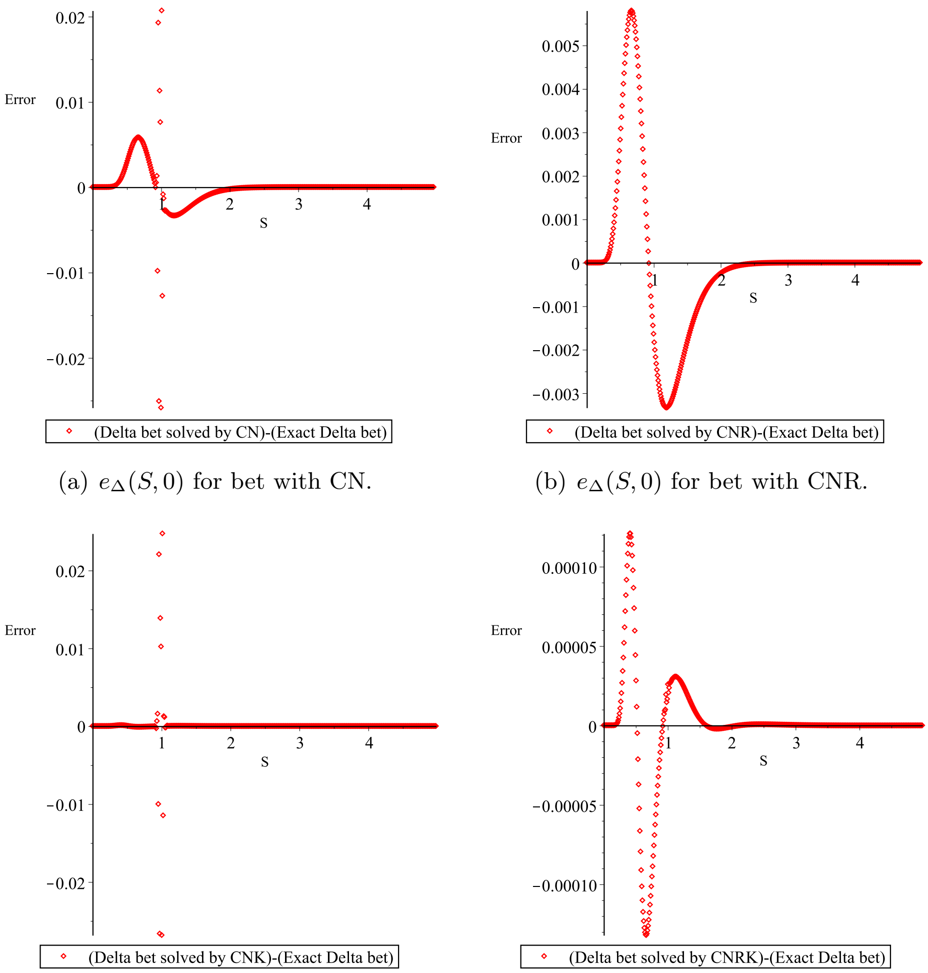
(a) $e(S,0)$ for bet with CN.

(b) $e(S,0)$ for bet with CNR.

(c) $e(S,0)$ for bet with CNK_α.

(d) $e(S,0)$ for bet with CNRK_α.

Figure 7: Solution error $e(S,0)$ at time $t=0$ as function of $S \in (0, S_{\max})$ for the bet option in the Giles case with $h = 0.01$ and $k = 5h$ solved with (a) CN with $K_\alpha = 0$, (b) CNR with $K_\alpha = 0$, (c) CNK_α with $K_\alpha = 0.5$ and (d) CNRK_α with $K_\alpha = 0.5$.



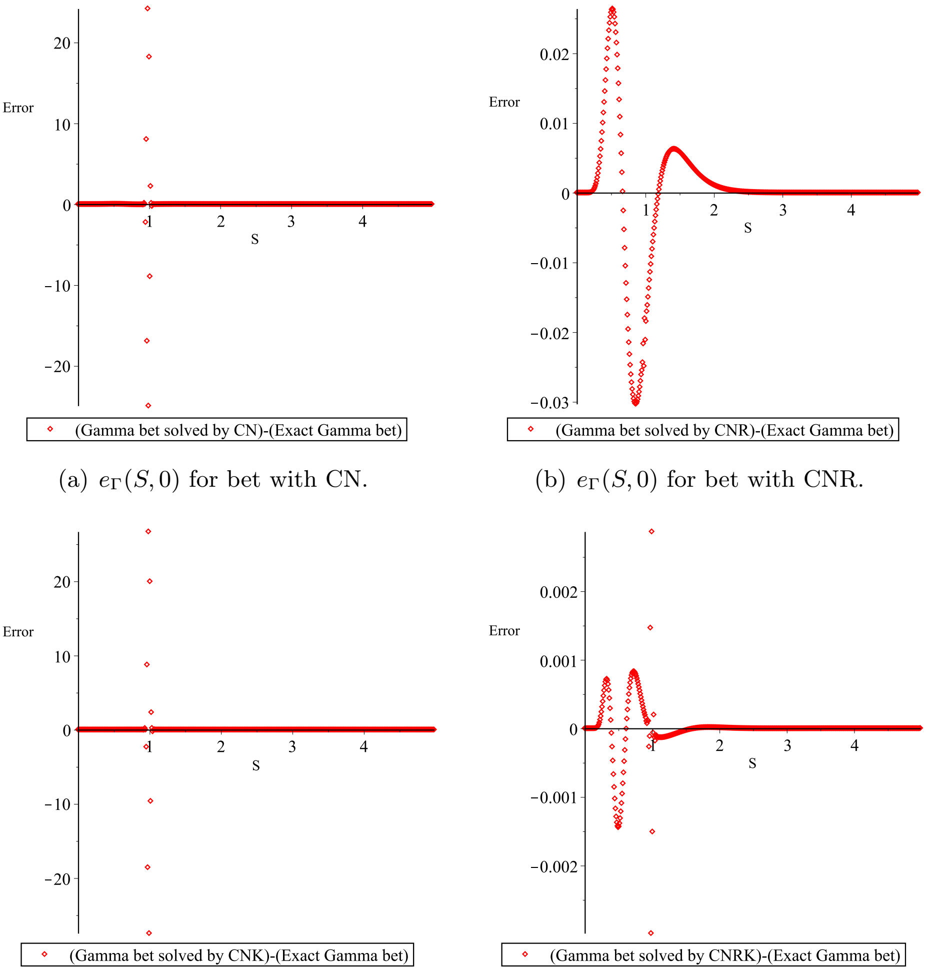
(a) $e_\Delta(S, 0)$ for bet with CN.

(b) $e_\Delta(S, 0)$ for bet with CNR.

(c) $e_\Delta(S, 0)$ for bet with CNK_α.

(d) $e_\Delta(S, 0)$ for bet with CNRK_α.

Figure 8: Delta error $e_\Delta(S, 0)$ at time $t = 0$ as function of $S \in (0, S_{\max})$ for the bet option in the Giles case with $h = 0.01$ and $k = 5h$ solved with CN with $K_\alpha = 0$ (a), CNR with $K_\alpha = 0$ (b), CNK_α with $K_\alpha = 0.5$ (c) and CNRK_α with $K_\alpha = 0.5$ (d).



(a) $e_\Gamma(S, 0)$ for bet with CN.

(b) $e_\Gamma(S, 0)$ for bet with CNR.

(c) $e_\Gamma(S, 0)$ for bet with CNK_α.

(d) $e_\Gamma(S, 0)$ for bet with CNRK_α.

Figure 9: Gamma error $e_\Gamma(S, 0)$ at time $t = 0$ as function of $S \in (0, S_{\max})$ for the bet option in the Giles case with $h = 0.01$ and $k = 5h$ solved with (a) CN with $K_\alpha = 0$, (b) CNR with $K_\alpha = 0$, (c) CNK_α with $K_\alpha = 0.5$ and (d) CNRK_α with $K_\alpha = 0.5$.

Table 2: The maximal solution, Delta and Gamma errors over $S \in (0, S_{\max})$ at time $t = 0$ for the bet option in the Giles case with $h = 0.01$ and $k = 5h$ ($dx = 0.01/S_{\max}$, for $E^0_{V,\text{nu}}$) solved with CN with $K_\alpha = 0$, CNR with $K_\alpha = 0$, CNK_α with $K_\alpha = 0.5$ and CNRK_α with $K_\alpha = 0.5$

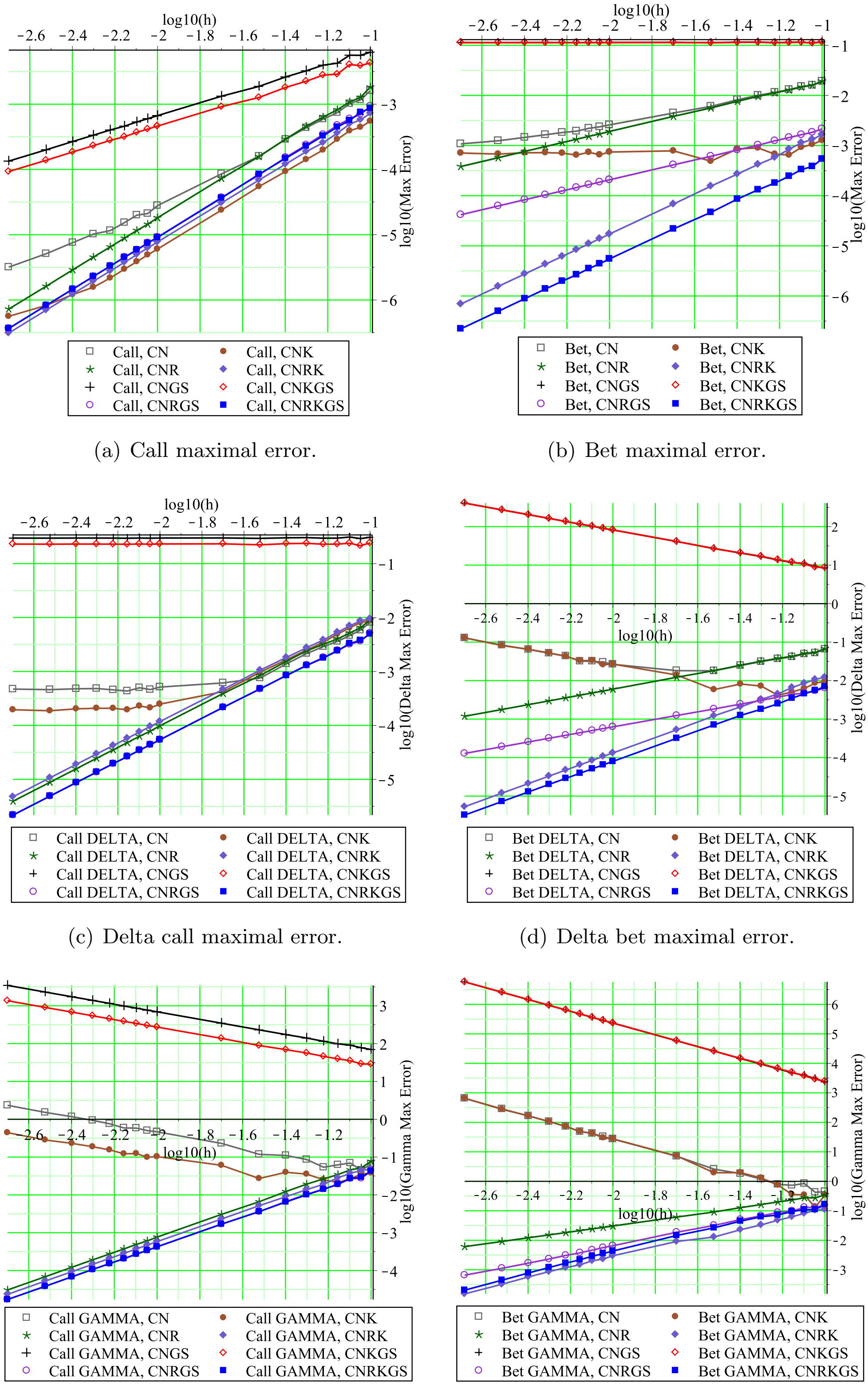
Methods	$E^0_{V,\text{nu}}$	E^0_V	E^0_Δ	E^0_Γ
CN	0.113659	0.00255428	0.0258461	24.9258
CNR	9.11740e-05	0.00191539	0.00580019	0.0303068
CNK_α	0.113888	0.000743987	0.0268447	27.4361
CNRK_α	5.48878e-06	1.71763e-05	0.000132096	0.00298739

The results show two features: A high frequency oscillation and a "bump" both occurring near $S = K$. For the standard CN method the oscillations are fairly small compared to the bump for the solution error, sizable for the Δ error and all dominating for the Γ error. Rannacher startup completely removes the oscillations for the solution and Δ error and very significantly reduces the oscillations for the Γ. Instead Rannacher startup does nothing to reduce the "bump". The K_α method reduces the size of the "bump" but does not remove the oscillation like the Rannacher startup. Finally it is seen how adding the K_α-optimization together with the Rannacher startup completely removes the oscillatory part of the solution and Δ error and significantly reduces it for the Γ error. On top of this the size of the bump is significantly reduced for both solution, Δ and Γ error. For the total error for the CNRK_α-method, including oscillation and bump, the maximal solution error is reduced by a factor of 100 with respect to the CNR solution error. This factor reduces to 44 for the Delta error and 10 for the Gamma error, but in all cases the reduction is at least an order of magnitude. These factors are computed from Table 2. It should be noted that our results for CN and CNR are completely consistent with those of [9, Fig. 2].

After establishing the merit of the CNRK_α-method for the bet option for one mesh, we turn to the question of whether this is just a very particular case? So we solve for both the call and the bet option with a number of different values of $h \in [0.002, 0.1]$ in the Giles case. Also we again take $\lambda^\star = 0.5$ corresponding to $k = 5h$. For the K_α-shifting methods we use $K_\alpha = 0.275$ for the call option and $K_\alpha = 0.5$ for the bet option. For the non K_α-shifting methods actually we take $K_\alpha = 0$ to show some sort of "worst case scenario". For the "true" non K_α-shifting methods, the error will fluctuate erratically between this worst case scenario curve and the curve for the optimal K_α depending on whether the actual K_α is far from or close to optimal.

Apart from uniform meshes we also compute for nonuniform meshes created with the mesh grading function (13) indicated with a [GS] for grid stretching after the acronym for the method. As above, for the nonuniform meshes we take $dx = h/S_{\max}$ to get the same number of elements in S and $k = 5h$. In order to visualize the wide intervals of h-values and corresponding maximal errors we show the results on double logarithmic scales. Instead of showing plots of the errors for all values of $S \in (0, S_{\max})$ at $t = 0$ as we did in Figures 6–9 we now only show maximal errors over $S \in (0, S_{\max})$ at $t = 0$ in Figure 10.

It is seen, that Rannacher time stepping is essential in order to obtain convergence and for all cases Rannacher time stepping combined with mesh grading decreases the error although not the order of convergence. Rannacher time stepping combined with K_α-optimization is clearly the better choice when it comes to order of convergence and combining also with mesh grading decreases the error further without increasing the order of convergence. Hence only the CNRK_α- and CNRK_αGS-methods can be recommended for general use. The CN, CNR and CNK_α-methods (with or without mesh grading) must be considered unsuited for general use even though of course they can be used in particular cases especially if only limited precision is required. The conclusion is, that the CNRK_αGS-method is the overall winner as a general



(a) Call maximal error.

(b) Bet maximal error.

(c) Delta call maximal error.

(d) Delta bet maximal error.

(e) Gamma call maximal error.

(f) Gamma bet maximal error.

Figure 10: Maximal solution (a)-(b), Δ (c)-(d) and Γ (e)-(f) errors at time $t = 0$ as function of $h \in (0.002, 0.1)$ for the call option with $K_\alpha = 0.275$ for the K_α methods and $K_\alpha = 0$ for the non K_α methods (left column) and bet option with $K_\alpha = 0.500$ for the K_α methods and $K_\alpha = 0$ for the non K_α methods (right column) in the Giles case with $k = 5h$ and $dx = h/S_{\max}$ for graded meshes. Each plot is showing error curves E^0_V, E^0_Δ or E^0_Γ respectively for uniform meshes for the 4 methods CN, CNR, CNK_α and CNRK_α and error curves $E^0_{V,\mathrm{nu}}$, $E^0_{\Delta,\mathrm{nu}}$ or $E^0_{\Gamma,\mathrm{nu}}$ respectively for graded meshes for the 4 methods CNGS, CNRGS, CNK_αGS and CNRK_αGS.

method for computing solution, Delta and Gamma values for put, call and bet options.

Our results for the call option in the left column of Figure 10 for CN and CNR are very similar in structure to those of Giles et al [9, Fig. 3]. In reality, the Giles et al results more resemble our results for CNK_α and CNRK_α. For the bet option shown in the right column of Figure 10 it is even more clear, that it is our results for CNK_α and CNRK_α that are comparable with the Giles et al results for CN and CNR [9, Fig. 4].

4 Order of Convergence

For problems without degenerations and singularities the maximal error with the Crank-Nicolson method should converge to zero as $\mathcal{O}(h^2) + \mathcal{O}(k^2)$ but the singular terminal conditions are known to decrease the orders of convergence for "computable" step sizes. The loss of convergence order is bigger the worse the singularity is and hence we focus on the bet option showing results for the error in the solution, the Δ and the Γ, the latter having the strongest singularity.

Figure 11 shows the maximal solution errors at time t = 0 for the bet option with logarithmic axes. The results are provided for the Giles case with $h \in (2^{-8}, 2^{-3})$ and $k \in (10^{-1.7}, 10^{-0.2})$. For the CN and CN$K_\alpha$ methods the (h, k)-plane is clearly divided into two regions with different behavior of the error: In one part a reduction in k reduces the error whereas a reduction in h increases the error. This part will be denoted the *bubble*. The rest of the (h, k)-plane is denoted the *asymptotic* part. The CNR and CNRK_α methods show no bubble part, only the asymptotic part. In order to estimate convergence orders in both h and k independently we use a weighted least squares fitting of the computational errors E^0_V of the form

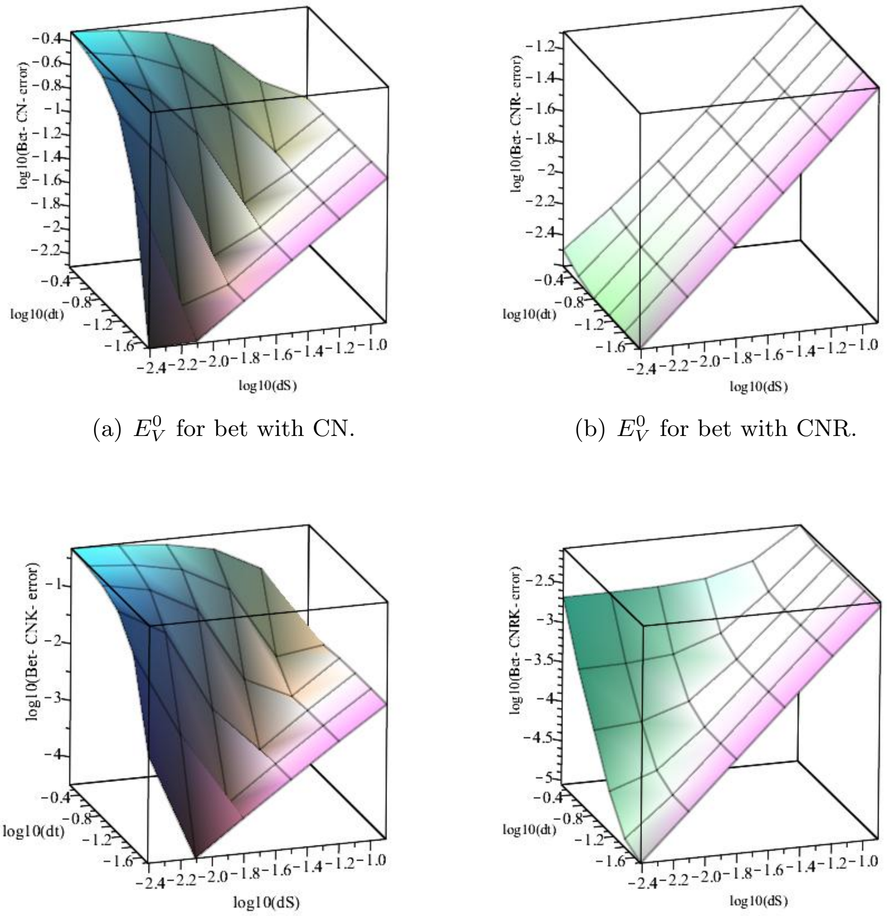
$$\min_{a,b,\alpha,\beta} \sum_{i,j} w_{i,j} \cdot ((E^0_V)_{i,j} - (a \cdot h_i^\alpha + b \cdot k_j^\beta))^2. \tag{14}$$

The stepsizes are recorded so that they decrease with increasing index, i.e. $h_{i+1} \le h_i$ and $k_{j+1} \le k_j$, and the simple weight function $w_{i,j} = i \cdot j$ putting higher weight on smaller step sizes is applied. Obviously selecting a different weight function may change the results somewhat. Separate fittings are made for the bubble and asymptotic parts. In the least squares minimizations the side conditions $0 \le a$, $0 \le b$, $0 \le \alpha \le 3$ and $0 \le \beta \le 3$ are imposed. In a few cases $\beta > 2$. In these cases β is restricted to $0 \le \beta \le 2$ and the least squares fitting is repeated. If the maximal fitting error is not increased on the leading digit, then the latter result is selected. The following convergence orders are computed for the error in the bet option:

$$\begin{aligned}
E^0_V[CN] &\simeq \begin{cases} 0.5 \cdot k^{0.5} & \text{bubble} \\ 0.7 \cdot h^{1.1} + 0.002 & \text{asymptotic} \end{cases} \\
E^0_V[CNR] &\simeq 0.7 \cdot h^{1.0} + 0.001 \cdot k^{0.5} \quad \text{asymptotic} \\
E^0_V[CNK_\alpha] &\simeq \begin{cases} 0.6 \cdot k^{0.7} & \text{bubble} \\ 0.4 \cdot h^{1.9} & \text{asymptotic} \end{cases} \\
E^0_V[CNRK_\alpha] &\simeq 0.4 \cdot h^{1.9} + 0.005 \cdot k^{2.0} \quad \text{asymptotic}
\end{aligned} \tag{15}$$

Similar computations are performed for the Delta and Gamma errors of the bet option but the convergence plots look very similar in structure to Figure 11 and are not shown here. Instead the approximate convergence results are given. The Delta errors for the bet option are computed as

$$E^0_\Delta[CN] \simeq \begin{cases} 77 \cdot k^{0.7} & \text{bubble} \\ 1.0 \cdot h^{0.8} & \text{asymptotic} \end{cases}$$



(a) E_V^0 for bet with CN.

(b) E_V^0 for bet with CNR.

(c) E_V^0 for bet with CNK_α.

(d) E_V^0 for bet with CNRK_α.

Figure 11: Maximal error E_V^0 over $S \in (0, S_{\max})$ at time $t = 0$ as function of the step sizes $h = dS$ and $k = dt$ for the bet option in the Giles case for (a) CN ($K_\alpha = 0$), (b) CNR ($K_\alpha = 0$), (c) CNK_α ($K_\alpha = 0.5$) and (d) CNRK_α ($K_\alpha = 0.5$)

$$
\begin{aligned}
E^0_\Delta[CNR] &\simeq 1.1\cdot h^{0.9}+0.06\cdot k^{2.0} \quad \text{asymptotic}\\
E^0_\Delta[CNK_\alpha] &\simeq \begin{cases} 77\cdot k^{0.7} & \text{bubble}\\ 3.6\cdot h^{1.9} & \text{asymptotic}\end{cases}\\
E^0_\Delta[CNRK_\alpha] &\simeq 3.7\cdot h^{1.9}+0.004\cdot k^{2.0} \quad \text{asymptotic}
\end{aligned}
\tag{16}
$$

and the Gamma errors for the bet option are computed as

$$
\begin{aligned}
E^0_\Gamma[CN] &\simeq \begin{cases} 36000\cdot k^{0.3} & \text{bubble}\\ 8.0\cdot h^{0.9}+0.9\cdot k^{2.0} & \text{asymptotic}\end{cases}\\
E^0_\Gamma[CNR] &\simeq 7.9\cdot h^{1.0}+1.0\cdot k^{1.2} \quad \text{asymptotic}\\
E^0_\Gamma[CNK_\alpha] &\simeq \begin{cases} 35000\cdot k^{0.3} & \text{bubble}\\ 18.3\cdot h^{1.7}+1.6\cdot k^{2.0} & \text{asymptotic}\end{cases}\\
E^0_\Gamma[CNRK_\alpha] &\simeq 17.0\cdot h^{1.7}+0.03\cdot k^{0.9} \quad \text{asymptotic}
\end{aligned}
\tag{17}
$$

We see evidence that the CN and CNR methods are missing one order of convergence in h i.e. in the S-direction. The convergence in k i.e. in the t-direction is quite imprecise. Because of the small coefficient the term is only visible for large values of k where the results are maybe not even in the asymptotic range. The CNK_α and CNRK_α methods reestablishes (almost) full quadratic convergence in h (1.9 for solution and Δ errors and 1.7 for Γ error). The convergence in k is also for these methods "problematic", but somewhat better than for the methods without K_α optimization.

Similar calculations for the call option results in the same conclusion only with a more perfect recovery of the optimal results since the singularity for the call option is weaker than for the bet option. Hence again, the CNRK_α method must be the one recommended for general use.

5 Conclusions and future work

We investigated the Crank-Nicolson finite difference method [CN] and simple improvements for European vanilla options (put, call, bet and butterfly spread).

We proposed the K_α method for uniform and nonuniform meshes with one or more singularities in the terminal condition and tested it with good results for uniform and graded meshes with 1 or 3 singularities.

We found that the Rannacher start up method removes high frequency oscillations in the CN-solution, Delta and Gamma error around the "bump" in the maximal error (see Figure 1) and partially reestablishes the optimal second order convergence in the t-direction of the CN method. Instead it does not decrease the size of the error bump or improve the order of convergence in the S-direction.

The K_α method instead reduces the size of the bump in the CN-error and partially reestablishes the optimal second order convergence in the S-direction of the CN method. Instead it does not remove the high frequency oscillations around the bump or improve the order of convergence in the t-direction.

The CN method with the addition of both the Rannacher and the K_α method removes the high frequency oscillations around the maximal error in the solution, Delta and Gamma and significantly reduces the size of the error bump. Further it partially reestablishes the optimal second order convergence in the S and t-direction of the CN method.

Finally we found that mesh grading further reduces the maximal error in all methods without changing orders of convergence thus establishing the merit of utilizing nonuniform meshes in the S-variable.

The Rannacher and K_α methods can be included into any finite difference scheme with very low cost, and is expected to give similar improvements as for the Crank-Nicolson method.

We have also shown that the optimal K_α-values depend on the option but that they are almost independent of the parameters (in particular of the interest r, the volatility σ and the step sizes h and k). For the call option the optimal K_α lies in $(0.2, 0.3)$ or $(0.7, 0.8)$ for the solution error but the error is not very sensitive to values of K_α in $(0.2, 0.8)$. Values outside this interval may instead lead to significant increases in the error. For the Δ and Γ errors for the call the choice of K_α is almost insignificant but does have an optimal value at $K_\alpha = 0$ (and 1) and 0.3 respectively. For the bet option the optimal K_α is 0.5 for the solution, Δ as well as the Γ error and should be picked in $(0.45, 0.55)$. Values outside this interval may lead to significant increases in the error.

For possible future work we are planning to apply K_α-shifting to nonuniform S-grids considering both graded meshes and adaptive meshes. We intend to do this for finite difference grids, but also to extend to the finite element method, where theoretical results are more easily obtained.

Also we will apply K_α-shifting to problems without closed form solutions such as American options, Asian options, basket options, options with variable parameters (such as σ and γ) and options from a generalized Black-Scholes world taking into consideration for example nonvanishing trading cost, influence from trading volume on stock prices and other features leading to “nonlinear volatility” options.

References

[1] P. Wilmott, , S. Howison, and J. Dewynne, “*The Mathematics of Financial Derivatives*”, Cambridge University Press, 1995.

[2] D. Tavella, and C. Randall, “*Pricing Financial Instruments: The Finite Difference Method*”, Wiley series in financial engineering, 2000.

[3] R. Rannacher, “*Finite element solution of diffusion problems with irregular data*”, Numerische Mathematik, 43, 1984, pp. 309–327.

[4] D.M. Pooley, K.R. Vetzal, and P.A. Forsyth, “*Convergence remedies for non-smooth payoffs in option pricing*”, Journal of Computational Finance, 6(4), 2003, pp. 25–40.

[5] C. Reisinger, and A. Whitley, “*The impact of a natural time change on the convergence of the CrankNicolson scheme*”, IMA Journal of Numerical Analysis, 34, 2014, pp. 1156–1192.

[6] O. Østerby, “*Five ways of reducing the Crank-Nicolson oscillations*”, BIT Numerical Mathematics, 43, 2003, pp. 811–822.

[7] C.W. Oosterlee and C.C. Leentvaar and X. Huang, “*Accurate American option pricing by grid stretching and high order finite differences*”, Working papers, DIAM, Delft University of Technology, the Netherlands, 2005.

[8] D.Y. Tangman, A. Gopaul and M. Bhuruth, “*Numerical pricing of options using high-order compact finite difference schemes*”, Journal of Computational and Applied Mathematics,Vol. 218.2, 2008, pp. 270–280,

[9] M.B. Giles and R. Carter, “*Convergence analysis of Crank-Nicolson and Rannacher time-marching*”, Journal of Computational Finance, 9(4) 2006, pp. 89–112.