

# The Damped Crank-Nicolson Time-Marching Scheme for the Adaptive Solution of the Black-Scholes Equation

Christian Goll<sup>†</sup>    Rolf Rannacher<sup>‡</sup>    Winnifried Wollner<sup>§</sup>

January 3, 2013

**Abstract:** This article is concerned with the derivation of a residual-based a posteriori error estimator and mesh-adaptation strategies for the space-time finite element approximation of parabolic problems with irregular data. Typical applications arise in mathematical finance where the Black-Scholes equation is used for modeling the pricing of European options. A conforming finite element discretization in space is combined with second-order time discretization by a damped Crank-Nicolson scheme for coping with data irregularities in the model. The a posteriori error analysis is developed within the general framework of the Dual Weighted Residual (DWR) method for sensitivity-based goal-oriented error estimation and mesh optimization. In particular, the correct form of the dual problem with damping is considered.

**Keywords:** Black-Scholes equation, space-time finite elements, damped Crank-Nicolson method, goal-oriented adaptivity, DWR-method, adjoint consistent

## 1 Introduction

In this article, we are mainly concerned with the derivation of an a posteriori error estimator for a damped Crank-Nicolson scheme in the spirit of the Dual Weighted Residual (DWR) method for goal-oriented mesh adaptation introduced in Becker and Rannacher (1996, 2001) (see also Bangert and Rannacher (2003)).

To avoid technicalities unnecessary for the understanding of the main points of the article, we focus on a rather simple model problem in computational finance although more complex situations could be handled by the described method as well. As a model

---

<sup>†</sup>[christian.goll@iwr.uni-heidelberg.de](mailto:christian.goll@iwr.uni-heidelberg.de), Institut für Angewandte Mathematik, Universität Heidelberg, Im Neuenheimer Feld 293/294, D-69120 Heidelberg, Germany.

<sup>‡</sup>[rolf.rannacher@iwr.uni-heidelberg.de](mailto:rolf.rannacher@iwr.uni-heidelberg.de), Institut für Angewandte Mathematik, Universität Heidelberg, Im Neuenheimer Feld 293/294, D-69120 Heidelberg, Germany.

<sup>§</sup>[winnifried.wollner@math.uni-hamburg.de](mailto:winnifried.wollner@math.uni-hamburg.de), Department of Mathematics, University of Hamburg, Bundesstr. 55, D-20145, Hamburg

application, we have in mind a multivariate Black-Scholes partial differential equation for the pricing of European plain vanilla options.

In this case, due to possible irregularities of the initial data, the use of the second-order Crank-Nicolson method requires a certain amount of damping such as proposed in Luskin and Rannacher (1982) and Rannacher (1984) in order to compensate for the known weak stability properties of this scheme. The importance of damping has also been recognized in computational finance, see, eg, Pooley et al. (2003), Duffy (2004), Carter and Giles (2006), Forsyth and Vetzal (2002) and the references cited therein. This nonstandard time discretization is combined with a standard conforming finite element approach in space. The key to the application of the DWR method to this space-time discretization is its reformulation as a conforming Galerkin method in space-time. The resulting strategy for the simultaneous adaptation of time step and spatial mesh size is illustrated by some simple test examples.

To calculate sensitivities of the solution it is required to solve a, so-called, dual problem whose initial data are given depending on the the quantity of interest for the sensitivities. These dual initial data are rather irregular for quantities typically of interest in computational finance such as the price of the option at a certain time and price of the underlying asset, or derivatives of the price, so-called, Greeks. This irregularity makes damping of the dual equation even more necessary than for the primal problem itself. Thus, in order to obtain a consistent discretization, ie, one where the dual to the discrete problem is also the discrete of the dual problem, one has to modify the damped Crank-Nicolson scheme even further.

This modified scheme is not only of interest for the application of adaptive methods, but in any case where sensitivities of the solution are required. Such situations arise for instance in calibration problems where local volatilities are adjusted to given (known) option prices.

The use of mesh adaptivity based on a posteriori error estimates is well accepted in the context of finite element discretization of partial differential equations. There are two main approaches in this context: error estimation with respect to natural “energy norms” induced by the given variational problem, see, eg, Verfürth (1996), Ainsworth and Oden (2000) and Babuška and Strouboulis (2001) for surveys. As an alternative approach “goal-oriented” error estimation with respect to a pre-assigned “quantity of interest” is going back to Eriksson et al. (1996), Becker and Rannacher (1996, 2001). For parabolic problems a posteriori error estimates with respect to global norms have been derived, eg, in Akrivis et al. (2006), Chen and Feng (2004), Eriksson and Johnson (1991, 1995), Lang (2001) and Picasso (1998). The case of goal-oriented error estimation for space-time finite element discretization of parabolic problems was considered in Schmich and Vexler (2008), see also Meidner and Vexler (2007) for the case of parabolic optimal control problems. In Ern et al. (2004) the authors apply a dual weighted residual error estimator for a space-time finite element discretization of the one-dimensional Black-Scholes equation using continuous finite elements in space and piece-wise constant or linear discontinuous finite elements in time. A similar setting is considered in Fofas and Larson (2008). However, because of the choice of the time discretization, in particular the use of a discontinuous Galerkin methods in time for the dual problem, the above

mentioned articles did not need to consider the necessity of damping due to irregular primal and dual initial values.

Error estimation and adaptivity in the framework of finite difference schemes applied to the Black-Scholes equation is studied in Achdou and Pironneau (2005), Pironneau and Hecht (2000), Lötstedt et al. (2007) and Persson and von Sydow (2007). In the same context temporal adaptivity for a  $\theta$ -scheme is considered in Khaliq et al. (2008).

However, none of the afore mentioned articles take care of the important case of the Crank-Nicolson method with damping. Such damping is required in order to avoid oscillations in the solution in the case of irregular initial data. A thorough analysis of the effect of irregular initial data on the behavior of the standard Crank-Nicolson scheme and strategies for dealing with this complication can be found in Luskin and Rannacher (1982) and Rannacher (1984).

Such irregular initial data appears naturally in the Black-Scholes model; see for instance Black and Scholes (1973) and Merton (1973). Therefore, we will consider goal-oriented error estimation for the damped Crank-Nicolson method for the Black-Scholes equation as an instructive example. We focused this analysis on the damped Crank-Nicolson method because it is most popular in financial engineering, see for example Pooley et al. (2003), Duffy (2004), Carter and Giles (2006), Forsyth and Vetzal (2002) and the references cited therein. The adjoint time stepping scheme proposed in this article introduces additional damping in the time stepping scheme in order to assert smoothing properties for the adjoint scheme. The smoothing properties of the adjoint scheme are required here in order to accurately calculate the sensitivity factors. The most interesting property of the described method is that the order of computing sensitivities and discretizing the equation does not matter, ie, the discrete sensitivities are in fact approximations to the continuous ones.

Although, we have limited our discussion here to the case of a European option in order to avoid technicalities obstructing the view on the, already quite technical, derivation of an appropriate adjoint scheme one should note that an extension of the techniques is possible to a variety of other options. For example, residual type estimators for American options have been considered in Allegretto et al. (2006); Moon et al. (2007); Nochetto et al. (2010). Dual weighted residual error estimates can be extended to the case of American options using the techniques developed for the treatment of variational inequalities, see, eg, Blum and Suttmeier (2000); Suttmeier (2008).

This article is structured as follows: In Section 2, we describe the problem under consideration together with its time and space discretization. In Section 3, we derive the time-stepping scheme for the dual/adjoint problem. In particular, we reformulate the primal scheme from Section 2 as a Galerkin method in time. Then, we discuss a posteriori error estimates and their separation into local spatial and temporal contributions in Section 4. Finally, in Section 5, we conclude with some results of numerical experiments, which confirm our claim on the need for damping of the dual problem.

## 2 The Problem and its Discretization

### 2.1 The Black-Scholes Equation

We begin by a brief review of the Black-Scholes equation for describing the price of a plain vanilla European option. For an introduction of this model, we refer the reader to the original papers from Black and Scholes (1973) and Merton (1973), or the book of Kwok (1998) and the references cited therein for the multivariate case. Here, we are particularly concerned with the fair price  $u$  of an European plain vanilla option on a basket of  $d$  risky underlying assets. In the following, we consider an option with *maturity*  $T > 0$  and *strike price*  $K > 0$ . For the sake of simplicity, we assume the *interest rate*  $r > 0$  and the *volatility* of the  $i$ -th asset  $\sigma_i > 0$ ,  $1 \leq i \leq d$ , to be constant. Further, we assume the matrix  $\rho = (\rho_{ij})$  of the *correlation factors*  $\rho_{ij}$  with  $-1 \leq \rho_{ij} \leq 1$  for  $1 \leq i, j \leq d$ , to be positive definite. Of course  $\rho$  is symmetric with  $\rho_{ii} = 1$ . As an abbreviation, we define the symmetric positive definite Matrix  $\Xi := (\sigma_i \rho_{ij} \sigma_j)$ .

With  $(t, x) \in I = (0, T] \times \mathbb{R}_+^d$  denoting the prices  $x$  of the underlying assets at the time  $t$ , the problem of determining the fair price  $u$  of such an option is (after a time reversal) given by the following equation:

$$\partial_t u - \frac{1}{2} \sum_{i,j=1}^d \Xi_{ij} x_i x_j \partial_{x_i} \partial_{x_j} u - r \sum_{i=1}^d x_i \partial_{x_i} u + ru = 0 \quad \text{in } (0, T] \times \mathbb{R}_+^d, \quad (2.1a)$$

$$u(0) = u_0 \quad \text{in } \mathbb{R}_+^d. \quad (2.1b)$$

Where the initial condition  $u_0 \in C^0(\mathbb{R}_+^d)$  (ie, the *payoff*) is given depending of the type of the option. For example:

$$u_0 := \begin{cases} \max(\sum_{i=1}^d x_i - K, 0), & u \text{ is a } Call, \\ \max(K - \sum_{i=1}^d x_i, 0), & u \text{ is a } Put. \end{cases}$$

where  $x = (x_i) \in \mathbb{R}_+^d$ .

Using standard arguments, see, eg, Friedman (1983), the existence of a solution to this problem can be shown. This solution is unique provided an additional growth condition

$$|u(t, x)| \leq \begin{cases} \sum_{i=1}^d x_i, & u \text{ is a } Call, \\ K, & u \text{ is a } Put, \end{cases} \quad \text{for } (t, x) \in [0, T] \times \mathbb{R}_+^d,$$

is required.

In order to discretize (2.1) it is desirable to consider a bounded domain. For this purpose let  $\bar{x}_i > 0$ , ( $i = 1, \dots, d$ ) be given and consider the domain

$$\Omega := (0, \bar{x}_1) \times \dots \times (0, \bar{x}_d).$$

We now require additional boundary values on

$$\Gamma_D := \partial\Omega \cap \mathbb{R}_+^d.$$

These values are chosen compatible with  $u_0$  using knowledge on the asymptotic behavior of the solution to (2.1). For the two types of options considered here these values are given as follows:

$$g(t, x) := \begin{cases} \sum_{i=1}^d x_i - Ke^{-rt}, & u \text{ is a Call,} \\ 0, & u \text{ is a Put,} \end{cases} \quad (2.2)$$

see for instance Pironneau and Achdou (2009).

Then, we consider the following problem on a bounded polygonal domain:

$$\partial_t u - \frac{1}{2} \sum_{i,j=1}^d \Xi_{ij} x_i x_j \partial_{x_i} \partial_{x_j} u - r \sum_{i=1}^d x_i \partial_{x_i} u + ru = 0 \quad \text{in } (0, T] \times \Omega, \quad (2.3a)$$

$$u|_{\Gamma_D} = g \quad \text{on } [0, T] \times \Gamma_D, \quad (2.3b)$$

$$u(0) = u_0 \quad \text{in } \Omega. \quad (2.3c)$$

The error introduced by cutting of the domain can be made arbitrary small by choosing the domain  $\Omega$  sufficiently large as been shown in Kangro and Nicolaides (2000). In particular, there holds:

**Theorem 1.** *Let  $u$  be a regular solution of (2.1) and let  $u_{\text{co}}$  be a regular solution of (2.3). Then, for any point  $(t, x) \in [0, T] \times \Omega$  such that*

$$\ln \frac{\bar{x}_i}{x_i} \geq (\sigma_i^2 + 2r)t \geq 0$$

the following error estimate holds:

$$|u(t, s) - u_{\text{co}}(t, s)| \leq \|u - g\|_{L^\infty((t, T) \times \Gamma_D)} \cdot \left( \sum_{i=1}^d \exp \left( - \frac{\ln \frac{\bar{x}_i}{x_i} (\ln \frac{\bar{x}_i}{x_i} - (\sigma_i^2 + 2r)t)}{2\sigma_i^2 t} \right) \right).$$

In order to obtain a variational formulation, we follow Achdou and Pironneau (2005) and define

$$V := \{ v \in L^2(\Omega) \mid x_i \partial_{x_i} v \in L^2(\Omega), \ 1 \leq i \leq d \}.$$

Then, with the scalar product  $(\cdot, \cdot)$  of  $L^2(\Omega)$ , we define a scalar product on  $V$  by

$$(u, v)_V := (u, v) + \sum_{i=1}^d (x_i \partial_{x_i} u, x_i \partial_{x_i} v).$$

Thereby  $V$  becomes a Hilbert space with norm  $\|v\|_V := (v, v)_V^{1/2}$ . In order to include boundary values, we introduce the space

$$V_0 := \overline{C_0^\infty(\Omega)}^{\|\cdot\|_V} = \{v \in V \mid v|_{\Gamma_D} = 0\}.$$

The semi-norm  $|\cdot|_V := (\sum_{i=1}^d \|x_i \partial_{x_i} \cdot\|_\Omega^2)^{1/2}$  is a norm on  $V_0$ , which is equivalent to  $\|\cdot\|_V$  due to a Poincaré type inequality.

Let  $H := L^2(\Omega)$ ,  $Y = V$  or  $Y = V_0$  and  $Y'$  be the topological dual of  $Y$ . Then,  $Y \hookrightarrow H \hookrightarrow Y'$  forms a Gelfand triple. We define the function space  $W(Y) := W(0, T)(Y)$  through

$$W(0, T)(Y) := \{w \in L^2(0, T; Y) \mid \partial_t w \in L^2(0, T; Y')\}.$$

For the Hilbert space  $W(Y)$  the embedding  $W(Y) \subset C^0([0, T]; H)$  holds, see, eg, Wloka (1987, Theorem 25.5). Further, we define the functions  $A : \mathbb{R}^d \rightarrow \mathbb{R}^{d \times d}$  and  $\beta : \mathbb{R}^d \rightarrow \mathbb{R}^d$  by

$$Ax := \left(\frac{1}{2}\Xi_{ij}x_i x_j\right)_{ij=1}^d, \quad \beta x := \left(\left(\Xi_{ii} + \frac{1}{2}\sum_{j=1, j \neq i}^d \Xi_{ij} - r\right)x_i\right)_{i=1}^d,$$

and the bilinear forms  $\bar{a} : V \times V \rightarrow \mathbb{R}$  and  $a : W(V) \times W(V) \rightarrow \mathbb{R}$  by

$$\begin{aligned} \bar{a}(v, w) &:= (A\nabla v, \nabla w) + (\beta \cdot \nabla v, w) + r(v, w), \\ a(v, w) &:= \int_0^T \bar{a}(v(t), w(t)) dt. \end{aligned}$$

With this notation, we can state the weak formulation of (2.3): *Given  $u_0 \in L^2(\Omega)$ , find  $u \in \tilde{g} + W(V_0)$  such that*

$$((\partial_t u, \varphi)) + a(u, \varphi) + (u(0), \varphi(0)) = (u_0, \varphi(0)) \quad \forall \varphi \in W(V_0), \quad (2.4)$$

where we denoted by  $((\cdot, \cdot))$  the scalar product on  $L^2(I \times \Omega)$  and  $\tilde{g} \in W(V)$  is any continuation of the boundary data  $g$ .

It has been shown for  $d = 1$  in Achdou and Pironneau (2005) and is otherwise clear by definition that  $\bar{a}$  is continuous and fulfills Gårdings inequality in the space  $V_0$ , ie, there exist constants  $C, c, \lambda > 0$  such that for any  $v, w \in V_0$  there holds:

$$\bar{a}(v, w) \leq C|v|_V|w|_V, \quad \bar{a}(v, v) \geq c|v|_V^2 - \lambda\|v\|^2.$$

Hence, by standard arguments there exists a unique solution  $u \in \tilde{g} + W(V_0)$  of (2.4), see, eg, Wloka (1987, Theorem 26.1).

*Remark 2.1.* At this point, we remark that in the presented framework of option pricing one is particularly interested in the value of the solution today at the actual price of the asset  $x_0$ , ie, in the value  $u(T, x_0)$ . Other quantities of interest are, so-called, *Greeks*, which are basically partial derivatives of the solution  $u$ , also evaluated at the point  $(T, x_0)$ . For example, the *Greek Delta* of the solution  $u$  is defined as  $\partial_x u$ . For further information about the sensitivities and their role in option pricing, we refer to Hull (2006).

## 2.2 Semi-discretization in Time

In this section, we describe the semi-discretization in time of (2.4) by means of a Galerkin method. We choose time points  $t_l$  for  $0 \leq l \leq M$  such that

$$0 = t_0 < \dots < t_l < \dots < t_M = T.$$

Then, we set  $I_m = (t_{m-1}, t_m]$  and  $k_m = t_m - t_{m-1}$  for  $1 \leq m \leq M$  to obtain a partition  $\mathcal{T}_k = \{I_l \mid 1 \leq l \leq M\}$  of  $I$ , ie,

$$\bar{I} = \{0\} \cup \bigcup_{I_l \in \mathcal{T}_k} I_l.$$

In the next Section 2.2.1, we will recall the continuous formulation of the Crank-Nicolson time-stepping scheme. It is crucial, for our later analysis to note that the Crank-Nicolson method can be embedded into the context of Galerkin methods (cG(1)-method), see Section 3.1. We continue by formulating the damped Crank-Nicolson method to cover non regular initial values. We note that this can be embedded into the Galerkin framework as well, see Section 3.2, which is one of the contributions of this article and allows for an easy way to calculate adjoint information. This will then be used for purposes of error estimation in Section 4 but will be equally important in the context of calibration problems.

### 2.2.1 The damped Crank-Nicolson Scheme

We will first restate the well known Crank-Nicolson time marching scheme. To this end let  $\tilde{g}_k^m \in V$  for  $0 \leq m \leq M$  be some continuation of the boundary data, eg,  $\tilde{g}_k^m = g(t_m)$ . Then, the Crank-Nicolson scheme takes the form: *Given  $u_0 \in V$ , find  $u_k^m \in \tilde{g}_k^m + V_0$  for  $0 \leq m \leq M$  such that*

$$(u_k^0 - u_0, \psi) = 0 \quad \forall \psi \in L^2(\Omega), \quad (2.5)$$

$$(u_k^m - u_k^{m-1}, \psi) + \frac{1}{2}k_m \bar{a}(u_k^m + u_k^{m-1}, \psi) = 0 \quad \forall \psi \in V_0, \quad 1 \leq m \leq M. \quad (2.6)$$

It is immediately clear from (2.5) that we may only consider initial data  $u_0 \in V$  in order to have a well posed semi-discrete problem. This is in contrast to the continuous problem (2.4) in which initial conditions in  $L^2(\Omega)$  were possible. Furthermore, even for initial data in  $V$  convergence of optimal order with respect to  $k$  is usually not obtained. In order to ensure optimal convergence one may replace one (or several) steps of the Crank-Nicolson method by two steps of an implicit Euler scheme, see Luskin and Rannacher (1982) or Rannacher (1984).

*Remark 2.2.* Let us make a remark on the number of damping steps. Even though the theorems in the aforementioned works state that two damping steps (ie, two implicit Euler-steps) are sufficient for initial data in  $L^2(\Omega)$ , sometimes more damping steps are needed. The number of damping steps in fact depends on the lack of regularity in the initial conditions. For example, if the initial conditions show a 'kink' or a jump, ie, they are in  $L^2(\Omega)$ , two damping steps are sufficient. However, if we consider less regular initial data like a Dirac functional or derivatives thereof, we need to apply three or four damping steps respectively. Such irregular initial conditions arise naturally in the adjoint equation for the evaluation of point-values or derivatives as they appear in the, so-called, Greeks.

In fact, a careful analysis of Rannacher (1984) shows that for initial conditions  $u_0 \in H^m$  with  $m \leq 2$ , at least  $2 - m$  damping steps are required to gain optimal rates of convergence. Because we replace one step of the Crank-Nicolson scheme by two implicit Euler steps, we consider only even numbers of damping steps in this paper.

According to Remark 2.2, it is clear that the precise number of damping steps needs to be adjusted to the given initial data. Thus, for the description of the time-stepping scheme, we select  $L$  intervals,  $0 \leq L \leq M$ , of the partition  $\mathcal{T}_k$  on which we intend to damp the method. The number  $L$  then needs to be picked according to the comments above depending on the regularity of the initial data.

Let  $J_L$  be the corresponding set of indices,  $J_L = \{l_1, \dots, l_L\} \subset \{1, \dots, M\}$ . For  $l \in J_L$ , we split the interval. Hence, we set  $t_{l-1/2} := t_l - \frac{1}{2}k_l$ . Then, we define the set  $J^0$  of intervals on which we use an implicit Euler step as well as the set  $J^1$  on which we use the Crank-Nicolson method by

$$J^0 = \{l, l - \frac{1}{2} \mid l \in J_L\}, \quad J = J^0 \cup \{n \in \mathbb{N} \mid 1 \leq n \leq M\}, \quad J^1 = J \setminus J^0. \quad (2.7)$$

By this, we also redefine the intervals  $I_m$  for indices in  $J$ . Therefore let

$$I_m = \begin{cases} (t_{m-1}, t_m], & m \in J^1, \\ (t_{m-1/2}, t_m], & m \in J^0, \end{cases} \quad \text{and} \quad k_m = |I_m|.$$

This defines a new partition  $\hat{\mathcal{T}}_k = \{I_l \mid l \in J\}$ . We are now able to formulate the damped Crank-Nicolson method: *Given  $u_0$ , find  $u_k^m \in \tilde{g}_k^m + V_0$  such that for  $m \in J \cup \{0\}$  the following holds:*

$$(u_k^0, \psi) = (u_0, \psi) \quad \forall \psi \in L^2(\Omega), \quad \text{if } m = 0, \quad (2.8a)$$

$$(u_k^m, \psi) + k_m \bar{a}(u_k^m, \psi) = (u_k^{m-1/2}, \psi) \quad \forall \psi \in V_0, \quad \text{if } m \in J^0, \quad (2.8b)$$

$$(u_k^m, \psi) + \frac{1}{2}k_m \bar{a}(u_k^m + u_k^{m-1}, \psi) = (u_k^{m-1}, \psi) \quad \forall \psi \in V_0, \quad \text{if } m \in J^1. \quad (2.8c)$$

### 2.3 Spatial Discretization

We discretize the infinite dimensional function space  $V_0$  by means of  $V_0$ -conforming finite elements. For this purpose, for each time point  $t_m$  with  $m \in J \cup \{0\}$  let  $\mathcal{T}_h^m = \{K\}$  be a shape regular non-overlapping partition of  $\bar{\Omega}$  into (closed)  $d$ -dimensional triangular or quadrilateral cells  $K$ . For details on this construction, we refer to the usual finite element literature, eg, Ciarlet (1987). The discretization parameter  $h$  is defined as  $h = \max_{m \in J \cup \{0\}} \max_{K \in \mathcal{T}_h^m} \text{diam}(K)$ . Then, for  $s \in \mathbb{N}$ ,  $s \geq 1$ , we define the finite dimensional spaces  $V_h^{s,m}$  by

$$V_h^{s,m} := \{v \in C^0(\bar{\Omega}) \mid v|_K \in \mathcal{Q}^s(K), \forall K \in \mathcal{T}_h^m\} \subset V,$$

where  $\mathcal{Q}^s(K)$  denotes the usual space of isoparametric  $P_s$  or  $Q_s$  finite elements of polynomial degree  $s$ . Further, we set  $V_{h,0}^{s,m} = V_h^{s,m} \cap V_0$ . With this, we can discretize (2.8) to get a fully discrete time-marching scheme:

*Given  $u_0$ , find  $u_{kh}^m \in \tilde{g}_k^m + V_{h,0}^{s,m}$  such that for  $m \in J \cup \{0\}$  there holds*

$$(u_{kh}^0, \psi_h) = (u_0 - \tilde{g}_k^0, \psi_h) \quad \forall \psi_h \in V_{h,0}^{s,0}, \quad \text{if } m = 0 \quad (2.9a)$$

$$(u_{kh}^m, \psi_h) + k_m \bar{a}(u_{kh}^m, \psi_h) = (u_{kh}^{m-1/2}, \psi_h) \quad \forall \psi_h \in V_{h,0}^{s,m}, \quad \text{if } m \in J^0, \quad (2.9b)$$

$$(u_{kh}^m, \psi_h) + \frac{1}{2}k_m \bar{a}(u_{kh}^m + u_{kh}^{m-1}, \psi_h) = (u_{kh}^{m-1}, \psi_h) \quad \forall \psi_h \in V_{h,0}^{s,m}, \quad \text{if } m \in J^1. \quad (2.9c)$$

*Remark 2.3.* For readers unfamiliar with the finite element notation, we should that the above equations can easily be written in a matrix-vector form by expanding the solution  $u_{kh}^m$  into a basis and then evaluating the integrals for appropriate pairs of basis functions. For a more detailed introduction into the finite element method see Brenner and Scott (2007, Chapter 0.6) as well as to Brenner and Scott (2007, Chapter 0.5) where the connection to finite differences is discussed in more detail.

### 3 The Dual Time Stepping Scheme

Now, we come to the important issue of deriving the dual time stepping scheme. Although one could in principle derive the dual time stepping scheme directly from the primal one. Unfortunately, the finite difference discretization of the time derivatives may cause trouble in this calculation. This is because then it needs to be checked whether the above formulation (2.9) is adjoint consistent, ie, whether the calculated discrete adjoint corresponds to a reasonable discretization of the continuous adjoint. For the numerical problems associated with this if adjoint consistency is not given, we refer to Sachs and Schu (2012).

This is why we take the trouble to reformulate the above time-stepping scheme in terms of a Galerkin method. It is well known that Galerkin methods are adjoint consistent by construction. Reader uninterested in the details may wish to skip this derivation and continue directly in Section 3.4.

#### 3.1 The cG( $r$ )-Method

Let  $Y = V$  or  $Y = V_0$ . For  $r \in \mathbb{N}, r \geq 1$ , we define the following semi-discrete spaces  $X_k^r(Y)$  and  $\tilde{X}_k^{r-1}(Y)$  by

$$X_k^r(Y) = \{\varphi_k \in C^0(\bar{I}, L^2(\Omega)) \mid \varphi_k|_{I_m} \in P_r(I_m, Y), 1 \leq m \leq M\}, \quad (3.1)$$

$$\tilde{X}_k^{r-1}(Y) = \{\varphi_k : \bar{I} \rightarrow Y \mid \varphi_k|_{I_m} \in P_{r-1}(I_m, Y), 1 \leq m \leq M; \varphi_k(0) \in L^2(\Omega)\}. \quad (3.2)$$

Here, we denote by  $P_r(I_m, Y)$  the space of polynomials of degree lower or equal to  $r$  on  $I_m$  with values in  $Y$ . The space  $X_k^r(Y)$  can be identified with a subspace of  $W(0, T)(Y)$ . Due to the possible discontinuities for functions  $\varphi_k \in \tilde{X}_k^{r-1}(Y)$  at the time points  $t_l$ ,  $1 \leq l \leq M$ ,  $\tilde{X}_k^{r-1}(Y)$  is not a subspace of  $W(0, T)(Y)$ , but it may be viewed as a subspace of  $L^2(I, Y)$ .

In order to deal with the discontinuities in  $\tilde{X}_k^{r-1}(Y)$ , we introduce the following standard notation:

$$v^{l,+} = \lim_{\varepsilon \searrow 0} v(t_l + \varepsilon), \quad v^{l,-} = \lim_{\varepsilon \nearrow 0} v(t_l + \varepsilon) = v(t_l), \quad [v]_l = v^{l,+} - v^{l,-},$$

for functions  $v \in \tilde{X}_k^{r-1}(Y)$  and any  $0 \leq l \leq M$ .

Now, let  $g_k \in \{\varphi_k \in C^0([0, T], L^2(\Gamma_D)) \mid \varphi_k|_{I_m} \in P(I_m, L^2(\Gamma_D))\}$  be an interpolation of the boundary data  $g$  such that

$$(g - g_k)(t_l) = 0, \quad 0 \leq l \leq M.$$

Further, let  $\tilde{g}_k \in X_k^r(V)$  be a continuation of these boundary data.

By replacing trial and test space in (2.4) by  $X_k^r(V_0)$  and  $\tilde{X}_k^{r-1}(V_0)$ , respectively, we obtain the following semi-discretized form of the cG( $r$ )-method: *Given  $u_0 \in V$ , find  $u_k \in \tilde{g}_k + X_k^r(V_0)$  such that*

$$((\partial_t u_k, \varphi_k)) + a(u_k, \varphi_k) + (u_k(0) - u_0, \varphi_k^{0,-}) = 0 \quad \forall \varphi_k \in \tilde{X}_k^{r-1}(V_0). \quad (3.3)$$

We will now consider the case  $r = 1$  which is frequently used for the solution of this problem. We will specify (3.3) for this case and see that we recover the well known Crank-Nicolson scheme (2.5). For functions  $v_k \in X_k^1(V)$ , we set

$$v_k^l := v_k(t_l) = v_k^{l,-} \in V, \quad 0 \leq l \leq M.$$

Hence for  $t \in I_m$ ,  $1 \leq m \leq M$ ,

$$v_k(t) = k_m^{-1}(t_m - t)v_k^{m-1} + k_m^{-1}(t - t_{m-1})v_k^m, \quad \partial_t v_k(t) = k_m^{-1}(v_k^m - v_k^{m-1}).$$

Since the coefficients in  $\bar{a}$  do not depend on time, using test functions  $\varphi_k$  that are non zero on one interval  $I_m$  only, we obtain the Crank-Nicolson time marching scheme already given in (2.5): *Given  $u_0 \in V$ , find  $u_k^m \in \tilde{g}_k^m + V_0$  for  $0 \leq m \leq M$  such that*

$$\begin{aligned} (u_k^0 - u_0, \psi) &= 0 \quad \forall \psi \in L^2(\Omega), \\ (u_k^m - u_k^{m-1}, \psi) + \frac{1}{2}k_m \bar{a}(u_k^m + u_k^{m-1}, \psi) &= 0 \quad \forall \psi \in V_0, \quad 1 \leq m \leq M. \end{aligned}$$

Then, the solution  $u_k$  to (3.3) is defined on  $t \in \bar{I}$  by linear interpolation of the discrete values  $u_k^m$ , ie,

$$u_k(t) = u_k^0 \chi_{\{0\}}(t) + \sum_{m=1}^M (k_m^{-1}(t_m - t)u_k^{m-1} + k_m^{-1}(t - t_{m-1})u_k^m) \chi_{I_m}(t).$$

### 3.2 Damped Crank-Nicolson Scheme as Galerkin Method

For the cG( $r$ ) method it is again clear from (3.3) that we may only consider initial data  $u_0 \in V$  in order to have a well posed semi-discrete problem (3.3).

As in Section 2.2.1, we define the set  $J^0$  of intervals on which we use an implicit Euler step. This scheme is now interpreted as a step of a discontinuous Galerkin method of order zero (dG(0)-method). Further, we define the set  $J^1$  on which we use the cG(1) method by (2.7).

Now, with the same notation as in Section 2.2.1, we recall the formulation of the damped Crank-Nicolson method from (2.8): *Given  $u_0$ , find  $u_k^m \in \tilde{g}_k^m + V_0$  such that for  $m \in J \cup \{0\}$  the following holds:*

$$(u_k^0, \psi) = (u_0, \psi) \quad \forall \psi \in L^2(\Omega), \quad m = 0, \quad (3.4a)$$

$$(u_k^m, \psi) + k_m \bar{a}(u_k^m, \psi) = (u_k^{m-1/2}, \psi) \quad \forall \psi \in V_0, \quad m \in J^0, \quad (3.4b)$$

$$(u_k^m, \psi) + \frac{1}{2}k_m \bar{a}(u_k^m + u_k^{m-1}, \psi) = (u_k^{m-1}, \psi) \quad \forall \psi \in V_0, \quad m \in J^1. \quad (3.4c)$$

In order to derive our error estimates it is convenient to reformulate this scheme as a Galerkin method. To this end, we have to properly define our trial space. As the implicit Euler method corresponds to a dG(0) method, we will have to allow for possible discontinuities on the left side of the intervals where we employ the dG(0) method. Hence, we set

$$J^{\text{dc}} = \{l - \frac{1}{2} \mid l \in J^0\}, \quad J^c = (J \cup \{0\}) \setminus J^{\text{dc}}.$$

Here  $J^{\text{dc}}$  denotes the time points where discontinuities may occur, while  $J^c$  are the time points where the semi-discrete solution is continuous. Then, we define for  $r \geq 1$

$$\begin{aligned} \hat{X}_k^r(Y) = \{ \varphi_k : \bar{I} \rightarrow Y \mid \varphi_k|_{I_m} \in P_r(I_m, Y), \varphi_k|_{I_l} \in P_{r-1}(I_l, Y), \\ [\varphi_k]_n = 0, l \in J^0, m \in J^1, n \in J^c \} \subset L^2(I, Y). \end{aligned} \quad (3.5)$$

Let, in abuse of notation,

$$\begin{aligned} g_k \in \{ \varphi_k : \bar{I} \rightarrow L^2(\Gamma_D) \mid \varphi_k|_{I_m} \in P_r(I_m, L^2(\Gamma_D)), \varphi_k|_{I_l} \in P_{r-1}(I_l, L^2(\Gamma_D)), \\ [\varphi_k]_n = 0, l \in J^0, m \in J^1, n \in J^c \}, \end{aligned}$$

be an interpolation of the boundary data  $g$  such that

$$(g - g_k)(t_l) = 0, \quad l \in J \cup \{0\}. \quad (3.6)$$

Once again, we denote by  $\tilde{g}_k \in \hat{X}_k^1(V)$  a continuation of the boundary data  $g_k$ . Now, we can state the Galerkin version of (3.4): *Given  $u_0$ , find  $u_k \in \tilde{g}_k + \hat{X}_k^1(V_0)$  such that*

$$\sum_{m \in J} ((\partial_t u_k, \varphi_k))_m + a(u_k, \varphi_k) + \sum_{l \in J^{\text{dc}}} ([u_k]_l, \varphi_k^{l,+}) + (u_k(0) - u_0, \varphi_k^{0,-}) = 0 \quad (3.7)$$

holds for all  $\varphi_k \in \hat{X}_k^0(V_0)$ . For  $m \in J$ , we denote the scalar product in  $L^2(I_m \times \Omega)$  by  $((\cdot, \cdot))_m$ .

### 3.3 Spatial Discretization

Now, in order to discretize the problem completely, it is no longer sufficient to discretize the infinite dimensional function space  $V_0$  by means of  $V_0$ -conforming finite elements as we did with the space  $V_{h,0}^{s,m}$  in Section 2.3. Instead, we need to find completely discrete spaces corresponding to  $\hat{X}_k^r$  and  $\hat{X}_k^{r-1}$ . To do so, we proceed analogously to (3.2) by setting

$$\tilde{X}_{kh,0}^{r,s} := \{ \varphi_{kh} : \bar{I} \rightarrow Y \mid \varphi_{kh}|_{I_m} \in P_r(I_m, V_{h,0}^{s,m}), m \in J; \varphi_{kh}(0) \in V_{h,0}^{s,0} \}.$$

In order to discretize the trial space, we have to ensure continuity at time points  $t_l$  for  $l \in J^c$ . To this end, we use the construction described in Schmich and Vexler (2008). Let  $\{\tau_0, \tau_1, \dots, \tau_r\}$  be a Lagrangian basis of  $P_r(I_m, \mathbb{R})$ ,  $m \in J^1$  such that

$$\tau_0(t_{m-1}) = 1, \quad \tau_0(t_m) = 0, \quad \tau_k(t_{m-1}) = 0 \quad (k = 1, \dots, r). \quad (3.8)$$

In the case  $r = 1$  a basis fulfilling (3.8) is given by

$$\tau_0(t) := k_m^{-1}(t_m - t), \quad \tau_1(t) = k_m^{-1}(t - t_{m-1}), \quad t \in I_m.$$

Notice that for  $m \in J^1$  it holds  $m - 1 \in J^c$ . We define the spaces  $X_{kh,0}^{r,s,m}$  on the sub-intervals  $I_m$  by

$$X_{kh,0}^{r,s,m} := \text{span} \{ \tau_i v_i \mid v_0 \in V_{h,0}^{s,m-1}, v_j \in V_{h,0}^{s,m}, j \geq 1, i = 0, \dots, r \}.$$

By definition  $X_{kh,0}^{r,s,m} \subset P_r(I_m, V)$ . Then, the fully discretized trial space is analogously to (3.5) given by

$$\begin{aligned} \hat{X}_{kh,0}^{r,s} := \{ \varphi_{kh} : \bar{I} \rightarrow Y \mid \varphi_{kh}|_{I_m} \in X_{kh,0}^{r,s,m}, \varphi_{kh}|_{I_l} \in P_{r-1}(I_l, V_{h,0}^{s,l}), [\varphi_{kh}]_n = 0, \\ \text{for } l \in J^0, m \in J^1, n \in J^c \} \subset \hat{X}_k^r(V_0). \end{aligned}$$

Analogously to the above construction, we define the spaces  $\tilde{X}_{kh}^{r,s}, \hat{X}_{kh}^{r,s}$  and  $X_{kh}^{r,s,m}$  through replacing  $V_{h,0}^{s,m}$  by  $V_h^{s,m}$  in the above definitions.

With these preparations, we can now state the fully discretized version of (3.7). We remark that due to the form of the boundary data (2.2), we can choose the continuation in (3.6) such that  $\tilde{g}_k \in \hat{X}_{kh}^{1,s}$ .

Then, the fully discretized problem reads as follows: *Given  $u_0$ , find  $u_{kh} \in \tilde{g}_k + \hat{X}_{kh,0}^{1,s}$  such that*

$$\begin{aligned} \sum_{m \in J} ((\partial_t u_{kh}, \varphi_{kh}))_m + a(u_{kh}, \varphi_{kh}) \\ + \sum_{l \in \bar{J}^{dc}} ([u_{kh}]_l, \varphi_{kh}^{l,+}) + (u_{kh}(0) - u_0, \varphi_{kh}^{0,-}) = 0, \end{aligned} \quad (3.9)$$

for all  $\varphi_{kh} \in \tilde{X}_{kh,0}^{0,s}$ . Again, taking test functions  $\varphi_{kh}$  with support on only one of the subintervals  $I_m$  one readily sees that this is equivalent to the damped Crank-Nicolson scheme given in (2.9).

### 3.4 Dual Time Marching Scheme

For the purpose of sensitivity/duality-based error estimation, we will have to consider a dual problem of the form: *Find  $z \in W(V_0)$  such that*

$$-((\varphi, \partial_t z)) + a(\varphi, z) + (\varphi(T), z(T)) = \mathcal{J}(\varphi(T)) \quad \forall \varphi \in W(V_0). \quad (3.10)$$

with a given linear functional  $\mathcal{J} \in V'$ . In addition, we require semi-discretized and fully discretized versions of this dual problem and corresponding solutions  $z_k$  and  $z_{kh}$ , which have to be determined in  $\tilde{X}_k^0(V_0)$  and  $\tilde{X}_{kh}^{0,s}(V_0)$ , respectively such that  $z_k$  and  $z_{kh}$  are feasible as test functions in (3.7) and (3.9), respectively. To this end, we use the formal dual problems to (3.7) and (3.9), respectively. In the case of (3.7) this leads us the following dual time marching scheme: *For  $l \in J \cup \{0\}$  find  $z_k^l \in V_0$  such that for any  $\psi \in V_0$  there holds:*

i. For  $l = M$ :

$$\text{a) } M \in J^0 : \quad (\psi, z_k^M) + k_M \bar{a}(\psi, z_k^M) = \mathcal{J}(\psi), \quad (3.11a)$$

$$\text{b) } M \in J^1 : \quad (\psi, z_k^M) + \frac{1}{2}k_M \bar{a}(\psi, z_k^M) = \mathcal{J}(\psi), \quad (3.11b)$$

ii. For  $0 < l < M$ :

$$\text{a) } l \in J^{\text{dc}}, l \in J^1 : \quad (\psi, z_k^l) + \frac{1}{2}k_l \bar{a}(\psi, z_k^l) = (\psi, z_k^{l+1/2}), \quad (3.11c)$$

$$\text{b) } l \in J^{\text{dc}}, l \in J^0 : \quad (\psi, z_k^l) + k_l \bar{a}(\psi, z_k^l) = (\psi, z_k^{l+1/2}), \quad (3.11d)$$

$$\text{c) } l \in J^c, l \in J^0 : \quad (\psi, z_k^l) + k_l \bar{a}(\psi, z_k^l) + \frac{1}{2}k_{l+1} \bar{a}(\psi, z_k^{l+1}) = (\psi, z_k^{l+1}), \quad (3.11e)$$

$$\text{d) } l \in J^c, l \in J^1 : \quad (\psi, z_k^l) + \frac{1}{2}k_l \bar{a}(\psi, z_k^l) = (\psi, z_k^{l+1}) - \frac{1}{2}k_{l+1} \bar{a}(\psi, z_k^{l+1}), \quad (3.11f)$$

iii. For  $l = 0$ :

$$\text{a) } 0 \in J^{\text{dc}} : \quad (\psi, z_k^0) = (\psi, z_k^1), \quad (3.11g)$$

$$\text{b) } 0 \in J^c : \quad (\psi, z_k^0) = (\psi, z_k^1) - \frac{1}{2}k_1 \bar{a}(\psi, z_k^1). \quad (3.11h)$$

The dual scheme corresponding to the fully discretized problem (2.9) or (3.9) reads exactly the same except that  $z_{kh}, \psi_h \in V_{h,0}^{s,m}$ .

*Remark 3.1.* We remark that the above time marching schemes with a constant step size and  $J^0 = \emptyset$  coincide with a Crank-Nicolson method whose time intervals are shifted by  $\frac{1}{2}k$ .

## 4 A Posteriori Error Estimation

After the initial considerations in the previous section, we are now prepared to derive dual weighted a posteriori error estimates for both the temporal as well as the spatial error with respect to a given quantity of interest  $\mathcal{J}(u)$ . Let  $u$ ,  $u_k$  and  $u_{kh}$  be the solutions to (2.4), (3.7), and (3.9), respectively. We intend to derive error estimators  $\eta_h$  and  $\eta_k$  such that the following holds:

$$\mathcal{J}(u) - \mathcal{J}(u_{kh}) = \{\mathcal{J}(u) - \mathcal{J}(u_k)\} + \{\mathcal{J}(u_k) - \mathcal{J}(u_{kh})\} \approx \eta_k + \eta_h.$$

For simplicity and because this situation is of particular interest in financial mathematics, we assume  $\mathcal{J} : V \rightarrow \mathbb{R}$  to be a continuous linear functional. Two examples which we consider in Section 5 are the value of an option and the value of the Delta of an option at the time  $t = T$  (ie, today) for a given price of the underlying  $x_0$ . In this cases the functionals would read  $\mathcal{J}(u(T)) = u(T, x_0)$  and  $\mathcal{J}(u(T)) = \partial_x u(T, x_0)$  respectively.

*Remark 4.1.* In fact, the above given functionals are not continuous on  $V$ . To cope with this difficulty, one can either use local mean values of these quantities, or do an analysis in a slightly modified setting, where primal and dual variables live in different function spaces. Since the resulting error indicators will not be effected by these technicalities we refrain from the these modifications in the following.

For notational simplicity, we will introduce two additional bilinear forms  $B : W(V) \times W(V) \rightarrow \mathbb{R}$  and  $\tilde{B} : \{W(V) \cup \hat{X}_k^1\} \times \{W(V) \cup \tilde{X}_k^0\} \rightarrow \mathbb{R}$  by

$$\begin{aligned} B(v, w) &:= ((\partial_t v, w)) + a(v, w) + (v(0), w(0)), \\ \tilde{B}(v_k, w_k) &:= \sum_{m \in J} ((\partial_t v_k, w_k))_m + a(v_k, w_k) + \sum_{l \in J^{\text{dc}}} ([v_k]_l, w_k^{l,+}) + (v_k(0), w_k(0)). \end{aligned}$$

Then, the solutions  $u \in \tilde{g} + W(V_0)$ ,  $u_k \in \tilde{g}_k + \hat{X}_k^1(V_0)$ , and  $u_{kh} \in \tilde{g}_k + \hat{X}_{kh,0}^{1,1}$  to (2.4), (3.7), and (3.9) can equivalently be obtained by

$$B(u, \varphi) = (u_0, \varphi(0)) \quad \forall \varphi \in W(V_0), \quad (4.1a)$$

$$\tilde{B}(u_k, \varphi_k) = (u_0, \varphi_k(0)) \quad \forall \varphi_k \in \tilde{X}_k^0(V_0), \quad (4.1b)$$

$$\tilde{B}(u_{kh}, \varphi_{kh}) = (u_0, \varphi_{kh}(0)) \quad \forall \varphi_{kh} \in \tilde{X}_{kh,0}^{0,1}. \quad (4.1c)$$

Analogously, the dual solutions  $z \in W(V_0)$ ,  $z_k \in \tilde{X}_k^0(V_0)$ , and  $z_{kh} \in \tilde{X}_{kh,0}^{0,1}$  given by (3.10) and (3.11) are also given by

$$B(\varphi, z) = \mathcal{J}(\varphi(T)) \quad \forall \varphi \in W(V_0), \quad (4.2a)$$

$$\tilde{B}(\varphi_k, z_k) = \mathcal{J}(\varphi_k(T)) \quad \forall \varphi_k \in \hat{X}_k^1(V_0), \quad (4.2b)$$

$$\tilde{B}(\varphi_{kh}, z_{kh}) = \mathcal{J}(\varphi_{kh}(T)) \quad \forall \varphi_{kh} \in \hat{X}_{kh,0}^{1,1}. \quad (4.2c)$$

Then, we define the corresponding Lagrange functionals  $\mathcal{L} : W(V) \times W(V_0) \rightarrow \mathbb{R}$  and  $\tilde{\mathcal{L}} : \hat{X}_k^1(V) \times \tilde{X}_k^0(V_0) \rightarrow \mathbb{R}$  by

$$\mathcal{L}(\delta u, \delta z) := \mathcal{J}(\delta u(T)) - B(\delta u, \delta z) + (u_0, \delta z(0)), \quad (4.3a)$$

$$\tilde{\mathcal{L}}(\delta u_k, \delta z_k) := \mathcal{J}(\delta u_k(T)) - \tilde{B}(\delta u_k, \delta z_k) + (u_0, \delta z_k(0)). \quad (4.3b)$$

From (4.3), we obtain using (4.1) and (4.2) that the *primal* solutions  $u$ ,  $u_k$  and  $u_{kh}$  together with their *adjoint/dual* counterparts  $z$ ,  $z_k$  and  $z_{kh}$  are stationary points of these Lagrangians in the sense that

$$\mathcal{L}'(u, z)(\delta u, \delta z) = 0 \quad \forall (\delta u, \delta z) \in W(V_0) \times W(V_0), \quad (4.4a)$$

$$\tilde{\mathcal{L}}'(u_k, z_k)(\delta u_k, \delta z_k) = 0 \quad \forall (\delta u_k, \delta z_k) \in \hat{X}_k^1(V_0) \times \tilde{X}_k^0(V_0), \quad (4.4b)$$

$$\tilde{\mathcal{L}}'(u_{kh}, z_{kh})(\delta u_{kh}, \delta z_{kh}) = 0 \quad \forall (\delta u_{kh}, \delta z_{kh}) \in \hat{X}_{kh,0}^{1,1} \times \tilde{X}_{kh,0}^{0,1}. \quad (4.4c)$$

We note that for functions  $\delta z \in W(V_0) \cup \tilde{X}_k^0(V_0)$  and  $\delta u \in W(V_0) \cup \hat{X}_k^1(V_0)$  the corresponding *residuals* of the primal and dual equations are defined by

$$\rho(u)(\delta z) = \tilde{\mathcal{L}}'_z(u, z)(\delta z) = -\tilde{B}(u, \delta z) + (u_0, \delta z(0)),$$

$$\rho'(z)(\delta u) = \tilde{\mathcal{L}}'_u(u, z)(\delta u) = -\tilde{B}(\delta u, z) + \mathcal{J}(\delta u(T)).$$

With this notation, we recall the following basic theorem from Schmich and Vexler (2008) with a minor modification taking care of the time dependent Dirichlet data.

**Theorem 2.** Let  $(u, z) \in \tilde{g} + W(V_0) \times W(V_0)$ ,  $(u_k, z_k) \in \tilde{g}_k + \hat{X}_k^1(V_0) \times \tilde{X}_k^0(V_0)$  and  $(u_{kh}, z_{kh}) \in \tilde{g}_k + \hat{X}_{kh,0}^{1,1} \times \tilde{X}_{kh,0}^{0,1}$  be solutions to (4.4). Then, the following error representation hold for the temporal and spatial discretization errors:

$$\begin{aligned}\mathcal{J}(u(T)) - \mathcal{J}(u_k(T)) &= \frac{1}{2} (\rho(u_k)(z - \psi_k) + \rho'(z_k)(u - \varphi_k)) + \mathcal{B}_k, \\ \mathcal{J}(u_k(T)) - \mathcal{J}(u_{kh}(T)) &= \frac{1}{2} (\rho(u_{kh})(z_k - \psi_{kh}) + \rho'(z_{kh})(u_k - \varphi_{kh})).\end{aligned}$$

The functions  $(\varphi_k, \psi_k) \in \tilde{g}_k + \hat{X}_k^1(V_0) \times \tilde{X}_k^0(V_0)$  and  $(\varphi_{kh}, \psi_{kh}) \in \tilde{g}_k + \hat{X}_{kh,0}^{1,1} \times \tilde{X}_{kh,0}^{0,1}$  can be chosen arbitrarily. The term  $\mathcal{B}_k$  is defined by

$$\mathcal{B}_k = \inf_{\varphi \in \tilde{g} + W(V_0)} -\frac{1}{2} \tilde{B}(\varphi - \tilde{g}_k)(z)$$

and measures the oscillation of the time dependent Dirichlet data.

#### 4.1 Evaluation of the Error Identities

The representation of the error in Theorem 2 is not yet useful for estimating the error, because it contains the unknown semi-discrete solutions  $(u_k, z_k)$  and the weights  $(u - \varphi_k, z - \psi_k)$ ,  $(u_k - \varphi_{kh}, z_k - \psi_{kh})$  as well as the data oscillation term  $\mathcal{B}_k$ . This problem is circumvented by simply replacing the semi-discrete solution  $(u_k, z_k)$  by the fully discrete solution  $(u_{kh}, z_{kh})$  in the first argument of the residual. This approximation leads to reasonable estimates as is demonstrated in Schmich and Vexler (2008). Second, we neglect the data oscillation term  $\mathcal{B}_k$ . Although this is not feasible in general, here the oscillating term is at worst given as a piece-wise linear approximation of  $e^{-rt}$  for  $t \in I$  and is concentrated near the boundary where the weight  $z$  is small. The resulting approximate error representations read as follows:

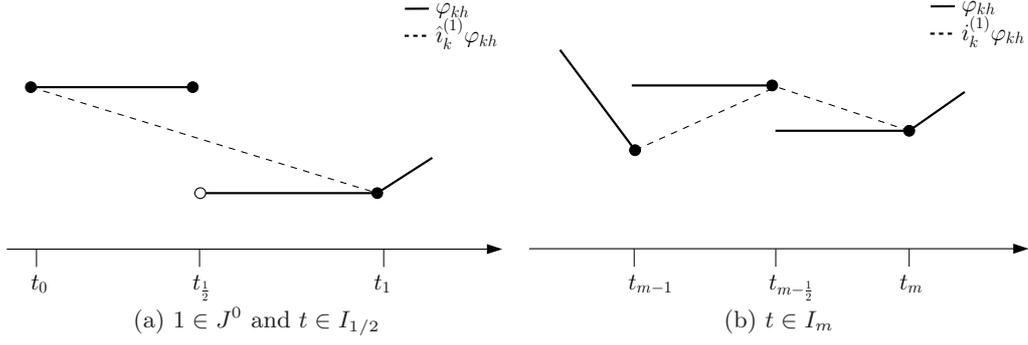
$$\begin{aligned}\mathcal{J}(u) - \mathcal{J}(u_k) &\approx \frac{1}{2} (\rho(u_{kh})(z - \psi_k) + \rho'(z_{kh})(u - \varphi_k)), \\ \mathcal{J}(u_k) - \mathcal{J}(u_{kh}) &= \frac{1}{2} (\rho(u_{kh})(z_k - \psi_{kh}) + \rho'(z_{kh})(u_k - \varphi_{kh})).\end{aligned}$$

Now, we turn to the remaining approximation of the weights. There are several possible ways to do this, see Becker and Rannacher (2001). Here, we use local post-processing of the computed solution on the current space-time mesh through higher order interpolation on mesh patches. This means, we define the approximate errors

$$\begin{aligned}\Pi_k^{(u)} u_{kh} &\approx u - \varphi_k, & \Pi_k^{(z)} z_{kh} &\approx z - \psi_k, \\ \Pi_h^{(u)} u_{kh} &\approx u_k - \varphi_{kh}, & \Pi_h^{(z)} z_{kh} &\approx z_k - \psi_{kh}.\end{aligned}$$

Here the temporal approximations are defined by

$$\begin{aligned}\Pi_k^{(u)} : \hat{X}_{kh}^{1,1} &\rightarrow \hat{X}_{kh}^{2,1} \cap C^0(I; V), & \Pi_k^{(u)} &:= \hat{i}_{2k}^{(2)} - \text{id}, \\ \Pi_k^{(z)} : \tilde{X}_{kh,0}^{0,1} &\rightarrow \tilde{X}_{kh,0}^{1,1}, & \Pi_k^{(z)} &:= \hat{i}_k^{(1)} - \text{id},\end{aligned}$$


 Figure 1: Sketch of  $\hat{i}_k^{(1)} \varphi_{kh}$  on  $I_m$ .

and the spatial approximations by

$$\begin{aligned} \Pi_h^{(u)} : \hat{X}_{kh}^{1,1} &\rightarrow \hat{X}_{kh}^{1,2}, & \Pi_h^{(u)} &:= i_{2h}^{(2)} - \text{id}, \\ \Pi_h^{(z)} : \tilde{X}_{kh,0}^{0,1} &\rightarrow \tilde{X}_{kh,0}^{0,2}, & \Pi_h^{(z)} &:= i_{2h}^{(2)} - \text{id}. \end{aligned}$$

Here,  $\text{id}$  denotes the identity mapping on the corresponding space. The higher-order interpolation operators  $\hat{i}_k^{(2)}$  and  $\hat{i}_k^{(1)}$  are defined as follows:

$$\begin{aligned} \hat{i}_k^{(1)} \varphi_{kh}(t) &:= \begin{cases} \frac{t_1-t}{t_1-t_0} \varphi_{kh}^0 + \frac{t-t_0}{t_1-t_0} \varphi_{kh}^1, & 1 \in J^0, t \in I_{1/2}, \\ \hat{i}_k^{(1)} \varphi_{kh}(t), & \text{otherwise,} \end{cases} \\ \hat{i}_{2k}^{(2)} \varphi_{kh}(t) &:= \begin{cases} \hat{i}_k^{(1)} \varphi_{kh}(t), & t \in I_m, m \in J^0, \\ \hat{i}_{2k}^{(2)} \varphi_{kh}(t), & t \in I_m, m \in J^1. \end{cases} \end{aligned}$$

*Remark 4.2.* Except for the time point  $t_{1/2}$  the function  $\Pi_k^{(z)} z_{kh}$  is continuous for any  $z_{kh} \in \tilde{X}_{kh,0}^{0,1}$ . The special treatment for  $t_{1/2}$  is due to the fact that in case  $1 \in J_L$ , we have  $0 \in J^{\text{dc}}$  and hence  $z_{kh}^{1/2} = z_{kh}^0$  because of (3.11g). This is why  $(i_k^{(1)} - \text{id}) z_{kh}^{1/2} \approx 0$  is not a good approximation for the interpolation error, see Figure 1.

The temporal interpolation operator  $i_k^{(1)}$  is defined as follows:

$$i_k^{(1)} \varphi_{kh}(t) := \begin{cases} \frac{t_m-t}{k_m} \varphi_{kh}^{m-1} + \frac{t-t_{m-1}}{k_m} \varphi_{kh}^m, & t \in I_m, m \in J^1, \\ \frac{t_m-t}{k_m} \varphi_{kh}^{m-1/2} + \frac{t-t_{m-1/2}}{k_m} \varphi_{kh}^m, & t \in I_m, m \in J^0. \end{cases}$$

A sketch of the corresponding operator  $\hat{i}_k^{(1)}$  can be seen in Figure 1.

To define  $i_{2k}^{(2)}$ , we have to consider several different cases. Let  $t \in I_m$  with  $m \in J$ . Then, we define  $i_{2k}^{(2)} \varphi_{kh}(t)$  by

(a)  $m - 1 \in 2\mathbb{N}$ ,  $m + 1 \in J^1$ :

$$\begin{aligned} i_{2k}^{(2)} \varphi_{kh}(t) &:= \frac{(t_m - t)(t_{m+1} - t)}{k_m(k_m + k_{m+1})} \varphi_{kh}^{m-1} + \frac{(t - t_{m-1})(t_{m+1} - t)}{k_m k_{m+1}} \varphi_{kh}^m \\ &\quad + \frac{(t - t_m)(t - t_{m-1})}{k_{m+1}(k_m + k_{m+1})} \varphi_{kh}^{m+1}, \end{aligned}$$

(b)  $m - 1 \in 2\mathbb{N}$ ,  $m + 1/2 \in J^0$ :

$$\begin{aligned} i_{2k}^{(2)} \varphi_{kh}(t) &:= \frac{(t_m - t)(t_{m+1/2} - t)}{k_m(k_m + k_{m+1/2})} \varphi_{kh}^{m-1} + \frac{(t - t_{m-1})(t_{m+1/2} - t)}{k_m k_{m+1/2}} \varphi_{kh}^m \\ &\quad + \frac{(t - t_m)(t - t_{m-1})}{k_{m+1/2}(k_m + k_{m+1/2})} \varphi_{kh}^{m+1/2}, \end{aligned}$$

(c)  $m \in 2\mathbb{N}$ ,  $m - 1 \in J^1$ :

$$\begin{aligned} i_{2k}^{(2)} \varphi_{kh}(t) &:= \frac{(t_{m-1} - t)(t_m - t)}{k_{m-1}(k_{m-1} + k_m)} \varphi_{kh}^{m-2} + \frac{(t - t_{m-2})(t_m - t)}{k_{m-1} k_m} \varphi_{kh}^{m-1} \\ &\quad + \frac{(t - t_{m-1})(t - t_{m-2})}{k_m(k_{m-1} + k_m)} \varphi_{kh}^m, \end{aligned}$$

(d)  $m \in 2\mathbb{N}$ ,  $m - 1 \in J^0$ :

$$\begin{aligned} i_{2k}^{(2)} \varphi_{kh}(t) &:= \frac{(t_{m-1} - t)(t_m - t)}{k_{m-1}(k_{m-1} + k_m)} \varphi_{kh}^{m-3/2} + \frac{(t - t_{m-3/2})(t_m - t)}{k_{m-1} k_m} \varphi_{kh}^{m-1} \\ &\quad + \frac{(t - t_{m-1})(t - t_{m-3/2})}{k_m(k_{m-1} + k_m)} \varphi_{kh}^m. \end{aligned}$$

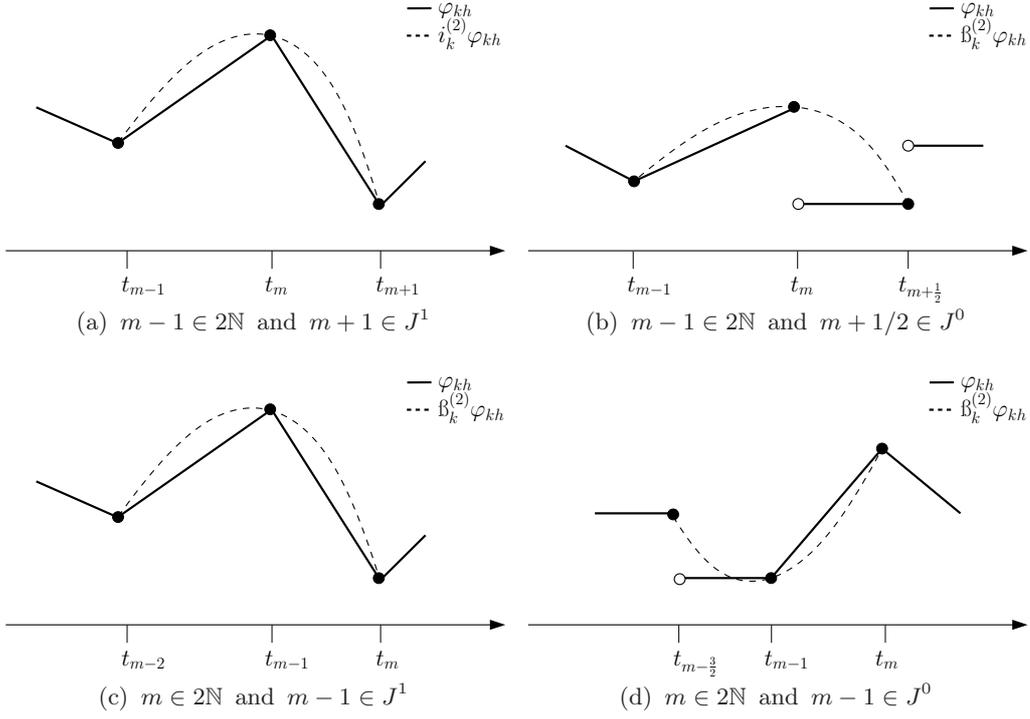
A sketch of the operator  $i_{2k}^{(2)} \varphi_{kh}$  is shown in Figure 2.

For the construction of the spatial interpolation operator  $i_{2h}^{(2)}$ , we assume that all spatial meshes possess a patch structure. That is for all  $m$ , we assume that  $\mathcal{T}_h^m$  is given as a global uniform refinement of a mesh  $\mathcal{T}_{2h}^m$ . Then, for each element  $K$  in  $\mathcal{T}_h^m$  there exist  $2^d - 1$  neighbors such that their union defines an element  $\tilde{K} \in \mathcal{T}_{2h}^m$ . Thereby, we can define a piece-wise (with respect to  $\mathcal{T}_{2h}^m$ ) quadratic interpolation  $i_{2h}^{(2)}: V_h^{1,m} \rightarrow V_{2h}^{2,m}$ . By application of this for all  $m \in J$ , we obtain the desired interpolation for  $\varphi_{kh} \in \tilde{X}_{kh}^{1,1}$  or  $\varphi_{kh} \in \tilde{X}_{kh,0}^{0,1}$  by

$$(i_{2h}^{(2)} \varphi_{kh})(t) := i_{2h}^{(2)}(\varphi_{kh}(t)).$$

As a consequence of the previous considerations, we can now define a computable error estimator  $\eta = \eta_k + \eta_h$ , where the temporal error estimator  $\eta_k$  and the spatial error estimator  $\eta_h$  are given by

$$\begin{aligned} \eta_k &:= \frac{1}{2} (\rho(u_{kh})(\Pi_k^{(z)} z_{kh}) + \rho'(z_{kh})(\Pi_k^{(u)} u_{kh})), \\ \eta_h &:= \frac{1}{2} (\rho(u_{kh})(\Pi_h^{(z)} z_{kh}) + \rho'(z_{kh})(\Pi_h^{(u)} u_{kh})). \end{aligned}$$


 Figure 2: Sketch of  $i_{2k}^{(2)} \varphi_{kh}$  on  $I_m$ .

Then, with this notation, we have the following approximate a posteriori error estimate:

$$\mathcal{J}(u) - \mathcal{J}(u_{kh}) \approx \eta := \eta_k + \eta_h.$$

*Remark 4.3.* The derivation of the above a posteriori error estimator includes various approximations, particularly the approximation of the weights by local post-processing. This may be seen as a weak point generally inherent to the DWR method. However, there is some theoretical justification through superconvergence on uniform meshes, cf. Bangert and Rannacher (2003), and by the rich practical experience also on locally adapted meshes.

## 4.2 Localization of the Error Estimates

In order to use the error estimators  $\eta_h$  and  $\eta_k$  for local mesh refinement, we need to localize them to element-wise contributions.

#### 4.2.1 Localization in Time

In order to localize the error estimates in time, we define for functions  $v, w$  and  $m \in J$  the following abbreviation:

$$a_m(v, w) := \int_{I_m} \bar{a}(v(t), w(t)) dt.$$

Then, we define the local primal and dual residuals  $\rho_m(u_{kh}, \delta z)$  and  $\rho'_m(z_{kh}, \delta u)$  for  $m \in J \cup \{0\}$ ,  $\delta u \in \hat{X}_{kh,0}^{2,2}$  and  $\delta z \in \tilde{X}_{kh,0}^{1,1} \cup \tilde{X}_{kh,0}^{0,2}$  as follows:

- The local primal residual  $\rho_m(\delta u)(\delta z)$  is defined by

- (a) For  $m \in J^0$ :

$$\rho_m(u_{kh}, \delta z) = -((\partial_t u_{kh}, \delta z))_m + a_m(u_{kh}, \delta z) + ([u_{kh}]_{m-1/2}, \delta z^{m-1/2,+}),$$

- (b) For  $m \in J^1$ :

$$\rho_m(u_{kh}, \delta z) = -((\partial_t u_{kh}, \delta z))_m + a_m(u_{kh}, \delta z),$$

- (c) For  $m = 0, 1/2 \in J^0$ :

$$\rho_0(u_{kh}, \delta z) = (u_0 - u_{kh}(0), \delta z^{0,-}) + ([u_{kh}]_0, \delta z^{0,+}),$$

- (d) For  $m = 0, 1 \in J^1$ :

$$\rho_0(u_{kh}, \delta z) = (u_0 - u_{kh}(0), \delta z^{0,-}).$$

- The local dual residual  $\rho'_m(z_{kh})(\delta u)$  is defined by

- (a) For  $m = M$ :

$$\rho'_M(z_{kh}, \delta u) := \mathcal{J}(\delta u) - a_M(\delta u, z_{kh}) - (\delta u(t_M), z_{kh}^{M,-}),$$

- (b) For  $m \neq M, m \neq 0$ :

$$\rho'_m(z_{kh}, \delta u) := -a_m(\delta u, z_{kh}) - (\delta u(t_m), [z_{kh}]_m),$$

- (c) For  $m = 0$ :

$$\rho'_0(z_{kh}, \delta u) := -(\delta u(0), [z_{kh}]_0).$$

Now, we can define the local primal and dual error indicators  $\hat{\eta}_k^{p,m}$ ,  $\hat{\eta}_k^{d,m}$ ,  $\eta_h^{p,l}$  and  $\eta_h^{d,l}$  for  $m \in J$  and  $l \in J \cup \{0\}$  as follows:

$$\begin{aligned} \hat{\eta}_k^{p,m} &:= \rho_m(u_{kh}, \Pi_k^{(z)} z_{kh}), & \hat{\eta}_k^{d,m} &:= \rho'_m(z_{kh}, \Pi_k^{(u)} u_{kh}), \\ \eta_h^{p,l} &:= \rho_l(u_{kh}, \Pi_h^{(z)} z_{kh}), & \eta_h^{d,l} &:= \rho'_l(z_{kh}, \Pi_h^{(u)} u_{kh}). \end{aligned}$$

In particular, we can split the global time error estimator into local contributions  $\hat{\eta}_k^m$  such that

$$\eta_k = \frac{1}{2} \sum_{m \in J} \hat{\eta}_k^m, \quad \hat{\eta}_k^m = \hat{\eta}_k^{p,m} + \hat{\eta}_k^{d,m}. \quad (4.5)$$

The global spatial error estimator can then be split into the contributions  $\eta_h^l$  from the different time intervals

$$\eta_h = \frac{1}{2} \sum_{l \in J \cup \{0\}} \eta_h^l, \quad \eta_h^l = \eta_h^{p,l} + \eta_h^{d,l}. \quad (4.6)$$

The local indicators  $\hat{\eta}_k^m$  from (4.5) can immediately be used to stir temporal refinement. In contrast the indicators  $\eta_h^l$  from (4.6) are not local in space, yet.

However, we do not intend to refine the temporal mesh  $\hat{\mathcal{T}}_k$  but rather  $\mathcal{T}_k$ . To this end, we add the temporal indicators on two consecutive intervals in  $J^0$  to obtain

$$\eta_k^m := \begin{cases} \hat{\eta}_k^m, & m \in J^1, \\ \hat{\eta}_k^m + \hat{\eta}_k^{m-1/2}, & m \in J_L. \end{cases}$$

#### 4.2.2 Localization in Space

We will now consider the localization of  $\eta_h^l$  into element-wise contributions. A direct localization of  $\eta_h^l$  as done in the temporal case leads to problems due to the oscillatory behavior of the cell residuals, see, eg, Carstensen and Verfürth (1999). To overcome this problem, we integrate by parts and obtain

$$\bar{a}(v, w) = \sum_{K \in \mathcal{T}_h^l} \{ (A_{\text{BS}} v, w)_K + \frac{1}{2} ([n \cdot (A \nabla v)], w)_{\partial K} \}, \quad (4.7)$$

$$\bar{a}(v, w) = \sum_{K \in \mathcal{T}_h^l} \{ (v, A'_{\text{BS}} w)_K + \frac{1}{2} (v, [n \cdot (A \nabla w)])_{\partial K} \}, \quad (4.8)$$

for arbitrary functions  $v, w \in V_{h,0}^{s,l}$  with  $l \in J \cup \{0\}$ . Here,  $A_{\text{BS}}$  and  $A'_{\text{BS}}$  denote the strong form of the Black-Scholes operator, ie, with the notation of (2.4) and functions  $\tilde{v}, \tilde{w} \in C^\infty(K)$  on  $K \in \mathcal{T}_h^l$ , it holds

$$\begin{aligned} A_{\text{BS}} \tilde{v} &:= -\nabla \cdot (A \nabla \tilde{v}) - \beta \cdot \nabla \tilde{v} + r \tilde{v}, \\ A'_{\text{BS}} \tilde{w} &:= -\nabla \cdot (A \nabla \tilde{w}) - \nabla \cdot (\beta \tilde{w}) + r \tilde{w}. \end{aligned}$$

With these preparations, we can localize  $\eta_h^{p,l}$  and  $\eta_h^{d,l}$  to element-wise contributions at time  $t_l$  as

$$\eta_h^{p,l} = \sum_{K \in \mathcal{T}_h^l} \eta_{h,K}^{p,l}, \quad \eta_h^{d,l} = \sum_{K \in \mathcal{T}_h^l} \eta_{h,K}^{d,l}. \quad (4.9)$$

For  $m \in J$  and  $K \in \mathcal{T}_h^m$ , we denote the scalar product on  $L^2(I_m \times K)$  by  $((\cdot, \cdot))_{m,K}$  and the scalar product on  $L^2(I_m \times \partial K)$  by  $((\cdot, \cdot))_{m,\partial K}$ . Then, the summands  $\eta_{h,K}^{p,m}$  in (4.9) are defined by

(a) For  $m \in J^0$ :

$$\begin{aligned} \eta_{h,K}^{p,m} = & -((\partial_t u_{kh} + A_{BS} u_{kh}, \Pi_h^{(z)} z_{kh}))_{m,K} - \frac{1}{2}([n \cdot (A \nabla u_{kh}(t))], \Pi_h^{(z)} z_{kh})_{m,\partial K} \\ & + ([u_{kh}]_{m-1/2}, (\Pi_h^{(z)} z_{kh})^{m-1/2,+})_K, \end{aligned}$$

(b) For  $m \in J^1$ :

$$\eta_{h,K}^{p,m} = -((\partial_t u_{kh} + A_{BS} u_{kh}, \Pi_h^{(z)} z_{kh}))_{m,K},$$

(c) For  $m = 0, 1/2 \in J^0$ :

$$\eta_{h,K}^{p,0} = (u_0 - u_{kh}(0), (\Pi_h^{(z)} z_{kh})^{0,-})_K + ([u_{kh}]_0, (\Pi_h^{(z)} z_{kh})^{0,+})_K,$$

(d) For  $m = 0, 1 \in J^1$ :

$$\eta_{h,K}^{p,0} = (u_0 - u_{kh}(0), (\Pi_h^{(z)} z_{kh})^{0,-})_K,$$

and the summands  $\eta_{h,K}^{d,m}$  are defined by

(a) For  $m = M$ :

$$\begin{aligned} \eta_{h,K}^{d,M} := & \mathcal{J}(\Pi_h^{(u)} u_{kh}|_K) - ((\Pi_h^{(u)} u_{kh}, A'_{BS} z_{kh}))_{M,K} \\ & - \frac{1}{2}((\Pi_h^{(u)} u_{kh}, [n \cdot (A \nabla z_{kh})]))_{M,\partial K} - (\Pi_h^{(u)} u_{kh}(t_M), z_{kh}^{M,-})_K, \end{aligned}$$

(b) For  $m \neq M, m \neq 0$ :

$$\begin{aligned} \eta_{h,K}^{d,m} := & -((\Pi_h^{(u)} u_{kh}, A'_{BS} z_{kh}))_{m,K} - \frac{1}{2}((\Pi_h^{(u)} u_{kh}(t), [n \cdot (A \nabla z_{kh}(t))]))_{m,\partial K} \\ & - (\Pi_h^{(u)} u_{kh}(t_m), [z_{kh}]_m)_K, \end{aligned}$$

(c) For  $m = 0$ :

$$\eta_{h,K}^{d,0} := -(\Pi_h^{(u)} u_{kh}(0), [z_{kh}]_0)_K.$$

Hence, we can rewrite  $\eta_h$  as follows:

$$\eta_h = \frac{1}{2} \sum_{l \in J \cup \{0\}} \sum_{K \in \mathcal{T}_h^l} \eta_{h,K}^l, \quad \eta_{h,K}^l = \eta_{h,K}^{p,l} + \eta_{h,K}^{d,l}.$$

Since we are concerned with one spatial mesh for all time points  $t_l$  with  $l \in J \cup \{0\}$ , we sum all element indicators for one element over time and obtain  $\eta_{h,K} = \sum_{l \in J \cup \{0\}} \eta_{h,K}^l$ . Then, we have

$$\eta_h = \frac{1}{2} \sum_{l \in J \cup \{0\}} \sum_{K \in \mathcal{T}_h^l} \eta_{h,K}^l = \sum_{K \in \mathcal{T}_h} \eta_{h,K}.$$

Thereby, we have obtained two sets of local error indicators  $\Sigma_k$  and  $\Sigma_h$  given by

$$\Sigma_k := \{\eta_k^m | 1 \leq m \leq M\}, \quad \Sigma_h := \{\eta_{h,K} | K \in \mathcal{T}_h\}.$$

On this basis, we can now stir the simultaneous adaptation of space and time discretization.

### 4.3 Refinement Cycle

We will now briefly describe the algorithm for space-time adaptivity based on the above a posteriori error estimates. In order to obtain an efficient algorithm, we attempt to balance the temporal and spatial error contributions. Hence for some given parameter  $\kappa \geq 1$ , we attempt to choose  $\mathcal{T}_k$  and  $\mathcal{T}_h$  such that

$$\frac{1}{\kappa} \leq \frac{\eta_h}{\eta_k} \leq \kappa.$$

In our computations, we have chosen  $\kappa = 4$ . In addition, for a sequence  $k_n$  and  $h_n$  denote the cardinality of the corresponding triangulations by  $M_n = \text{card}(\mathcal{T}_{k_n})$  and  $N_n = \text{card}(\mathcal{T}_{h_n})$ . Then, the algorithm reads as described in Algorithm 1.

There are several techniques available in order to choose a subset of elements in  $\mathcal{T}_k$  or  $\mathcal{T}_h$  to be refined in Algorithm 1. Here, we will employ a refinement strategy based upon minimization of the product of expected error and computational effort required for the solution on the new mesh, see Richter (2005). However, due to the way we evaluate the weights in the error estimate, we require that both temporal and spatial mesh satisfy a patch structure after refinement.

---

#### Algorithm 1 Space time adaptive finite elements

---

- 1: Choose parameters  $M_{\max}, N_{\max} \in \mathbb{N}$ ,  $\kappa \geq 1$  and  $\text{TOL} > 0$ .
  - 2: Choose initial meshes  $\mathcal{T}_{k_0}$  and  $\mathcal{T}_{h_0}$ .
  - 3: Choose a subset of damped intervals  $J_L^0$ .
  - 4: Let  $n = 0$ .
  - 5: **loop**
  - 6:   Determine  $\hat{\mathcal{T}}_{k_n}$  from  $J_L^n$  and  $\mathcal{T}_{k_h}$ .
  - 7:   Compute primal and dual solution  $u_{k_n h_n}$  and  $z_{k_n h_n}$ .
  - 8:   Compute local error indicators  $\Sigma_{k_n}, \Sigma_{h_n}$  and error estimates  $\eta_{k_n}$  and  $\eta_{h_n}$ .
  - 9:   **if**  $M_n \geq M_{\max}$  or  $N_n \geq N_{\max}$  or  $\eta_{k_n} + \eta_{h_n} \leq \text{TOL}$  **then**
  - 10:     **return**
  - 11:   **else**
  - 12:     **if**  $|\eta_{k_n}| > \kappa |\eta_{h_n}|$  **then**
  - 13:       Use  $\Sigma_{k_n}$  to refine  $\mathcal{T}_{k_n}$ .
  - 14:     **else if**  $|\eta_{h_n}| > \kappa |\eta_{k_n}|$  **then**
  - 15:       Use  $\Sigma_{h_n}$  to refine  $\mathcal{T}_{h_n}$ .
  - 16:     **else**
  - 17:       Use  $\Sigma_{k_n}$  and  $\Sigma_{h_n}$  to refine both  $\mathcal{T}_{k_n}$  and  $\mathcal{T}_{h_n}$ .
  - 18:    Compute  $J_L^{n+1}$ .
  - 19:    Let  $n = n + 1$ .
- 

## 5 Numerical Examples

We will now test the proposed algorithm for mesh adaptation and discuss the effect of the damping in the Crank-Nicolson scheme on the error estimation by some numerical

examples in one and two space dimensions. As a quantity of interest, we consider a point evaluation of the solution  $u$  or the Delta of  $u$ , ie,  $\partial_x u$ , as mentioned in Remark 2.1.

In order to evaluate the quality of the error estimator, we define the effectivity index

$$I_{\text{eff}} = \frac{\eta}{\mathcal{J}((u - u_{kh})(T))}.$$

In order to discuss the effect of the damping, we will consider several choices of the damping intervals. Hence for  $m_p, m_d \in \mathbb{N}$ , we denote by  $dk\text{-}m_p\text{-}m_d$  a damped Crank-Nicolson method where on the first  $m_p$  and the last  $m_d$  intervals damping is used, ie, the set  $J_L$  defined in Section 3.2 is given by  $J_L = \{1, \dots, m_p\} \cup \{M + 1 - m_d, \dots, M\}$ . The reason to allow for damping at the end of the time interval is that we have to solve the dual problem (3.11), which corresponds on the undamped intervals to a Crank-Nicolson method in the interval midpoints. Since the initial condition for the dual problem is given as  $\mathcal{J}$  it is reasonable—and as we will see necessary—to apply damping to the dual problem, see also our discussion at the beginning of Section 2.2.1

For the subsequent computations, we will consider plain vanilla European options. In particular, we will consider a 1D call and a 2D put with the values given in Table 1. The data in Table 1 for the 1D call is taken from Ern et al. (2004), the exact value  $u(T, x_0)$  has been computed using an analytic formula. The 2D put comes from Reisinger (2004). Because the value of  $u(T, x_0)$  is unknown in this case, we computed a reference value on a very fine mesh ( $M = 512$ ,  $N = 800065$ ) using the  $dk\text{-}2\text{-}0$  method. We note that the initial condition in Reisinger (2004) was chosen as  $\max(K - 0.5x_1 - 0.5x_2, 0)$  in contrast to the default value stated earlier in this paper. In order to have better comparability, we do our computations with the changed initial condition.

Table 1: Data for the test cases: The data for the call option is taken from Ern et al. (2004). The data for the put option on two uncorrelated shares is taken from Reisinger (2004).

	$x_0$	$K$	$T$	$\sigma$	$\bar{x}$	$r$	$u(T, x_0)$	$\partial_x u(T, x_0)$
1d-Call	100	100	1	0.2	200	$\log(1.1)$	$\approx 12.9927$	$\approx 0.7179$
2d-Put	(25,25)	25	1	$(\frac{1}{2}, \frac{3}{10})$	(100,100)	0.05	$\approx 2.2692$	

In both cases the choice of  $x_0$  was motivated by the fact that the computation of  $u(T, x_0)$  is most challenging near the strike price  $K$ .

The computations were done using the C++-library `deal.II`, see Bangerth et al. (2007) together with UMFPACK Davis (2004) for the solution of the linear equations. All the integrals are evaluated with a numerical quadrature formula, and all integrals involving only piece-wise polynomials are evaluated exactly. However, we did not compute the exact projection of the initial conditions for the 2d put onto the finite element space. We remark that there is a slight sensitivity of the error estimator regarding the numerical integration of the initial data. For the numerical calculations we used a high order Gaussian quadrature rule (order seven) to compute the projection of the initial data.

*Remark 5.1.* We should note that in the example considered here the memory of our workstation has been more than sufficient to store both the primal and dual solutions in main memory. Due to the local nature of the reconstruction process involved it is at any time sufficient to have the primal and dual solution at three neighboring time points to be able to calculate the reconstructions  $\Pi_h^{(u)}$  and  $\Pi_h^{(z)}$ . Thus, it is possible to store most of the solution on the hard-disc if main memory is a limiting factor. Furthermore, if such techniques fail there are, so-called, check-pointing strategies available that allow a trade of between memory requirements and computational time, see, eg, Becker et al. (2007); Berggren et al. (1996); Griewank (1991). Finally, for real high-dimensional problems different spatial discretizations like sparse grids may be employed, see Bungartz and Griebel (2004).

### 5.1 Independence of the Error Estimators $\eta_k$ , $\eta_h$ and Effects of Damping

We begin by considering the behavior of the temporal error estimate  $\eta_k$  under spatial refinement and the behavior of the spatial error estimate  $\eta_h$  under temporal refinement. The results for the 1d case are shown in Table 2 the results in 2D are given in Table 3.

Table 2: Invariance of  $\eta_h$  under temporal refinement (a) and invariance of  $\eta_k$  under spatial refinement (b) for the 1D call. (Computation of  $u(T, x_0)$ )

M	$\eta_h$		N	$\eta_k$	
	dk-1-0	dk-1-1		dk-1-0	dk-1-1
4	$1.68 \cdot 10^{-3}$	$1.88 \cdot 10^{-3}$	33	$1.02 \cdot 10^{-4}$	$3.20 \cdot 10^{-4}$
8	$1.79 \cdot 10^{-3}$	$1.85 \cdot 10^{-3}$	65	$1.02 \cdot 10^{-4}$	$3.18 \cdot 10^{-4}$
16	$1.87 \cdot 10^{-3}$	$1.84 \cdot 10^{-3}$	129	$1.01 \cdot 10^{-4}$	$3.17 \cdot 10^{-4}$
32	$1.84 \cdot 10^{-3}$	$1.84 \cdot 10^{-3}$	257	$-2.46 \cdot 10^{-4}$	$3.18 \cdot 10^{-4}$
64	$1.84 \cdot 10^{-3}$	$1.84 \cdot 10^{-3}$	513	$-1.59 \cdot 10^{-5}$	$3.14 \cdot 10^{-4}$
128	$1.84 \cdot 10^{-3}$	$1.84 \cdot 10^{-3}$	1025	$-1.36 \cdot 10^{-4}$	$3.17 \cdot 10^{-4}$

(a) Behavior of  $\eta_h$  for  $N = 129$ .

(b) Behavior of  $\eta_k$  for  $M = 50$ .

Table 3: Invariance of  $\eta_h$  under temporal refinement (a) and invariance of  $\eta_k$  under spatial refinement (b) for the 2D put. (Computation of  $u(T, x_0)$ )

M	$\eta_h$		N	$\eta_k$	
	dk-1-0	dk-1-1		dk-1-0	dk-1-1
8	$7.55 \cdot 10^{-4}$	$7.74 \cdot 10^{-4}$	81	$2.12 \cdot 10^{-4}$	$1.09 \cdot 10^{-3}$
16	$7.81 \cdot 10^{-4}$	$7.72 \cdot 10^{-4}$	289	$2.56 \cdot 10^{-4}$	$9.77 \cdot 10^{-4}$
32	$7.72 \cdot 10^{-4}$	$7.71 \cdot 10^{-4}$	1089	$2.48 \cdot 10^{-4}$	$9.44 \cdot 10^{-4}$
64	$7.71 \cdot 10^{-4}$	$7.71 \cdot 10^{-4}$	4225	$-4.76 \cdot 10^{-4}$	$9.43 \cdot 10^{-4}$
128	$7.71 \cdot 10^{-4}$	$7.71 \cdot 10^{-4}$	16641	$-8.49 \cdot 10^{-4}$	$9.38 \cdot 10^{-4}$

(a) Behavior of  $\eta_h$  for  $N = 16641$ .

(b) Behavior of  $\eta_k$  for  $M = 16$ .

We observe that the spatial error estimate remains unchanged under temporal refinement. In contrast, the temporal error estimate remains constant under spatial refinement only for the  $dk-1-1$  method while it changes drastically in the case of the  $dk-1-0$  method. In fact, for these examples the temporal error estimate remains invariant under spatial refinement as long as damping is applied to the dual equation on an interval  $I_m$  with  $m > 1$ . Already from this example, we can see the importance of appropriate dual damping.

After these initial considerations, we will show the numerical quality of the temporal error estimator under refinement in time. To this end, we consider a very fine spatial discretization in order to ensure that the temporal error is dominant. In Table 4, we compare the effectivity indices for the  $dk-1-0$  and  $dk-1-1$  method.

Table 4: Comparison of the effectivity of the temporal error estimator  $\eta_k$  for dominant temporal error under uniform refinement in time for the 1D call. In fact, we have  $\eta_h \approx 1.1 \cdot 10^{-7}$  for the present situations. (Computation of  $u(T, x_0)$ )

M	dk-1-0			dk-1-1		
	$ \mathcal{J}(e) $	$\eta_k$	$I_{\text{eff}}$	$ \mathcal{J}(e) $	$\eta_k$	$I_{\text{eff}}$
8	$4.17 \cdot 10^{-3}$	$-3.67 \cdot 10^{-3}$	-0.88	$1.35 \cdot 10^{-2}$	$1.23 \cdot 10^{-2}$	0.91
16	$1.13 \cdot 10^{-3}$	$-1.02 \cdot 10^{-3}$	-0.90	$3.36 \cdot 10^{-3}$	$3.07 \cdot 10^{-3}$	0.92
32	$2.94 \cdot 10^{-4}$	$-2.69 \cdot 10^{-4}$	-0.91	$8.38 \cdot 10^{-4}$	$7.73 \cdot 10^{-4}$	0.92
64	$7.51 \cdot 10^{-5}$	$-6.94 \cdot 10^{-5}$	-0.92	$2.09 \cdot 10^{-4}$	$1.94 \cdot 10^{-4}$	0.93
128	$1.90 \cdot 10^{-5}$	$-1.77 \cdot 10^{-5}$	-0.92	$5.24 \cdot 10^{-5}$	$4.86 \cdot 10^{-5}$	0.93
256	$4.87 \cdot 10^{-6}$	$-4.55 \cdot 10^{-6}$	-0.91	$1.32 \cdot 10^{-5}$	$1.22 \cdot 10^{-5}$	0.93

As we have observed already in Table 2 and Table 3, there is a change in the sign of the error estimator for the  $dk-1-0$  method once the temporal error becomes dominant. From Table 4, we infer that the change in sign is not due to behavior of the real error, but is rather due to the bad quality of the error estimator. We observe that this effect does not occur for the  $dk-1-1$  method where we obtain good error estimates. This shows the importance of sufficient damping not only for the computation of the value  $u(x_0, T)$ , which is computed equally well by both methods but even better for the error estimation. In particular, we see that the  $dk-1-0$  method is unsuitable for adaptive computation, or even error estimation alone, in the context of the Black-Scholes equation.

In the next test, we consider the quality of the spatial error estimation in the regime of dominant spatial error. By the previous results from Table 4, we only consider the  $dk-1-1$  method.

In Table 5 the results are depicted for the 1D call. The quality of the estimate is remarkably good. However, as the situation in 1D is quite different from the situation in 2D. We consider additionally the 2D put.

Table 5: Comparison of the effectivity of the spatial error estimator  $\eta_h$  for dominant spatial error under uniform refinement for the 1D call (Computation of  $u(T, x_0)$ ) using the  $dk$ -1-1 method. In the present case, we have  $\eta_k \approx 3.2 \cdot 10^{-6}$ .

N	$ \mathcal{J}(e) $	$\eta_h$	$I_{\text{eff}}$
9	$5.42 \cdot 10^{-1}$	$6.79 \cdot 10^{-1}$	1.25
17	$1.20 \cdot 10^{-1}$	$1.57 \cdot 10^{-1}$	1.30
33	$2.95 \cdot 10^{-2}$	$3.08 \cdot 10^{-2}$	1.04
65	$7.35 \cdot 10^{-3}$	$7.42 \cdot 10^{-3}$	1.01
129	$1.84 \cdot 10^{-3}$	$1.84 \cdot 10^{-3}$	1.00
257	$4.62 \cdot 10^{-4}$	$4.59 \cdot 10^{-4}$	1.00
513	$1.18 \cdot 10^{-4}$	$1.15 \cdot 10^{-4}$	1.00

We note that the effectivity index in Table 6 is only 0.68. This is due to the fact, that the recovery of the singular dual solution for the weights  $z - z_{kh}$  is not satisfactory. This can be circumvented by using only the dual residual for the purpose of error estimation, see, eg, Bangert and Rannacher (2003). When employing this trick the effectivity rises to 0.98 and is hence quite satisfactory.

Table 6: Comparison of the effectivity of the spatial error estimator  $\eta_h$  for dominant spatial error under uniform refinement for the 2d put (Computation of  $u(T, x_0)$ ) using the  $dk$ -1-1 method. For the present configuration the temporal error estimator gives  $\eta_k \approx 2.7 \cdot 10^{-6}$ .

N	$ \mathcal{J}(e) $	$\eta_h$	$I_{\text{eff}}$
81	$2.62 \cdot 10^{-1}$	$1.58 \cdot 10^{-1}$	0.60
289	$7.08 \cdot 10^{-2}$	$4.83 \cdot 10^{-2}$	0.68
1 089	$1.80 \cdot 10^{-2}$	$1.23 \cdot 10^{-2}$	0.68
4 225	$4.51 \cdot 10^{-3}$	$3.08 \cdot 10^{-3}$	0.68
16 641	$1.13 \cdot 10^{-3}$	$7.71 \cdot 10^{-4}$	0.69
66 049	$2.82 \cdot 10^{-4}$	$1.93 \cdot 10^{-4}$	0.69

## 5.2 Effect of Mesh Adaptation

We now consider the use of our error indicators to stir mesh refinement. We begin with temporal adaptation where we choose the spatial mesh fine enough to assert dominant temporal error.

In Table 7, we compare the error  $|\mathcal{J}(e)|$  as well as the effectivity of the error estimator and the mesh adaptation for the undamped Crank-Nicolson scheme, eg, the  $dk$ -0-0 method, with the results of the  $dk$ -1-0 and  $dk$ -1-1 method on a 1D spatial mesh with 16 385 cells.

Table 7: Comparison of different damping strategies onto adaptive temporal refinement on a uniform spatial mesh with 16 385 cells. The spatial error is of size  $10^{-7}$ . (Computation of  $u(T, x_0)$ )

dk-0-0			dk-1-0			dk-1-1		
$M$	$ \mathcal{J}(e) $	$I_{\text{eff}}$	$M$	$ \mathcal{J}(e) $	$I_{\text{eff}}$	$M$	$ \mathcal{J}(e) $	$I_{\text{eff}}$
4	$4.8 \cdot 10^{-1}$	1230.79	4	$1.4 \cdot 10^{-2}$	-0.89	4	$5.6 \cdot 10^{-2}$	0.93
8	$2.5 \cdot 10^{-1}$	1202.86	8	$4.2 \cdot 10^{-3}$	-0.88	8	$1.4 \cdot 10^{-2}$	0.91
16	$1.2 \cdot 10^{-1}$	1189.18	10	$1.5 \cdot 10^{-3}$	5.33	12	$2.0 \cdot 10^{-3}$	0.92
32	$6.2 \cdot 10^{-2}$	1179.15	20	$2.9 \cdot 10^{-4}$	7.32	20	$3.3 \cdot 10^{-4}$	0.91
64	$3.1 \cdot 10^{-2}$	1167.05	22	$7.4 \cdot 10^{-4}$	4.31	24	$4.5 \cdot 10^{-4}$	0.94
128	$1.6 \cdot 10^{-2}$	1148.03	24	$8.7 \cdot 10^{-4}$	4.17	40	$1.4 \cdot 10^{-4}$	0.96
256	$7.6 \cdot 10^{-3}$	1117.78	48	$2.0 \cdot 10^{-4}$	4.75	72	$3.7 \cdot 10^{-5}$	0.98
512	$3.6 \cdot 10^{-3}$	1079.07	50	$2.2 \cdot 10^{-4}$	6.14	76	$4.9 \cdot 10^{-5}$	0.94

We see from Table 7 that as expected the convergence order of the undamped Crank-Nicolson method is only of order  $\mathcal{O}(k)$  while that of the damped versions is in each case of order  $\mathcal{O}(k^2)$ . However, as already seen for global refinement, only for sufficient damping of the dual equation the error estimator is capable of estimating the error correctly. In particular, local refinement according to the error estimator does not help in obtaining better efficiency if insufficient dual damping is considered. This is in contrast to the behavior of the spatial error estimator. In Table 8 the results are shown for adaptive spatial refinement on a fine time mesh. We see that the spatial error estimate is getting better under adaptive refinement (compare Table 6) since the spatial singularity in the dual solution is then resolved appropriately.

Table 8: Effectivity index for the 2D put on adaptively refined spatial meshes on an equidistant temporal mesh with  $M = 300$  such that  $\eta_k \approx 3 \cdot 10^{-6}$  with the dk-1-1 method. (Computation of  $u(T, x_0)$ )

N	$ \mathcal{J}(e) $	$\eta_h$	$I_{\text{eff}}$
81	$2.62 \cdot 10^{-1}$	$1.58 \cdot 10^{-1}$	0.60
137	$7.13 \cdot 10^{-2}$	$4.59 \cdot 10^{-2}$	0.64
207	$2.22 \cdot 10^{-2}$	$1.10 \cdot 10^{-2}$	0.50
523	$6.07 \cdot 10^{-3}$	$3.99 \cdot 10^{-3}$	0.66
1157	$2.06 \cdot 10^{-3}$	$1.41 \cdot 10^{-3}$	0.69
2503	$7.60 \cdot 10^{-4}$	$5.99 \cdot 10^{-4}$	0.79
4991	$3.16 \cdot 10^{-4}$	$2.63 \cdot 10^{-4}$	0.84
10255	$1.41 \cdot 10^{-4}$	$1.23 \cdot 10^{-4}$	0.89

### 5.3 Simultaneous Mesh Adaption

Finally, we consider the behavior of the adaptive Algorithm 1. We begin with the point evaluation of the 1D call in Table 9. In Table 10, the results for a point evaluation of the two dimensional put are depicted.

The 1D computations were started on a spatial mesh with  $N = 9$  and a temporal mesh with  $M = 4$ . Similarly the 2D computations were started on a spatial mesh with  $N = 81$  and a temporal mesh with  $M = 4$ . For reasons of brevity, we have only displayed the results in the vicinity of an error of  $10^{-3}$  in the price.

Table 9: Comparison of simultaneous space time adaptive refinement with the  $dk-1-1$  method for the 1d call. (Computation of  $u(T, x_0)$ )

global nonadaptive refinement				local adaptive refinement			
N	M	$ \mathcal{J}(e) $	$I_{\text{eff}}$	N	M	$ \mathcal{J}(e) $	$I_{\text{eff}}$
65	32	$8.19 \cdot 10^{-3}$	1.00	28	12	$1.12 \cdot 10^{-2}$	0.94
129	64	$2.04 \cdot 10^{-3}$	0.99	48	12	$4.36 \cdot 10^{-3}$	0.96
257	128	$5.11 \cdot 10^{-4}$	0.99	79	20	$1.05 \cdot 10^{-3}$	0.98
513	256	$1.28 \cdot 10^{-4}$	0.99	113	24	$1.21 \cdot 10^{-4}$	0.83

Table 10: Comparison of simultaneous space time adaptive refinement with the  $dk-1-1$  method for the 2D put.

global nonadaptive refinement				local adaptive refinement			
N	M	$ \mathcal{J}(e) $	$I_{\text{eff}}$	N	M	$ \mathcal{J}(e) $	$I_{\text{eff}}$
1 089	16	$1.90 \cdot 10^{-2}$	0.70	207	8	$2.64 \cdot 10^{-2}$	0.56
4 225	32	$4.77 \cdot 10^{-3}$	0.70	523	12	$6.57 \cdot 10^{-3}$	0.68
16 641	64	$1.19 \cdot 10^{-3}$	0.70	1157	12	$2.56 \cdot 10^{-3}$	0.73
66 049	128	$2.95 \cdot 10^{-4}$	0.70	2515	20	$8.14 \cdot 10^{-4}$	0.80
263 169	256	$7.15 \cdot 10^{-5}$	0.73	4971	20	$3.71 \cdot 10^{-4}$	0.84

Finally, in Table 11 the results for the evaluation of the Greek Delta in  $(x_0, T)$  of the one dimensional call are depicted. Notice, that in this case, due to the even greater lack of smoothness of the functional  $\mathcal{J}$  (in comparison to a point evaluation), we employ four damping steps near  $T$ , ie, we use the  $dk-1-2$  method, for a discussion on the required number of damping steps we refer to the beginning of Section 2.2.1.

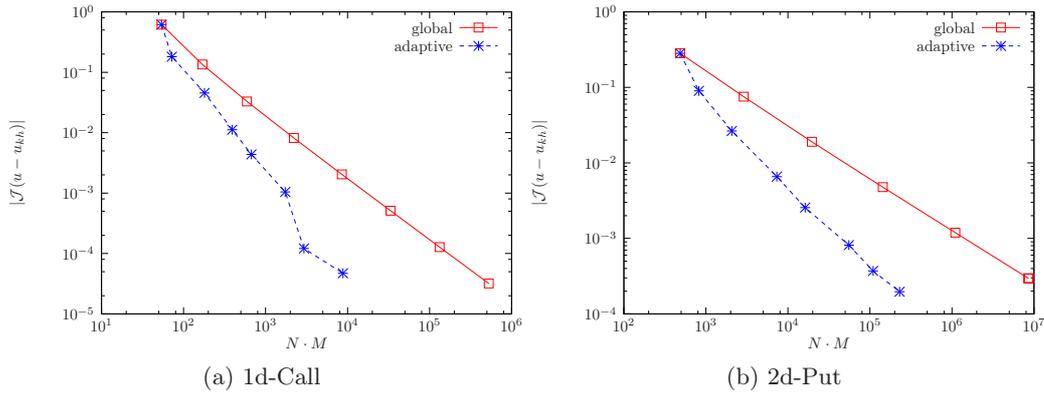


Figure 3: Comparison of mesh efficiency of global uniform refinement and adaptive refinement with the  $dk-1-1$  method for the point evaluation.

Table 11: Comparison of simultaneous space time adaptive refinement with the  $dk-1-2$  method for the point evaluation of the Greek Delta of the 1D call.

global nonadaptive refinement				local adaptive refinement			
N	M	$ \mathcal{J}(e) $	$I_{\text{eff}}$	N	M	$ \mathcal{J}(e) $	$I_{\text{eff}}$
9	4	$4.66 \cdot 10^{-2}$	0.52	14	4	$4.47 \cdot 10^{-3}$	1.40
17	8	$1.36 \cdot 10^{-2}$	1.03	21	8	$1.59 \cdot 10^{-3}$	1.51
33	16	$3.68 \cdot 10^{-3}$	1.20	28	12	$4.84 \cdot 10^{-6}$	34.05
65	32	$9.41 \cdot 10^{-4}$	1.26	33	20	$1.04 \cdot 10^{-4}$	-0.60
129	64	$2.37 \cdot 10^{-4}$	1.27	39	24	$1.62 \cdot 10^{-4}$	0.50
257	128	$5.93 \cdot 10^{-5}$	1.28	50	40	$9.50 \cdot 10^{-5}$	0.84
513	256	$1.48 \cdot 10^{-5}$	1.28	60	40	$7.01 \cdot 10^{-5}$	1.28
1025	512	$3.71 \cdot 10^{-6}$	1.28	68	40	$4.62 \cdot 10^{-5}$	1.10
2049	1024	$9.26 \cdot 10^{-7}$	1.28	97	40	$1.34 \cdot 10^{-5}$	0.98

In all tables, we can clearly see that the adaptive refinement allows for substantial savings with respect to the degrees of freedom.

Finally, we would like to comment on the gain in computation time. It is a non trivial task to do a fair numerical comparison given that either codes for uniform as well as adaptive computations can be tuned and the tuning for one of the tasks may be inappropriate for the other. Thus, we refrain from displaying numbers of computation times and instead provide a worst case complexity analysis which still yields that the adaptive procedure is by far superior.

For the moment, let us assume that we have a solver of optimal complexity for the linear systems, ie, the cost for computing the primal solution with  $M$  time-steps and  $N$  spatial degrees of freedom is proportional to  $N \cdot M$ . Further, to calculate the refinement indicators, we need one additional solution of the dual problem which in this case is as costly as the primal one and the calculation of the refinement indicators which is comparable to the evaluation of a residual. For convenience, we will assume that it is as

expensive as a complete solution. Finally, the adaptive algorithm will need less unknowns for the same accuracy, but as we can see in the Tables needs a few more solutions to get the final mesh. From what we can see in the Tables, at most twice the number of meshes is needed. Thus, in total, if we denote the amount of 'work' needed by global and adaptive refinement with  $\omega_{\text{glob}}$  and  $\omega_{\text{adap}}$ , we get

$$\omega_{\text{glob}} = N_{\text{glob}}M_{\text{glob}}, \quad \omega_{\text{adap}} = 6N_{\text{adap}}M_{\text{adap}}.$$

In Table 12, we compare the effort needed to compute the solution of the numerical examples with global and adaptive mesh refinement down to some pre selected accuracy. We see that solving the equation with global mesh refinement needs between 6 times in 1d and 28 times more work than using the DWR method.

Table 12: Comparison of the effort of solving the numerical examples with global and adaptive refinement.

	global nonadaptive refinement			local adaptive refinement			$\omega_{\text{glob}}/\omega_{\text{adap}}$
	N	M	$ \mathcal{J}(e) $	N	M	$ \mathcal{J}(e) $	
1d-Call	513	256	$1.28 \cdot 10^{-4}$	113	24	$1.21 \cdot 10^{-4}$	$\approx 8$
2d-Put	66 049	128	$2.95 \cdot 10^{-4}$	4971	20	$3.71 \cdot 10^{-4}$	$\approx 28$
1d-Greek	513	256	$1.48 \cdot 10^{-5}$	97	40	$1.34 \cdot 10^{-5}$	$\approx 6$

Moreover, if a non optimal solver is used, then the work for the solution is no longer  $N \cdot M$  but in fact higher, ie,  $N^2 \cdot M$  for a direct sparse solver. This would in turn yield a time overhead of a factor 100 to 300 for computations using a uniform refinement instead of adaptivity.

## Acknowledgment

For this work the first author has received a research student scholarship by the MAThematics Center Heidelberg (MATCH) at the University of Heidelberg. This support is gratefully acknowledged.

## References

- Y. Achdou and O. Pironneau. *Computational Methods for Option Pricing*. Frontiers in applied mathematics ; 30. Society for Industrial and Applied Mathematics (SIAM), 2005.
- M. Ainsworth and J. T. Oden. *A Posteriori Error Estimation in Finite Element Analysis*. Pure and Applied Mathematics (New York). Wiley-Interscience [John Wiley & Sons], New York, 2000.
- G. Akrivis, C. Makridakis, and R. H. Nochetto. A posteriori error estimates for the Crank-Nicolson method for parabolic equations. *Math. Comp.*, 75, 2006.

- W. Allegretto, Y. Lin, and N. Yan. A posteriori error analysis for FEM of American options. *Discrete Contin. Dyn. Syst. Ser. B*, 6(5):957–978, 2006.
- I. Babuška and T. Strouboulis. *The Finite Element Method and its Reliability*. Numerical Mathematics and Scientific Computation. The Clarendon Press Oxford University Press, New York, 2001.
- W. Bangert and R. Rannacher. *Adaptive Finite Element Methods for Differential Equations*. Lectures in Mathematics ETH Zürich. Birkhäuser, Basel, Boston, Berlin, 2003.
- W. Bangerth, R. Hartmann, and G. Kanschat. deal.II — a general-purpose object-oriented finite element library. *ACM Trans. Math. Softw.*, 33(4):24, 2007.
- R. Becker and R. Rannacher. A feed-back approach to error control in finite element methods: Basic analysis and examples. *East-West J. Numer. Math.* 4, 4:237–264, 1996.
- R. Becker and R. Rannacher. An optimal control approach to a posteriori error estimation. In A. Iserles, editor, *Acta Numerica 2001*, pages 1–102. Cambridge University Press, 2001.
- R. Becker, D. Meidner, and B. Vexler. Efficient numerical solution of parabolic optimization problems by finite element methods. *Optim. Methods Softw.*, 22(5):813–833, 2007.
- M. Berggren, R. Glowinski, and J. Lions. A computational approach to controllability issues for flow-related models. (I): Pointwise control of the viscous burgers equation. *Int. J. Comput. Fluid Dyn.*, 7(3):237–252, 1996.
- F. Black and M. Scholes. The pricing of options and corporate liabilities. *Journal of Political Economy*, 81(3):637, 1973.
- H. Blum and F.-T. Suttmeier. Weighted error estimates for finite element solutions of variational inequalities. *Computing*, 65(2):119–134, 2000.
- S. Brenner and L. Scott. *The Mathematical Theory of Finite Element Methods*. Texts in Applied Mathematics. Springer-Verlag, 2007.
- H.-J. Bungartz and M. Griebel. Sparse grids. *Acta Numer.*, 13:147–269, 2004.
- C. Carstensen and R. Verfürth. Edge residuals dominate a posteriori error estimates for low order finite element methods. *SIAM J. Numer. Anal.*, 36(5):1571–1587, 1999.
- R. Carter and M. B. Giles. Convergence analysis of Crank-Nicolson and Rannacher time-marching. *Journal of Computational Finance*, 9(4):89–112, 2006.
- Z. Chen and J. Feng. An adaptive finite element algorithm with reliable and efficient error control for linear parabolic problems. *Math. Comp.*, 73(247):1167–1193 (electronic), 2004.

- P. G. Ciarlet. *The Finite Element Method for Elliptic Problems*. North-Holland, Amsterdam, 2. edition, 1987.
- T. A. Davis. Algorithm 832: Umfpack v4.3—an unsymmetric-pattern multifrontal method. *ACM Trans. Math. Softw.*, 30(2):196–199, 2004. ISSN 0098-3500.
- D. J. Duffy. A critique of the Crank Nicolson scheme strengths and weaknesses for financial instrument pricing. *Wilmott Magazine*, 12:68–76, 2004.
- K. Eriksson and C. Johnson. Adaptive finite element methods for parabolic problems. I. A linear model problem. *SIAM J. Numer. Anal.*, 28(1):43–77, 1991.
- K. Eriksson and C. Johnson. Adaptive finite element methods for parabolic problems. II. Optimal error estimates in  $L_\infty L_2$  and  $L_\infty L_\infty$ . *SIAM J. Numer. Anal.*, 32(3):706–740, 1995.
- K. Eriksson, D. Estep, P. Hansbo, and C. Johnson. *Computational Differential Equations*. Cambridge University Press, Cambridge, 1996.
- A. Ern, S. Villeneuve, and A. Zanette. Adaptive finite element methods for local volatility european option pricing. *International Journal of Theoretical and Applied Finance*, 7(6):659–684, 2004.
- P. A. Forsyth and K. R. Vetzal. Quadratic convergence for valuing american options using a penalty method. *SIAM J. Sci. Comput.*, 23:2095–2122, 2002.
- G. Foufas and M. G. Larson. Valuing asian options using the finite element method and duality techniques. *J. Comput. Appl. Math.*, 222:144–158, 2008.
- A. Friedman. *Partial Differential Equations of Parabolic Type*. Krieger, Malabar, Florida, 1983.
- A. Griewank. Achieving logarithmic growth of temporal and spatial complexity in reverse automatic differentiation. *Optim. Methods Software*, 1(1):35–54, 1991.
- J. Hull. *Options, Futures, and other Derivatives*. Pearson, Prentice Hall, Upper Saddle River, NJ, 6. ed., internat. ed. edition, 2006.
- R. Kangro and R. Nicolaides. Far field boundary conditions for black–scholes equations. *SIAM J. Numer. Anal.*, 38(4):1357–1368, 2000.
- A. Q. M. Khaliq, D. A. Voss, and K. Kazmi. Adaptive [theta]-Methods for Pricing American Options. *J. Comput. Appl. Math.*, 222:210–227, 2008.
- Y.-K. Kwok. *Mathematical Models of Financial Derivatives*. Springer Finance. Springer-Verlag Singapore, Singapore, 1998.
- J. Lang. *Adaptive Multilevel Solution of Nonlinear Parabolic PDE Systems*, volume 16 of *Lecture Notes in Computational Science and Engineering*. Springer-Verlag, Berlin, 2001. Theory, algorithm, and applications.

- P. Lötstedt, J. Persson, L. von Sydow, and J. Tysk. Space-time adaptive finite difference method for european multi-asset options. *Comput. Math. Appl.*, 53:1159–1180, 2007.
- M. Luskin and R. Rannacher. On the smoothing property of the crank-nicolson scheme. *Applicable Analysis: An International Journal*, 14(2):117–135, 1982.
- D. Meidner and B. Vexler. Adaptive space-time finite element methods for parabolic optimization problems. *SIAM J. Control Optim.*, 46(1):116–142, 2007.
- R. C. Merton. Theory of rational option pricing. *The Bell Journal of Economics and Management Science*, 4(1):141–183, 1973.
- K.-S. Moon, R. H. Nochetto, T. von Petersdorff, and C.-s. Zhang. A posteriori error analysis for parabolic variational inequalities. *M2AN Math. Model. Numer. Anal.*, 41(3):485–511, 2007.
- R. H. Nochetto, T. von Petersdorff, and C.-S. Zhang. A posteriori error analysis for a class of integral equations and variational inequalities. *Numer. Math.*, 116(3):519–552, 2010.
- J. Persson and L. von Sydow. Pricing european multi-asset options using a space-time adaptive FD-method. *Computing and Visualization in Science*, 10(4):173–183, 2007.
- M. Picasso. Adaptive finite elements for a linear parabolic problem. *Comput. Methods Appl. Mech. Engrg.*, 167(3-4):223–237, 1998.
- O. Pironneau and Y. Achdou. Partial differential equations for option pricing. volume 15 of *Handbook of Numerical Analysis*, pages 369, 371, 373 – 375, 377–401, 403–495. Elsevier, 2009.
- O. Pironneau and F. Hecht. Mesh adaption for the Black and Scholes equations. *East-West J. Numer. Math.*, 8(1):25–35, 2000.
- D. M. Pooley, K. R. Vetzal, and P. A. Forsyth. Remedies for non-smooth payoffs in option pricing. *J. Computational Finance*, 6:25–40, 2003.
- R. Rannacher. Finite element solution of diffusion problems with irregular data. *Numerische Mathematik*, 43(2):309–327, 1984.
- C. Reisinger. *Numerische Methoden für hochdimensionale parabolische Gleichungen am Beispiel von Optionspreisaufgaben*. PhD thesis, Mathematisch-Naturwissenschaftliche Gesamtfakultät, Universität Heidelberg, 2004.
- T. Richter. *Parallel Multigrid Method for Adaptive Finite Elements with Application to 3D Flow Problems*. PhD thesis, Mathematisch-Naturwissenschaftliche Gesamtfakultät, Universität Heidelberg, 2005.

- E. W. Sachs and M. Schu. Gradient computation for model calibration with pointwise observations. In K. Kunisch, editor, *International Workshop on Control and Optimization of PDEs*, International Series of Numerical Mathematics. Birkhäuser, 2012. to appear.
- M. Schmich and B. Vexler. Adaptivity with dynamic meshes for space-time finite element discretizations of parabolic equations. *SIAM J. Sci. Comput.*, 30(1):369–393, 2008.
- F.-T. Suttmeier. *Numerical Solution of Variational Inequalities by Adaptive Finite Elements*. Vieweg+Teubner, Wiesbaden, 2008.
- R. Verfürth. *A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques*. Wiley/Teubner, New York-Stuttgart, 1996.
- J. Wloka. *Partial Differential Equations*. Cambridge University Press, Cambridge, 1987.