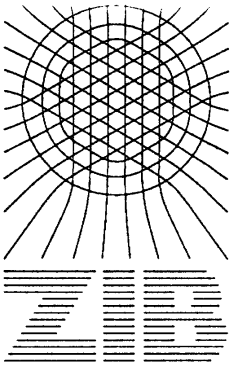
Konrad-Zuse-Zentrum für Informationstechnik Berlin



Folkmar A. Bornemann

An Adaptive Multilevel Approach to Parabolic Equations in Two Space Dimensions

Technical Report TR 91-7 (Juni 1991)

Herausgegeben vom
Konrad-Zuse-Zentrum für Informationstechnik Berlin
Heilbronner Str. 10
1000 Berlin 31
Verantwortlich: Dr. Klaus André
Umschlagsatz und Druck: Rabe KG Buch-und Offsetdruck Berlin

ISSN 0933-789X

An Adaptive Multilevel Approach to Parabolic Equations in Two Space Dimensions

FOLKMAR A. BORNEMANN

Konrad–Zuse–Zentrum für Informationstechnik Berlin, Heilbronner Strasse 10, D-1000 Berlin 31, Federal Republic of Germany

June 12, 1991

ABSTRACT

A new adaptive multilevel approach for linear parabolic partial differential equations is presented, which is able to handle complicated space geometries, discontinuous coefficients, inconsistent initial data. Discretization in time first (Rothe's method) with order and stepsize control is perturbed by an adaptive finite element discretization of the elliptic subproblems, whose errors are controlled independently. Thus the high standards of solving adaptively ordinary differential equations and elliptic boundary value problems are combined. A theory of time discretization in Hilbert space is developed which yields to an optimal variable order method based on a multiplicative error correction. The problem of an efficient solution of the singularly perturbed elliptic subproblems and the problem of error estimation for them can be uniquely solved within the framework of preconditioning. A multilevel nodal basis preconditioner is derived, which allows the use of highly nonuniform triangulations. Implementation issues are discussed in detail. Numerous numerical examples in one and two space dimensions clearly show the significant perspectives opened by the new algorithmic approach. Finally an application of the method is given in the area of hyperthermia, a recent clinical method for cancer therapy.

Für B_S

"Ein guter Engel wird immer nötig sein, was immer du tust."

L. Wittgenstein,
Bemerkungen über die Grundlagen der Mathematik, VII.16,
Suhrkamp, Frankfurt a. M., 1984

Contents

INTRODUCTION

In the presence of complicated space geometries, discontinuous coefficients, inconsistent initial data etc., the numerical solution of parabolic problems in two space dimensions requires a *sophisticated* reduction of the computational amount of work. This reduction will be even more important in three space dimensions, for which it would be the only hope to break through the complexity barrier of many important problems of the natural sciences and technology. A nowadays increasingly important concept of such an amount of work reduction is *adaptivity*, i.e., the automatic choice of the degrees of freedom, such as the automatic distribution of nodes in a triangulation or the local order of a discretization. If we compute a *family* of approximations to an infinite dimensional problem with different local discretization parameters or orders instead of a single approximation, we speak of *multilevel* methods. Such a computation of simultaneous approximations allows to construct effective error estimates, which support the adaptation control. Moreover the construction of fast iterative solvers for arising linear systems of very high dimension becomes possible by multilevel techniques.

In the field of ordinary differential equations a high standard of adaptive multilevel algorithms has been reached by the state-of-the-art solvers with order *and* stepsize control, e.g., extrapolation methods, cf. [24]. For stationary scalar elliptic boundary value problems a similar standard has been obtained in 2D by the adaptive finite element methods with a multilevel (multigrid) iterative solution process, cf. the work of BANK, YSERENTANT, DEUFLHARD and their collaborators [5, 6, 7, 8, 9, 10, 11, 12, 25, 33]. In the opinion of the author this thesis will present an approach of a comparable standard for linear scalar selfadjoint parabolic problems in 1D and 2D. The restriction to parabolic problems is understood as a first step towards more general time dependent partial differential equations.

A widespread method for the adaptive solution of parabolic problems is the *method of lines*. Since there is in general no space mesh, which is a good and efficient one for all time layers, the space mesh has to be updated (regridded) appropriately from time to time. A rather advanced approach of *statical regridding* is due to BIETERMAN/BABUŠKA [13, 14, 15]: At *fixed* time–points an error estimate for the whole parabolic problem decides where to regrid. However, this error estimator is given for the 1D case only and an automatic choice of the regridding times is missing. The *moving finite element* variant of the method of lines due to MILLER/MILLER [36, 37] (*dynamical regridding*) is restricted to a fixed number of grid points and moreover to the 1D case for *geometrical reasons*; for the 2D case inherent difficulties and drawbacks

occur, cf. [57], which make this approach likely to be not feasible in three space dimensions. Still quite common is the use of space–time elements, cf. the work of FLAHERTY, JOHNSON and their collaborators [2, 22, 27, 35], which increase the geometrical complexity by one dimension.

However, in this thesis we favor an approach which strictly separates time and space — *time is not just another dimension of space.* This fact is backed by the semigroup solution of parabolic equations, which leads to the approach we suggested for the first time in [16]:

- The parabolic initial boundary value problem can be considered as an abstract Cauchy problem in an appropriate function space.
- A variable–step variable–order discretization in time applied in that function space gives rise to an approximation of the known standard for ordinary differential equations.
- Discretization of the elliptic subproblems is considered as a perturbation, which can be controlled independently of the time discretization.

This approach separates time and space in exactly the same way as semigroup theory does and glues them together just as semigroup theory does — thus making a combination of the standards from ordinary differential equations and elliptic boundary values problems possible and, equally important, *natural.* Time and space discretization have — besides their perturbation character — no influence on each other.

In his former work [16], the author suggested an extrapolated implicit Euler scheme for the construction of the variable–step variable–order method in function space, which turned out to be a good choice for a 1D implementation only.

This thesis now exploits the full advantage of our approach in the 2D case by the construction of a variable order discretization in time with an optimal amount of work. Moreover, the restriction to the 2D case is mainly due to reasons of programming and data structures, it nowhere seriously enters into the developed theory which easily extends — if not already independent of the space dimension — to the 3D case.

In order to apply the proposed approach to actual 2D problems, it was necessary to construct an adaptive finite element solver for the arising singularly perturbed elliptic problems. The singular perturbation results from the time step of the discretization in time; standard solvers run into difficulties for small time steps, which occur in transient phases. Two devices had to be re-constructed:

- Error estimator
- Linear solver

Both devices have to behave well — uniform in the time step. Using a multilevel iteration as linear solver rises the question of a *proper preconditioner*. As it turns out this preconditioner is the key to the error estimator as well.

Because of its use of orthogonal projections a recently presented preconditioner for elliptic equations due to BRAMBLE/PASCIAK/XU [19] — extended to the case of highly nonuniform meshes by YSERENTANT [56] — is ideally suited as conceptual base for our purposes. Moreover this concept is not restricted to certain space dimensions, like hierarchical basis preconditioners, but is easily extended to higher dimensions. For this type of preconditioner the question of an effective implementation in the presence of highly nonuniform meshes had to be studied for the first time. We developed a kind of algebraic description of triangulations and nodal bases functions which permits to handle and prove implementation details easily.

Numerous numerical experiments on model problems have proved the algorithm to be very robust, reliable and efficient in 1D and 2D. However, model problems tend to isolate the different kind of difficulties or to test for difficulties other than those arising in real applications. In order to prove (mainly in view of possible future extensions) the applicability of our approach to real life problems we did some computations on the Bio–Heat–Transfer equation. This equation plays a prominent role in planning *hyperthermia*, a recent clinical method for the treatment of malignancies (cancer), which at this time is in an experimental status. The numerical solution of this equation shows the following typical difficulties *in combination*:

- nasty complicated problem geometry: re–entrant boundary corners, a lot of different inner regions, many nodal points in the initial coarse triangulation, etc.
- discontinuous coefficients due to different regions
- inconsistent initial data.

Surely a fast and reliable solution on a workstation is important for an experimental planning phase which studies the involved model parameters. Needless to say, that in an actual clinical treatment with *on line* control computation, a fast and reliable solution, which permits to react in reasonable time, would be of vital interest. In computations starting from 2D computer tomography cross sections we have obtained — within clinical tolerances — a fast solution, which gives enough time to react interactively.

Outline of the paper

The paper is divided into three major parts: Time, Space and Results.

The Time Part. In this part we review in Sections 1 and 3 our approach as introduced in [16, 17]. In Section 2 we present our new optimal variable order time discretization, which is based on a multiplicative error correction.

This first part is based on material already published by the author, [17, 18]; however some changes should be indicated:

- Theorem 1.2, which is [18, Theorem 1.4], has been given a shorter proof, which is now independent of the more general theoretical setting of [17].
- Section 2.1 is an extension of [18, Section 2.1] and has been totally rewritten. It consists now of a comprehensive account on the discretizations based on the multiplicative error correction. Lemma 2.1 on the Laguerre polynomials, which we only conjectured in [18], is completely proven now.

The Space Part. This part contains entirely new material and is devoted to the solution of the singularly perturbed elliptic subproblems in 2D.

Section 4 introduces the notation and the formalism to handle highly nonuniform triangulations and finite element spaces. Also a first discussion of preconditioning and corresponding iterative solution may be found.

Section 5 discusses on a rather abstract level error estimation for general Galerkin methods and explains why preconditioning is the key to an effective error estimation.

Section 6 is devoted to the construction of a preconditioner on the base of the elliptic preconditioner of BRAMBLE/PASCIAK/XU [19]. We first deal with the case of an elliptic operator with no Helmholtz term and natural boundary conditions outside the Dirichlet boundary piece. The thus developed preconditioner, which gives a smooth transient from diagonal preconditioning of the mass matrix to a preconditioner of the stiffness matrix, is thereafter extended to the presence of a Helmholtz term and general Cauchy boundary conditions. For the need of error estimation we present the preconditioning of quadratic elements. This leads us to the discussion of the error estimation, where the abstract considerations of Section 5 will find their counterpart. We close the section by a detailed and careful derivation of the actual implementation of our preconditioner. This implementation follows naturally from the mathematical description of the preconditioner with

the help of the formalism which describes triangulations and finite element spaces. We obtain a result, which states that the complexity of a preconditioner multiplication is the same as the multiplication with a sparse matrix of constant bandwidth.

The Result Part. First in Section 7 some algorithmic details for the 1D and 2D case are given. They include such important issues as the optimal choice of certain parameters, the discussion of possible orders for the time discretization in dependence of the imposed accuracy, a stop criterion for the time error iteration, a stabilization of orthogonal projections and the direct solver on the coarsest triangulation in 2D. The latter becomes important when the starting grid already consists of "many" nodes.

Section 8 contains numerical computations on model problems in 1D and 2D. The 1D examples have already been published in [18]. These model problem computations show a lot of carefully chosen details, which back the developed theory.

Section 9 finally gives a real life application of our method in the cancer therapy method *hyperthermia*. This shows the full applicability of our method to the given problem class.

ACKNOWLEDGMENTS

I express my deep gratitude towards my mentor P. Deuflhard, whose way of intellectual approach to numerical problems had been an influence, which cannot be overestimated. He also taught me to keep important applications in mind and had thought of hyperthermia at a time when my method was a mere vision.

Many thanks to the staff of the Konrad–Zuse–Zentrum Berlin (ZIB) who provided a very comfortable and productive atmosphere.

Special thanks to my colleagues M. Wulkow and R. Kornhuber for numerous discussions and proof reading of a primary version of the manuscript. From time to time their encouragement was very healthy.

Many thanks to C. Lubich for discussions on the matter of time discretization.

Finally I like to announce that the computations of Section 9, *hyperthermia*, were done in cooperation with the Klinikum Rudolf Virchow, Freie Universität Berlin. I thank J. Nadobny for making available the computer tomography data, for computing the E–field and for enduring my impatience.

I. Multilevel Discretization in Time

1. Preliminary Draft of the Algorithm

1.1. The Problem

We are concerned with linear scalar selfadjoint parabolic initial–boundary value problems:

Given a domain $\Omega \subset \mathbb{R}^d$ with Lipschitz boundary $\partial\Omega = \Gamma_D \dot{\cup} \Gamma_C$, a time $T_{\text{fin}} > 0$, solve

$$
\begin{aligned}
&\text{i)} \quad \phi(x)\frac{\partial u(t,x)}{\partial t} + A(x,\partial)u(t,x) = f(t,x), \qquad x \in \Omega, \;\; t \in]0, T_{\text{fin}}];\\
&\text{ii)} \quad u(t,\cdot)|_{\Gamma_D} = g(t,\cdot), \qquad t \in]0, T_{\text{fin}}];\\
&\text{iii)} \quad C(x,\partial)u(t,x)|_{x\in\Gamma_C} = \xi(t,x)|_{x\in\Gamma_C}, \qquad t \in]0, T_{\text{fin}}];\\
&\text{iv)} \quad u(0,\cdot) = u_0.
\end{aligned}
\tag{1.1}
$$

Here $A(x,\partial)$ denotes a *formally selfadjoint* elliptic operator of second order, which has a principal part in divergence form:

$$
A(x,\partial)u(x) = -\sum_{i,k=1}^{d} \partial_k \left(a_{ik}(x)\partial_i u(x)\right) + q(x)u(x),
$$

where $a_{ik} = a_{ki}$. Moreover $C(x,\partial)$ denotes the corresponding Cauchy boundary operator

$$
C(x,\partial)u(x) = -\sum_{i,k=1}^{d} n_k(x)a_{ik}(x)\partial_i u(x) - \zeta(x)u(x),
$$

where $n = (n_1, \ldots, n_d)^T$ is the outer unit normal on $\partial\Omega$.

Notation. The norms of the Sobolev spaces $H^s(\Omega)$ will be denoted by $\|\cdot\|_s$, their seminorms by $|\cdot|_s$, the norms of the spaces $W^{s,p}(\Omega)$ by $\|\cdot\|_{s,p}$ and the inner product of $L^2(\Omega)$ will be denoted by $(\cdot,\cdot)$. For a function $\psi \in L^\infty(\Omega) = W^{0,\infty}(\Omega)$ with $\psi \geq 0$ a.e. we abbreviate

$$
\psi_{\max} = \|\psi\|_{0,\infty} \text{ and } \psi_{\min} = 1/\|1/\psi\|_{0,\infty}.
$$

We make the following *assumptions*:

1. Ω has Lipschitz boundary, i.e., $\Omega \in C^{0,1}$. Furthermore Γ_D is a closed subset of $\partial\Omega$.

2. $\phi, q, \zeta, a_{ik} \in L^\infty(\Omega)$.

3. $\phi, q, \zeta \geq 0$ a.e., moreover $\phi_{\min} > 0$.

4. $A(x, \partial)$ is strongly elliptic, such that there are constants $0 < \delta \leq \Delta$

$$\delta \sum_{i=1}^{d} \xi_i^2 \leq \sum_{i,k=1}^{d} a_{ik}(x)\xi_i\xi_k \leq \Delta \sum_{i=1}^{d} \xi_i^2$$

for all $\xi \in \mathbb{R}^d$ and almost all $x \in \Omega$.

5. $f(t, \cdot), u_0 \in L^2(\Omega)$ for all $t \in [0, T_{\text{fin}}]$.

6. $g, \xi \in C^1\left([0, T_{\text{fin}}], H^{1/2}(\partial\Omega)\right)$.

By means of assumption 6 and the known properties of the trace operator, we can take by a simple transformation the case that

$$g, \xi \equiv 0.$$

For ease of representation we will assume mostly in this paper

- the temporally homogeneous case $f_t \equiv 0$
- $\phi \equiv 1$.

Important. The extension to the case $f_t \not\equiv 0$ will be discussed in Section 2.2.3 and the extension to the case $\phi \not\equiv 1$ in Section 9.2.2.

We introduce the space of weak solutions

$$H_D^1(\Omega) = \left\{u \in H^1(\Omega) \mid u|_{\Gamma_D} = 0\right\},$$

(the restriction is understood in the sense of traces), which is — due to assumption 1 — a closed subspace of $H^1(\Omega)$, cf. [21, VII§2.2.1], and therefore a Hilbert space. We now consider the following continuous symmetric bilinear form $a(\cdot, \cdot)$ on $H_D^1(\Omega) \times H_D^1(\Omega)$:

$$a(u, v) = \sum_{i,k=1}^{d} \int_\Omega a_{ik}\partial_i u \partial_k v \, dx + \int_\Omega q\, uv \, dx + \int_{\Gamma_C} \zeta\, uv \, d\sigma,$$

$u, v \in H_D^1(\Omega)$. Thus, both the operator $A(x, \partial)$ and the boundary conditions are incorporated in this form. For the following the property of $H_D^1(\Omega)$-*ellipticity* of the form $a(\cdot, \cdot)$ will be important: There is a constant $c_1 > 0$ such that

$$a(u, u) \geq c_1 \|u\|_1^2 \quad \text{for all } u \in H_D^1(\Omega).$$

The next Lemma will give some conditions for the $H_D^1(\Omega)$-ellipticity of the form $a(\cdot, \cdot)$.

LEMMA 1.1. *Each of the following cases guarantees the $H^1_D(\Omega)$–ellipticity of the form $a(\cdot,\cdot)$:*

i) The Cauchy boundary piece is empty, $\Gamma_C = \emptyset$. In this case we estimate for $u \in H^1_D(\Omega)$

$$a(u,u) \geq \frac{\delta}{1 + d_\Omega^2/2} \|u\|_1^2$$

and

$$a(u,u) \geq \frac{2\delta}{d_\Omega^2} \|u\|_0^2.$$

Here d_Ω denotes the band width of a strip containing Ω.

ii) $q_{\min} > 0$. In this case we estimate for $u \in H^1_D(\Omega)$

$$a(u,u) \geq \min(\delta, q_{\min}) \|u\|_1^2$$

and

$$a(u,u) \geq q_{\min} \|u\|_0^2.$$

iii) $\operatorname{mes}(\Gamma_D) > 0$.

iv) $\operatorname{mes}(\Gamma_C) > 0$ *and* $\zeta_{\min} > 0$.

Proof. By assumption 4 we can estimate

$$(*) \qquad a(u,u) \geq \delta |u|_1^2 + q_{\min} \|u\|_0^2 + \zeta_{\min} \int_{\Gamma_C} u^2 d\sigma.$$

i) If $\Gamma_C = \emptyset$ we have that $H^1_D(\Omega) = H^1_0(\Omega)$. Hence the assertion follows from (*) and from the $H^1_0(\Omega)$ Poincaré inequality

$$\|u\|_0^2 \leq \frac{d_\Omega^2}{2} |u|_1^2 \qquad \text{for } u \in H^1_0(\Omega),$$

cf. [21, IV§7, Prop. 1].

ii) Follows trivially from (*).

iii) [21, IV§7, Remark 4] states the equivalence of $|\cdot|_1$ and $\|\cdot\|_1$ on $H^1_D(\Omega)$ for $\operatorname{mes}(\Gamma_D) > 0$. Thus (*) proves the assertion.

iv) [48, Theorem 28.5] states, that if $\operatorname{mes}(\Gamma_C) > 0$ the norms $\|\cdot\|_1$ and $\|\cdot\|$, which is defined on $H^1_D(\Omega)$ by

$$\|u\|^2 = |u|_1^2 + \int_{\Gamma_C} u^2 d\sigma,$$

$u \in H^1_D(\Omega)$, are equivalent. Again the assertion follows from (*). ∎

The next Theorem mainly serve the purpose to provide a concept of *solution* of the parabolic problem, which justifies our approach *without additional regularity assumptions.*

THEOREM 1.1. *Suppose that the bilinear form $a(\cdot,\cdot)$ is $H_D^1(\Omega)$-elliptic, then the following holds:*

a) *There is exactly one positive selfadjoint operator*

$$A : D_A \subset L^2(\Omega) \to L^2(\Omega)$$

satisfying

i) $D_A \subset H_D^1(\Omega)$,

ii) $a(u,v) = (Au,v)$ for all $u \in D_A$, $v \in H_D^1(\Omega)$.

Furthermore we have:

b) *The domain of definition D_A is dense in $H_D^1(\Omega)$ with respect to the Hilbert space topology of $H_D^1(\Omega)$.*

c) *For every $f \in L^2(\Omega)$ the solution $u \in H_D^1(\Omega)$ of the variational problem*

$$a(u,v) = (f,v) \quad \textit{for all } v \in H_D^1(\Omega)$$

exists and satisfies in addition:

$$u \in D_A, \quad Au = f.$$

d) *The square root $A^{1/2}$ of A exists with $D_{A^{1/2}} = H_D^1(\Omega)$ and satisfies*

$$a(u,v) = \left(A^{1/2}u, A^{1/2}v\right) \quad \textit{for all } u,v \in H_D^1(\Omega).$$

Proof. The assertions a) & b) are essentially the Friedrichs representation theorem of semibounded symmetric bilinear forms in Hilbert space, consult e.g., KATO [32, pp. 322f.]. The solution $u \in H_D^1(\Omega)$ of the variational problem exists by the Lax–Milgram Lemma and the rest of assertion c) holds again by the Friedrichs representation theorem. For assertion d) consult e.g., KATO [32, pp. 331f.]. ∎

REMARK 1.1. Let $f \in L^2(\Omega)$. By means of the above theorem we observe that the *weak* solution u of the elliptic boundary–value problem

$$\begin{aligned} &\text{i)} \quad A(x,\partial)u(x) = f(x), \qquad x \in \Omega, \\ &\text{ii)} \quad u|_{\Gamma_D} = 0, \\ &\text{iii)} \quad C(x,\partial)u(x)|_{x\in\Gamma_C} = 0, \end{aligned} \tag{1.2}$$

exists and is given as

$$u = A^{-1} f \in D_A \subset H_D^1(\Omega).$$

Therefore we call A the *weak representation* of the differential operator $A(x, \partial)$ imposed with the boundary conditions.

Since the weak representation operator A is positive selfadjoint the fractional powers A^α, $\alpha \geq 0$, exist and the corresponding domains of definition

$$\dot{H}^{2\alpha} = D_{A^\alpha}$$

equipped with the inner product

$$(u, v)_{\dot{H}^{2\alpha}} = (A^\alpha u, A^\alpha v) \qquad \text{for all } u, v \in \dot{H}^{2\alpha},$$

define a scale of Hilbert spaces for which the embeddings

$$\dot{H}^\alpha \hookrightarrow \dot{H}^\beta, \qquad \alpha > \beta,$$

are continuous. Hence Theorem 1.1 states that

$$D_A = \dot{H}^2 \hookrightarrow \dot{H}^1 = D_{A^{1/2}} = H_D^1(\Omega).$$

In some sense the space $\dot{H}^2$ fully describes the *regularity* of weak solutions of the problem (1.2) since

$$\|u\|_{\dot{H}^2} = \|f\|_0.$$

The term of $H^{1+s}(\Omega)$–regularity, $s \geq 0$, may now be expressed as the existence of a continuous embedding

$$\dot{H}^2 \hookrightarrow H^{1+s}(\Omega) \cap H_D^1(\Omega).$$

EXAMPLE 1.1. By making the weak assumption

$$a_{ik} \in C^{0,t}(\bar{\Omega}) \qquad \text{for some } 0 < t \leq 1,$$

we gain the following regularity result due to NEČAS [38] for the case $\Gamma_C = \emptyset$:

$$\dot{H}^2 \hookrightarrow H^{1+s}(\Omega) \cap H_0^1(\Omega) \qquad \text{for all } 0 \leq s < \min(t, \frac{1}{2}).$$

Imposing in addition

$$\Omega \text{ is convex}, \quad t = 1,$$

yields full regularity

$$\dot{H}^2 \hookrightarrow H^2(\Omega) \cap H_0^1(\Omega),$$

a result due to KADLEC [31].

With the help of the weak representation operator A we may restate our parabolic problem (1.1) as the following *abstract Cauchy problem* in $L^2(\Omega)$:

$$\text{(1.3)} \qquad \begin{aligned} &\text{i)} \quad u' + Au = f, \\ &\text{ii)} \quad u(0) = u_0. \end{aligned}$$

If we denote the *holomorphic semigroup* [29, 42] of contractions generated by the negative selfadjoint operator $(-A)$ as

$$\mathcal{U}(t) = \exp(-tA),$$

the solution $u \in C^\infty(]0, T_{\text{fin}}], \dot{H}^2)$ of (1.3) is given by

$$\begin{aligned} &\text{i)} \quad u(t) = [w - \mathcal{U}(t)w] + \mathcal{U}(t)u_0, \qquad \text{where} \\ &\text{ii)} \quad w = A^{-1}f \in \dot{H}^2. \end{aligned}$$

Exactly *this* solution will be approximated by our algorithm.

1.2. Semidiscretization in Time

As mentioned in the introduction and discussed in [17], the initial–value character of the abstract Cauchy problem requires the discretization in time *first*, which is often called *Rothe's method* in the literature, cf. [30, 39, 46]. The principle of a variable–step, variable–order discretization in time will be explained first assuming that the spatial elliptic subproblems can be solved exactly.

We consider *linear* single step methods of the form

$$u_{j+1} = \Phi(u_j, \tau), \qquad j = 0, 1, \ldots .$$

Applied to the scalar differential equation

$$y' = -zy$$

they give rise to *rational* approximations $r_\Phi(z) = \Phi(1, 1)$ to $\exp(-z)$. The rational approximation r_Φ is said to be of *order* $p \geq 1$ whenever

$$r_\Phi(z) = e^{-z} + \mathcal{O}(z^{p+1}) \qquad \text{for } z \to 0,$$

and to be of *exact order* $p \geq 1$ if r_Φ is of order p but not of order $p + 1$.

In order to be able to apply the single step method to the abstract Cauchy problem (1.3) we have to make demands on the stability. The approximation r_Φ is said to be

- *strongly A_0-stable* if

$$|r_\Phi(z)| < 1 \quad \text{for } z > 0, \text{ and } |r_\Phi(\infty)| < 1;$$

- *strongly A_ϑ-stable*, $0 < \vartheta \le \pi/2$, if

$$|r_\Phi(z)| \le 1 \quad \text{for } z \in \Sigma_\vartheta, \text{ and } |r_\Phi(\infty)| < 1;$$

- *L_ϑ-stable*, $0 \le \vartheta \le \pi/2$, if it is strongly A_ϑ-stable with

$$r_\Phi(\infty) = 0.$$

Here Σ_ϑ denotes

$$\Sigma_\vartheta = \{z \in \mathbb{C} \mid |\arg z| \le \vartheta\}.$$

Note the implications: L_ϑ-stable $\Rightarrow$ strongly $A_{\tilde\vartheta}$-stable $\Rightarrow$ strongly A_0-stable for $0 \le \tilde\vartheta \le \vartheta$.

For proving the applicability of a corresponding single step method to the abstract Cauchy problem (1.3) we need the following special case of a lemma due to LUBICH [34, Lemma 6.3], whose proof may be found in [17, Lemma 2.4] as well. It is stated there for A_ϑ-stable approximations with $\vartheta > 0$ only, but is valid with the same proof for strongly A_0-stable approximations.

LEMMA 1.2. (LUBICH [34]). *Let $r(z)$ be a strongly A_0-stable approximation of order p to $\exp(-z)$. There is an $\eta > 0$ such that for $0 \le z \le \eta$ the following asymptotic expansion holds*

$$r(z)^n = e^{-nz}\left[1 + P_p(nz)z^p + \ldots + P_N(nz)z^N\right] + R_{N+1}(n,z).$$

Here the P_j are polynomials of degree $j-p+1$, $P_j(0) = 0$ and the remainder satisfies

$$|R_{N+1}(n,z)| \le Ce^{-nz/2}z^{N+1}.$$

The applicability to the abstract Cauchy problem can now be stated.

THEOREM 1.2. *Given a strongly A_0-stable rational approximation $r(z)$ to $\exp(-z)$ of order p, the single step method*

$$\Phi_\tau(u,\tau) = r(\tau A)u + \Big(I - r(\tau A)\Big)A^{-1}f \tag{1.4}$$

is well defined for $\tau > 0$ and the sequence $u_{n+1} = \Phi(u_n,\tau)$, $n = 1,2,\ldots$, approximates the solution of the abstract Cauchy problem (1.3) at $t_n = n\tau$ with an error of

$$\|u_n - u(t_n)\|_0 \le C\tau^p t_n^{\min(1,\alpha-p)}\|u_0\|_{\dot H^{2\alpha}}. \tag{1.5}$$

Proof. Since we study the error due to discretization in *time* by a *linear* single–step method, it is enough to consider the homogeneous case $f = 0$ only. Simply subtract the stationary solution $v \in \dot{H}^2$ of $Av = f$ and observe that this commutes by linearity with the discretization in time. Due to the continuity of the embeddings $\dot{H}^\alpha \hookrightarrow \dot{H}^\beta$, $\alpha > \beta$, we can restrict ourselves to the case $\alpha \leq p+1$.

Now put

$$u_n - u(t_n) = e_p(t_n)\tau^p + e_{p+1}(t_n;\tau)$$

with $e_p(t) = P_p(tA)A^p u(t)$, where $P_p(z) = \pi_p z$ is the linear polynomial of Lemma 1.2. For $u_0 \in \dot{H}^{2\alpha}$ we get

$$\begin{aligned} \|e_p(t)\|_0 &\leq |\pi_p|\, t\, \|A^{p+1-\alpha}\mathcal{U}(t)A^\alpha u_0\|_0 \\ &\leq C_0 t^{\alpha-p}\|u_0\|_{\dot{H}^{2\alpha}}, \end{aligned}$$

since $\alpha \leq p+1$. Putting $\varphi(z) = R_{p+1}(n,z)z^{-\alpha}$, R_{p+1} as in Lemma 1.2, we obtain

$$\|e_{p+1}(t_n;\tau)\|_0 \leq \tau^\alpha \|\varphi(\tau A)\|_{\mathcal{L}(L^2,L^2)}\|u_0\|_{\dot{H}^{2\alpha}}.$$

The spectral theorem for selfadjoint operators yields

$$\|\varphi(\tau A)\|_{\mathcal{L}(L^2,L^2)} \leq \sup_{z\geq 0}|\varphi(z)|$$

since A is positive by Theorem 1.1. By Lemma 1.2 we can estimate for some $\eta > 0$

$$|\varphi(z)| \leq C_1 z^{p+1-\alpha}e^{-nz/2} \leq C_2 n^{\alpha-(p+1)}, \qquad 0 \leq z \leq \eta.$$

The strong A_0–stability of r yields the existence of some $r_0 < 1$, such that

$$\sup_{z\geq\eta}|r(z)| \leq r_0.$$

Hence we can estimate for $z \geq \eta$

$$\begin{aligned} |\varphi(z)| &\leq z^{-\alpha}\left(|r^n(z)| + e^{-nz}(1+|\pi_p|nz^{p+1})\right) \\ &\leq \eta^{-\alpha}(r_0^n + e^{-n\eta}) + C_3 n^{\alpha-p}e^{-n\eta/2} \\ &\leq C_4\varrho^n \text{ for some } 0 < \varrho < 1 \\ &\leq C_5 n^{\alpha-(p+1)}. \end{aligned}$$

Summarizing we have established the estimate

$$\|\varphi(\tau A)\|_{\mathcal{L}(L^2,L^2)} \leq C_6 n^{\alpha-(p+1)},$$

which implies

$$\|e_{p+1}(t_n;\tau)\|_0 \le C_6\tau^{p+1}t_n^{\alpha-(p+1)}\|u_0\|_{\dot{H}^{2\alpha}}$$

and hence

$$\begin{aligned}\|u_n - u(t_n)\|_0 &\le \|e_p(t_n)\|_0\tau^p + \|e_{p+1}(t_n;\tau)\|_0\\ &\le C_7\left(\tau^p t_n^{\alpha-p} + \tau^{p+1}t_n^{\alpha-(p+1)}\right)\|u_0\|_{\dot{H}^{2\alpha}}\\ &\le 2C_7\tau^p t_n^{\alpha-p}\|u_0\|_{\dot{H}^{2\alpha}},\end{aligned}$$

observing $\tau/t = 1/n \le 1$ for $n \ge 1$. ∎

REMARKS 1.1. Theorem 1.2 is actually the case $N = p-1$ of the general asymptotic expansion result [17, Theorem 2.7], which handles with m–sectorial operators in Hilbert space. The proof given here is simplified to the case of a positive selfadjoint operator, which enables us to use the spectral theorem for that operators instead of the Dunford–Taylor calculus.

Note, however, that we are forced to use the term $e_p(t)\tau^p$ of the asymptotic expansion in the proof, since otherwise we could only prove the weaker estimate

$$\|u_n - u(t_n)\|_0 \le C\tau^p t_n^{\min(0,\alpha-p)}\|u_0\|_{\dot{H}^{2\alpha}}.$$

Thus the result of Lemma 1.2 about asymptotic expansions is *inevitable* even if one is not interested in asymptotic expansions in the operator case.

A detailed discussion of estimate (1.5) in some concrete situations can be found in Examples 8.3 and 8.5.

Consider now a sequence $r_j(z)$, $j = 1,2,\ldots$, of A_0–stable rational approximations to $\exp(-z)$ of increasing order j together with the corresponding single step methods $\Phi_j = \Phi_{r_j}$. A variable–step variable–order method for the abstract Cauchy problem can be described as the following device:

Given an initial approximation $u^0 = \tilde{u}(t)$ at time t, a tolerance TOL, time step τ and suggested order k, the method computes the sequence

$$u^j = \Phi_j(u^0,\tau), \qquad j = 1,\ldots,k+1,$$

which approximates with successive higher order the solution $\tilde{u}(t+\tau)$ of the Cauchy problem with initial data $\tilde{u}(t)$ at time τ.

As in [23] we get the error estimates

$$\epsilon_j = \|u^{j+1} - u^j\|_0 \doteq \|\tilde{u}(t+\tau) - u^j\|_0,$$

such that further the approximation u^{k+1} is accepted if

$$\epsilon_k < \mathsf{TOL}\,.$$

Comparison of the ϵ_j with the a priori estimate (1.5) gives new time steps

$$\tau_j = \sqrt[j+1]{\frac{\mathsf{TOL}}{\epsilon_j}}\tau \tag{1.6}$$

for the orders $j = 1, \ldots, k$. As new order k^* together with $\tau^* = \tau_{k^*}$ as new time step we take that order, which minimizes the amount of work per unit step, i.e.

$$\frac{A_{k^*+1}}{\tau^*} = \min_{1 \le j \le k} \frac{A_{j+1}}{\tau_j}. \tag{1.7}$$

Here A_j measures the amount of work for computing the sequence $u^1, \ldots, u^j$.

Repeatedly application of this procedure yields the approximate orbit in Hilbert space.

1.3. The Control of Spatial Perturbations

Computation of $u^j = \Phi_j(\tilde{u}(t), \tau)$ requires the *weak* solution of several elliptic problems due to the denominator of the rational functions $r_j(z)$. In general we cannot get the exact functions u^j but *perturbed* functions

$$\hat{u}^j = u^j + \delta_j \qquad j = 1, \ldots, k+1,$$

with perturbations $\delta_j \in L^2(\Omega)$. The following requirements are reasonable:

Keep the perturbations δ_j below a certain level such that

- the approximation $\hat{u}^{k+1}$ is good enough with respect to TOL,
- the generated time–step sequence is nearly the same as in the case of *no* perturbations.

These requirements ensure that the problem dependent time–stepping in Hilbert space is preserved.

It can be met if we assume that we are able to compute time–error estimates

$$\hat{\epsilon}_j = \epsilon_j + \theta_j \qquad j = 1, \ldots, k,$$

as well as estimates $[\theta_j], [\delta_j]$ of the spatial perturbations $|\theta_j|, \|\delta_j\|_0$.

We then proceed as follows: Compute time steps with respect to $\varrho\,\mathsf{TOL}$ instead of TOL, where $0 < \varrho < 1$. The approximation $\hat{u}^{k+1}$ is accepted if

$$\text{(1.8)} \qquad \begin{aligned} &\text{i)} \quad \hat{\epsilon}_k + [\delta_{k+1}] < \mathsf{TOL}, \\ &\text{ii)} \quad [\theta_j] < \frac{1}{4}\hat{\epsilon}_j \qquad j = 1, \ldots, k. \end{aligned}$$

Implementing this computable control criterion (1.8) yields $\hat{u}^{k+1}$ accurate enough and time steps

$$\hat{\tau}_j = \sqrt[j+1]{\frac{\mathsf{TOL}}{\hat{\epsilon}_j}}\tau,$$

varying in comparison to the corresponding *exact* time steps τ_j as

$$\frac{1}{1.8}\,\tau_j \le \hat{\tau}_j \le 1.3\,\tau_j,$$

provided that $[\theta_j] \doteq |\theta_j|, [\delta_j] \doteq \|\delta_j\|_0$.

In order to make a passing through the criterion (1.8) possible we have to impose accuracies

$$\text{(1.9)} \qquad \mathsf{eps}_j = \chi(j,k)\,(1-\varrho)\,\mathsf{TOL}$$

to the elliptic problems arising in the computation of u^j.

EXAMPLE 1.2. The extrapolated implicit Euler scheme yields as shown in [17]

$$\chi(j,k) = \frac{1}{j}\alpha_j^{k+1},$$

with coefficients α_j^k quite *small* for higher k, for instance $\alpha_5^5 = 6.5_{10}-3$. Thus extrapolation *amplifies* spatial perturbations. This amplification is due to the fact that we build higher and higher order *differences* whose perturbations stay in the order of magnitude of the initial perturbation — but do not decrease like the differences.

2. VARIABLE–ORDER TIME DISCRETIZATION BASED ON A MULTIPLICATIVE ERROR CORRECTION

The drawbacks, which are discussed in [18, Introduction] as well as in Example 1.2, of extrapolation methods or related methods like deferred corrections are a result of the fact that the error estimation is built as a *difference* of two approximations of different order:

$$\eta_j = u^{j+1} - u^j,$$

— a fact which is very similar to the "cancellation effect".

On the contrary, we will derive a method which computes η_j *directly* in such a way that the higher order approximation is given as

$$u^{j+1} = u^j + \eta_j,$$

in order to avoid any cancellation.

2.1. Two Families of Rational Approximations to $\exp(-z)$

In order to achieve the above mentioned structure we write the corresponding rational approximation $r_j(z)$ to e^{-z} as

$$r_{j+1}(z) = r_j(z) + \rho_j(z).$$

The following requirement on the correction term $\rho_j(z)$ will be essential:

> The corrections ρ_{j+1} should be obtained *multiplicatively*, i.e.,
>
> $$\rho_{j+1}(z) = \gamma_{j+1}\rho(z)\rho_j(z), \qquad j = 1, 2, \ldots$$
>
> with a rational function ρ and coefficients γ_j.

Discussion of the Requirement. This specific shape of a *multiplicative error correction* is motivated — besides the effect of avoidance of differences — by the aims of the least possible need of memory and the least possible effort of work:

1. We only need to memorize the actual approximation $r_j(\tau A)$ and the last correction $\rho_{j-1}(\tau A)$ in order to get a new correction $\rho_j(\tau A)$ and thereafter a new approximation $r_{j+1}(\tau A)$;
2. We always have to perform the *same* type of elliptic problem, that is the evaluation of $\rho(\tau A)$, in contrast, e.g., to extrapolation methods which have to compute the *different* elliptic problems $(I + \tau/j\, A)^{-1}$ for varying j.

2.1.1. *Derivation of the Approximations*

We want to start with the implicit Euler, which belongs to parabolic problems [16], i.e.,

$$r_1(z) = \frac{1}{1+z}.$$

In order to evaluate $\rho(\tau A)$ the *same* elliptic problem as in $r_1(\tau A)$ should be solved. Thus the denominator of the rational function ρ has to be chosen as the denominator of r_1, which yields

$$\rho(z) = \frac{\pi(z)}{1+z},$$

with $\pi(z)$ a polynomial in z. Summing up the different correction terms we obtain

$$r_{j+1}(z) = r_1(z) + \sum_{k=0}^{j-1} \bar{\alpha}_k \rho^k(z) \rho_1(z), \qquad j = 1, 2, \ldots,$$

with $\bar{\alpha}_k = \prod_{l=1}^{k} \gamma_{l+1}$ for $k \geq 1$, $\bar{\alpha}_0 = 1$. Finally we have to choose $1+z$ as denominator for ρ_1 as well. We will see later on, that in this way only *two different* families of rational functions $\{r_j\}_j$ are obtainable. Thus we will not discuss a general ansatz for ρ_1, but only those two which technically simplify the matter:

$$\text{(2.1)} \qquad \begin{aligned} &\text{i)} \quad \rho_1^A(z) &&= \gamma_1 \rho^\nu(z), \\ &\text{ii)} \quad \rho_1^L(z) &&= \gamma_1 \rho^\nu(z) r_1(z). \end{aligned}$$

The coefficient $\gamma_1 \neq 0$ and the integer $\nu \geq 0$ will be specified later on. The family generated by ρ_1^A will be called to be of type (A), the family generated by ρ_1^L to be of type (L). The letter A will stand for strong A_0–stability, the letter L for L_0–stability.

Consistency of the approximation, i.e. $r_j(0) = 1$, yields

$$\rho(0) = \pi(0) = 0,$$

because of $\bar{\alpha}_0 = r_1(0) = 1$ and $\gamma_1 \neq 0$. Moreover the minimal stability requirement $r_j(z) = \mathcal{O}(1)$ for $z \to \infty$ yields for $k \geq 1$

$$\rho^k(z) = \mathcal{O}(z) \qquad \text{as } z \to \infty.$$

Thus $\deg \pi \leq 1$. Since we have not yet specified the coefficients $\{\bar{\alpha}_k\}_{k \geq 0}$, we get

$$\rho(z) = \frac{z}{1+z} = 1 - r_1(z).$$

2.1.2. *The Family of Type (L)*

Introducing

$$\text{(2.2)} \qquad \begin{aligned} &\text{i)} \quad \alpha_{k+\nu} &&= \gamma_1 \bar{\alpha}_k + \beta_{k+\nu}, \quad &&k \geq 0, \\ &\text{ii)} \quad \alpha_k &&= \beta_k, &&k = 0, \ldots, \nu - 1, \end{aligned}$$

with $\beta_0 = 1, \beta_k = 0$ for $k \geq 1$, we gain the relation

$$r_j(z) = r_1(z) \sum_{k=0}^{j-2+\nu} \alpha_k \rho^k(z), \qquad j = 1,2,\ldots . \tag{2.3}$$

Our considerations led so far to the following problem: Find coefficients $\{\alpha_k\}_{k\geq 0}$ such that

$$e^{-z} = \frac{1}{1+z} \sum_{k=0}^{\infty} \alpha_k \left(\frac{z}{1+z}\right)^k .$$

Upon introducing

$$w = \frac{z}{1+z} \tag{2.4}$$

we observe that the $\{\alpha_k\}_{k\geq 0}$ should be generated by the function

$$\frac{1}{1-w} \exp\left(\frac{w}{w-1}\right).$$

This function is intimately connected with the *Laguerre polynomials*, since

$$\frac{1}{1-w} \exp\left(\frac{xw}{w-1}\right) = \sum_{k=0}^{\infty} L_k(x) w^k, \qquad |w| < 1 , \tag{2.5}$$

where $L_k(\cdot)$ denotes the Laguerre polynomial of degree k. Thus we have

$$e^{-z} = \frac{1}{1+z} \sum_{k=0}^{\infty} L_k(1) \left(\frac{z}{1+z}\right)^k , \qquad \Re z > -\frac{1}{2}, \tag{2.6}$$

and get

$$\alpha_k = L_k(1), \qquad k = 0,1,\ldots .$$

Therefore $\alpha_0 = 1$, $\alpha_1 = 0$ and $\alpha_2 = -1/2$, which implies by (2.2) that $\nu = 2$ and $\gamma_1 = -1/2$. By (2.3) our rational approximation $r_j(z)$ is given as

$$r_j^L(z) = \frac{1}{1+z} \sum_{k=0}^{j} L_k(1) \left(\frac{z}{1+z}\right)^k . \tag{2.7}$$

To end up with a recurrence formula for the rational functions r_j^L of type (L), we trace the derivation backward:

$$
\begin{aligned}
&\text{i)} \quad r_1^L(z) &&= \frac{1}{1+z} \\
&\text{ii)} \quad \rho_1^L(z) &&= -\frac{1}{2}\frac{z^2}{(1+z)^2} r_1^L(z) \\
&\text{iii)} \quad r_{j+1}^L(z) &&= r_j^L(z) + \rho_j^L(z), \qquad j = 1,2,\dots \\
&\text{iv)} \quad \rho_{j+1}^L(z) &&= \gamma_{j+1}^L \frac{z}{1+z} \rho_j^L(z), \qquad j = 1,2,\dots \\
&\text{v)} \quad \gamma_{j+1}^L &&= \frac{L_{j+2}(1)}{L_{j+1}(1)}, \qquad j = 1,2,\dots
\end{aligned} \tag{2.8}
$$

2.1.3. The Family of Type (A)

Observing $r_1(z) = -1 - \rho(z)$ and introducing

$$
\begin{aligned}
&\text{i)} \quad \alpha_{k+\nu} &&= \gamma_1 \bar{\alpha}_k + \beta_{k+\nu}, \quad k \geq 0, \\
&\text{ii)} \quad \alpha_k &&= \beta_k, \qquad\qquad k = 0,\dots,\nu-1,
\end{aligned} \tag{2.9}
$$

with $\beta_0 = 1, \beta_1 = -1, \beta_k = 0$ for $k \geq 2$, we gain the relation

$$
r_j(z) = \sum_{k=0}^{j-2+\nu} \alpha_k \rho^k(z), \qquad j = 1,2,\dots . \tag{2.10}
$$

Therefore we have to find coefficients $\{\alpha_k\}_{k\geq 0}$ such that

$$
e^{-z} = \sum_{k=0}^{\infty} \alpha_k \left(\frac{z}{1+z}\right)^k .
$$

We now take (2.6) in the form

$$
\begin{aligned}
e^{-z} &= \left(1 - \frac{z}{1+z}\right) \sum_{k=0}^{\infty} L_k(1) \left(\frac{z}{1+z}\right)^k \\
&= \sum_{k=0}^{\infty} (L_k(1) - L_{k-1}(1)) \left(\frac{z}{1+z}\right)^k \\
&= L_0(1) + \sum_{k=1}^{\infty} \frac{L_k'(1)}{k} \left(\frac{z}{1+z}\right)^k ,
\end{aligned}
$$

since $xL_k'(x) = k(L_k(x) - L_{k-1}(x))$. Hence $\alpha_0 = L_0(1)$ and

$$
\alpha_k = \frac{L_k'(1)}{k}, \qquad k = 1,2,\dots ,
$$

which yields $\alpha_0 = 1$, $\alpha_1 = -1$ and $\alpha_2 = -1/2$. We thus obtain by (2.9) that $\nu = 2$ and $\gamma_1 = -1/2$ for the family of type (A) as well. By (2.10) our rational approximation $r_j(z)$ is given as

$$r_j^A(z) = L_0(1) + \sum_{k=1}^{j} \frac{L_k'(1)}{k} \left(\frac{z}{1+z}\right)^k. \tag{2.11}$$

The recurrence formula for the rational functions r_j^A of type (A) is now obtained as:

$$\begin{aligned}
&\text{i)} \quad r_1^A(z) &&= \frac{1}{1+z} \\
&\text{ii)} \quad \rho_1^A(z) &&= -\frac{1}{2}\frac{z^2}{(1+z)^2} \\
&\text{iii)} \quad r_{j+1}^A(z) &&= r_j^A(z) + \rho_j^A(z), \quad j = 1,2,\ldots \\
&\text{iv)} \quad \rho_{j+1}^A(z) &&= \gamma_{j+1}^A \frac{z}{1+z} \rho_j^A(z), \quad j = 1,2,\ldots \\
&\text{v)} \quad \gamma_{j+1}^A &&= \frac{j+1}{j+2}\frac{L_{j+2}'(1)}{L_{j+1}'(1)}, \quad j = 1,2,\ldots
\end{aligned} \tag{2.12}$$

2.1.4. Properties of the Laguerre Polynomials at $x = 1$

In order to derive properties about our families of rational approximations to $\exp(-z)$ we have to study the Laguerre polynomials $L_k(x)$ at $x = 1$ more closely. Note that we denote by $L_n^{(m)}(\cdot)$ the m^{th} derivative of $L_n(\cdot)$ and *not* the generalized Laguerre polynomial.

LEMMA 2.1.

a) We have $|L_n(x)| \leq 1$ for all $x \in [0,1]$, $n \geq 0$.

b) The value $L_n^{(m)}(1)$ is an integer if and only if $n \leq m+1$; $n, m \geq 0$. In the case $n \leq m+1$ we have

$$L_n^{(m)}(1) = \begin{cases} 0 & \text{if } n < m \\ (-1)^m & \text{if } n = m \\ (-1)^m m & \text{if } n = m+1 \end{cases}$$

Proof.

a) Starting with the *confluent hypergeometric equation*

$$xy'' + (1-x)y' + ny = 0,$$

which has the solution $L_n(x)$, we obtain as derivative of

$$v_n(x) = L_n^2(x) + \frac{x}{n}L_n'^2(x)$$

the expression

$$v_n'(x) = \frac{2x-1}{n}L_n'^2(x).$$

Hence $v_n'(x) \leq 0$ for $0 \leq x \leq 1/2$, which implies

$$v_n(x) \leq v_n(0) = L_n^2(0) = 1,$$

i.e.,

$$(*) \qquad L_n^2(x) + \frac{x}{n}L_n'^2(x) \leq 1 \qquad \text{for } 0 \leq x \leq \frac{1}{2}.$$

Next introduce the transformed $\tilde{L}_n(x) = x^{1/4}e^{-x/2}L_n(x)$ and

$$\begin{aligned} \tilde{v}_n(x) &= \tilde{L}_n^2(x) + \frac{x}{n}\tilde{L}_n'^2(x) \\ (**) \qquad &= x^{1/2}e^{-x}\left[L_n^2(x) + \left(\frac{1-2x}{4\sqrt{nx}}L_n(x) + \sqrt{\frac{x}{n}}L_n'(x)\right)^2\right]. \end{aligned}$$

Routine calculation yields

$$\tilde{v}_n'(x) = \frac{4(x-1)^2 - 5}{8nx}\tilde{L}_n(x)\tilde{L}_n'(x).$$

By means of the evident inequality $2ab \leq a^2 + b^2$ for real a, b we obtain

$$2\sqrt{\frac{x}{n}}\tilde{L}_n(x)\tilde{L}_n'(x) \leq \tilde{L}_n^2(x) + \frac{x}{n}\tilde{L}_n'^2(x) = \tilde{v}_n(x),$$

hence the differential inequality

$$\tilde{v}_n'(x) \leq \frac{5}{16\sqrt{n}}x^{-3/2}\tilde{v}_n(x), \quad 0 \leq x \leq 1,$$

since $5 \geq 5 - 4(x-1)^2 \geq 1$ and $\tilde{v}_n(x) \geq 0$ for $0 \leq x \leq 1$. Thus we get for $x \geq 1/n$ that

$$\tilde{v}_n(x) \leq \tilde{v}_n(\tfrac{1}{n}) \exp\left(\int_{1/n}^{x} \frac{5}{16\sqrt{n}}\xi^{-3/2}d\xi\right) = \tilde{v}_n(\tfrac{1}{n}) \exp\left(\tfrac{5}{8}\left(1 - \tfrac{1}{\sqrt{nx}}\right)\right),$$

which implies

$$\tilde{v}_n(1) \leq \tilde{v}_n(1/n)e^{5/8}.$$

For $n \geq 2$ we establish by means of (*) and (**) that

$$\begin{aligned}\tilde{v}_n(1/n) &\leq n^{-1/2}e^{-1/n}\left[1+\left(\frac{1-2/n}{4}+1\right)^2\right] \\ &\leq \frac{41}{16}n^{-1/2}.\end{aligned}$$

Hence we obtain for $n \geq 2$

$$\tilde{v}_n(1) \leq \frac{41}{16}e^{5/8}n^{-1/2}$$

and observing $|\tilde{L}_n(1)| \leq \sqrt{\tilde{v}_n(1)}$

$$|L_n(1)| \leq \frac{1}{4}e^{13/16}\sqrt{41}n^{-1/4}.$$

Thus we can guarantee $|L_n(1)| \leq 1$ if the right hand side is below 1, which means

$$n \geq \frac{1681}{256}e^{13/4} \doteq 169.35,$$

i.e., $n \geq 170$. For $n = 0, 1, \ldots, 169$ one can prove $|L_n(1)| \leq 1$ by direct computation, e.g., with the formula manipulating language REDUCE. Finally formula [1, 22.12.7]

$$L_n(\xi x) = \sum_{j=0}^{n}\binom{n}{j}\xi^j(1-\xi)^{n-j}L_j(x)$$

shows that the value $L_n(\xi)$, $\xi \in [0,1]$, is a convex combination of the values $L_0(1), \ldots, L_n(1)$. Thus $|L_n(x)| \leq 1$, $n \geq 0$, $x \in [0,1]$ follows from $|L_n(1)| \leq 1$, $n \geq 0$.

b) Since

$$L_n(x) = \frac{(-1)^n}{n!}x^n + \frac{(-1)^{n-1}}{(n-1)!}nx^{n-1} + \cdots$$

we get the asserted values for $L_n^{(m)}(1)$, $n \leq m+1$. Now assume that we have

$$L_n^{(m)}(1) = (-1)^m \sum_{j=0}^{n-m}\binom{n}{n-m-j}\frac{(-1)^j}{j!} \in \mathbb{Z}$$

for some $n \geq m+1$. Multiplication with $(n-m)!$ yields

$$\sum_{j=0}^{n-m} \binom{n}{n-m-j}(-1)^j(n-m)(n-m-1)\ldots(j+1) = (-1)^m(n-m)!\,L_n^{(m)}(1),$$

a pure integer expression. Since $n-m$ divides the right hand side and each term of the sum with $j+1 \leq n-m$, it has also to divide the term with $j = n-m$, i.e.,

$$n-m \mid (-1)^{n-m},$$

which implies that $n-m=1$.

■

REMARK 2.1. The technique of estimation in part a) of the proof is the standard way to obtain asymptotic properties for the classical orthogonal polynomials, cf. [40, §7.2] where the estimate

$$|L_n(x)| \leq Cn^{-1/4}, \qquad \text{for } x \in [x_1, x_2], \text{ given } 0 < x_1 < x_2 < \infty$$

can be found. For our purposes we traced the proof more quantitatively in order to get some knowledge about C.

2.1.5. *Properties of the Approximations*

Our approximations are in fact special cases of the so called *RD–Padé approximations* to e^{-z} introduced by NØRSETT [41]. (RD = restricted denominator).

DEFINITION 2.1. *For given real $\sigma \neq 0$ a rational approximation to e^{-z} of the form*

$$R_p^j(z;\sigma) = \frac{\sum_{k=0}^{j} a_k z^k}{(1+\sigma z)^p}$$

of order at least $j \leq p$ is called a (j,p)–RD–Padé approximation.

LEMMA 2.2. (NØRSETT [41]). *Given $\sigma \neq 0$ there is only one (j,p)–RD–Padé approximation, $p-1 \leq j \leq p$, which is*

$$R_p^j(z;\sigma) = \frac{\sum_{k=0}^{j}(-1)^{p-k}L_p^{(p-k)}(1/\sigma)(\sigma z)^k}{(1+\sigma z)^p}.$$

The order of $R_p^j(\cdot\,;\sigma)$ *is* $j+1$ *if and only if*

$$L_p(1/\sigma) = 0 \quad \textit{in the case } j = p-1$$

resp.

$$L_p'(1/\sigma) = 0 \quad \textit{in the case } j = p,$$

otherwise the order is j.

Proof. Corollary 2.1 and Theorem 2.1 of [41]. ∎

REMARK 2.2. The uniqueness statement of Lemma 2.2 is the reason why we can obtain only two families of rational approximations by means of the multiplicative error correction — once we have chosen the denominator to be a power of $1+z$. Hence it is sufficient to only consider the special choices of ρ_1 made in (2.1).

Our main result is now

THEOREM 2.1. *Let* $j \geq 1$.

a) We have that

$$r_j^A(z) = R_j^j(z;1) = \frac{\sum_{k=0}^{j}(-1)^{j-k}L_j^{(j-k)}(1)z^k}{(1+z)^j} \tag{2.13}$$

and

$$r_j^L(z) = R_{j+1}^j(z;1) = \frac{\sum_{k=0}^{j}(-1)^{j+1-k}L_{j+1}^{(j+1-k)}(1)z^k}{(1+z)^{j+1}}. \tag{2.14}$$

b) The approximations r_j^A *are of exact order* j, *are computable by means of the recurrence formula (2.12) and they are strongly* A_0*-stable.*

c) The approximations r_j^L *are of exact order* j, *are computable by means of the recurrence formula (2.8) and they are* L_0*-stable.*

Proof.

a) By construction r_j^A, r_j^L as given by (2.11,2.7) are of order at least j and they have a denominator which is a power of $1+z$. Thus by definition they are RD–Padé approximations and therefore uniquely given by the expressions (2.13,2.14) as stated in Lemma 2.2.

b) According to Lemma 2.2 $r_j^A = R_j^j(\cdot\,;1)$ has order j if and only if $L_j'(1) \neq 0$. This is indeed the case for $j \geq 1$ (and only for those j) by part b) of Lemma 2.1. The same condition assures the recursion (2.12) to be computable. The strong A_0–stability will be proven at the end of c).

c) We obtain by means of Lemma 2.2 that $r_j^L = R_{j+1}^j(\cdot\,;1)$ has order j if and only if $L_{j+1}(1) \neq 0$. This is the case for $j \geq 1$ (and only for those j) by part b) of Lemma 2.1. The fact $L_n(1) \neq 0$ for $n \geq 2$ assures the computability of the recursion (2.8) as well.

Using the transformation (2.4) we obtain

$$\begin{aligned} r_j^L(z) &= \frac{1}{1+z}\sum_{k=0}^{j} L_k(1)\left(\frac{z}{1+z}\right)^k \\ &= (1-w)\sum_{k=0}^{j} L_k(1)w^k. \end{aligned}$$

The interval $z \in]0,\infty[$ is mapped to $w \in]0,1[$ which yields

$$\begin{aligned} |r_j^L(z)| &\leq (1-w)\sum_{k=0}^{j} |L_k(1)|w^k \\ &\leq (1-w)\sum_{k=0}^{j} w^k \\ &< 1, \end{aligned}$$

whenever $z > 0$, since $|L_k(1)| \leq 1$ for $k \geq 0$ by part a) of Lemma (2.1). This proves the L_0–stability of the r_j^L, $j \geq 1$, since $r_j^L(\infty) = 0$.

Comparison of (2.13) with (2.14) yields

$$r_j^A(z) = r_{j-1}^L(z) + L_j(1)\left(\frac{z}{1+z}\right)^j$$

for $j \geq 2$. We thus obtain for $z > 0$

$$\begin{aligned} |r_j^A(z)| &\leq (1-w)\sum_{k=0}^{j-1} w^k + |L_j(1)|w^j \\ &= 1 + (|L_j(1)| - 1)w^j \\ &< 1, \end{aligned}$$

since $|L_j(1)| < 1$ for $j \geq 2$ by Lemma 2.1 (Note part b)). Moreover

$$|r_j^A(\infty)| = |L_j(1)| < 1 \text{ for } j \geq 2.$$

Hence the approximations r_j^A are strongly A_0–stable for $j \geq 2$, $r_1^A = r_1^L$ is L_0–stable anyway.

■

REMARK 2.3. A direct proof of equations (2.13,2.14) is possible by using (2.11,2.7) and the relation

$$\sum_{k=0}^{j} \binom{p-k-1}{j-k} L_k(x) = (-1)^{p-j} L_p^{(p-j)}(x),$$

which can be obtained by differentiating $p-j$ times with respect to x in the generating function formula (2.5).

For some applications sharper stabilities of the r_j could be of interest. For this reason we include in Table I the angles of strong A_ϑ–stability of $r_j^{A,L}$ for $j = 1, \ldots, 10$.

TABLE I.
LOWER BOUNDS FOR THE ANGLES OF STRONG A_ϑ–STABILITY (DEGREES)

j	1	2	3	4	5	6	7	8	9	10
$\vartheta(r_j^A)$	90.00	90.00	90.00	90.00	89.98	89.90	89.76	89.64	89.63	89.73
$\vartheta(r_j^L)$	90.00	90.00	89.45	88.94	89.15	89.67	89.76	89.55	89.25	88.96

We close by listing in Table II the first coefficients γ_j of the recurrences (2.12,2.8).

TABLE II.
THE COEFFICIENTS γ_j FOR $j = 2, \ldots, 9$.

j	numerator of γ_j^A	denominator of γ_j^A	numerator of γ_j^L	denominator of γ_j^L
2	1	3	4	3
3	-1	4	15	16
4	19	5	56	75
5	151	114	185	336
6	1091	1057	204	1295
7	7841	8728	-6209	1632
8	56519	70569	112400	55881
9	396271	565190	1520271	1124000

2.2. The Variable–Order Single Step Method in Hilbert Space

2.2.1. *Application of the Family of Type (L)*

Since for the family of type (L) $r_j^L(\infty) = 0$, the mapping

$$r_j^L(\tau A) : \dot{H}^\alpha \to \dot{H}^{\alpha+2}$$

exists and is continuous. Thus the family of type (L) models the effect of *parabolic smoothing* and is the method of choice for temporally homogeneous processes. Now we will explain the single step methods corresponding to the family of type (L). Given

$$u^0 = \tilde{u}(t)$$

and a time step $\tau \geq 0$ the recurrence (2.8) yields

$$\text{(2.15)} \qquad \begin{aligned} &\text{i)} \quad u^1 = r_1^L(\tau A)u^0 + \left(I - r_1^L(\tau A)\right) A^{-1}f \\ &\text{ii)} \quad \eta_1 = -\frac{1}{2}\left(\tau A(I+\tau A)^{-1}\right)^2 (u^1 - A^{-1}f) \\ &\text{iii)} \quad u^{j+1} = u^j + \eta_j \qquad\qquad j = 1,2,\ldots \\ &\text{iv)} \quad \eta_{j+1} = \gamma_{j+1}^L \tau A(I+\tau A)^{-1}\eta_j \qquad j = 1,2,\ldots, \end{aligned}$$

if we remember the construction (1.4) of single step methods from the rational function. The *update relation* (iv) specifies the meaning of what we called a direct computation of the error corrections η_j.

If we make use of the relation

$$I - (I+\tau A)^{-1} = \tau A(I+\tau A)^{-1},$$

we are able to find a simpler expression for the terms u^1, η_1:

$$\text{(2.16)} \qquad \begin{aligned} &\text{i)} \quad u^1 = (I+\tau A)^{-1}(u^0 + \tau f) \\ &\text{ii)} \quad \eta_0 = u^1 - u^0 \\ &\text{iii)} \quad \eta_1 = \frac{1}{2}\tau A(I+\tau A)^{-2}\eta_0 \end{aligned}$$

Remark 2.4. Another version of representing u^1, η_0 would be

$$\begin{aligned} &\text{i)} \quad \eta_0 = \tau(I+\tau A)^{-1}(f - Au^0) \\ &\text{ii)} \quad u^1 = u^0 + \eta_0, \end{aligned}$$

which puts the difference at a more desirable place. However, this is *only* possible if $u^0 \in \dot{H}^2$.

By means of the representation (2.15) we observe that for

$$u^0, f \in L^2(\Omega)$$

the approximations and corrections possess the necessary regularity:

$$u^j, \eta_j \in \dot{H}^2 \qquad \text{for } j \geq 1.$$

2.2.2. *Application of the Family of Type (A)*

Here only the mapping

$$r_j^A(\tau A) : \dot{H}^\alpha \to \dot{H}^\alpha$$

exists and is continuous. Thus the family of type (A) would the method of choice for processes, which do not increase smoothness. Let us briefly indicate the single step methods corresponding to the family of type (A). For simplicity we assume that

$$u^0 = \tilde{u}(t) \in \dot{H}^2.$$

Given a time step $\tau \geq 0$ the recurrence (2.12) yields by observing Remark 2.4

$$\begin{aligned} &\text{i)} \quad \eta_0 = \tau(I + \tau A)^{-1}(f - Au^0) \\ &\text{ii)} \quad u^{j+1} = u^j + \eta_j \qquad\qquad j = 1, 2, \ldots \\ &\text{iii)} \quad \eta_{j+1} = \gamma_{j+1}^A \tau A(I + \tau A)^{-1}\eta_j \quad j = 0, 1, \ldots . \end{aligned} \tag{2.17}$$

2.2.3. *Time Discretization Pair for Temporally Inhomogeneous Problems*

Let us discuss very briefly what has to be done in the case of temporally inhomogeneous problems, i.e., where the right–hand side f and possibly the boundary conditions depend on the time variable t. We restrict ourselves to a pair of discretizations in time which gives at each time step solutions u^1, u^2 of order 1 and 2 resp. — provided f and u^0 are sufficiently smooth. Since in temporally inhomogeneous problems smoothing can not be in general expected, we take the family of type (A) of rational approximations. For r_1^A we evaluate the time dependent f as in the implicit Euler scheme, for r_2^A by means of the trapezoidal rule. Here u^2 can be shown to be of order 2 for smooth u^0, f by analyzing the corresponding Runge–Kutta method.

For the parabolic problem we thus get as a reasonable device

$$
\begin{aligned}
&\text{i)} \quad u^1 &&= (I+\tau A)^{-1}(u_0+\tau f(\tau)),\\
(2.18) \qquad &\text{ii)} \quad \eta_1 &&= \frac{1}{2}\tau(I+\tau A)^{-1}\left(A(u^1-u^0)-(f(\tau)-f(0))\right),\\
&\text{iii)} \quad u^2 &&= u^1+\eta_1.
\end{aligned}
$$

2.2.4. Interpretation of the Rational Expressions

Since A is the weak representation of the elliptic operator $A(x,\partial)$, problems of the kind

$$u=(I+\tau A)^{-1}w, \qquad w\in L^2(\Omega)$$

are *equivalent* to the variational problem

$$(u,v)+\tau a(u,v)=(w,v) \qquad \text{for all } \ v\in H_D^1(\Omega);$$

whereas problems of the kind

$$\eta=\tau A(I+\tau A)^{-1}\zeta, \qquad \zeta\in \dot{H}^2$$

are equivalent to the variational problem

$$(\eta,v)+\tau a(\eta,v)=\tau a(\zeta,v) \qquad \text{for all } \ v\in H_D^1(\Omega).$$

The equivalence is backed by Theorem 1.1.

3. The Matching of Spatial Errors and Algorithmic Details

In this section we will specify the perturbation concept introduced in the preliminary draft of Section 1 for the family of single step methods *corresponding to the family r_j^L of type (L)*. To this end we derive expressions for the estimators $[\theta_j]$, $[\delta_j]$ as well as for the accuracy function χ. Finally we discuss some algorithmic details.

3.1. The Perturbation Estimators $[\theta_j]$, $[\delta_j]$

In order to realize (2.15) we have to approximate the arising (weak) elliptic problems. Since we have seen that they are equivalent to the variational forms, an adaptive FEM method is ideally suited for our purposes. However, it has to fulfill certain requirements already discussed in [17], Section 4. For

the following the main point of interest is that the elliptic solver may solve within a given accuracy eps and delivers an error estimate. Then we can proceed as follows:

By using the elliptic solver within the given accuracy eps we first get an approximation of u^1

$$\hat{u}^1 = u^1 + \delta_1 \in L^2(\Omega),$$

together with an estimate $[\delta_1] \leq \text{eps}$ of $\|\delta_1\|_0$.

Next we *fix* the triangulation chosen by the elliptic solver in order to compute $\hat{u}^1$ and compute

$$\begin{aligned} \hat{\eta}_1 &= \frac{1}{2}\tau A(I+\tau A)^{-2}(\hat{u}^1 - u^0) + \hat{\omega}_1 + \frac{1}{2}\tau A(I+\tau A)^{-1}\hat{\omega}_0 \\ &= \eta_1 + \omega_1 \end{aligned}$$

on that triangulation. Here $\hat{\omega}_0$ is the error of the approximation $\tilde{\eta}_1$ of

$$(I+\tau A)^{-1}(\hat{u}^1 - u^0),$$

and $\hat{\omega}_1$ denotes the error made while solving the second elliptic problem

$$\frac{1}{2}\tau A(I+\tau A)^{-1}\tilde{\eta}_1.$$

Hence the elliptic solver provides estimates $[\hat{\omega}_0], [\hat{\omega}_1]$ of the norms of the corresponding errors. We gain the representation

$$\omega_1 = \frac{1}{2}\tau A(I+\tau A)^{-2}\delta_1 + \frac{1}{2}\tau A(I+\tau A)^{-1}\hat{\omega}_0 + \hat{\omega}_1$$

and can derive, by using the important estimates

$$\|\tau A(I+\tau A)^{-1}\|, \|(I+\tau A)^{-1}\| \leq 1,$$

the estimator

$$[\theta_1] = \frac{1}{2}([\delta_1] + [\hat{\omega}_0]) + [\hat{\omega}_1].$$

By successively computing

$$\begin{aligned} \hat{\eta}_{j+1} &= \gamma_{j+1}^L \tau A(I+\tau A)^{-1}\hat{\eta}_j + \hat{\omega}_{j+1} \\ &= \eta_{j+1} + \omega_{j+1} \end{aligned}$$

where $\hat{\omega}_{j+1}$ denotes the error made by the elliptic solver, we get

$$\omega_{j+1} = \gamma_{j+1}^L \tau A(I+\tau A)^{-1}\omega_j + \hat{\omega}_{j+1}.$$

This yields the estimator

$$[\theta_{j+1}] = |\gamma_{j+1}^L|[\theta_j] + [\hat{\omega}_{j+1}]. \tag{3.1}$$

Herein $[\hat{\omega}_{j+1}]$ denotes the error estimate given by the elliptic solver. The spatial perturbations of the correction η_j give rise to perturbations of the approximations u^{j+1} since we compute

$$\begin{aligned} \hat{u}^{j+1} &= \hat{u}^j + \hat{\eta}_j \\ &= u^{j+1} + \delta_{j+1}. \end{aligned}$$

Thus we end up with the successive estimates

$$[\delta_{j+1}] = [\delta_j] + [\theta_j]. \tag{3.2}$$

Together with the computationally available time–error estimates

$$\hat{\epsilon}_j = \|\hat{\eta}_j\|_0,$$

we have described the whole family of required estimators.

3.2. The Accuracy Function χ

We have to determine the accuracy eps, in order to make a passing through the criterion (1.8) possible. As shown in Section 1 this determination will result in a relation between the accuracy eps and the parabolic accuracy TOL of the form (1.9).

To this end we observe, that the effect of the perturbation δ_1 will dominate the effects due to the perturbations $\hat{\omega}_j$ in general. *This observation can be backed by a careful frequency analysis of some model problems and is the reason why we fix the triangulation after the computation of* $\hat{u}^1$. Hence it is reasonable to determine eps by the following procedure:

Set $[\delta_1] = \mathsf{eps}_{k+1}$ and $[\hat{\omega}_j] = 0$ for $j = 1, 2, \ldots$. Compute eps_{k+1} in such a way that

$$(1 - \varrho)\,\mathsf{TOL} = [\delta_{k+1}].$$

By using the recurrences (3.1) and (3.2) and the explicit formula for the coefficients γ_j^L (2.8v) we get

$$\begin{aligned} [\delta_{k+1}] &= \left(1 + \sum_{j=1}^{k} \prod_{i=1}^{j} |\gamma_i^L|\right) \mathsf{eps}_{k+1} \\ &= \sum_{j=0}^{k+1} |L_j(1)| \cdot \mathsf{eps}_{k+1}. \end{aligned}$$

We obtain the same result by using the representation (2.7) of $r_{k+1}^L(z)$, if we assume that the error δ_1 is due to the term ahead of the sum.

Hence in order to compute the sequence $\hat{u}^1, \ldots, \hat{u}^{k+1}$ for some $k \geq 1$ we have to impose the elliptic accuracy given by

$$\mathsf{eps} = (1 - \varrho)\chi(k)\,\mathsf{TOL} \tag{3.3}$$

with

$$\chi(k) = \left(\sum_{j=0}^{k+1} |L_j(1)| \right)^{-1}.$$

We can estimate this factor from below *a priori* if we remember that $L_0(1) = 1$, $L_1(1) = 0$ and $|L_j(1)| \leq 1$ for $j \geq 2$ as stated in Lemma 2.1:

$$\chi(k) \geq \frac{1}{k+1}, \qquad \text{for } k \geq 1.$$

This result is highly satisfactory if we compare it with the function χ obtained for extrapolation methods, cf. Example 1.2. The relevant first values of $\chi(k)$ are even more satisfying as shown in Table III.

TABLE III.
THE COEFFICIENTS $\chi(k)^{-1}$ FOR $k = 1, \ldots, 9$.

k	1	2	3	4	5	6	7	8	9
$\chi(k)^{-1}$	1.5	2.2	2.8	3.3	3.5	3.6	3.7	4.0	4.4

REMARK 3.1. The question of the "correct" value for the elliptic accuracy eps is *not* a question of *guaranteeing* the pass through the control criterion (1.8) for *all* possible situations, which yields far too *pessimistic* values and in turn much more effort than needed. However, it is a question of making the pass through *possible* for a large class of *realistic*, i.e. quite probable, situations. This yields more *optimistic* accuracies, and it is intended by some heuristic considerations as well as experience that it is not *too optimistic*. Such unreasonable optimistic accuracies would cause that too much time–step and order suggestions are withdrawn, which in turn leads to more work than needed. Looking at the elliptic accuracy (3.3) and the assumptions leading to it, we should bear in mind that balance.

3.3. Algorithmic Details for Arbitrary Dimension

3.3.1. Information Theoretic Standard Model

As discussed in Deuflhard [23] for ODE's and by the author in [16] for parabolic equations, time–step and order control along the draft of Section 1 becomes a reliable device if we supplement it by an *information theoretic standard model* as introduced in [23]. By comparing the computed time–error estimates $\hat{\epsilon}_j$ with the standard behavior predicted by that model we can implement three devices:

- convergence monitor
- order window
- device for possible increase of order greater than the computed k

For the meaning of these terms we refer to [23].

In [16, 23] the information theoretic standard model is derived for extrapolation methods, but it needs only little change for our new time-discretization: Just replace the coefficients $\alpha(k,q)$ of formula (3.8) in [23] by

$$\alpha(k,q) = \sqrt[k+1]{\frac{(\varrho\,\mathsf{TOL})^{\frac{k+2}{q+2}}}{\varrho\,\mathsf{TOL}}}.$$

3.3.2. Consistency Estimator

To avoid step–size reductions in transient phases we can use a consistency estimator as introduced by the author in [17]. We estimate the "maximal" value of α, such that

$$u^0 \in \dot{H}^{2\alpha}.$$

In view of Theorem 1.2 we can use exactly the same consistency estimator as in [17].

3.3.3. Optimal Choice of the Factor ϱ

We want to optimize the factor ϱ with respect to the expected work. This can be done at least locally in time direction as follows: The local amount of work to realize our algorithm depends on ϱ roughly by means of the factor

$$\frac{1}{(1-\varrho)^{d/2}\sqrt{\varrho}},$$

where d denotes the dimension of the spatial part. This factor may be obtained if we assume the lowest order in time and if we model the work, that the elliptic solver needs to achieve a certain accuracy in the L^2–norm, as in the case of quasi–uniform triangulations. Minimizing that factor gives the optimal value

$$\varrho_d = \frac{1}{d+1}. \tag{3.4}$$

II. Multilevel Discretization of the Elliptic Subproblems

In the Sections 1.3 and 3 we treated the elliptic solver mainly as a *black box*. In fact we required only two things

1. The elliptic solver is started with a given accuracy eps, and supplies us with the error estimates $[\delta_j]$, $[\hat{\omega}_j]$ (cf. Section 3.1).

2. The amount of work A_{j+1} as occurring in (1.7) should be computable.

As another *inevitable* feature turns out to be:

In order to realize the first requirement, it is reasonable in view of Section 2.2.4 to use an adaptive multilevel FEM–method. Such a method contains the following three modules:

- error–estimator
- linear solver
- refinement–strategy

Since we are dealing with the one–parameter family of elliptic problems

$$u + \tau A u = f$$

we have to require:

3. The the error–estimator and the linear solver should behave well *independent of* τ, especially in the vicinity of $\tau = 0$.

Complexity considerations lead us to use as linear solver a *multilevel iteration* as in [25]. Thus the question of a proper *preconditioner* arises. It turns out that preconditioning is the key to the error estimation as well.

4. Triangulations and the Finite Element Discretization

4.1. The Singularly Perturbed Elliptic Problems

Let $\Omega \subset \mathbb{R}^2$ be a bounded simply connected polygonal domain. This implies that

$$\Omega \in C^{0,1},$$

i.e., assumption 1 of Section 1.1 is fulfilled.

As we have seen in Section 2.2.4 the elliptic problems resulting from discretization in time of the parabolic problem can always be given in the following variational form: Find $u \in H_D^1(\Omega)$ such that

$$(u,v) + \tau a(u,v) = \theta_0^*(v) + \tau\theta_1^*(v) \quad \text{for all } v \in H_D^1(\Omega).$$

Here $\theta_0^*, \theta_1^* \in L^2(\Omega) \subset H_D^{-1}(\Omega) = (H_D^1(\Omega))^*$. In order to obtain a convex mean, which will be needed later, we scale with $1+\tau$ to get the equivalent problem: Find $u \in H_D^1(\Omega)$ such that

$$a_\tau(u,v) = \theta_\tau^*(v) \quad \text{for all } v \in H_D^1(\Omega), \tag{4.1}$$

where we made use of the following notation:

$$\text{i)} \quad a_\tau(u,v) = \frac{1}{1+\tau}(u,v) + \frac{\tau}{1+\tau}a(u,v), \quad u,v \in H_D^1(\Omega),$$

$$\text{ii)} \quad \theta_\tau^*(v) = \frac{1}{1+\tau}\theta_0^*(v) + \frac{\tau}{1+\tau}\theta_1^*(v), \quad v \in H_D^1(\Omega).$$

Remark 4.1. Note that this scaling is *not invariant* to a linear scale of the time variable. Thus the question of an appropriate scaling of given physical examples arises. This can be answered in a satisfying way as discussed in Section 9.2.3.

Furthermore we get as the corresponding weak representation (cf. Sec. 1.1) the positive selfadjoint operator

$$\Lambda = \frac{1}{1+\tau}I + \frac{\tau}{1+\tau}A, \tag{4.2}$$

with the domain of definition

$$D(\Lambda) = \begin{cases} \dot{H}^2, & \tau > 0 \\ L^2(\Omega), & \tau = 0 \end{cases}.$$

The *energy norm* $\|\Lambda^{1/2}\cdot\|_0$ will be denoted by $\|\cdot\|_\Lambda$.

Case $\tau = 0$. In this case problem (4.1) reads as

$$(u, v) = \theta_0^*(v) \quad \text{for all } v \in H_D^1(\Omega).$$

The corresponding energy norm is the L^2–norm:

$$\|u\|_\Lambda = \|u\|_0, \qquad u \in L^2(\Omega).$$

Case $\tau = \infty$. In this case problem (4.1) reduces to the stationary problem

$$a(u, v) = \theta_1^*(v), \qquad v \in H_D^1(\Omega).$$

The corresponding energy norm is the energy norm of the elliptic operator A:

$$\|u\|_\Lambda = \|u\|_{\dot{H}^1}, \qquad u \in H_D^1(\Omega).$$

4.2. Triangulations and the Finite Element Spaces

4.2.1. Triangulations

A *triangulation* $\mathcal{T}$ of the polygonal domain Ω is given as the set of triangles resulting from a simplicial partition of Ω.

We start with a coarse triangulation $\mathcal{T}_0$ of Ω with the property that the Dirichlet boundary piece Γ_D is composed of edges of triangles $T \in \mathcal{T}_0$. The triangulation $\mathcal{T}_0$ is refined several times, giving a family of *nested* triangulations $\mathcal{T}_0, \mathcal{T}_1, \ldots, \mathcal{T}_j$. A triangle of $\mathcal{T}_{k+1}$ is either a triangle of $\mathcal{T}_k$ or is generated by subdividing a triangle of $\mathcal{T}_k$ into four congruent triangles or into two triangles by connecting one of its vertices with the midpoint of the opposite side. The first case is called a regular or *red* refinement and the resulting triangles as well as the triangles of the initial triangulation are called regular triangles. The second case is an irregular or *green* refinement and results in two so–called irregular triangles.

However, the irregular refinement has the character of a *closure* which we force by the following *rule*:

(T1) Each new vertex of $\mathcal{T}_k$, i.e., a vertex which does not belong to $\mathcal{T}_{k-1}$, is a vertex of a triangle which was generated by regular refinement.

The irregular refinement is potentially dangerous because interior angles are reduced. Therefore, we add the following rule:

(T2) Irregular triangles may not be further refined.

This rule insures that every triangle of any triangulation $\mathcal{T}_k$ is geometrically similar to a triangle of the initial triangulation $\mathcal{T}_0$ or to a green refinement of a triangle in $\mathcal{T}_0$. These triangulations are meanwhile standard and have been introduced by BANK et al. in [6, 11].

The index of the final triangulation will always be denoted by j and will be fixed in most of the following considerations.

By the *depth* of a triangle

$$T \in \bigcup_{k=0}^{j} \mathcal{T}_k$$

we mean the number of successive ancestors in the family of triangulations. If we add the rule

(T3) Only triangles of depth $k-1$ are refined for the construction of $\mathcal{T}_k$,

we get the following expression for the depth of a triangle $T \in \bigcup_{k=0}^{j} \mathcal{T}_k$

$$\text{depth}(T) = \min\{0 \leq k \leq j \mid T \in \mathcal{T}_k\}.$$

Equipped with rule (T3) we can uniquely reconstruct the sequence $\mathcal{T}_1, \ldots, \mathcal{T}_{j-1}$ from the knowledge of the initial triangulation $\mathcal{T}_0$ and the final triangulation $\mathcal{T}_j$ *alone*, without knowing the actual dynamic refinement process leading to $\mathcal{T}_j$ in an adaptive algorithm, see [25]. However, if we choose the data-structures representing the triangulation cleverly, the sequence $\mathcal{T}_0, \mathcal{T}_1, \ldots, \mathcal{T}_j$ is *implicitly* given. For example this is the case in the adaptive FEM code KASKADE, cf. ROITZSCH [44, 45] or LEINEN [33].

4.2.2. Notation in Connection with Finite Element Spaces

Corresponding to the triangulations $\mathcal{T}_k$ we have *finite element* spaces $\mathcal{S}_k$. $\mathcal{S}_k$ consists of all functions which are linear on each triangle $T \in \mathcal{T}_k$ and continuous on Ω. Furthermore they vanish on the Dirichlet boundary piece Γ_D. Because the triangulations are nested we have

$$\mathcal{S}_0 \subset \mathcal{S}_1 \subset \ldots \subset \mathcal{S}_j \subset H^1_D(\Omega).$$

Let $\mathcal{N}_k = \{x_1^{(k)}, \ldots, x_{n_k}^{(k)}\}$ be the set of vertices of triangles in $\mathcal{T}_k$, which do not lie on the Dirichlet boundary piece Γ_D.

The nodal basis. The set $\Gamma_k = \{\psi_1^{(k)}, \ldots, \psi_{n_k}^{(k)}\}$ of nodal basis functions, where

$$\psi_i^{(k)}(x_l^{(k)}) = \delta_{il} \quad \text{for } 1 \leq i, l \leq n_k,$$

forms a basis of $\mathcal{S}_k$. For $\psi \in \Gamma_k$ we denote by $x_\psi \in \mathcal{N}_k$ the *supporting point* of ψ, i.e.

$$\psi(x_\psi) = 1.$$

Structuring of the nodal bases of varying index k. We set

$$\begin{aligned} \text{i)} \quad \Psi &= \bigcup_{k=0}^{j} \Gamma_k, \\ \text{ii)} \quad \Psi_0 &= \Gamma_0, \\ \text{iii)} \quad \Psi_k &= \Gamma_k \setminus \Gamma_{k-1}, \qquad \text{whenever } 1 \leq k \leq j. \end{aligned}$$

It should be stressed that we split the set of nodal basis functions rather than the set of nodal points as done in hierarchical basis approaches. For $\psi \in \Psi$ we denote the set of indices, for which a nodal basis function ψ occurs, by

$$K_\psi = \{k |\ \psi \in \Gamma_k\}.$$

Here we abbreviate the first resp. the last occurrence of ψ in a set Γ_k by

$$\begin{aligned} \text{i)} \quad k_\psi^0 &= \min K_\psi, \\ \text{ii)} \quad k_\psi^1 &= \max K_\psi. \end{aligned}$$

The duality map. According to the Theorem of Fréchet–Riesz the duality map

$$\begin{aligned} \mathcal{I}_k : \mathcal{S}_k &\to \mathcal{S}_k^* \\ u &\mapsto u^* = (u, \cdot) \end{aligned}$$

is an isometrical isomorphism.
The dual basis. On $\mathcal{S}_k^*$ a natural basis is given by the canonical dual basis $\Gamma_k^* = \{\psi_* |\ \psi \in \Gamma_k\}$ to the basis Γ_k of $\mathcal{S}_k$. As usual ψ_* is defined as the *evaluation functional* at x_ψ:

$$\begin{aligned} \psi_* : \mathcal{S}_k &\to \mathbb{R} \\ u &\mapsto u(x_\psi), \end{aligned}$$

such that $\psi_*(\varphi) = \delta_{\psi\varphi}$ for all $\psi, \varphi \in \mathcal{S}_k$. The choice of these bases will be called the *natural representation* of the spaces $\mathcal{S}_k$ and $\mathcal{S}_k^*$.
The orthogonal L^2–projections. The orthogonal L^2 projections $\pi_k : L^2(\Omega) \to \mathcal{S}_k$, for $0 \leq k \leq j$ are given for $u \in L^2(\Omega)$ as

$$(\pi_k u, v) = (u, v) \qquad \text{for all } v \in \mathcal{S}_k.$$

Restricted L^2–norm. For a measurable subset $\Omega_1 \subset \Omega$ we denote the L^2 norm restricted to that Ω_1 by

$$\|u\|_{0;\Omega_1}^2 = \int_{\Omega_1} u^2(x)dx \qquad \text{for } u \in L^2(\Omega).$$

4.2.3. Some Technical Results

For later purposes we state the following technical Lemmas.

LEMMA 4.1.

i) For $\psi \in \Gamma_k$ we have

$$\text{supp } \psi = \bigcup \{T \in \mathcal{T}_k |\ x_\psi \in T\}$$

and

$$|\text{supp } \psi| = 3\ (1, \psi). \tag{4.3}$$

ii) If $x \in \mathcal{N}_k$ is the supporting point of both $\psi \in \Gamma_k$ and $\bar{\psi} \in \Gamma_{k+1}$ we get

$$\text{supp } \bar{\psi} \subset \text{supp } \psi,$$

with equality if and only if $\bar{\psi} = \psi$.

iii) For $\psi \in \Psi$ the set K_ψ is connected, that is

$$K_\psi = \{k |\ k_\psi^0 \leq k \leq k_\psi^1\}.$$

iv) The set Ψ is the disjoint union of the Ψ_k, $0 \leq k \leq j$ and the number of elements in it is bounded by

$$\#\Psi \leq 2n_j - n_0,$$

no matter how the sequence $\{n_k\}_k$ actually progresses. This is the best possible estimate in terms of n_j and n_0 only.

Proof. We only prove the less trivial assertions.

i) Formula (4.3) is just the well known formula for the volume of a pyramid.

ii) Take any triangle $T_0 \in \mathcal{T}_{k+1}$ with $T_0 \subset \text{supp } \bar{\psi}$. There is a unique triangle $T \in \mathcal{T}_k$ with $T_0 \subset T$. Since $x \in T_0 \subset T$ we have $T \subset \text{supp } \psi$. By i) we thus get supp $\bar{\psi} \subset$ supp ψ. If moreover supp $\bar{\psi} =$ supp ψ we get $\psi(x_\partial) = 0$ for $x_\partial \in \mathcal{N}_{k+1} \setminus \{x\}$ since supp $\bar{\psi} \cap (\mathcal{N}_{k+1} \setminus \{x\}) \subset \partial(\text{supp } \bar{\psi})$ and

$$\psi|_{\partial(\text{supp } \psi)} = 0.$$

By $\psi \in \mathcal{S}_k \subset \mathcal{S}_{k+1}$ we get $\psi = \bar{\psi}$.

iii) Take $\psi \in \Psi$. Assume that there is a k with $k_\psi^0 < k < k_\psi^1$ and $\psi \notin \Gamma_k$. This k can be chosen to be maximal, so that $\psi \in \Gamma_{k+1}$. Since $x_\psi \in \mathcal{N}_{k_\psi^0} \subset \mathcal{N}_k$ there is a $\bar{\psi} \in \Gamma_k$ with $x_{\bar{\psi}} = x_\psi$. This implies by ii) that $\operatorname{supp} \psi \subset \operatorname{supp} \bar{\psi} \subset \operatorname{supp} \psi$. Hence $\operatorname{supp} \psi = \operatorname{supp} \bar{\psi}$, which in turn implies by ii) that $\psi = \bar{\psi}$, a contradiction.

iv) Clearly $\psi \in \Psi$ implies $\psi \in \Psi_{k_\psi^0}$ which gives reason to

$$\Psi = \bigcup_{k=0}^{j} \Psi_k.$$

If there would be a $\psi \in \Psi_k \cap \Psi_l$ for some $k < l$ we would deduce that $\psi \notin \Gamma_{l-1}$ but $\psi \in \Gamma_k \cap \Gamma_l$, a contradiction to iii).

The set Ψ_{k+1} consists of all $\psi \in \Gamma_{k+1}$ supported by the vertices of regular triangles $T \in \mathcal{T}_{k+1} \setminus \mathcal{T}_k$, a result stated in Lemma 6.10 below. These vertices are *exactly* the vertices and midpoints of edges of triangles $T \in \mathcal{T}_k$, which have been regularly refined in transition to $\mathcal{T}_{k+1}$. Denote the set of these edges by $\mathcal{E}_k$, the sets of vertices resp. midpoints of these edges by $\operatorname{ver}(\mathcal{E}_k)$ resp. $\operatorname{mid}(\mathcal{E}_k)$. Each vertex of $\operatorname{ver}(\mathcal{E}_k)$ belongs to *at least two* edges of $\mathcal{E}_k$, thus $\#\operatorname{ver}(\mathcal{E}_k) \leq \frac{1}{2} \cdot 2 \cdot \#\mathcal{E}_k$. By means of rule (T1) we obtain $\operatorname{mid}(\mathcal{E}_k) = \mathcal{N}_{k+1} \setminus \mathcal{N}_k$, which implies $\#\mathcal{E}_k = \#\operatorname{mid}(\mathcal{E}_k) = \#(\mathcal{N}_{k+1} \setminus \mathcal{N}_k) = n_{k+1} - n_k$. Thus we have $\#\Psi_{k+1} = \#\operatorname{ver}(\mathcal{E}_k) + \#\operatorname{mid}(\mathcal{E}_k) \leq 2(n_{k+1} - n_k)$. This yields

$$\begin{aligned} \#\Psi &= \sum_{k=0}^{j} \#\Psi_k \\ &\leq n_0 + 2\sum_{k=1}^{j} (n_k - n_{k-1}) \\ &= 2n_j - n_0. \end{aligned}$$

Easy examples show that this is the best possible bound in terms of n_j and n_0 only. ∎

LEMMA 4.2. *For $u^* \in \mathcal{S}_l^*$ we have for $0 \leq k \leq l$*

$$\pi_k \mathcal{I}_l^{-1} u^* = \mathcal{I}_k^{-1} \left(u^*|_{\mathcal{S}_k} \right).$$

Proof. For $v \in \mathcal{S}_k$ holds both

$$u^*(v) = (\mathcal{I}_l^{-1} u^*, v) = (\pi_k \mathcal{I}_l^{-1} u^*, v)$$

and

$$u^*(v) = u^*|_{\mathcal{S}_k}(v) = \left(\mathcal{I}_k^{-1}\left(u^*|_{\mathcal{S}_k}\right), v\right).$$

Thus we obtain the assertion. ∎

4.3. The Finite Element Discretization

The finite element (FEM) discrete solution $u_k \in \mathcal{S}_k$ is given as the Galerkin approximation to the variational problem (4.1)

$$a_\tau(u_k, v_k) = f^*(v_k) \quad \text{for all } v_k \in \mathcal{S}_k. \tag{4.4}$$

Here $f^* \in \mathcal{S}_j^*$ denotes an approximation of θ_τ^* on $\mathcal{S}_j$. Due to the Theorem of Fréchet–Riesz there are *symmetric positive definite* linear operators $A_k, \Lambda_k : \mathcal{S}_k \to \mathcal{S}_k$, such that for given $u, v \in \mathcal{S}_k$

$$(A_k u, v) = a(u, v)$$

resp.

$$(\Lambda_k u, v) = a_\tau(u, v).$$

For $0 \le k \le l$ we obtain the relations

$$A_k = \pi_k A_l|_{\mathcal{S}_k}$$

and

$$\Lambda_k = \pi_k \Lambda_l|_{\mathcal{S}_k} = \frac{1}{1+\tau} I_k + \frac{\tau}{1+\tau} A_k,$$

where I_k denotes the identity on the space $\mathcal{S}_k$. Problem (4.4) is now given as

$$\mathcal{I}_k \Lambda_k u_k = f^*|_{\mathcal{S}_k}. \tag{4.5}$$

Usage of Lemma 4.2 yields as the solution operator $f^* \mapsto u_k$

$$\rho_k = \Lambda_k^{-1} \pi_k \mathcal{I}_j^{-1} : \mathcal{S}_j^* \to \mathcal{S}_k. \tag{4.6}$$

4.4. The Solution Process and Requirements for a Preconditioner

Computationally problem (4.4) is realized for $k = j$ as follows: We have

$$u_j = \sum_{i=1}^{n_j} u_j(x_{\psi_i^{(j)}}) \psi_i^{(j)},$$

which implies the equivalence of (4.4) and

$$\sum_{i=1}^{n_j} u_j(x_{\psi_i^{(j)}})\, a_\tau(\psi_i^{(j)}, \psi_l^{(j)}) = f^*(\psi_l^{(j)}) \qquad \text{for all } 1 \le l \le n_j.$$

By introducing the *mass matrix* $\mathrm{M} = (m_{il})_{il}$ with

$$m_{il} = (\psi_i^{(j)}, \psi_l^{(j)})$$

and the *stiffness matrix* $\mathrm{A} = (a_{il})_{il}$ with

$$a_{il} = a(\psi_i^{(j)}, \psi_l^{(j)})$$

for $1 \le i, l \le n_j$, we gain as problem matrix A_τ the following convex combination of M and A:

$$\mathrm{A}_\tau = \frac{1}{1+\tau}\mathrm{M} + \frac{\tau}{1+\tau}\mathrm{A}.$$

Introducing the vectors $\vec{u} = \left(u_j(x_{\psi_i^{(j)}})\right)_i$ and $\vec{f} = \left(f^*(\psi_l^{(j)})\right)_l$ we obtain the computational problem

$$\mathrm{A}_\tau \vec{u} = \vec{f}. \tag{4.7}$$

However, this linear equation on $\mathbb{R}^{n_j}$ is just the natural matrix representation of the linear problem (4.5) in the case $k = j$

$$\mathcal{I}_j \Lambda_j u_j = f^*. \tag{4.8}$$

This fact is the reason why we have stressed the importance of the natural representation of the dual pair $(\mathcal{S}_j, \mathcal{S}_j^*)$, which will serve as a rather elegant method to describe the computational problem.

The large linear system (4.7) has to be solved iteratively. Since the involved matrices are symmetric positive definite a preconditioned conjugate gradient (CG) method is the method of choice.

We require several features for a preconditioning matrix B_τ:

(P1) The spectral condition number $\kappa = \kappa(\mathrm{B}_\tau \mathrm{A}_\tau)$ should only grow in j like $j^{2\nu}$, where $0 \le \nu \le 1$. Further it should remain bounded *independently* of the time step $\tau \ge 0$. These properties should neither depend (severely) on the shape of the domain under consideration nor on any quasi–uniformity of the triangulations.

(P2) The cost of computing $\mathrm{B}_\tau \vec{r}$ should be proportional to the dimension n_j.

By requirement (P1) the number $\ell(\epsilon)$ of iterations necessary to reduce the error in the energy norm of A_τ by the factor ϵ is bounded by

$$\ell(\epsilon) \leq \frac{1}{2}\sqrt{\kappa}\left|\log\frac{\epsilon}{2}\right| = \mathcal{O}(j^\nu) \quad , \ 0 \leq \nu \leq 1,$$

independently of τ. If we solve each of the linear problems only as accurate as the discretization on the corresponding triangulation is expected to be, we end up with an *overall complexity* of

$$\mathcal{O}(j^{\nu+\sigma} n_j), \qquad 0 \leq \sigma \leq 1,$$

in view of requirement (P2) — an idea due to DEUFLHARD et al. [25] and implemented in the adaptive FEM solver KASKADE. The exponent σ is connected with the progression of unknowns during refinement: $\sigma = 0$ in the case of geometrical progression, whereas $\sigma = 1$ in the case of pure arithmetical progression. *Note that we do not propose to force the number of unknowns to progress geometrically* — for a reason discussed in Example 8.3, cf. especially Fig. 19.

Reliable time–step control requires that the locally arising systems of ordinary differential equations, as which our algorithm can be viewed in each time–layer, are smooth, thus leading to

(P3) The matrix B_τ should depend *smoothly* on $\tau \geq 0$.

Finally we do not want to analyze the problem in matrix notation but in the corresponding operator version (4.8). If we introduce the operator $\Theta_j^* : \mathcal{S}_j^* \to \mathcal{S}_j$, whose matrix in the natural representation of $(\mathcal{S}_j, \mathcal{S}_j^*)$ is given by B_τ, the preconditioned CG–method can be written in its untransformed fashion as follows:

4.4.1. Preconditioned CG–method on $(\mathcal{S}_j, \mathcal{S}_j^*)$.

We want to solve

$$\mathcal{I}_j \Lambda_j u = f^*.$$

Given a start iteration $\tilde{u}_0$ we set

$$p_0 = r_0 = \Theta_j^* \left(f^* - \mathcal{I}_j \Lambda_j \tilde{u}_0\right).$$

For $k = 0, 1, \ldots$ we iterate

$$\begin{aligned} \tilde{u}_{k+1} &= \tilde{u}_k + \alpha_k p_k \\ p_{k+1} &= r_{k+1} - \beta_k p_k \end{aligned}$$

with

$$\begin{aligned} r_{k+1} &= \Theta_j^* \left(f^* - \mathcal{I}_j \Lambda_j \tilde{u}_{k+1} \right) \\ \alpha_k &= \frac{(\mathcal{I}_j \Lambda_j r_k)(p_k)}{(\mathcal{I}_j \Lambda_j p_k) \left(\Theta_j^* \mathcal{I}_j \Lambda_j p_k \right)} \\ \beta_k &= \frac{(\mathcal{I}_j \Lambda_j r_{k+1}) \left(\Theta_j^* \mathcal{I}_j \Lambda_j p_k \right)}{(\mathcal{I}_j \Lambda_j p_k) \left(\Theta_j^* \mathcal{I}_j \Lambda_j p_k \right)}. \end{aligned}$$

In the natural representation of $(\mathcal{S}_j, \mathcal{S}_j^*)$ we directly get the computationally available version of the preconditioned CG–method. We thus have to require that the operator Θ_j^* is in fact already given in that representation, that means

(P4) The operator Θ_j^* should be given in such a form that directly allows to reconstruct the matrix B_τ without any further effort.

4.5. Quadratic Elements

For the use of error estimation the space of piecewise quadratic elements on $\mathcal{T}_j$ will be needed later on. Here we introduce the corresponding *notation:*

The space $\mathcal{S}_Q$ consists of all functions which are a quadratic polynomial on each triangle $T \in \mathcal{T}_j$ and which are continuous on Ω. Furthermore they vanish on the Dirichlet boundary piece Γ_D, such that

$$\mathcal{S}_Q \subset H_D^1(\Omega).$$

Let $\mathcal{N}_Q$ be the set of *midpoints* of edges belonging to $\mathcal{T}_j$ but not to the Dirichlet boundary piece Γ_D. Take the (quadratic) *hierarchical* basis Γ_Q, which consists of those $\psi \in \mathcal{S}_Q$, for which

$$\psi(x) = 0$$

for all $x \in \mathcal{N}_j$ and

$$\psi(x_\psi) = 1$$

for *exactly one* $x_\psi \in \mathcal{N}_Q$, the *supporting point* of ψ. With $\mathcal{V}_Q = \operatorname{span} \Gamma_Q$ we gain the direct composition

$$\mathcal{S}_Q = \mathcal{S}_j \oplus \mathcal{V}_Q.$$

The operators $I_Q, A_Q, \Lambda_Q : \mathcal{S}_Q \to \mathcal{S}_Q$ and $\mathcal{I}_Q : \mathcal{S}_Q \to \mathcal{S}_Q^*$ have the analogous meaning to I_j, A_j, Λ_j and $\mathcal{I}_j$.

5. Error Estimation — Basic Considerations

In this section we explain our concept of deriving error estimates for the elliptic subproblems. It clearly splits in two independent parts. First properties of the Galerkin approximation play a prominent role, while in the second part only finite dimensional linear problems are involved, where preconditioning very naturally comes into play. We will see that a good preconditioner of the linear systems gives rise to good error estimations. The whole section is written in a rather abstract spirit, which shows that the results are valid for Galerkin approximations in general. However, the notation which is used tries to keep a balance between the needs of the general case and those of the specific considerations of Section 6.4, where the now introduced concept is actually applied to the elliptic subproblems.

5.1. Deviation Estimates Imply Error Estimates

Let $u \in H^1_D(\Omega)$ be the solution of problem (4.1). Consider finite subspaces $\mathcal{S}^\flat \subset \mathcal{S}^\sharp$ of $H^1_D(\Omega)$. To $\sigma \in \{\flat, \sharp\}$ there correspond Galerkin approximations $u^\sigma \in \mathcal{S}^\sigma$ to u fulfilling

$$a_\tau(u^\sigma, v^\sigma) = \theta^*_\tau(v^\sigma), \qquad \text{for all } v^\sigma \in \mathcal{S}^\sigma.$$

Furthermore assume that we have an approximation $\hat{u} \in \mathcal{S}^\flat$ to $u^\flat$. The following notion will be the basis of our analysis.

Definition 5.1. *The pair $(\mathcal{S}^\flat, \mathcal{S}^\sharp)$ has the β–approximation property with respect to u, if there is a $\beta \in]0,1[$ such that*

$$\|u - u^\sharp\|_\Lambda \leq \beta \|u - u^\flat\|_\Lambda.$$

Theorem 5.1. *Whenever the pair $(\mathcal{S}^\flat, \mathcal{S}^\sharp)$ fulfills the β–approximation property with respect to u, we obtain the error estimate*

$$\|u^\sharp - \hat{u}\|_\Lambda \leq \|u - \hat{u}\|_\Lambda \leq \gamma \|u^\sharp - \hat{u}\|_\Lambda,$$

where

$$\gamma = \sqrt{1 + \frac{1}{1 - \beta^2}}.$$

Proof. Orthogonality with respect to the inner product $a_\tau(\cdot,\cdot)$ yields

$$(*) \qquad \|u-\hat{u}\|_\Lambda^2 = \|u-u^\sharp\|_\Lambda^2 + \|u^\sharp-\hat{u}\|_\Lambda^2,$$

since $\hat{u} \in \mathcal{S}^\sharp$. Thus the left inequality is proven. Similarly we get

$$(**) \qquad \|u-u^\flat\|_\Lambda^2 \le \|u-\hat{u}\|_\Lambda^2 = \|u-u^\flat\|_\Lambda^2 + \|u^\flat-\hat{u}\|_\Lambda^2.$$

Thus by (*) and the β–approximation property

$$\|u-u^\flat\|_\Lambda^2 \le \frac{1}{1-\beta^2}\|u^\sharp-\hat{u}\|_\Lambda^2.$$

Now (**) implies

$$\begin{aligned} \|u-\hat{u}\|_\Lambda^2 &\le \|u-u^\flat\|_\Lambda^2 + \|u^\sharp-\hat{u}\|_\Lambda^2 \\ &\le \left(1+\frac{1}{1-\beta^2}\right)\|u^\sharp-\hat{u}\|_\Lambda^2, \end{aligned}$$

which yields the right inequality. ∎

REMARK 5.1. The values of γ behave quite moderately: For example

$$\sqrt{2} \le \gamma \le 2 \quad \text{for} \quad \beta \le \frac{1}{3}\sqrt{6} \doteq 0.816,$$

or even

$$\sqrt{2} \le \gamma \le 10 \quad \text{for} \quad \beta \le \frac{1}{99}\sqrt{9702} \doteq 0.995.$$

Thus the energy norm of the $\sharp$*–deviation* $u^\sharp - \hat{u}$ is a good estimator for the energy norm of the error $u - \hat{u}$. However, we do not actually want to compute $u^\sharp$. Therefore our next objective will be the construction of a deviation estimator.

5.2. PRECONDITIONING IMPLIES DEVIATION ESTIMATES

Given a finite dimensional space $\mathcal{S} \subset H_D^1(\Omega)$, there exists — according to the theorem of Fréchet–Riesz — a symmetric positive definite operator $\Lambda : \mathcal{S} \to \mathcal{S}$ such that

$$a_\tau(u,v) = (\Lambda u, v) \quad \text{for all } u,v \in \mathcal{S}.$$

This operator should not be confused with the weak representation operator Λ on $H_D^1(\Omega)$.

The Galerkin approximation $u \in \mathcal{S}$ to the solution of problem (4.1) is the solution of

$$\Lambda u = f, \tag{5.1}$$

where $(f, \cdot) = \theta_\tau^*|_{\mathcal{S}}$.

In general inverting of Λ is very expansive. Thus we are forced to use an iterative method like the preconditioned CG–method. As a preconditioner can serve any symmetric positive definite operator Θ, which obeys the requirements (P1) and (P2) of Section 4.4. In our rather abstract setting they read as follows:

(P1') Θ behaves spectrally like Λ^{-1}, which means that $\kappa(\Theta\Lambda)$ is of "moderate" size.

(P2') Θr for $r \in \mathcal{S}$ should be easily and cheaply to obtain.

The condition number $\kappa(\Theta\Lambda)$ can be estimated recalling the following Lemma, which may be found, e.g., as Lemma 2.2 in Xu [53].

LEMMA 5.1. *Assume that Λ and Θ are both symmetric positive definite with respect to $(\cdot,\cdot)$ and μ_0 and μ_1 are two positive constants. The following inequalities, which hold for all $u \in \mathcal{S}$, are mutually equivalent:*

$$\begin{array}{lccccc} \text{i)} & \mu_0(\Lambda u, u) & \leq & (\Lambda\Theta\Lambda u, u) & \leq & \mu_1(\Lambda u, u), \\ \text{ii)} & \mu_0(\Theta u, u) & \leq & (\Theta\Lambda\Theta u, u) & \leq & \mu_1(\Theta u, u), \\ \text{iii)} & \mu_1^{-1}(\Lambda u, u) & \leq & (\Theta^{-1} u, u) & \leq & \mu_0^{-1}(\Lambda u, u), \\ \text{iv)} & \mu_1^{-1}(\Theta u, u) & \leq & (\Lambda^{-1} u, u) & \leq & \mu_0^{-1}(\Theta u, u). \end{array} \tag{5.2}$$

If any of the above inequalities holds, then

$$\kappa(\Theta\Lambda) \leq \frac{\mu_1}{\mu_0}.$$

Application of the CG–method to (5.1) with preconditioner Θ yields after some iterations an approximation $\hat{u}$ to u. We introduce the *deviation*

$$d = u - \hat{u},$$

which obeys the defect equation:

$$\Lambda d = r = f - \Lambda\hat{u}.$$

The next theorem states the *connection* of preconditioning and deviation estimation.

THEOREM 5.2. *With the notation introduced above we gain*

$$\|d\|_\Lambda = \zeta \|r\|_\Theta, \tag{5.3}$$

where the constant ζ *is bounded within*

$$\zeta \in \left[\frac{1}{\sqrt{\mu_1}}, \frac{1}{\sqrt{\mu_0}} \right].$$

Proof. By (5.2.iv) we get

$$\begin{aligned} \|d\|_\Lambda^2 &= (d, \Lambda d) \\ &= (\Lambda^{-1} r, r) \\ &= \zeta^2 (r, \Theta r) \\ &= \zeta^2 \|r\|_\Theta^2, \end{aligned}$$

where $\zeta^2 \in [\, 1/\mu_1, 1/\mu_0 \,]$. ∎

REMARK 5.2. Note that by requirement (P2') the value $\|r\|_\Theta$ can be obtained easily and cheaply. So it can serve as *deviation estimator.*

REMARK 5.3. We want to measure the *quality* of an error estimate like (5.3). Therefore we introduce the *quality indicator*

$$\kappa_\zeta = \frac{\zeta_{\max}}{\zeta_{\min}}$$

whenever $\zeta \in [\zeta_{\min}, \zeta_{\max}]$. A good error–estimator is therefore characterized by $\kappa_\zeta \approx 1$, since $\kappa_\zeta = 1$ would mean that we have computed the size exactly — with the exception of *gauging.* Theorem 5.2 together with Lemma 5.1 now states that

$$\kappa_\zeta = \sqrt{\kappa(\Theta\Lambda)},$$

Thus we gain the same number which governs the number of CG–iterations necessary for diminishing the error by a given factor.

REMARK 5.4. In requirement (P1) of Section 4.4 we have included the independency of the condition number from the time step τ. Thus we get an error estimate which behaves *uniformly* well with respect to τ. This feature has been an important demand on an error estimator.

6. The Multilevel Preconditioner

As we have seen, both — multilevel iterative solution of the linear system (4.7) and the error estimation — demands to construct a preconditioner, which obeys the requirements (P1)–(P4) of Section 4.4. This section is devoted to construct such a preconditioner for piecewise linear as well as for piecewise quadratic elements. These two FEM spaces will take the role of $\mathcal{S}^\flat$ resp. $\mathcal{S}^\sharp$ of Section 5.1, thus leading to an error estimator.

6.1. A Preconditioner for Piecewise Linear Elements

We first restrict the discussion to forms $a(\cdot,\cdot)$ which consists only of the principal part, i.e., $q \equiv 0$ and $\zeta \equiv 0$. Thus there is no Helmholtz term present and the boundary conditions on Γ_C are natural boundary conditions.

However, we do not exclude $\mathrm{mes}(\Gamma_D) = 0$ *in this section.* This will be important for the discussion of the next Section 6.2. Thus the form $a(\cdot,\cdot)$ might even be *not* $H^1_D(\Omega)$–elliptic. In the pure elliptic case, i.e., the case where no term due to discretization in time is present, we would surely have to assume $\mathrm{mes}(\Gamma_D) > 0$. But our operator Λ is $H^1_D(\Omega)$–elliptic for $0 < \tau < \infty$ by Lemma 1.1 anyway.

For the finite element discretization of the *purely elliptic* problem

$$A_j u_j = f \quad \text{on } \mathcal{S}_j$$

two good preconditioners B_j are known:

- the *hierarchical basis preconditioner* due to Yserentant [54],
- the *multilevel nodal basis preconditioner* due to Xu [52, 53, 19].

They are both based on a subspace decomposition of $\mathcal{S}_j$, which means

$$\mathcal{S}_j = \mathcal{S}_0 \oplus \mathcal{V}_1 \oplus \ldots \oplus \mathcal{V}_j, \qquad \mathcal{V}_k \subset \mathcal{S}_k \quad \text{for } 1 \le k \le j. \tag{6.1}$$

Both preconditioners lead to condition numbers $\kappa(B_j A_j) = \mathcal{O}(j^2)$. However, if we handle instead the problem resulting from time discretization of a parabolic problem, we end up with the finite element equation (4.5) which is for $k = j$ equivalent to

$$\Lambda_j u_j = f \quad \text{on } \mathcal{S}_j. \tag{6.2}$$

A straightforward generalization of the preconditioners by just taking Λ_j instead of A_j is not possible since for $\tau \downarrow 0$ the ellipticity constant of the

problem, which seriously enters $\kappa(B_j A_j)$, vanishes. On the other hand for $\tau = 0$ there is no need of preconditioning at all, since then $\Lambda_j = I_j$.

YSERENTANT suggested in [55] a τ–dependent version of his hierarchical basis preconditioner using local *Courant–numbers*, which allow locally to switch between the nodal and the hierarchical basis. However, this yields to a *non–smooth* dependence of the preconditioner on τ which is not desirable in the context of time–discretization, compare the requirement (P3) for a preconditioner as discussed in the last section.

XU suggested in [52] a natural τ–dependent version of the multilevel nodal basis preconditioner depending smoothly on τ. However, he considers only the case of *quasi uniform* triangulations and moreover it is not at all clear whether multiplying this τ–dependent preconditioner by a vector can be realized within $\mathcal{O}(n_j)$ operations, as was required in (P2). This is also true if one uses XU's ideas together with the version of the multilevel nodal basis preconditioner for highly nonuniform triangulations by YSERENTANT [56].

However, by some modifications of XU's and YSERENTANT's constructions it is possible to overcome the above mentioned difficulties as will be shown in this section. Besides that we intend to clarify some aspects of their original constructions.

6.1.1. *A Preconditioner Based on an Orthogonal Splitting of the Finite Element Spaces*

XU specifies the subspace decomposition (6.1) as follows

$$\mathcal{V}_k = (\pi_k - \pi_{k-1})\mathcal{S}_j \qquad \text{for } 1 \leq k \leq j,$$

thus ending up with an *orthogonal* decomposition. His main discovery was now that *on each $\mathcal{V}_k$ the operator A_k spectrally behaves like a constant.*

This means that the symmetric positive definite operator

$$B_j^{-1} = A_0\pi_0 + \sum_{k=1}^{j} \alpha_k(\pi_k - \pi_{k-1})$$

is spectrally close to A, if the constants $\alpha_k \geq 0$ are chosen in a correct manner. XU chose $\alpha_k = \varrho(A_k)$, the spectral radius of A_k, and could show that $\kappa(B_j A_j) = \mathcal{O}(j^2)$ if certain assumptions are satisfied. In the case of quasi uniform triangulations these assumptions are valid and the spectral radii are computable, whereas in the case of highly nonuniform triangulations this approach for determinating the coefficients is not at all easy to pursue. However, YSERENTANT [56] was able by choosing generically

$$\alpha_k = 4^k \tag{6.3}$$

to prove the following Lemma.

LEMMA 6.1. (YSERENTANT [56, Theorem 4.6]) *There are positive constants K_0 and K_1 with*

$$\text{i)} \quad \mu_0(B_j^{-1}u,u) \le (A_j u,u) \le \mu_1(B_j^{-1}u,u), \quad \textit{where}$$

$$\text{ii)} \quad \mu_0 = \frac{\delta}{\Delta}\frac{K_0}{j+1},$$

$$\text{iii)} \quad \mu_1 = \Delta K_1(j+1),$$

for all functions $u \in \mathcal{S}_j$. Furthermore $\mu_0 \le 1 \le \mu_1$ holds. The constants δ, Δ as introduced in assumption 4 of Section 1.1 describe the coefficient matrix a_{ik} of the elliptic operator $A(x,D)$, whereas the constants K_0, K_1 depend only on the local geometry of the initial triangulation $\mathcal{T}_0$ and they are independent of the maximal depth j of the final triangulation.

REMARK 6.1. This Lemma is also valid for the case $\text{mes}(\Gamma_D) = 0$. YSERENTANT assumes in his paper $\text{mes}(\Gamma_D) > 0$, as natural for the elliptic problem, but this assumption nowhere enters the proof of Lemma 6.1. Surely then A_j and B_j^{-1} will only be positive *semi*–definite and

$$\|\cdot\|_{A_j} = (A_j\cdot,\cdot)^{1/2}, \|\cdot\|_{B_j^{-1}} = (B_j^{-1}\cdot,\cdot)^{1/2}$$

will only be seminorms then. According to the above Lemma 6.1 these seminorms must have the same null–spaces, a fact which can also be seen by the $H^1(\Omega)$ Poincaré inequality. This inequality specifies the null–space as $\text{span}\{1\} \subset \mathcal{S}_j$.

We also make the choice (6.3) and turn now to the problem of a τ–dependent preconditioning of (6.2). By the symmetry of B_j^{-1} and A_j we get that the symmetric positive *definite* operator

$$\Theta_j^{-1} = \frac{1}{1+\tau} I_j + \frac{\tau}{1+\tau} B_j^{-1}$$

should be spectrally close to Λ_j. Since $I_j = \pi_0 + \sum_{k=1}^{j}(\pi_k - \pi_{k-1})$ the representation

$$\Theta_j^{-1} = \Lambda_0\pi_0 + \sum_{k=1}^{j} \lambda_k(\pi_k - \pi_{k-1}) \tag{6.4}$$

holds, where

$$\lambda_k = \frac{1+\tau\alpha_k}{1+\tau}. \tag{6.5}$$

COROLLARY 6.1. *The following inequalities hold for all functions* $u \in \mathcal{S}_j$*:*

$$\mu_0(\Theta_j^{-1}u, u) \leq (\Lambda_j u, u) \leq \mu_1(\Theta_j^{-1}u, u).$$

The constants μ_0, μ_1 *are taken from Lemma 6.1.*

Proof. Follows directly from Lemma 6.1, representation (6.4) and Lemma 5.1. ∎

We observe that Θ_j^{-1} is in fact a *direct* sum corresponding to the decomposition (6.1):

$$\Theta_j^{-1} = \Lambda_0 \oplus \bigoplus_{k=1}^{j} \lambda_k \iota_k,$$

where ι_k denotes the identity on the space $\mathcal{V}_k$. Hence

$$\Theta_j = \Lambda_0^{-1} \oplus \bigoplus_{k=1}^{j} \lambda_k^{-1} \iota_k$$

corresponding to (6.1), which is

$$\Theta_j = \Lambda_0^{-1}\pi_0 + \sum_{k=1}^{j} \lambda_k^{-1}(\pi_k - \pi_{k-1})$$

on $\mathcal{S}_j$. Next we are interested in representing Θ_j as a sum of the projections π_k with *positive* coefficients. This makes difficulties for the projection π_0. Because of the relation

$$\Lambda_0^{-1}\pi_0 + \frac{\tau}{1+\tau} A_0 \Lambda_0^{-1}\pi_0 = \frac{\tau}{1+\tau}\Lambda_0^{-1}\pi_0 + \pi_0$$

we can remedy this difficulty by adding the symmetric positive semi-definite operator

$$\frac{\tau}{1+\tau} A_0 \Lambda_0^{-1}\pi_0$$

to Θ_j. Thus we introduce the operator

$$\begin{aligned} \bar{\Theta}_j &= \Theta_j + \frac{\tau}{1+\tau} A_0 \Lambda_0^{-1}\pi_0 \\ &= \frac{\tau}{1+\tau}\Lambda_0^{-1}\pi_0 + \sum_{k=0}^{j} \vartheta_k \pi_k, \end{aligned}$$

with

$$0 \leq \vartheta_k = \begin{cases} \lambda_k^{-1} - \lambda_{k+1}^{-1} & \text{if } k < j \\ \lambda_j^{-1} & \text{if } k = j, \end{cases} \tag{6.6}$$

where $0 \leq k \leq j$. The change of Θ_j into $\bar{\Theta}_j$ does not change the quality of preconditioning, since both operators are spectrally equivalent in fact.

LEMMA 6.2. *The following inequalities hold for all* $u \in \mathcal{S}_j$

$$(\Theta_j u, u) \leq (\bar{\Theta}_j u, u) \leq (1 + \Delta K_1)(\Theta_j u, u).$$

Here K_1 *denotes the constant of Lemma 6.1.*

Proof. The left inequality is obvious since the added operator is symmetric positive semi–definite. By using the inverse inequality Lemma 3.3 of YSERENTANT [56] we get for $u \in \mathcal{S}_j$

$$\begin{aligned}
\|\pi_0 u\|_0^2 &= \|\Lambda_0^{-1/2}\pi_0 u\|_{\Lambda_0}^2 \\
&= \frac{1}{1+\tau}\left(\|\Lambda_0^{-1/2}\pi_0 u\|_0^2 + \tau\|\Lambda_0^{-1/2}\pi_0 u\|_{H^1}^2\right) \\
&\leq \frac{1}{1+\tau}(1+\tau\Delta K_1)\,\|\Lambda_0^{-1/2}\pi_0 u\|_0^2 \\
&\leq \Delta K_1 \|\Lambda_0^{-1/2}\pi_0 u\|_0^2,
\end{aligned}$$

since $\Delta K_1 \geq 1$. Here the the constant K_1 of Lemma 6.1 is exactly the constant of the inverse inequality which only depends on the local shape geometry of the initial triangulation $\mathcal{T}_0$. Hence

$$(\pi_0 u, u) \leq \Delta K_1 (\Lambda_0^{-1}\pi_0 u, u).$$

Therefore we get

$$\left(\frac{\tau}{1+\tau}\Lambda_0^{-1}\pi_0 u, u\right) + (\pi_0 u, u) \leq (1 + \Delta K_1)(\Lambda_0^{-1}\pi_0 u, u),$$

which in turn implies the right inequality. ∎

6.1.2. *A Computationally Available Spectrally Equivalent Preconditioner*

However, our computational problem *is not* problem (6.2), but as we have seen in Section 4.4 problem (4.8). Identification of problems (6.2) and (4.8) would mean to compute $\mathcal{I}_j^{-1}$, i.e in the natural representation of $(\mathcal{S}_j, \mathcal{S}_j^*)$ to invert the *mass matrix*, a problem of the same complexity as (6.2) itself. Hence it would be far more desirable to get a cheap and easy representable expression for $\bar{\Theta}_j\mathcal{I}_j^{-1}$ to fulfill requirement (P4). However, by Lemma 4.2 we obtain

$$\pi_k \mathcal{I}_j^{-1} u^* = \mathcal{I}_k^{-1}(u^*|_{S_k}) \tag{6.7}$$

for $u^* \in \mathcal{S}_j$, $k \leq j$, which means that we can compute $\pi_k\mathcal{I}_j^{-1}$ only by inverting $\mathcal{I}_k$, i.e., a mass matrix of dimension n_k. Thus we are led to replace $\mathcal{I}_k$ by

an easily invertible $\hat{\mathcal{I}}_k$, the duality map with respect to a new inner product $(\cdot,\cdot)_k$ on $\mathcal{S}_k$. A rather simple possibility of inverting $\hat{\mathcal{I}}_k$ is given when $\hat{\mathcal{I}}_k$ is a diagonal matrix in the natural representation. This requires that the nodal basis functions of Γ_k are mutually *orthogonal* with respect to the new inner product, i.e.,

$$(u,\psi)_k = (\psi,\psi)_k u(x_\psi) = (\psi,\psi)_k \psi_*(u) \tag{6.8}$$

for $\psi \in \Gamma_k$, $u \in \mathcal{S}_k$. Thus the new inner product $(\cdot,\cdot)_k$ has to be a weighted Euclidian product in the basis Γ_k. We now exploit the advantage of the usage of $\hat{\mathcal{I}}_k$ by defining an operator $\hat{\pi}_k : \mathcal{S}_j \to \mathcal{S}_k$ through

$$\hat{\pi}_k \mathcal{I}_j^{-1} u^* = \hat{\mathcal{I}}_k^{-1}(u^*|_{\mathcal{S}_k}), \tag{6.9}$$

in analogy to relation (6.7). Replacement of π_k by $\hat{\pi}_k$ in the preconditioner would only be reasonable if these operators are spectrally equivalent. Hence the remaining degrees of freedom in the new inner product, the weights, are chosen in a way that the new inner product resembles the L^2 inner product as much as possible, which yields us to the construction of a *discrete L^2 inner product* by using a quadrature rule with nodes in the vertices of $\mathcal{T}_j$, i.e., for $u, v \in \mathcal{S}_k$

$$(u,v)_k = \frac{1}{3} \sum_{T \in \mathcal{T}_k} |T| \sum_{x \in \mathcal{N}_k \cap T} (uv)(x).$$

This discrete L^2 product satisfies the following stability property yielding the spectral equivalence mentioned above.

LEMMA 6.3. *The inequalities*

$$(u,u) \le (u,u)_k \le 4(u,u)$$

hold for all $u \in \mathcal{S}_k$.

Proof. For $T \in \mathcal{T}_k$ let $\varphi_T : T_E \to T$ be the affine transformation which maps the unit triangle $T_E = \{(x_1,x_2) |\ x_1, x_2 \ge 0, x_1 + x_2 = 1\}$ vertex by vertex to T. The well known integral transformation theorem gives

$$\int_T u^2(x)dx = \int_{T_E} u^2(\varphi_T(\xi)) |\det D\varphi_T| d\xi,$$

where

$$|\det D\varphi_T| = \frac{|T|}{|T_E|} = 2|T|.$$

On the three–dimensional space $\mathcal{S}_{T_E} = \{\hat{u} |\ \hat{u} \text{ linear on } T_E\}$ the norms $\|\cdot\|_{0;T_E}$ and $\|\cdot\|_{\mathcal{N}_E}$ are equivalent, where

$$\|\hat{u}\|^2_{\mathcal{N}_E} = \hat{u}^2(0,0) + \hat{u}^2(0,1) + \hat{u}^2(1,0).$$

Thus there are positive constants ν_0, ν_1 with

$$\nu_0 \|\hat{u}\|^2_{0;T_E} \leq \|\hat{u}\|^2_{\mathcal{N}_E} \leq \nu_1 \|\hat{u}\|^2_{0;T_E}$$

for all $\hat{u} \in \mathcal{S}_{T_E}$. A simple direct computation shows that $\nu_0 = 6$ and $\nu_1 = 24$ are the best possible constants. With $\hat{u} = u \circ \varphi_T$ we therefore get

$$6\|u\|^2_{0;T} \leq 2|T| \sum_{x \in \mathcal{N}_k \cap T} u^2(x) \leq 24\|u\|^2_{0;T}.$$

Summing over all $T \in \mathcal{T}_k$ gives

$$6\|u\|^2_0 \leq 2 \sum_{T \in \mathcal{T}_k} |T| \sum_{x \in \mathcal{N}_k \cap T} u^2(x) \leq 24\|u\|^2_0.$$

Division by 6 gives the assertion. ∎

REMARK 6.2. We have included the proof which consists of standard arguments because one can find in the literature the *wrong* statement that the constants in the inequalities, which are never specified, depend on a lower bound for the interior angles of the triangles $T \in \mathcal{T}_k$.

The next definition is just an equivalent formulation of (6.9) and is due to XU.

DEFINITION 6.1. *The L^2 quasi–projection with respect to $(u,v)_k$ is given by the operator*

$$\hat{\pi}_k : L^2(\Omega) \to \mathcal{S}_k$$

for which

$$(\hat{\pi}_k u, v)_k = (u,v) \quad \text{for } u \in L^2(\Omega), v \in \mathcal{S}_k.$$

The next Lemma states that we have reached the desired properties for the L^2 quasi–projections $\hat{\pi}_k$.

LEMMA 6.4. *The L^2 quasi–projection $\hat{\pi}_k$ is explicitly given as*

$$\hat{\pi}_k u = \sum_{\psi \in \Gamma_k} \frac{(u,\psi)}{(1,\psi)} \psi \tag{6.10}$$

for $u \in L^2(\Omega)$. We further have for all $u \in \mathcal{S}_j$

$$(\hat{\pi}_k u, u) \leq (\pi_k u, u) \leq 4(\hat{\pi}_k u, u). \tag{6.11}$$

Thus the L^2 quasi–projections $\hat{\pi}_k$ and the L^2 projections π_k are spectrally equivalent, uniformly with respect to k. Finally the operator $\pi_k^* = \hat{\pi}_k \mathcal{I}_j^{-1} : \mathcal{S}_j^* \to \mathcal{S}_k$ has the representation

$$\pi_k^* u^* = \sum_{\psi \in \Gamma_k} \frac{u^*(\psi)}{(1,\psi)} \psi \tag{6.12}$$

for $u^* \in \mathcal{S}_j^*$.

Proof. Equation (6.8) gives for $u \in L^2(\Omega)$ and $\psi \in \Gamma_k$ that

$$(\hat{\pi}_k u, \psi)_k = (\psi, \psi)_k \; \hat{\pi}_k u(x_\psi).$$

By definition of $(\cdot,\cdot)_k$ we get

$$\begin{aligned} (\psi,\psi)_k &= \frac{1}{3} \sum_{x_\psi \in T \in \mathcal{T}_k} |T| \\ &= \frac{1}{3} |\text{supp } \psi| \\ &= (1,\psi). \end{aligned}$$

Hence the definition of the $\hat{\pi}_k$ gives

$$\hat{\pi}_k u(x_\psi) = \frac{(\hat{\pi}_k u, \psi)_k}{(\psi,\psi)_k} = \frac{(u,\psi)}{(1,\psi)}.$$

Therefore

$$\begin{aligned} \hat{\pi}_k u &= \sum_{\psi \in \Gamma_k} (\hat{\pi}_k u)(x_\psi) \psi \\ &= \sum_{\psi \in \Gamma_k} \frac{(u,\psi)}{(1,\psi)} \psi, \end{aligned}$$

which is (6.10). Next we define the operator $\sigma_k : \mathcal{S}_k \to \mathcal{S}_k$ such that

$$(\sigma_k u, v) = (u, v)_k$$

for all $u, v \in \mathcal{S}_k$. It is straightforward to check that

$$\hat{\pi}_k|_{\mathcal{S}_k} = \sigma_k^{-1}.$$

Lemma 6.3 states

$$(u,u) \leq (\sigma_k u, u) \leq 4(u,u)$$

for all $u \in \mathcal{S}_k$, hence by Lemma 5.1 the relation (6.11) for $u \in \mathcal{S}_k$. Replacing u by $\pi_k u$ we get (6.11) because of $\hat{\pi}_k \pi_k = \hat{\pi}_k$. Relation (6.12) is finally a direct consequence of (6.10). ∎

REMARK 6.3. The operators M_k of YSERENTANT [56] are just the L^2 quasi–projections $\hat{\pi}_k$; XU [52] denotes them by Π_k.

In view of Theorem 5.2 we take $2\hat{\pi}_k$ to replace π_k in the preconditioner $\bar{\Theta}_j$ and therefore reach the following preconditioner of Λ_j

$$\hat{\Theta}_j = \frac{\tau}{1+\tau}\Lambda_0^{-1}\pi_0 + 2\sum_{k=0}^{j} \vartheta_k \hat{\pi}_k, \tag{6.13}$$

which is spectrally equivalent to $\bar{\Theta}_j$ and for which $\hat{\Theta}_j \mathcal{I}_j^{-1}$ is computational available without inversion of the mass matrix.

COROLLARY 6.2. *The following inequalities hold for all functions $u \in \mathcal{S}_j$:*

$$\frac{1}{2}(\hat{\Theta}_j u, u) \leq (\bar{\Theta}_j u, u) \leq 2(\hat{\Theta}_j u, u).$$

Proof. Follows directly from Lemma 6.4. ∎

6.1.3. Reduction of the Number of Terms

However, the realization of $\hat{\Theta}_j \mathcal{I}_j^{-1} u^*$ for $u^* \in \mathcal{S}_j^*$ as suggested by the representation (6.13) would need at least

$$\mathcal{O}(\sum_{k=1}^{j} n_k)$$

operations since every $\hat{\pi}_k$ needs $\#\Gamma_k = n_k$ summations. For non–geometrical progression of the n_k, which we did not exclude, $\sum_{k=1}^{j} n_j$ will not be $\mathcal{O}(n_j)$ as desired, in the case of pure arithmetical progression it is even $\mathcal{O}(n_j^2)$.

For this reason we finally look whether the number of operations can be reduced to an effort of $\mathcal{O}(n_j)$. This objective can be achieved by a proper rearrangement of the terms in (6.13). We get for $u \in \mathcal{S}_j$

$$\begin{aligned}\hat{\Theta}_j u &= \frac{\tau}{1+\tau}\Lambda_0^{-1}\pi_0 u + 2\sum_{k=0}^{j}\vartheta_k \sum_{\psi\in\Gamma_k}\frac{(u,\psi)}{(1,\psi)}\psi \\ &= \frac{\tau}{1+\tau}\Lambda_0^{-1}\pi_0 u + 2\sum_{\psi\in\Psi}\left(\sum_{k\in K_\psi}\vartheta_k\right)\frac{(u,\psi)}{(1,\psi)}\psi \\ &= \frac{\tau}{1+\tau}\Lambda_0^{-1}\pi_0 u + 2\sum_{\psi\in\Psi}\vartheta(\psi)\frac{(u,\psi)}{(1,\psi)}\psi\end{aligned}$$

with

$$\vartheta(\psi) = \sum_{k\in K_\psi}\vartheta_k.$$

These numbers have a simple expression as the following Lemma shows.

LEMMA 6.5. *For $\psi \in \Psi$ we have*

$$\vartheta(\psi) = \begin{cases} \lambda_{k_\psi^0}^{-1} - \lambda_{k_\psi^1+1}^{-1}, & \text{whenever } 0 \le k_\psi^0 \le k_\psi^1 < j \\ \lambda_{k_\psi^0}^{-1}, & \text{whenever } 0 \le k_\psi^0,\ k_\psi^1 = j. \end{cases} \tag{6.14}$$

Proof. Since the set K_ψ of depths in which ψ occurs in the corresponding nodal basis is connected according to Lemma 4.1

$$K_\psi = \{k|\ k_\psi^0 \le k \le k_\psi^1\},$$

we get

$$\vartheta(\psi) = \sum_{k=k_\psi^0}^{k_\psi^1}\vartheta_k,$$

reducing to (6.14) in view of the telescope character of the sum due to the definition (6.6) of the ϑ_k. ∎

Thus our final preconditioner $\Theta_j^* = \hat{\Theta}_j\mathcal{I}_j^{-1}$ is given by

$$\Theta_j^* u^* = \frac{\tau}{1+\tau}\rho_0 u^* + 2\sum_{\psi\in\Psi}\vartheta(\psi)\frac{u^*(\psi)}{(1,\psi)}\psi \quad \text{for all } u^* \in \mathcal{S}_j^*. \tag{6.15}$$

Here we made use of (4.6), the representation of the solution operator of depth 0. Thus our preconditioner is just the sum of the damped FEM–solution on the coarsest triangulation $\mathcal{T}_0$ with right hand side u^* and according to Lemma 4.1 $\#\Psi \leq 2n_j$ additional simple terms. This estimate of the number of terms suggests that the sum of these additional terms can be computed within $\mathcal{O}(n_j)$ operations, which is in fact true, but will be discussed in detail in Section 6.5.

6.1.4. Summary

Summarizing our results so far we can state the main Theorem of this section.

THEOREM 6.1. *For all functions $u \in \mathcal{S}_j$ and for all numbers $\tau \geq 0$ the following inequalities hold*

$$\hat{\mu}_0(\Lambda_j^{-1}u, u) \leq (\hat{\Theta}_j u, u) \leq \hat{\mu}_1(\Lambda_j^{-1}u, u)$$

where

$$\hat{\mu}_0 = \frac{\mu_0}{2} \quad \text{and} \quad \hat{\mu}_1 = 2\mu_1(1 + \Delta K_1).$$

The constants μ_0, μ_1, K_1 are from Lemma 6.1.

Moreover we have

$$\kappa\left(\hat{\Theta}_j \Lambda_j\right) \leq 4(1 + \Delta K_1)\frac{\mu_1}{\mu_0} = \mathcal{O}(j^2),$$

independently of τ.

Specification of the case $\tau = 0$ gives for all $u \in \mathcal{S}_j$

$$\frac{1}{2}\,(\Lambda_j^{-1}u, u)\Big|_{\tau=0} \leq (\hat{\Theta}_j u, u)\Big|_{\tau=0} \leq 2\,(\Lambda_j^{-1}u, u)\Big|_{\tau=0}.$$

Here we get

$$\kappa\left(\hat{\Theta}_j \Lambda_j\right)\Big|_{\tau=0} \leq 4.$$

Proof. Follows from Corollary 6.1, Lemma 6.2 and Corollary 6.2. The estimate of the condition number follows from Lemma 5.1. The case $\tau = 0$ can be treated by Lemma 6.4 since then $\hat{\Theta}_j|_{\tau=0} = 2\hat{\pi}_j$. ∎

REMARK 6.4. In view of the requirements (P1)–(P4) for a preconditioner, we can state the results of this section as follows. The matrix B_τ given by Θ_j^* in the natural representation of $(\mathcal{S}_j, \mathcal{S}_j^*)$ fulfills requirements (P1) and (P3) whereas Θ_j^* itself fulfills requirement (P4).

REMARK 6.5. The case $\tau = 0$ is the preconditioning of $\mathcal{I}_j$, the operator which is represented by the *mass matrix* in the natural bases Γ_j, Γ_j^*. Here Theorem 6.1 states that $\Theta_j^* = 2\pi_j^*$ is a natural choice of a preconditioner. But the operator $2\pi_j^*$ is given by

$$(2\pi_j^*)u^* = \sum_{\psi \in \Gamma_j} \frac{u^*(\psi)}{(\psi,\psi)}\psi$$

since $(\psi,\psi) = \frac{1}{6}|\text{supp } \psi| = \frac{1}{2}(1,\psi)$ for $\psi \in \Gamma_j$. Hence $2\pi_j^*$ is represented in the natural bases by the matrix D^{-1} where $\mathrm{D} = \text{diag}(m_{11},\ldots,m_{n_jn_j})$ denotes the diagonal of the mass matrix M. Thus Theorem 6.1 gives in passing the result

COROLLARY 6.3. *The following holds for the diagonal preconditioner of the mass matrix:*

$$\kappa(\mathrm{D}^{-1}\mathrm{M}) \leq 4$$

and

$$\sigma(\mathrm{D}^{-1}\mathrm{M}) \subset [\frac{1}{2}, 2].$$

WATHEN [49] proved these results with a different technique.

REMARK 6.6. Since $\Lambda_j \to A_j$ for $\tau \to \infty$ and in turn, assuming now $\text{mes}(\Gamma_D) > 0$,

$$\hat{\Theta}_j \to \hat{B}_j = A_0^{-1}\pi_0 + 2\sum_{k=0}^{j}\left(\alpha_k^{-1} - \alpha_{k+1}^{-1}\right)\hat{\pi}_k,$$

Theorem 6.1 states that $\hat{B}_j$ is a good preconditioner for A_j. In fact the preconditioner

$$C_j = A_0^{-1}\pi_0 + \sum_{k=1}^{j}\alpha_k^{-1}\hat{\pi}_k$$

advocated by YSERENTANT in [55] is spectrally equivalent to $\hat{B}_j$:

$$(C_j u, u) \leq (\hat{B}_j u, u) \leq \frac{3}{2}(1 + \Delta K_1)(C_j u, u).$$

Comparison with Remark 6.5 shows that Θ_j^* *provides a continuous transition from the diagonal preconditioner of the mass matrix to the multilevel nodal basis preconditioner of the stiffness matrix.*

REMARK 6.7. Using Theorem 10.7 of XU [52] we can prove *at least* for the case of quasi–uniform triangulations that whenever our elliptic problem is $H^{1+\alpha}(\Omega)$–regular, $\alpha \in]0,1]$, we get

$$\kappa\left(\hat{\Theta}_j \Lambda_j\right) = \mathcal{O}(j^{\min(1/\alpha,2)}),$$

with constants *independent* of $\tau \geq 0$. This result seems to be also true for certain highly nonuniform triangulation as Example 8.5 will show later on.

6.2. EXTENSION TO HELMHOLTZ TERMS AND GENERAL CAUCHY BOUNDARY CONDITIONS

Until now we considered the bilinear form $a(\cdot,\cdot)$ to consist only of the principal part, i.e., $q \equiv 0$ and $\zeta \equiv 0$. In this section we get rid of this restriction.

Denote by

$$\Lambda_P = \frac{1}{1+\tau} I + \frac{\tau}{1+\tau} A_P$$

the operator belonging to the principal part $a_P(\cdot,\cdot)$ of $a(\cdot,\cdot)$, i.e., exactly the operator which we considered in the last Section 6.1.

First assume $q \not\equiv 0$ but $\zeta \equiv 0$, which means that a Helmholtz term is present, but the Cauchy boundary conditions are still natural boundary conditions.

We have to consider the two cases

Case I. $\text{mes}(\Gamma_D) > 0$

Case II. $q_{\min} > 0$,

either of which makes the form $a(\cdot,\cdot)$ $H^1_D(\Omega)$–elliptic by Lemma 1.1.

Case I and II will give two different versions of preconditioning, which will be discussed next. For the discussion of Case II it is essential that we did not exclude $\text{mes}(\Gamma_D) = 0$ in the last Section 6.1.

6.2.1. Version I of a Helmholtz Preconditioner

By Lemma 1.1.iii Case I implies the $H^1_D(\Omega)$–ellipticity of the form $a_P(\cdot,\cdot)$, i.e., there is a constant $c_{1P} > 0$ such that

$$a_P(u,u) \geq c_{1P}\|u\|_1^2$$

for $u \in H_D^1(\Omega)$. Thus we can estimate for $u \in H_D^1(\Omega)$

$$\begin{aligned} a_P(u,u) &\leq a(u,u) = a_P(u,u) + (q\,u,u) \\ &\leq a_P(u,u) + q_{\max}\|u\|_0^2 \\ &\leq \left(1 + \frac{q_{\max}}{c_{1P}}\right) a_P(u,u). \end{aligned}$$

This inequalities yield

$$((\Lambda_P)_j u, u) \leq (\Lambda_j u, u) \leq \left(1 + \frac{q_{\max}}{c_{1P}}\right) ((\Lambda_P)_j u, u)$$

for $u \in \mathcal{S}_j$. Hence using the preconditioner $\hat{\Theta}_j$, which was derived in Section 6.1, we obtain by Theorem 6.1

$$\kappa\left(\hat{\Theta}_j \Lambda_j\right) \leq 4\left(1 + \frac{q_{\max}}{c_{1P}}\right)(1 + \Delta K_1)\frac{\mu_1}{\mu_0},$$

which has the desired form.

REMARK 6.8. For $\Gamma_D = \partial\Omega$, $\Gamma_C = \emptyset$ we can estimate by Lemma 1.1.i as follows: For $u \in H_0^1(\Omega)$ we obtain

$$\begin{aligned} a(u,u) &= a_P(u,u) + (q\,u,u) \\ &\leq \left(1 + \frac{q_{\max} d_\Omega^2}{2\delta}\right) a_P(u,u), \end{aligned}$$

which yields

$$\kappa\left(\hat{\Theta}_j \Lambda_j\right) \leq 4\left(1 + \frac{q_{\max} d_\Omega^2}{2\delta}\right)(1 + \Delta K_1)\frac{\mu_1}{\mu_0}.$$

6.2.2. Version II of a Helmholtz Preconditioner

In Case II we proceed as follows: Defining for some

$$0 < q_{\min} \leq \bar{q} \leq q_{\max}$$

the operator

$$\tilde{\Lambda} = \frac{1 + \bar{q}\tau}{1 + \tau} I + \frac{\tau}{1 + \tau} A_P,$$

yields the estimate

$$\frac{q_{\min}}{\bar{q}}(\tilde{\Lambda}_j u, u) \leq (\Lambda_j u, u) \leq \frac{q_{\max}}{\bar{q}}(\tilde{\Lambda}_j u, u) \tag{6.16}$$

for $u \in \mathcal{S}_j$. Putting

$$\tilde{\lambda}_k = \frac{1 + \tau(\bar{q} + \alpha_k)}{1 + \tau}$$

and replacing the λ_k of (6.5) in the derivation of $\hat{\Theta}_j$ by this $\tilde{\lambda}_k$ we obtain a preconditioner $\tilde{\Theta}_j$ of $\tilde{\Lambda}_j$ with

$$\kappa\left(\tilde{\Theta}_j \tilde{\Lambda}_j\right) \leq 4(1 + \Delta K_1)\frac{\mu_1}{\mu_0},$$

which is the same bound as in Theorem 6.1 for $\kappa\left(\hat{\Theta}_j(\Lambda_P)_j\right)$. This works since we never used in Section 6.1 the specific form of the λ_k, but only the fact that

$$\lambda_{k+1} \geq \lambda_k > 0.$$

By the above inequality we can estimate

$$\kappa\left(\tilde{\Theta}_j \Lambda_j\right) \leq 4\frac{q_{\max}}{q_{\min}}(1 + \Delta K_1)\frac{\mu_1}{\mu_0},$$

which distinguishes $\tilde{\Theta}_j$ as preconditioner of the required form. The actual choice of $\bar{q}$ should be made in order to gauge the estimate (6.16) as

$$q_{\min}/\bar{q} = \bar{q}/q_{\max},$$

i.e.,

$$\bar{q} = \sqrt{q_{\min} q_{\max}},$$

the geometric mean of the two bounds.

6.2.3. A Case of Doubt: Case I and Case II

If both cases, I and II, are present the question arises which version should be taken ? Algorithmically both versions are quite related since Version I can be interpreted as the case $\bar{q} = 0$ of Version II, which gives $\tilde{\Theta}_j = \hat{\Theta}_j$. So, what value of $\bar{q}$ shall be taken ?

For $\Gamma_C = \emptyset$ we can answer this question definitely by means of Remark 6.8: Version II should be taken if

$$q_{\min} > \frac{2\delta}{d_\Omega^2}. \tag{6.17}$$

An example of such a decision will be given in Section 9.2.3.

6.2.4. *The Case of General Cauchy Boundary Conditions*

Now we consider $\zeta \not\equiv 0$ and $\Gamma_C \neq \emptyset$, assuming one of the above cases: I or II.

Hence we have

$$a(u,v) = a_H(u,v) + \int_{\Gamma_C} \zeta\, uv\, d\sigma$$

for all $u, v \in H^1_D(\Omega)$. The — due to Case I or II $H^1_D(\Omega)$–elliptic — form $a_H(\cdot,\cdot)$ with

$$a_H(u,u) \geq c_{1H}\|u\|_1^2$$

for $u \in H^1_D(\Omega)$ induces an operator Λ_H for which we know by Sections 6.1 and 6.2.1/6.2.2 a preconditioner $(\Theta_H)_j$. Since we may estimate for $u \in H^1_D(\Omega)$ by the continuity of the trace operator $H^1(\Omega) \to H^{1/2}(\partial\Omega)$

$$\begin{aligned}
a_H(u,u) \;&\leq\; a(u,u) = a_H(u,u) + \int_{\Gamma_C} \zeta\, u^2\, d\sigma \\
&\leq\; a_H(u,u) + \zeta_{\max}\|u\|^2_{L^2(\Gamma_C)} \\
&\leq\; a_H(u,u) + \zeta_{\max}\|u\|^2_{H^{1/2}(\partial\Omega)} \\
&\leq\; a_H(u,u) + \zeta_{\max} K_{\text{trace}}\|u\|_1^2 \\
&\leq\; \left(1 + \frac{\zeta_{\max} K_{\text{trace}}}{c_{1H}}\right) a_H(u,u),
\end{aligned}$$

we obtain the estimate

$$\kappa\left((\Theta_H)_j \Lambda_j\right) \leq \left(1 + \frac{\zeta_{\max} K_{\text{trace}}}{c_{1H}}\right) \kappa\left((\Theta_H)_j (\Lambda_H)_j\right).$$

This tells us that $(\Theta_H)_j$ should be an as good preconditioner as in the case $\zeta \equiv 0$ for moderately behaving ζ.

6.3. A Preconditioner for Piecewise Quadratic Elements

We assume that Case I or Case II of Section 6.2 is true.

As introduced in Section 4.5 we consider the space of piecewise quadratic elements $\mathcal{S}_Q$ on the triangulation $\mathcal{T}_j$. The hierarchical splitting

$$\mathcal{S}_Q = \mathcal{S}_j \oplus \mathcal{V}_Q$$

shows that there is a *unique* decomposition of each $u \in \mathcal{S}_Q$ into

$$u = u_L + u_Q,$$

where $u_L \in \mathcal{S}_j$ and $u_Q \in \mathcal{V}_Q$. Based on that decomposition we introduce norms — corresponding to some inner products on $\mathcal{S}_Q$ — which are generated by the L^2–inner product on $\mathcal{S}_j$ and which are a scaled Euclidian norm on $\mathcal{V}_Q$: For $u \in \mathcal{S}_Q$ we define

$$
\begin{aligned}
&\text{i)} \quad \|u\|_{Q;0}^2 &&= \|u_L\|_0^2 + \sum_{\psi \in \Gamma_Q} (\psi, \psi) |u_Q(x_\psi)|^2, \\
&\text{ii)} \quad \|u\|_{Q;1}^2 &&= \|u_L\|_{\dot{H}^1}^2 + \sum_{\psi \in \Gamma_Q} (A_Q \psi, \psi)\, |u_Q(x_\psi)|^2, \\
&\text{iii)} \quad \|u\|_{Q;\Lambda}^2 &&= \|u_L\|_\Lambda^2 + \sum_{\psi \in \Gamma_Q} (\Lambda_Q \psi, \psi)\, |u_Q(x_\psi)|^2.
\end{aligned} \tag{6.18}
$$

By construction $\|u\|_{Q;\Lambda}^2$ is a convex combination of $\|u\|_{Q;0}^2$ and $\|u\|_{Q;1}^2$:

$$
\|u\|_{Q;\Lambda}^2 = \frac{1}{1+\tau} \|u\|_{Q;0}^2 + \frac{\tau}{1+\tau} \|u\|_{Q;1}^2. \tag{6.19}
$$

The following lemmas exhibit, that the norms (6.18) are in fact equivalent to $\|\cdot\|_0$, $\|\cdot\|_{\dot{H}^1}$ resp. $\|\cdot\|_\Lambda$ on $\mathcal{S}_Q$.

LEMMA 6.6. *For $u \in \mathcal{S}_Q$ we have*

$$
\gamma_0 \|u\|_{Q;0}^2 \le \|u\|_0^2 \le \gamma_1 \|u\|_{Q;0}^2,
$$

with

$$
\gamma_0 = \frac{1}{4} \left(6 - \sqrt{34}\right) \doteq 0.042
$$

and

$$
\gamma_1 = \frac{1}{4} \left(6 + \sqrt{34}\right) \doteq 2.96.
$$

These are the best constants in general.

Proof. The proof runs along the lines of the proof of Lemma 6.3. The computation of the constants is tedious but not very subtle. ∎

LEMMA 6.7. *There exists a positive constant $\bar{\gamma}_0$ such that*

$$
\bar{\gamma}_0 \|u\|_{Q;1}^2 \le \|u\|_{\dot{H}^1}^2 \le 4 \|u\|_{Q;1}^2,
$$

for $u \in \mathcal{S}_Q$. The constant $\bar{\gamma}_0$ depends only on a lower bound of the interior angles of $\mathcal{T}_0$ and on the bilinear form $a(\cdot, \cdot)$.

Proof. The assertion is proven by the arguments preceding [25, (2.22)] and the arguments of Section 6.2.4. ∎

COROLLARY 6.4. *For $u \in \mathcal{S}_Q$ we get*

$$\hat{\gamma}_0 \|u\|^2_{Q;\Lambda} \leq \|u\|^2_\Lambda \leq 4\|u\|^2_{Q;\Lambda},$$

with $\hat{\gamma}_0 = \min(\gamma_0, \bar{\gamma}_0)$, where γ_0,$\bar{\gamma}_0$ are the constants of the preceding two Lemmas.

Proof. Follows directly from the two foregoing Lemmas and the convex combination (6.19). ∎

REMARK 6.9. In *one* spatial dimension we have a complete knowledge of the involved constants. In fact we obtain for $u \in \mathcal{S}_Q$

$$\tilde{\gamma}_0 \|u\|^2_{Q;\Lambda} \leq \|u\|^2_\Lambda \leq \tilde{\gamma}_1 \|u\|^2_{Q;\Lambda},$$

with $\tilde{\gamma}_0 = \min(\gamma_0^*, \delta/\Delta)$ and $\tilde{\gamma}_1 = \max(\gamma_1^*, \min(2, \Delta/\delta))$, where

$$\gamma_0^* = \frac{1}{6}\left(6 - \sqrt{30}\right) \doteq 0.087,$$

and

$$\gamma_1^* = \frac{1}{6}\left(6 + \sqrt{30}\right) \doteq 1.91.$$

The proof exploits the fact that in 1D we have $\mathcal{S}_j \perp \mathcal{V}_Q$ with respect to the $\dot{H}^1$–inner product.

Due to the Theorem of Fréchet–Riesz there is a symmetric positive definite operator $H_\Lambda : \mathcal{S}_Q \to \mathcal{S}_Q$ such that

$$(H_\Lambda u, v) = (\Lambda_j u_L, v_L) + \sum_{\psi \in \Gamma_Q} (\Lambda_Q \psi, \psi) u_Q(x_\psi) v_Q(x_\psi),$$

for all $u, v \in \mathcal{S}_Q$, which implies that

$$(H_\Lambda u, u) = \|u\|^2_{Q;\Lambda}.$$

LEMMA 6.8. *The inverse of H_Λ is given by*

$$H_\Lambda^{-1} = \Theta_Q,$$

where

$$\Theta_Q u = \Lambda_j^{-1} \pi_j u + \sum_{\psi \in \Gamma_Q} \frac{(u, \psi)}{(\Lambda_Q \psi, \psi)} \psi, \tag{6.20}$$

for all $u \in \mathcal{S}_Q$.

Proof. Take $u \in \mathcal{S}_Q$.

1. Let $v \in \mathcal{S}_j$. Since $v_Q = 0$ and $v = v_L$ we get

$$\begin{aligned}(H_\Lambda \Theta_Q u, v) &= (\Lambda_j (\Theta_Q u)_L, v) \\ &= \left(\Lambda_j \Lambda_j^{-1} \pi_j u, v\right) \\ &= (\pi_j u, v) \\ &= (u, v).\end{aligned}$$

2. Let $v \in \mathcal{V}_Q$. Since $v_L = 0$ and $v = v_Q$ we get

$$\begin{aligned}(H_\Lambda \Theta_Q u, v) &= \sum_{\psi \in \Gamma_Q} (\Lambda_Q \psi, \psi)(\Theta_Q u)_Q(x_\psi) v(x_\psi) \\ &= \sum_{\psi \in \Gamma_Q} (\Lambda_Q \psi, \psi) \frac{(u, \psi)}{(\Lambda_Q \psi, \psi)} v(x_\psi) \\ &= \left(u, \sum_{\psi \in \Gamma_Q} v(x_\psi) \psi\right) \\ &= (u, v).\end{aligned}$$

3. Thus for $v \in \mathcal{S}_Q$

$$(H_\Lambda \Theta_Q u, v) = (u, v),$$

which implies $H_\Lambda \Theta_Q u = u$.

Since H_Λ is positive definite, the Lemma is proven. ∎

COROLLARY 6.5. *The following inequalities hold for all $u \in \mathcal{S}_Q$:*

$$\hat{\gamma}_0 (\Theta_Q^{-1} u, u) \leq (\Lambda_Q u, u) \leq 4 (\Theta_Q^{-1} u, u).$$

The constant $\hat{\gamma}_0$ is from Corollary 6.4.

Proof. This is simply a restatement of Corollary 6.4, since we observe that $(\Theta_Q^{-1} u, u) = (H_\Lambda u, u) = \|u\|^2_{Q;\Lambda}$. ∎

Thus the operator Θ_Q is a preconditioner of Λ_Q.

For ease of representation we restrict ourselves for the rest of this section to the case of $q \equiv 0$, $\zeta \equiv 0$, which was discussed in Section 6.1. The extension to the general case is obvious by means of Section 6.2.

In view of Section 6.1.2 the representation (6.20) suggests to use the operator $\hat{\Theta}_Q : \mathcal{S}_Q \to \mathcal{S}_Q$, defined by

$$\hat{\Theta}_Q u = \hat{\Theta}_j u + \sum_{\psi \in \Gamma_Q} \frac{(u,\psi)}{(\Lambda_Q \psi, \psi)} \psi,$$

as computationally available preconditioner. Note that the operator $\hat{\Theta}_j$ can be defined *on all* $L^2(\Omega)$.

However, in actual computation we use

$$\begin{aligned} \Theta_Q^* u^* &= \hat{\Theta}_Q \mathcal{I}_Q^{-1} u^* \\ &= \frac{\tau}{1+\tau} \rho_0 u^* + \sum_{\psi \in \Psi} \frac{\vartheta(\psi)}{(\psi,\psi)} u^*(\psi)\psi + \sum_{\psi \in \Gamma_Q} \frac{1}{(\Lambda_Q \psi, \psi)} u^*(\psi)\psi, \end{aligned}$$

where $u^* \in \mathcal{S}_Q^*$.

LEMMA 6.9. *For $u \in \mathcal{S}_Q$ we get*

$$\hat{\mu}_0(\Theta_Q u, u) \leq (\hat{\Theta}_Q u, u) \leq \hat{\mu}_1(\Theta_Q u, u),$$

where $\hat{\mu}_0$, $\hat{\mu}_1$ are from Theorem 6.1.

Proof. Since $\pi_0 = \pi_0 \pi_j$ and $\hat{\pi}_k = \hat{\pi}_k \pi_j$ for $k \leq j$, we have for $u \in \mathcal{S}_Q$

$$\Theta_Q u = \hat{\Theta}_j \pi_j u + \sum_{\psi \in \Gamma_Q} \frac{(u,\psi)}{(\Lambda_Q \psi, \psi)} \psi. \tag{6.21}$$

Therefore Theorem 6.1 yields the assertion. ∎

We summarize our results.

THEOREM 6.2. *For all functions $u \in \mathcal{S}_Q$ and for all numbers $\tau \geq 0$ the following inequalities hold*

$$\mu_0^Q(\Lambda_Q^{-1} u, u) \leq (\hat{\Theta}_Q u, u) \leq \mu_1^Q(\Lambda_Q^{-1} u, u)$$

where $\mu_0^Q = \hat{\gamma}_0 \hat{\mu}_0$ and $\mu_1^Q = 4\hat{\mu}_1$. The constants $\hat{\mu}_0$, $\hat{\mu}_1$ are from Theorem 6.1, the constant $\hat{\gamma}_0$ from Corollary 6.5.

Moreover we have

$$\kappa\left(\hat{\Theta}_Q \Lambda_Q\right) \leq 4 \frac{\hat{\mu}_1}{\hat{\gamma}_0 \hat{\mu}_0} = \mathcal{O}(j^2),$$

independently of τ.

Specification of the case $\tau = 0$ *gives for all* $u \in \mathcal{S}_Q$

$$\gamma_2 \left(\Lambda_Q^{-1} u, u\right)\Big|_{\tau=0} \leq \left(\hat{\Theta}_Q u, u\right)\Big|_{\tau=0} \leq \gamma_3 \left(\Lambda_Q^{-1} u, u\right)\Big|_{\tau=0},$$

with

$$\gamma_2 = 2\left(4 - \sqrt{15}\right) \doteq 0.25$$

and

$$\gamma_3 = 2\left(4 + \sqrt{15}\right) \doteq 15.75.$$

Here we get

$$\kappa\left(\hat{\Theta}_Q \Lambda_Q\right)\Big|_{\tau=0} \leq 62.$$

Proof. Follows from Corollary 6.5 and Lemma 6.9 according to Lemma 5.1. The case $\tau = 0$ follows by arguments similar to the proof of Lemma 6.6. ■

REMARK 6.10. Note that an application of Theorem 6.1 and Lemma 6.6 *alone* would give

$$\kappa\left(\hat{\Theta}_Q \Lambda_Q\right)\Big|_{\tau=0} \leq 4\frac{\gamma_1}{\gamma_0} \doteq 280,$$

which is an overestimation by a factor of 4.5.

6.4. Error Estimation — Specific Considerations

Here we specify the abstract considerations of Section 5 by choosing $\mathcal{S}^\flat = \mathcal{S}_j$ and $\mathcal{S}^\sharp = \mathcal{S}_Q$. Let $u^\sharp \in \mathcal{S}_Q$ be the solution of

$$\Lambda_Q u^\sharp = f_Q$$

and $u^\flat \in \mathcal{S}_j$ be the solution of

$$\Lambda_j u^\flat = f_j.$$

Furthermore let $\hat{u} \in \mathcal{S}_j$ be *arbitrary*. We have to analyze the specific expressions for the estimate of the *linear deviation* (necessary to control the preconditioned CG–iterations)

$$\|u^\flat - \hat{u}\|_\Lambda$$

and the *quadratic deviation* (serving as discretization estimate according to Section 5.1)

$$\|u^\sharp - \hat{u}\|_\Lambda.$$

Finally we have to discuss the β–approximation property of the pair $(\mathcal{S}_j, \mathcal{S}_Q)$.

6.4.1. The Linear and the Quadratic Deviation Estimate

Consider the *linear residual*

$$r^\flat = f_j - \Lambda_j \hat{u}$$

and the *quadratic residual*

$$r^\sharp = f_Q - \Lambda_Q \hat{u}.$$

Because of the structure of Galerkin approximations and $\mathcal{S}_j \subset \mathcal{S}_Q$ we gain

$$r^\flat = \pi_j r^\sharp. \tag{6.22}$$

The abstract discussion of Section 5.2 exhibits the linear–deviation estimate $\|r^\flat\|_{\hat{\Theta}_j}$, where

$$\|r^\flat\|^2_{\hat{\Theta}_j} = (\hat{\Theta}_j r^\flat, r^\flat).$$

Along the same lines we get the quadratic–deviation estimate $\|r^\sharp\|_{\hat{\Theta}_Q}$, where

$$\begin{aligned} \|r^\sharp\|^2_{\hat{\Theta}_Q} &= (\hat{\Theta}_Q r^\sharp, r^\sharp) \\ &= (\hat{\Theta}_j \pi_j r^\sharp, \pi_j r^\sharp) + \sum_{\psi \in \Gamma_Q} \frac{(r^\sharp, \psi)^2}{(\Lambda_Q \psi, \psi)}. \end{aligned}$$

Here we made use of the relation (6.21). By the projection property (6.22) we gain

$$\|r^\sharp\|^2_{\hat{\Theta}_Q} = \|r^\flat\|^2_{\hat{\Theta}_j} + \sum_{\psi \in \Gamma_Q} \eta^2_\psi,$$

where we define

$$\eta_\psi = \frac{|(r^\sharp, \psi)|}{\|\psi\|_\Lambda}$$

for every $\psi \in \Gamma_Q$.

THEOREM 6.3. *For any $\hat{u} \in \mathcal{S}_j$ we have*

$$\|u^\flat - \hat{u}\|_\Lambda = \zeta_\flat \|r^\flat\|_{\hat{\Theta}_j}$$

and

$$\|u^\sharp - \hat{u}\|_\Lambda = \zeta_\sharp \|r^\sharp\|_{\hat{\Theta}_Q}$$

with

$$\zeta_\flat \in \left[\frac{1}{\sqrt{\hat{\mu}_1}}, \frac{1}{\sqrt{\hat{\mu}_0}}\right]$$

and

$$\zeta_\sharp \in \left[\frac{1}{\sqrt{\mu_1^Q}}, \frac{1}{\sqrt{\mu_0^Q}}\right],$$

where the constants $\hat{\mu}_0$, $\hat{\mu}_1$ *are from Theorem 6.1 and* μ_0^Q, μ_1^Q *are from Theorem 6.2. These constants are independent of* $\tau \geq 0$. *Specification of the case* $\tau = 0$ *yields*

$$\zeta_\flat|_{\tau=0} \in \left[\frac{1}{2}\sqrt{2}, \sqrt{2}\right] \subset [0.7, 1.42]$$

and

$$\zeta_\sharp|_{\tau=0} \in \left[\frac{1}{\sqrt{\gamma_3}}, \frac{1}{\sqrt{\gamma_2}}\right] \subset [0.25, 2].$$

In this case the quality indicators κ_ζ *of Section 5.2 turn out to be*

$$\kappa_{\zeta_\flat}|_{\tau=0} \leq 2$$

and

$$\kappa_{\zeta_\sharp}|_{\tau=0} \leq 7.88.$$

Proof. Theorems 5.2, 6.1 and 6.2 yield the assertions. ∎

6.4.2. Refinement–Strategy

The values η_ψ may serve as *indicators* for an edge–oriented refinement–strategy, since there is a one-to-one correspondence between Γ_Q and the edges of $\mathcal{T}_j$ which does not belong to the Dirichlet boundary piece Γ_D. The indicators η_ψ are in fact exactly the same as in [25], in the notation of [25]

$$\eta_\psi = \left(D_{QQ}^{-1/2} r_Q\right)\Big|_{\text{edge containing } x_\psi}.$$

Now, an edge containing x_ψ is marked for local refinement if

$$\eta_\psi \geq \eta_{\text{thresh}}.$$

We favor for the computation of η_{thresh} a procedure due to [4]:

It uses a simple heuristic prediction scheme to forecast what may happen to η_ψ if the edge containing x_ψ is subdivided. This forecast relies on the assumption that *locally*

$$\eta_\psi = c_\psi h_\psi^{\lambda_\psi} \text{ as } h_\psi \to 0.$$

Here h_ψ denotes the length of the edge containing x_ψ. Suppose this edge was generated by subdividing an edge with local error η_ψ^{old}. A simple extrapolation yields

$$\eta_\psi^{\text{new}} = \frac{\eta_\psi^2}{\eta_\psi^{\text{old}}}$$

as prediction for the error after a new subdivision of the edge. Clearly now, we should — in order of equidistributing the error — refine only those edges which have an η_ψ–value above the *largest* predicted *new* η^{new}–value of the virtual next triangulation, hence

$$\eta_{\text{thresh}} = \max_\psi \eta_\psi^{\text{new}}.$$

To avoid a refinement of too many triangles when the estimated error is near the given elliptic tolerance eps, we actually take with

$$\text{cut} = \max_\psi \eta_\psi^{\text{new}}$$

the value

$$\eta_{\text{thresh}} = \max\left(\text{cut}, \frac{\text{eps}}{\epsilon}\sqrt{\eta_{\max}\,\text{cut}}\right),$$

where $\eta_{\max} = \max_\psi \eta_\psi$ and ϵ is the actually estimated error.

This procedure of computing η_{thresh} yields triangulations with far fewer nodal points than the procedure originally proposed in [25]. For detailed comparisons see [26].

6.4.3. *The β–Approximation Property of $(\mathcal{S}_j, \mathcal{S}_Q)$*

Consider first the case of *quasi–uniform* triangulations $\mathcal{T}_j$ with mesh parameter h: For $u \in H^3(\Omega)$ we have

$$\|u - u^\flat\|_\Lambda \leq Ch\|u\|_2 \tag{6.23}$$

and

$$\|u - u^\sharp\|_\Lambda \leq Ch^2\|u\|_3.$$

Because of the best approximation property of finite element solutions and $\mathcal{S}_j \subset \mathcal{S}_Q$ there is an $0 < \beta_h \leq 1$ such that

$$\|u - u^\sharp\|_\Lambda = \beta_h\|u - u^\flat\|_\Lambda.$$

Since the estimate (6.23) is optimal with respect to the power of h, we gain

$$\beta_h = \mathcal{O}(h).$$

DEFINITION 6.2. *The sequence of pairs $(\mathcal{S}_j, \mathcal{S}_Q)_j$ is called to have the asymptotic 0–approximation property if there is a sequence $(\beta_n)_n$ of reals with $0 < \beta_n \leq 1$ and $\lim_{n\to\infty} \beta_n = 0$, such that*

$$\|u - u^\sharp\|_\Lambda \leq \beta_{n_j} \|u - u^\flat\|_\Lambda$$

holds for every j.

Thus the pairs $(\mathcal{S}_j, \mathcal{S}_Q)_j$ belonging to a family of quasi–uniform triangulations have the asymptotic 0–approximation property.

Now consider *highly nonuniform triangulations* $\mathcal{T}_j$. They are assumed to be of type (h, γ, L), a notation due to BABUŠKA et al. [3]. If the multiindex γ underlies the restrictions of [3, Corollary 5.1], which depend on the interior angles of $\partial\Omega$ at the vertices of $\partial\Omega$, we want to call the family of triangulations *adequate*.

For those adequate triangulations [3, Corollary 5.1] states that

$$\|u - u^\flat\|_\Lambda \leq C n_j^{-1/2} \|f\|_0, \tag{6.24}$$

which seems to be the straight generalization of (6.23), but is an entirely nontrivial result proved by using weighted Besov spaces.

In view of this result we *conjecture*

> The pairs $(\mathcal{S}_j, \mathcal{S}_Q)_j$ have the asymptotic 0–approximation property, whenever the triangulations $\mathcal{T}_j$ are adequate.

DEFINITION 6.3. *A triangulation $\mathcal{T}_j$ is called fine, if the pair $(\mathcal{S}_j, \mathcal{S}_Q)$ fulfills the β–approximation property with $\beta \leq \frac{1}{3}\sqrt{6}$.*

The value $\frac{1}{3}\sqrt{6}$ has been chosen in view of Remark 5.1.

The above *heuristic* considerations mainly served the purpose to justify our following *expectation*:

> If the initial triangulation $\mathcal{T}_0$ is reflecting enough structure of the problem, such that the refinement process produces adequate triangulations, there is an index j_0, such that $\mathcal{T}_j$ is fine for $j \geq j_0$.

Of course a *proof* on the base of the solution process described in Section 4.4 and the refinement strategy of Section 6.4.2 would be very desirable, but seems to be extremely difficult and nearly untractable.

On the base of our expectation we can state:

THEOREM 6.4. *On a fine triangulation $\mathcal{T}_j$ we get for any $\hat{u} \in \mathcal{S}_j$*

$$\|u - \hat{u}\|_A = \zeta \|r_Q\|_{\hat{\Theta}_Q}$$

with

$$\zeta \in \left[\frac{1}{\sqrt{\mu_1^Q}}, \frac{2}{\sqrt{\mu_0^Q}} \right],$$

where the constants are from Theorem 6.2. This interval is independent of $\tau \geq 0$. Specification of $\tau = 0$ yields:

$$\zeta|_{\tau=0} \in [0.25, 4],$$

with quality indicator

$$\kappa_\zeta|_{\tau=0} \leq 16.$$

Proof. Theorems 6.3 and 5.1 with Remark 5.1 yield the assertions. ∎

REMARK 6.11. The *one dimensional* version of the above theorem can be stated with notation of [17, Section 4.1] as follows: On a fine 1D–grid Δ we get

$$\|u - u_\Delta\|_A = \zeta\,[\eta]$$

with

$$\zeta \in \left[\frac{1}{\sqrt{\tilde{\gamma}_1}}, \frac{2}{\sqrt{\tilde{\gamma}_0}} \right],$$

where the constants are from Remark 6.9. This interval is independent of $\tau \geq 0$. Only the definition of the indicators $[\eta_j]$ has to be changed slightly compared to [17, 4.12]:

$$[\eta_j] = \|\tilde{w}_j\|_A.$$

Comparing this result with [17, Theorem 4.2] we have given a *totally new justification* of the 1D error estimator of [17, Section 4.1]. Since we have a complete knowledge of the involved constants in the 1D case, let us specify them for the Laplacian $A(x, \partial) = -\dfrac{d^2}{dx^2}$: Here the interval is

$$\zeta \in [0.72, 6.78].$$

[17, Theorem 4.2] yields in this case $\zeta \geq 0.45$, with the specification of the constant K as given in [16].

6.5. Implementation of the Multilevel Preconditioner and Complexity Analysis

Here we are concerned with the assertion at the end of Section 6.1.3, that $\Theta_j^* u^*$ can be computed within $\mathcal{O}(n_j)$ operations, i.e., requirement (P2). This assertion needs both: specification as well as careful explanation.

The expression $\Theta_j^* u^*$ has to be realized in the natural representation of $(\mathcal{S}_j, \mathcal{S}_j^*)$, that is we have as

Input: The values $\{u^*(\psi)\}_{\psi \in \Gamma_j}$

and compute the

Output: The values $\left\{(\Theta_j^* u^*)(x_\psi)\right\}_{\psi \in \Gamma_j}$.

However the expression (6.15), restated as

$$\Theta_j^* u^* = \frac{\tau}{1+\tau} \rho_0 u^* + \sum_{\psi \in \Psi} \frac{6\vartheta(\psi)}{|\operatorname{supp} \psi|} u^*(\psi)\psi,$$

requires the values $u^*(\psi)$ for all $\psi \in \Psi$, whereas only $u^*(\psi)$ for $\psi \in \Gamma_j$ is given first. Furthermore this expression states the result as a linear combination of all $\psi \in \Psi$, whereas we need the result as a linear combination of the nodal basis functions $\psi \in \Gamma_j$, i.e., a situation just the other way round than in the input case.

Thus we are confronted with the *proper organization* of two tasks

- *Restricting* the linear form u^* on $\mathcal{S}_k$ for $0 \leq k < j$
- *Interpolating* the values of functions in $\mathcal{S}_k$ to $\mathcal{S}_j$.

As it turns out the main difficulty will be the organization of the restriction.

Since the FEM code data structures of KASKADE [45] deal with nodal points rather than with the nodal basis functions itself, we have to reformulate our problem.

6.5.1. Ordering of Nodal Points

The collection of supporting points of Ψ_k will be called

$$\mathcal{M}_k = \{x_\psi \mid \psi \in \Psi_k\} \subset \mathcal{N}_k.$$

Taking the disjoint union of them

$$\mathcal{M} = \{(x,k) \mid x \in \mathcal{M}_k,\, 0 \leq k \leq j\}$$

we get a mapping

$$\not{\psi} : \mathcal{M} \to \Psi,$$

such that $\not{\psi}(x,k) \in \Psi_k$ and $x_{\not{\psi}(x,k)} = x$, which gives a unique $\psi \in \Psi$. Clearly $\not{\psi}$ is one-to-one and onto, such that

$$\#\mathcal{M} = \#\Psi \leq 2n_j.$$

Now define mappings $S, U : \mathcal{M} \to \mathbb{R}$ and $\ell : \mathcal{M} \to \mathbb{N}$ such that

$$\begin{array}{lrcl} \text{i)} & S(x,k) & = & |\operatorname{supp} \not{\psi}(x,k)| = 3\,(1, \not{\psi}(x,k)) \\ \text{ii)} & U(x,k) & = & u^*(\not{\psi}(x,k)) \\ \text{iii)} & \ell(x,k) & = & k^1_{\not{\psi}(x,k)} \geq k, \end{array}$$

for $(x,k) \in \mathcal{M}$. Note that

$$k^0_{\not{\psi}(x,k)} = k.$$

Therefore

$$\Theta_j^* u^* = \frac{\tau}{1+\tau} V_0(x) \not{\psi}(x,0) + \sum_{k=0}^{j} \sum_{x \in \mathcal{M}_k} \frac{6\vartheta(x,k)}{S(x,k)} U(x,k) \not{\psi}(x,k),$$

where $\{V_0(x)\}_{x \in \mathcal{N}_0}$ is the *direct solution* of

$$\sum_{x \in \mathcal{N}_0} a_\tau(V_0(x) \not{\psi}(x,0), \not{\psi}(\bar{x},0)) = U(\bar{x},0)$$

for all $\bar{x} \in \mathcal{N}_0 = \mathcal{M}_0$; and where due to Lemma 6.5

$$\vartheta(x,k) = \vartheta(\not{\psi}(x,k)) = \lambda_k^{-1} - \lambda_{\ell(x,k)+1}^{-1}$$

for $(x,k) \in \mathcal{M}$ by formally setting $\lambda_{j+1}^{-1} = 0$. Defining for $(x,k) \in \mathcal{M}$:

$$V(x,k) = \begin{cases} V_0(x) + 6\dfrac{\vartheta(x,0)}{S(x,0)} U(x,0), & k = 0 \\ 6\dfrac{\vartheta(x,k)}{S(x,k)} U(x,k), & 0 < k \leq j, \end{cases}$$

we have

$$\Theta_j^* u^* = \sum_{k=0}^{j} \sum_{x \in \mathcal{M}_k} V(x,k) \not{\psi}(x,k).$$

Thus it is essential to realize the sum $\sum_{x \in \mathcal{M}_k}$ algorithmically.

6.5.2. Algorithmic Realization of the Loop: For all $x \in \mathcal{M}_k$.

Define for $0 \leq k \leq j$

$$\partial\mathcal{T}_k = \left\{ T \in \bigcup_{k=0}^{j} \mathcal{T}_k \,\middle|\, T \text{ regular and } \operatorname{depth} T = k \right\}.$$

Rule (T3) of Section 4.2.1 gives for $k > 0$ that

$$\partial\mathcal{T}_k = \{T \in \mathcal{T}_k \setminus \mathcal{T}_{k-1} \mid T \text{ regular}\}$$

and

$$\partial\mathcal{T}_0 = \mathcal{T}_0.$$

The realization of the set $\mathcal{M}_k$ is a consequence of the following Lemma.

LEMMA 6.10. *For $0 \leq k \leq j$ we get*

$$\mathcal{M}_k = \{x \in T \cap \mathcal{N}_k \mid T \in \partial\mathcal{T}_k\}.$$

Proof. The case $k = 0$ is trivial. Let $k > 0$.

1. Take $T^* \in \partial\mathcal{T}_k$ and $x \in T^* \cap \mathcal{N}_k$ a vertex. Thus T^* is generated by regular refinement of some triangle $T \in \mathcal{T}_{k-1}$. Moreover there is a $\psi \in \Gamma_k$ with $x = x_\psi$.

   Assuming that $\psi \in \Gamma_{k-1}$ would imply $x \in T \cap \mathcal{N}_{k-1}$ which yields $T \subset \operatorname{supp}\psi$ — in contradiction to $\psi|_{T \setminus T^*} \equiv 0$. Thus $\psi \notin \Gamma_{k-1}$, leading to $\psi \in \Psi_k$ and in turn to $x \in \mathcal{M}_k$.

2. Take $x = x_\psi \in \mathcal{M}_k$, $\psi \in \Psi_k$.

   Case A.: $x \notin \mathcal{N}_{k-1}$. Then x must be a vertex of a regular triangle $T \in \mathcal{T}_k \setminus \mathcal{T}_{k-1}$, according to rule (T1) of Section 4.2.1. Thus $T \in \partial\mathcal{T}_k$.

   Case B.: $x \in \mathcal{N}_{k-1}$. Then there is a $\bar{\psi} \in \Gamma_{k-1}$ with $x_{\bar{\psi}} = x$. Since $\psi \notin \Gamma_{k-1}$ we have according to Lemma 4.1.ii that $\operatorname{supp}\bar{\psi} \not\subset \operatorname{supp}\psi$. Thus there is a triangle $T \in \mathcal{T}_{k-1}$ with $x \in T$ and

   $$T \not\subset \operatorname{supp}\psi,$$

   and a triangle $T^* \in \mathcal{T}_k \setminus \mathcal{T}_{k-1}$ generated by subdivision of T, for which $x \in T^*$.

Case B.1.: T^* is *regular*. Thus $T^* \in \partial\mathcal{T}_k$.

Case B.2.: T^* is *irregular*. Take the unique vertex $x^* \in T^* \cap (\mathcal{N}_k \setminus \mathcal{N}_{k-1})$ and consider the edge $[x, x^*]$. There is just one other triangle $T^{**} \in \mathcal{T}_k \setminus \mathcal{T}_{k-1}$ with $T^{**} \cap T^* = [x, x^*]$. By definition of irregular refinement we obtain $T = T^* \cup T^{**}$. Now $T^*, T^{**} \subset \operatorname{supp}\psi$ implies $T \subset \operatorname{supp}\psi$, a contradiction.

■

This Lemma suggests a slight extension of just two data structures of KASKADE, see the *Programmer's Manual* [45].

1. The `TRIANGULATION`–data type gets the following additional entry:

`firstTriangleOfDepth`	`TR**`	array of pointers, the k^{th} entry is the pointer to the first triangle of the singly linked list of regular triangles of depth k.

2. The `TR`–data type (triangles) gets the following additional entry:

`nextOfSameDepth`	`TR*`	pointer to the next regular triangle of same depth in the list.

During the refinement process we can easily establish the following *singly linked list* of regular triangles of depth k:

$$\underbrace{\texttt{actTriang->firstTriangleOfDepth[k]}}_{\downarrow} \quad T_1 \xrightarrow{\text{nextOfSameDepth}} T_2 \xrightarrow{\text{nextOfSameDepth}} \dots$$

$$\dots \xrightarrow{\text{nextOfSameDepth}} T_{l_k} \xrightarrow{\text{nextOfSameDepth}} \texttt{nil}.$$

The assembly $\{T_1, \dots, T_{l_k}\}$ of entries of this list constitutes just the set $\partial\mathcal{T}_k$.

Now the expression

For all $x \in \mathcal{M}_k$ apply procedure P to x

can be realized by means of the *pseudo C–procedure* Algorithm 6.1.

The *marking* in Algorithm 6.1 is necessary in order to avoid that the procedure P is applied *more than one time* to a point x.

Algorithm 6.1 can be realized with $\mathcal{O}(d_{\max}(\#\mathcal{M}_k))$ pointer operations. The number $d_{\max}$ denotes the maximal degree of a vertex in the triangulation $\mathcal{T}_k$. Due to rule (T2) of Section 4.2.1 $d_{\max}$ depends only on $\mathcal{T}_0$.

ALGORITHM 6.1.

```
T = actTriang->firstTriangleOfDepth[k]
while (T!=nil)
      mark all vertices x ∈ N_k of T
      T = T->nextOfSameDepth
T = actTriang->firstTriangleOfDepth[k]
while (T!=nil)
      for all vertices x ∈ N_k of T do
          if x is marked do
             apply P to x
             unmark x
      T = T->nextOfSameDepth
```

RESULT 6.1. *With Algorithm 6.1 we can run through the set $\mathcal{M}_k$ in $\mathcal{O}(\#\mathcal{M}_k)$ pointer operations.*

6.5.3. Realization of the Interpolation

Assume that the function $V(x,k)$, $(x,k) \in \mathcal{M}$, has already been computed and is stored. The actual computation of the values $V(x,k)$, however, will be described in Section 6.4.6 and consists of the restriction operation. Interpolation is now the computation of

$$\left(\Theta_j^* u^*\right)(x), \qquad x \in \mathcal{N}_j,$$

values which have to be stored in the places denoted by `x->theta`. The pseudo C–procedure Algorithm 6.2 computes them. Comments are printed italic.

Algorithm 6.2 needs by Result 6.1

$$\mathcal{O}\left(\sum_{k=0}^{j} \#\mathcal{M}_k\right) = \mathcal{O}(\#\mathcal{M}) = \mathcal{O}(n_j)$$

operations.

RESULT 6.2. *Given the function $V : \mathcal{M} \to \mathbb{R}$, the expression $\Theta_j^* u^*$ can be computed in $\mathcal{O}(n_j)$ operations, where only the storage for the result is needed.*

ALGORITHM 6.2.

```
for all x ∈ N_0 (which is just M_0) do
    x->theta = V(x,0)
for k = 1 to j do
    for all x ∈ M_k \ N_{k-1} do
        (Note that x is midpoint of an edge e = [x_1, x_2] ⊂ T ∈ T_{k-1},
        where x_1, x_2 ∈ N_{k-1}.)
        x->theta = 1/2*(x_1->theta + x_2->theta)
    for all x ∈ M_k do
        x->theta += V(x,k)
    (Inductively one shows: hereafter one has
    that x->theta = (Σ_{l=0}^{k} Σ_{x̂∈M_l} V(x̂,l) ψ(x̂,l)) (x) for all x ∈ M_k.)
```

REMARK 6.12. In Algorithm 6.2 both vertices of the edge $e = [x_1, x_2]$ were assumed to be in $\mathcal{N}_{k-1}$. This is actually not the case, if one of them belongs to the Dirichlet boundary piece Γ_D. In order to avoid awkward case–studies, we have denoted that case not explicitly in the algorithm, but the change is obvious: Just handle all entries for vertices not belonging to $\mathcal{N}_{k-1}$ as zero.

The same remark should be kept in mind for all algorithms which follow.

6.5.4. *Adjoint Ordering of Nodal Points*

The realization of the restriction of u^*, i.e., the actual computation of V, needs, however, more effort. For a clear representation we introduce the notion of the *adjoint ordering* of the nodal points.

Define for $0 \leq k \leq j$

$$\Psi^+ = \{\psi \mid k_\psi^1 = k\}.$$

By Lemma 4.1 we get

$$\begin{aligned} \text{i)} \quad \Psi_j^+ &= \Gamma_j, \\ \text{ii)} \quad \Psi_k^+ &= \Gamma_k \setminus \Gamma_{k+1}, \qquad 0 \leq k < j. \end{aligned}$$

Lemma 4.1 also states that Ψ is the disjoint union of the Ψ_k^+. Note that the set Ψ_j^+ will be changed if we add a triangulation $\mathcal{T}_{j+1}$, thus the sets Ψ_k^+ are an *a posteriori* splitting of Ψ whereas the sets Ψ_k established an *a priori*

splitting. Now we introduce a set *adjoint* to $\mathcal{M}_k$,

$$\mathcal{M}_k^+ = \{x_\psi \mid \psi \in \Psi_k^+\}, \qquad 0 \le k \le j.$$

Taking the disjoint union of them

$$\mathcal{M}^+ = \left\{(x,k) \mid x \in \mathcal{M}_k^+,\, 0 \le k \le j\right\}$$

we get a bijection

$$\not{\psi}^+ : \mathcal{M}^+ \to \Psi,$$

such that $\not{\psi}^+(x,k) \in \Psi_k^+$ and $x_{\not{\psi}^+(x,k)} = x$, which gives a unique $\psi \in \Psi$. Thus $\#\mathcal{M} = \#\Psi \le 2n_j$ holds. Define mappings $S^+, U^+ : \mathcal{M}^+ \to \mathbb{R}$ such that

$$\begin{aligned} &\text{i)} \quad S^+(x,k) &&= |\operatorname{supp} \not{\psi}^+(x,k)| = 3\,(1, \not{\psi}^+(x,k)) \\ &\text{ii)} \quad U^+(x,k) &&= u^*(\not{\psi}^+(x,k)) \end{aligned}$$

for $(x,k) \in \mathcal{M}^+$. Note that

$$k^1_{\not{\psi}^+(x,k)} = k.$$

Let $(x,k) \in \mathcal{M}$. Then we have $\not{\psi}(x,k) \in \Psi^+_{\ell(x,k)}$, thus

$$\not{\psi}(x,k) = \not{\psi}^+(x, \ell(x,k)), \tag{6.25}$$

and therefore

$$S(x,k) = S^+(x,\ell(x,k)) \text{ and } U(x,k) = U^+(x,\ell(x,k)).$$

Thus the function V is given once that we have computed the functions S^+, U^+, ℓ.

6.5.5. *Characterization of the Adjoint Sets* $\mathcal{M}_k^+$

By construction

$$\mathcal{M}_j^+ = \mathcal{N}_j$$

holds. But what can be said about the sets $\mathcal{M}_k^+$ for $0 \le k < j$? The next Lemma answers this question.

LEMMA 6.11. *For* $0 \le k < j$ *we obtain*

$$\mathcal{M}_k^+ = \mathcal{M}_{k+1} \cap \mathcal{N}_k.$$

Proof.

1. Take $\psi \in \Psi_{k+1}^+ = \Gamma_k \setminus \Gamma_{k+1}$. There is an $\bar{\psi} \in \Psi_{k+1}$ with $x_\psi = x_{\bar{\psi}} \in \mathcal{N}_k$. Thus $x_\psi \in \mathcal{M}_{k+1} \cap \mathcal{N}_k$.

2. Take $x \in \mathcal{M}_{k+1} \cap \mathcal{N}_k$. Thus there is an $\bar{\psi} \in \Psi_{k+1}$ with $x = x_{\bar{\psi}}$. Since $x \in \mathcal{N}_k$ there is an $\psi \in \Gamma_k$ with $x = x_\psi$. Because of $\bar{\psi} \notin \Gamma_k$ we have $\psi \neq \bar{\psi}$, which implies $\psi \notin \Gamma_{k+1}$. Thus we have $\psi \in \Psi_k^+$ and $x \in \mathcal{M}_k^+$.

■

This Lemma enables us to state an inverse to relation (6.25).

LEMMA 6.12. *For $x \in \mathcal{M}_k^+$, $0 \leq k < j$ we get*

$$\not{\psi}^+(x,k) = \not{\psi}(x,k+1) + \frac{1}{2} \sum_{\hat{x} \in \partial_{k+1} x} \not{\psi}(\hat{x}, k+1),$$

where

$$\partial_{k+1} x = \partial(\operatorname{supp} \not{\psi}(x,k+1)) \cap (\mathcal{N}_{k+1} \setminus \mathcal{N}_k).$$

Proof. The assertion follows from Lemma 6.11 and the fact that

$$\left(\not{\psi}^+(x,k)\right)(\hat{x}) = \frac{1}{2}$$

for all $\hat{x} \in \partial_{k+1} x$,

$$\left(\not{\psi}^+(x,k)\right)(\hat{x}) = 0$$

for all $\hat{x} \in \mathcal{N}_{k+1} \setminus \partial_{k+1} x$, $\hat{x} \neq x$, and

$$\left(\not{\psi}^+(x,k)\right)(x) = 1.$$

■

6.5.6. *Realization of the Restriction*

Because of $\mathcal{M}_j^+ = \mathcal{N}_j$ and $\Psi_j^+ = \Gamma_j$ the values $\{U^+(x,j)\}_{x \in \mathcal{N}_j}$ are just the input values $\{u^*(\psi)\}_{\psi \in \Gamma_j}$. Moreover we know the values of V if we have computed S^+, U^+, ℓ. All this indicates why we have to use the adjoint ordering for the computation of the restrictions.

The values $\{S^+(x,j)\}_{x \in \mathcal{N}_j}$ can be obtained almost easily by observing

$$S^+(x,j) = \sum_{x \in T \in \mathcal{T}_j} |T|.$$

These areas $|T|$, where $T \in \mathcal{T}_j$, can be assumed to be known: One has just to compute $|T|$ for all $T \in \mathcal{T}_0$ once at the beginning, the rest is done during refinement: if T is the refinement of T^* put $|T| = |T^*|/2$ or $|T| = |T^*|/4$ — depending on whether the refinement was irregular or regular.

RESULT 6.3. *The values $S^+(x,j)$ can be computed in $\mathcal{O}(n_j)$ operations.*

Since we have to *store* the values of S^+, U^+, ℓ, we introduce two arrays of reals: `S+[x][k]`, `U+[x][k]`, where $(x,k) \in \mathcal{M}^+$, and one array of integers: ℓ`[x][k]`, where $(x,k) \in \mathcal{M}$. Furthermore we need one additional array of integers: `l[x]`, where $x \in \mathcal{N}_j$.

The pseudo C–procedure Algorithm 6.3 computes the values of S^+, U^+, ℓ. Again comments are printed italic.

Discussion of Algorithm 6.3. Besides the comments made in the presentation of the algorithm we discuss one point more closely:
Each step of the main loop does for *fixed* k the following: Assume that before entering the loop the algorithm is at k in a state that

Assumptions:

1. `S+[x][k]` = $S^+(x,k)$, `U+[x][k]` = $U^+(x,k)$, for all $x \in \mathcal{M}_k^+$.
2. For $\psi \in \Gamma_k$ we have that `l`$[x_\psi]$= k_ψ^1, i.e. $\psi \in \Psi^+_{\texttt{l}[x_\psi]}$.

In this state step k of the main loop computes

1. `S+[x][k-1]` = $S^+(x,k-1)$, `U+[x][k-1]` = $U^+(x,k-1)$, for all $x \in \mathcal{M}_{k-1}^+$.
2. For $x \in \mathcal{M}_{k-1}^+$ the additional array is set to
$$\texttt{l[x]} = k_\psi^1 \qquad \text{for } \psi = \psi^+(x,k-1).$$
3. For $x \in \mathcal{N}_{k-1} \setminus \mathcal{M}_{k-1}^+$ we have $k, k-1 \in K_\psi$, where $\psi \in \Gamma_{k-1}$ with $x = x_\psi$. Thus $\psi \in \Gamma_k$ such that by assumption 2 above we get
$$\texttt{l[x]} = k_\psi^1,$$
i.e. $\psi \in \Psi^+_{\texttt{l[x]}}$, since `l[x]` was not touched.
4. By assumption 2 we compute `l[x][k]`= $\ell(x,k)$.

Since the assumptions are valid for the first step $k = j$ by initialization, induction shows:

ALGORITHM 6.3.

Initialization:
for all $x \in \mathcal{N}_j$ *(which is just* $\mathcal{M}_j^+$*)* do
    S^+[x][j] = $S^+(x,j)$
    U^+[x][j] = $U^+(x,j)$
    l[x] = j
Main–Loop:
for k = j downto 1 do
    for all $x \in \mathcal{M}_k$ do
        (Take $\psi = \not\psi(x,k) \in \Psi_k$*. Since* $\psi \in \Psi^+_{l[x]}$ *and therefore* $k^1_\psi = l[x]$ *put:)*
        ℓ[x][k] = l[x]
        (Note that $l[x] \geq k$*.)*
        case:$x \in \mathcal{N}_{k-1}$
            (This is the case iff $x \in \mathcal{M}^+_{k-1}$*. Take* $\bar\psi = \not\psi^+(x,k-1)$*, then* $k^1_{\bar\psi} = k-1$*. Thus:)*
            l[x] = k-1
            (Now $\psi = \not\psi^+(x,\ell(x,k))$*. Thus:)*
            S^+[x][k-1] = S^+[x][ℓ[x][k]]
            U^+[x][k-1] = U^+[x][ℓ[x][k]]
        case:$x \in \mathcal{N}_k \setminus \mathcal{N}_{k-1}$
            (Thus $x \notin \mathcal{M}_i$ *for* $0 \leq i \leq k-1$*. But* x *is midpoint of an edge* $e = [x_1, x_2] \subset T \in \mathcal{T}_{k-1}$ *with* $x_1, x_2 \in \mathcal{N}_{k-1}$*, thus* $x_1, x_2 \in \mathcal{M}^+_{k-1}$ *according to rule (T1) of Section 4.2.1. Lemma 6.12 states that* ψ *contributes to exactly* $\not\psi^+(x_1, k-1)$ *and* $\not\psi^+(x_2, k-1)$*:)*
            S^+[x_1][k-1] += 1/2 * S^+[x][ℓ[x][k]]
            U^+[x_1][k-1] += 1/2 * U^+[x][ℓ[x][k]]
            S^+[x_2][k-1] += 1/2 * S^+[x][ℓ[x][k]]
            U^+[x_2][k-1] += 1/2 * U^+[x][ℓ[x][k]]
Values for $\mathcal{N}_0$*:*
for all $x \in \mathcal{N}_0$ *(which is just* $\mathcal{M}_0$*)* do
    ℓ[x][0] = l[x]

RESULT 6.4. *The Algorithm 6.3 computes S^+, U^+, ℓ in $\mathcal{O}(n_j)$ operations using additional storage of*

$$2(\#\Psi) \leq 4n_j \quad \textit{reals}$$

and

$$\#\Psi + n_j \leq 3n_j \quad \textit{integers.}$$

Here we used again Result 6.1 in order to estimate the number of operations by

$$\mathcal{O}\left(\sum_{k=0}^{j} \#\mathcal{M}_k\right) = \mathcal{O}(\#\mathcal{M}) = \mathcal{O}(n_j).$$

Summarizing all results up to now gives:

THEOREM 6.5. *The expression $\Theta_j^* u^*$ is computationally available in the natural representation of $(\mathcal{S}_j, \mathcal{S}_j^*)$ within $\mathcal{O}(n_j)$ operations using an additional amount of storage of not more than $4n_j$ reals and $3n_j$ integers — no matter how the n_k actually progress.*

Thus the preconditioner fulfills the missing requirement (P2).

REMARK 6.13. Note that we did not need a list of neighboring points to a given point – due to the special ordering of the nodal points. This ordering relies *only* on singly linked lists of triangles, which very well fit into the data structures of KASKADE.

REMARK 6.14. If we have to compute $\Theta_j^* u^*$ for *different* linear forms u^* during the iteration process, we have to update the array U+ only — due to the fact that S+ and ℓ depend on the triangulation only.

III. Algorithmic Details and Numerical Examples

7. Algorithmic Details

7.1. The 1D Case

We use the same elliptic solver as explained in [17, Section 4]. The measures for the amount of work as introduced in Section 1.2 should be chosen as

$$A_j = \frac{j+3}{\sqrt{\chi(j-1)}} \qquad j = 2, 3, \ldots .$$

Herein we assumed that creating the final mesh and solving for $\hat{u}^0$ is twice as expensive as the computation of one correction $\hat{\eta}_j$. Furthermore the amount of work principle stated in [17] has been used.

Knowing the amount of work in advance we are able to study the order control *qualitatively* in dependence of the imposed accuracy TOL – by using the information theoretic standard model of [23], which has been discussed in Section 3.3.1. This study shows that the minimal value of (1.7), that determines the optimal order, lies between neighbors which are nearly of the same size. In order to avoid a nasty oscillation between such neighboring orders we require that

$$\frac{A_{k+2}}{\tau_{k+1}} \leq \sigma \frac{A_{k+1}}{\tau_k}$$

before taking the order $k+1$ into account. The value

$$\sigma = 0.9$$

has turned out to be a good choice. Making this choice we gain the following nice result: The maximal possible order suggestion which we expect then is

$$k_{\max} = \lfloor 1 - \log_{10} \mathsf{TOL} \rfloor , \tag{7.1}$$

at least for tolerances $\mathsf{TOL} \in [10^{-6}, 10^{-1}]$. The numerical examples of Section 8.1 will confirm this a priori result.

7.2. The 2D Case

Here we discuss in detail the consequences of the stationary results of Section 6 for the time–stepping algorithm of Sections 1.3 and 3.

However, we have so far constructed an error estimator and linear solver belonging to the $\|\cdot\|_A$–norm and *not to the* L^2*–norm*, at least for $\tau \gg 0$. This relies on the fact that there is no iterative linear solver, at least to the knowledge of the author, comparable to the CG–method, which reduces the error with respect to the L^2–inner product. Since $\|u\|_A \geq \|u\|_0$ we can use the $\|\cdot\|_A$–norm as upper bound estimate for the L^2–norm and control the time–error nevertheless in the L^2–norm.

Usage of the $\|\cdot\|_A$–norm for the stationary elliptic subproblems instead of the L^2–norm has certain disadvantages with respect to the amount of work, but is the best we can do yet. The impact for the time–stepping procedure is discussed below.

7.2.1. *Optimal Choice of the Factor* ϱ

Because of the use of the stationary $\|\cdot\|_A$–norm we have to minimize instead of the term given in Section 3.3.3 the term

$$\frac{1}{(1-\varrho)^2\sqrt{\varrho}}$$

in view of the discussion in Section 6.4.2, especially estimate (6.24). This gives

$$\tilde{\varrho}_2 = \frac{1}{5}$$

instead of $\varrho_2 = \frac{1}{3}$ as in (3.4).

7.2.2. *The Amount of Work*

The measures for the amount of work introduced Section 1.2 should be chosen as

$$A_j = \frac{j+3}{\chi(j-1)^2}, \qquad j = 2, 3, \ldots.$$

Here we made again use of estimate (6.24), which states that n_j grows like eps^{-2}, where eps denotes the accuracy for the elliptic $\|\cdot\|_A$–norm error. Usage of the L^2–norm would give instead

$$A_j^{L^2} = \frac{j+3}{\chi(j-1)},$$

compare with the 1D result given in Section 7.1.

7.2.3. *Qualitative Study of the Order Control*

As in Section 7.1 for the 1D case we can study the order control mechanism in dependence of the imposed accuracy TOL. For the choice $\sigma = 0.9$ as in Section 7.1 we get the dependence of the maximal suggested order from TOL as listed in Table IV.

TABLE IV.
MAXIMAL ORDER $k_{\max}$ IN DEPENDENCE OF THE IMPOSED ACCURACY TOL; $\|\cdot\|_\Lambda$–NORM STATIONARY

TOL	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}
$k_{\max}$	1	1	1	2	3	6	7

It should be noted that $k_{\max} = 1$ means that we compute solutions u^1, u^2 of order 1 and 2 resp. at any time step. *Thus the time-step control is available anyway.*

RESULT 7.1. For tolerances TOL $\geq 10^{-3}$ the order control mechanism chooses the lowest possible order, since order switches do not pay off in terms of efficiency. For these tolerances we can restrict ourselves to the computation of u^1, u^2 at each time step because the order control would never decide to compute u^3. This result is still true if we choose $\sigma = 0.99$.

Quite a different result is obtained for the usage of the L^2–norm, as shown in Table V.

TABLE V.
MAXIMAL ORDER $k_{\max}$ IN DEPENDENCE OF THE IMPOSED ACCURACY TOL; $\|\cdot\|_0$–NORM STATIONARY

TOL	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}
$k_{\max}$	1	2	3	5	6	7	7

7.2.4. *Stop Criterion for the Linear Solver for the Time Error $\hat{\eta}_l$*

Approximation of the time error function η_l on the triangulation $\mathcal{T}_j$ gives the perturbed function $\hat{\eta}_l \in \mathcal{S}_j$ as discussed in Section 3.1. However, in the

2D case we compute $\hat{\eta}_l$ only with an iterative solver, which has to be stopped efficiently. If we choose the starting value

$$\hat{\eta}_{l,0} = 0,$$

which gives a residual r_0, the value

$$\|r_0\|_{\hat{\Theta}_j}$$

is a reasonable measure for the size of the error $\|\hat{\eta}_l\|_\Lambda = \hat{\epsilon}_l$, due to Theorem 6.3. In view of the control criterion (1.8(ii)) we iterate

$$\hat{\eta}_{l,1}, \ldots, \hat{\eta}_{l,\iota}$$

until the following *stop criterion* is fulfilled:

$$\|r_\iota\|_{\hat{\Theta}_j} \leq \frac{1}{10}\|r_0\|_{\hat{\Theta}_j}.$$

We have

$$\begin{aligned}[\theta_l] &= \|r_{\iota,Q}\|_{\hat{\Theta}_Q} \\ &= \|r_\iota\|_{\hat{\Theta}_j} + [\tilde{\theta}_l],\end{aligned}$$

where $[\theta_l]$ is the estimate of the spatial perturbation of the time error as introduced in (3.1) and $[\tilde{\theta}_l]$ is the estimate without the part due to the linear solver. In view of the stop criterion we replace (1.8(ii)) by the computationally available

$$[\tilde{\theta}_l] \leq \frac{3}{20}\|\hat{\eta}_{l,\iota}\|_0.$$

7.2.5. Stabilization of the L^2–Projection

In the case of an inconsistent start function u_0 the L^2–projection into S_j may be unstable, hence our whole stationary problem becomes unstable for τ small. Here we replace the L^2–inner product by the discrete L^2–inner product $(\cdot,\cdot)_k$ of Section 3.1.2 and assemble the mass matrix, stiffness matrix and the right–hand side with respect to that discrete inner product, which means the usage of the corresponding quadrature rule. Now the case $\tau = 0$ reduces to a simple stable interpolation. Moreover the local order of approximation is not touched for $\tau > 0$ since the quadrature rule is exact for piecewise linear functions on $\mathcal{T}_j$, cf. Theorem 4.1.6 of CIARLET [20].

Because of the construction of our preconditioner no property of it is lost by usage of this quadrature rule.

7.2.6. The Direct Solver

The iterative solution process described in Section 4.4 as well as the preconditioner itself requires a direct solution on the coarsest triangulation $\mathcal{T}_0$. Due to the complex geometries in applications, e.g., the one given in Section 9, the number of nodal points in $\mathcal{T}_0$ may be quite big, about 200–1000. In 3D the number would be even more big. Thus a sophisticated direct solver is indispensable.

We choose a Cholesky decomposition solver, which exploits the envelope structure of its L–factor. In order to make this nearly optimal, it is necessary to order the points in such a way, that the envelope is nearly minimal. The so–called *reverse Cuthill–McKee ordering* accomplishes that in a rather efficient way, cf. e.g., the textbook of GEORGE/LIU [28].

For a number of nodal points in $\mathcal{T}_0$ below about 1000 this choice turns out to be superior to the use of a fully sparse solver together with the nested dissection ordering.

In our application example of Section 9, where the number of nodal points in $\mathcal{T}_0$ is 351, the computing time for two direct decompositions and the RCM–ordering was only 0.66% = 3.4s of the total computing time for 16 time–steps of 8 min 53.5 s with a final number of 3688 nodal points. This relation would be nearly unchanged even if we would — by a fixed number of 3688 final points — have about 1000 nodal points in the coarsest triangulation. Thus the total complexity is independent of the number of nodal points — within the indicated range — in the coarsest triangulation.

8. Numerical Examples: Model Problems

8.1. Examples in One Space Dimension

In this section we will demonstrate the efficiency of the time discretization given by the family of type (L) by means of two 1D examples. This time discretization is implemented in the program KASTIO1, where the number 1 indicates the space dimension. It uses the same elliptic solver as the program KASTIX1 of the author [17], which is a realization of the extrapolated implicit Euler scheme as discretization in time. The superiority of the time discretization relying on the family of type (L) over the extrapolation technique is enlightened by comparison of the behavior of KASTIO1 and KASTIX1.

Notation. In the tables of this section we make use of the following — beside the notation introduced earlier in this paper:

Max. order k: During a run the program has computed a sequence $u_1, \ldots, u_{k+1}$ of approximations *at least for one time layer*. Thus the maximal given order of approximation is $k+1$ whereas the maximal order, for which an error estimation has been performed, is k.

$$
\begin{aligned}
n_{\text{step}} &= \text{no. of time steps,} \\
[N] &= \sum_{l=1}^{n_{\text{step}}} \text{no. of nodal points of the final grid } \Delta_{\text{fin},l}/n_{\text{step}}, \\
N_{\text{tot}} &= \sum_{l=1}^{n_{\text{step}}} \text{no. of nodal points of the grid } \Delta_{\text{fin},l}/1000, \\
\epsilon &= \max_{1 \le l \le n_{\text{step}}} \text{true relative } L^2\text{–error of the solution at time step } l, \\
\text{CPU} &= \text{computing time in seconds on a SPARC–station1+,} \\
\kappa &= \frac{\text{CPU}}{N_{tot}}.
\end{aligned}
$$

For the meaning of the mean value $[N]$ see [16]. Since it indicates the effort for every nodal point, κ is something like a *complexity index.*

Example 8.1.*Point–source.* This model problem has been proposed by the author in [16] to test the time–stepping procedure. We solve the homogeneous heat equation on the spatial interval $I = [-10, 10]$ with the following

approximate δ–function as initial data:

$$u_0(x) = 250 \exp(-250x^2).$$

The Dirichlet boundary conditions can be considered as zero for $t \leq 1$ to model the evolution of u_0 on the whole real axis, the solution computed by KASTIO1 can be seen in Fig. 1.

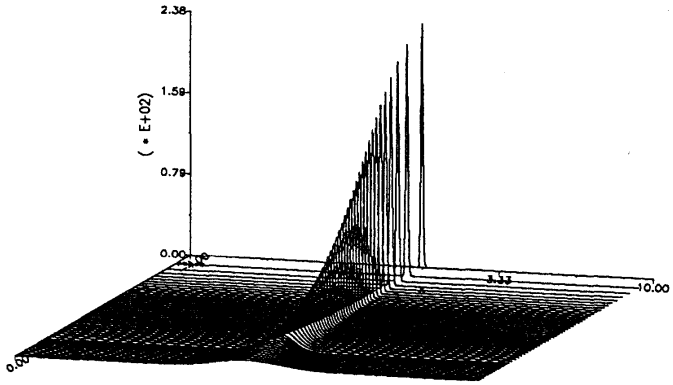


FIG. 1. Evolution of point–source, time in log–scale (Example 8.1).

Because of the exponential decay of the solution as shown in Fig. 1 we expect an increase of the time step according to a power law, which really occurs automatically in the performance of KASTIO1 as shown in Fig. 2; the corresponding development of the space mesh is shown in Fig. 3.

Comparison of Table VI with Table VII clearly shows a drawback of extrapolation: Increasingly higher cost while increasing the order. KASTIX1 needs more accurate TOL for increasing the order, and drops that order more quickly than KASTIO1. The slow increase of time steps as shown in Table VI for KASTIO1 means that the new time discretization is able to use the higher orders quite long — a feature, which had to be expected in view of the low cost of the higher orders. The maximal orders occurring in the runs of KASTIO1 nicely confirm the theoretical prediction (7.1) of Section 7.1. Moreover the complexity index κ behaves nearly constant for KASTIO1, thus we can speak of *multigrid complexity* of that program.

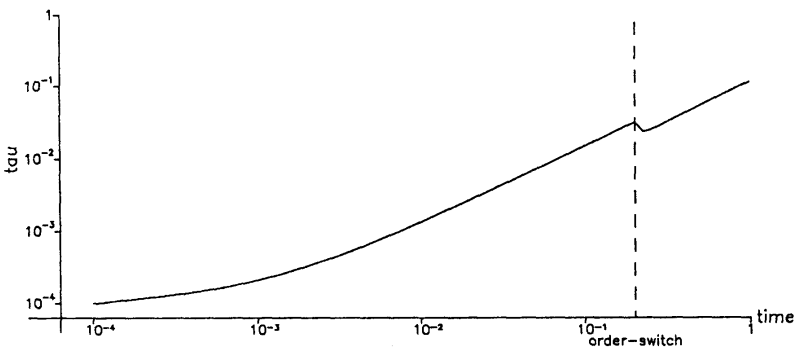


FIG. 2. Automatic increase of the time step (Example 8.1).

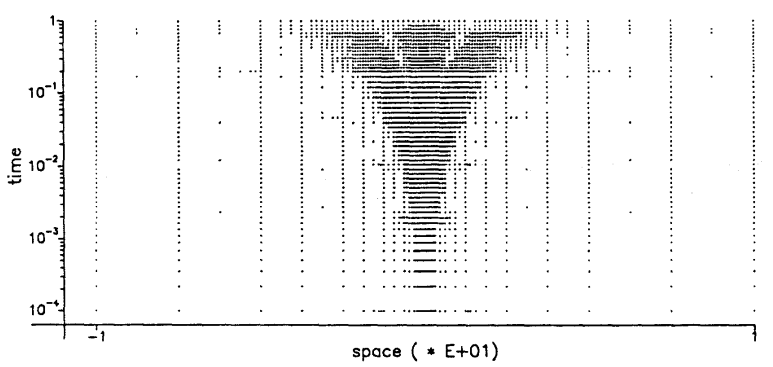


FIG. 3. Mesh development for the point-source (Example 8.1).

TABLE VI.
FAMILY OF TYPE(L)(KASTIO1):
PERFORMANCE FOR VARIABLE ORDER (EXAMPLE 8.1)

TOL	n_{step}	Max. order	$[N]$	ϵ	CPU	N_{tot}	κ
$*10^{-1}$	55	2	147	$4.90_{10}-2$	13	8	1.6
10^{-2}	66	3	513	$2.73_{10}-3$	61	34	1.8
10^{-3}	79	4	1748	$1.67_{10}-4$	289	138	2.1
10^{-4}	87	5	5632	$8.87_{10}-6$	1145	490	2.3

* run represented in Figs. 1–3.

TABLE VII.
EXTRAPOLATION(KASTIX1):
PERFORMANCE FOR VARIABLE ORDER (EXAMPLE 8.1)

TOL	n_{step}	Max. order	$[N]$	ϵ	CPU	N_{tot}	κ
10^{-1}	55	1	186	$3.96_{10}-2$	28	10	2.8
10^{-2}	118	1	634	$4.76_{10}-3$	286	75	3.8
10^{-3}	99	2	3758	$4.36_{10}-4$	2115	372	5.7
$*10^{-4}$	—	–	—	—	—	—	—

* run exceeds storage capabilities of the workstation used.

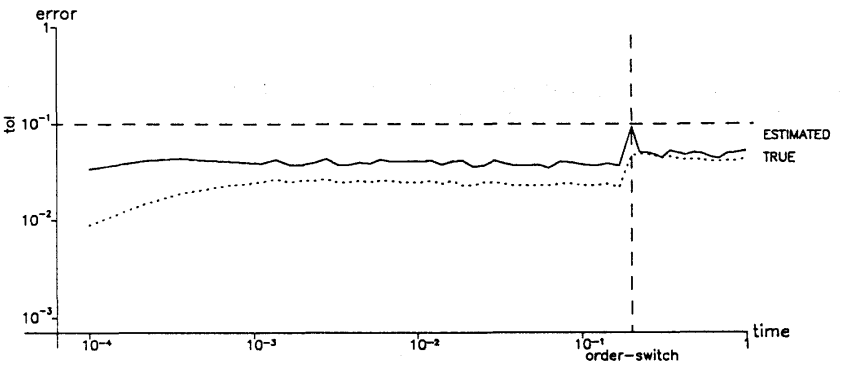


FIG. 4. Estimated vs. true error; KASTIO1 for TOL = 10^{-1} (Example 8.1).

Fig. 4 shows that the error estimation of KASTIO1 is very reliable. In the run the chosen order is 2 for $t < 0.2$ and 1 for $t \geq 0.2$. The jump of the error at this switching time nicely reflects the whole behavior of time–step and order control. Moreover it shows the quality of the error prediction for the next step, since the estimated error is just below the given tolerance.

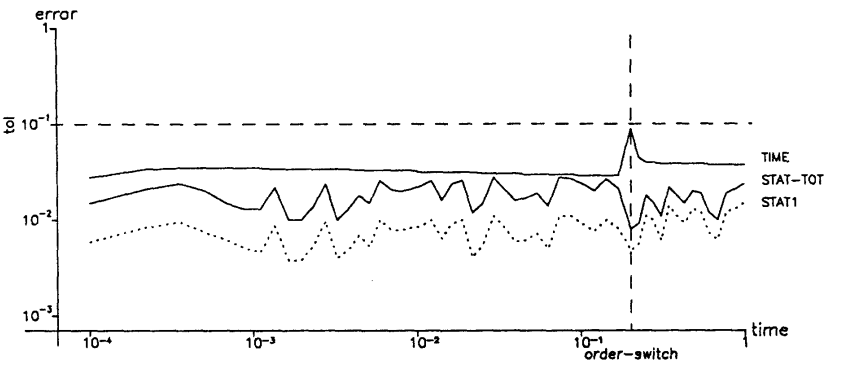


FIG. 5. The estimators $\hat{\epsilon}_j$, $[\delta_1]$ and $[\delta_{j+1}]$; KASTIO1 for TOL $= 10^{-2}$ (Ex. 8.1).

Fig. 5 shows the error estimators in more detail:

$$\begin{aligned} \hat{\epsilon}_j &= \text{TIME}, \\ [\delta_{j+1}] &= \text{STAT–TOT}, \\ [\delta_1] &= \text{STAT1}. \end{aligned}$$

Herein j denotes the actually chosen order. As expected we observe that the jump of the estimated error at $t = 0.2$ is due to $\hat{\epsilon}_j$. As long as the order remains constant the time–stepping procedure leaves the time–error component of the whole error nearly constant. The equal shape of the behavior of $[\delta_1]$ and $[\delta_{j+1}]$ backs the assertion that δ_1 dominates all other spatial perturbations — a feature detailly discussed in Section 3.2. This feature is shown more quantitatively in Fig. 6:

Herein the quotient $[\delta_{j+1}]/[\delta_1]$ is shown together with the corresponding propagation function $\chi(j)^{-1}$ (dotted line). It shows that our model of Section 3.2 slightly underestimates the error propagation.

EXAMPLE 8.2. *Inconsistent initial data.* This example is very challenging for the order and time–step control mechanism because of its transient phase.

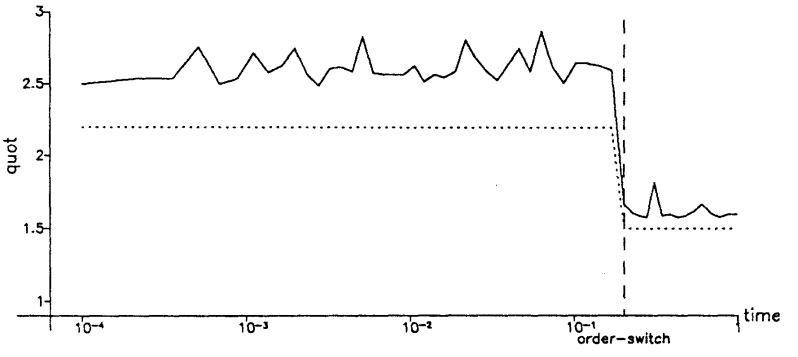


FIG. 6. The quotient $[\delta_{j+1}]/[\delta_1]$; KASTIO1 for TOL $= 10^{-3}$ (Example 8.1).

Moreover the solution runs into a stationary one. Thus we are able to study another drawback of the extrapolation method KASTIX1: KASTIX1 is not able to detect stationary phases.

The problem consists of the simple heat equation on the spatial interval $I = [0,1]$ with a simple time independent source term. We impose homogeneous Dirichlet boundary conditions and choose

$$u_0 \equiv 1.$$

Because u_0 does not satisfy the Dirichlet condition, the initial data are inconsistent. The source is chosen in order to get a stationary solution which is linear in $[0,0.5]$ and a parabola in $[0.5,1]$. The solution computed by KASTIO1 can be seen in Fig. 7.

Again we expect an increase of the time step according to a power law, which really occurs automatically in the performance of KASTIO1 as shown in Fig. 8. The corresponding development of the space mesh is shown in Fig. 9.

Comparison of Table VIII and Table IX shows that KASTIX1 chooses lower orders than KASTIO1 like in Example 8.1 and needs far more time steps. The latter observation can be explained by the above mentioned third drawback since the solution becomes stationary roughly at $t = 1$. For all tolerances KASTIO1 needs only 3 time steps to come from $t = 1$ to $t = 1000$, whereas KASTIX1 spends about 35 time steps for the same task (TOL $= 10^{-1}, 10^{-2}$). Moreover the need of computing time and of storage is much higher in case of the extrapolation method than in the case of the new time discretization.

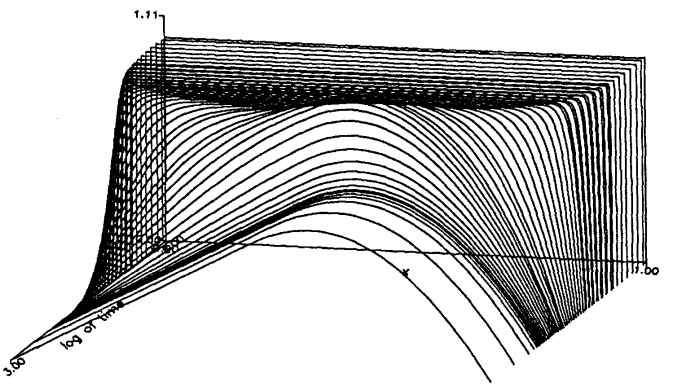


FIG. 7. Evolution of a boundary–inconsistency into a stationary solution, time in log–scale (Example 8.2).

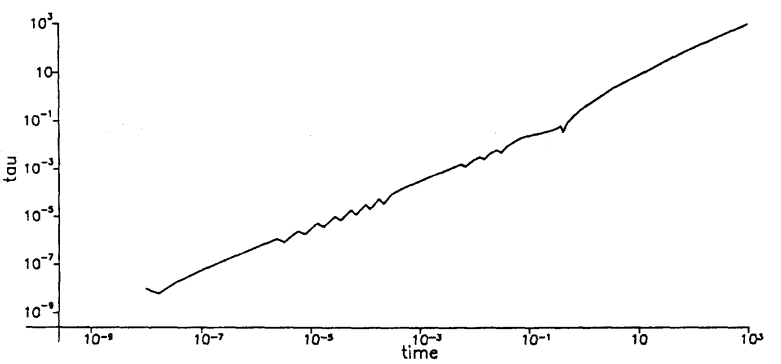


FIG. 8. Automatic increase of the time step (Example 8.2).

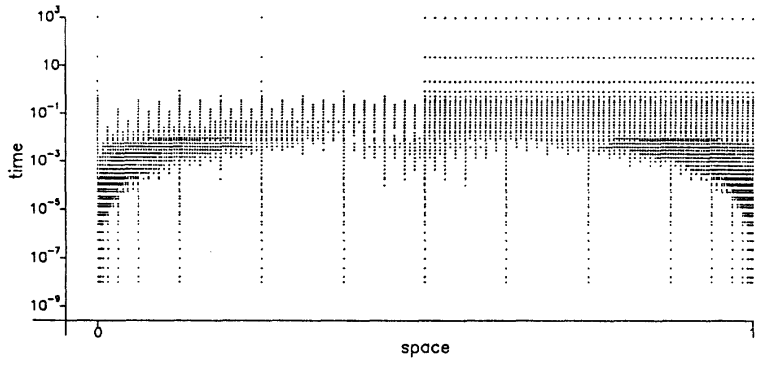


FIG. 9. Mesh development for the boundary–inconsistency (Example 8.2).

TABLE VIII.
FAMILY OF TYPE(L)(KASTIO1):
PERFORMANCE FOR VARIABLE ORDER (EXAMPLE 8.2)

TOL	n_{step}	Max. order	$[N]$	ϵ	CPU	N_{tot}	κ
10^{-1}	18	1	13	$2.49_{10}-2$	0.4	0.2	2.1
10^{-2}	25	2	31	$9.21_{10}-3$	1.1	0.8	1.4
$*10^{-3}$	52	3	89	$9.88_{10}-4$	5.9	4.6	1.3
10^{-4}	83	4	282	$9.93_{10}-5$	35.6	23.4	1.5
10^{-5}	79	6	1002	$9.81_{10}-6$	147.3	79.1	1.9

* run represented in Figs. 7–10.

TABLE IX.
EXTRAPOLATION(KASTIX1):
PERFORMANCE FOR VARIABLE ORDER (EXAMPLE 8.2)

TOL	n_{step}	Max. order	$[N]$	ϵ	CPU	N_{tot}	κ
10^{-1}	141	1	15	$8.03_{10}-2$	4.2	2.1	2.0
10^{-2}	131	1	33	$6.18_{10}-3$	8.9	4.2	2.1
10^{-3}	55	2	169	$9.85_{10}-4$	33.6	9.3	3.6
10^{-4}	164	2	483	$9.71_{10}-5$	524.1	79.0	6.6
$*10^{-5}$	—	-	—	—	—	—	—

* run exceeds storage capabilities of the workstation used.

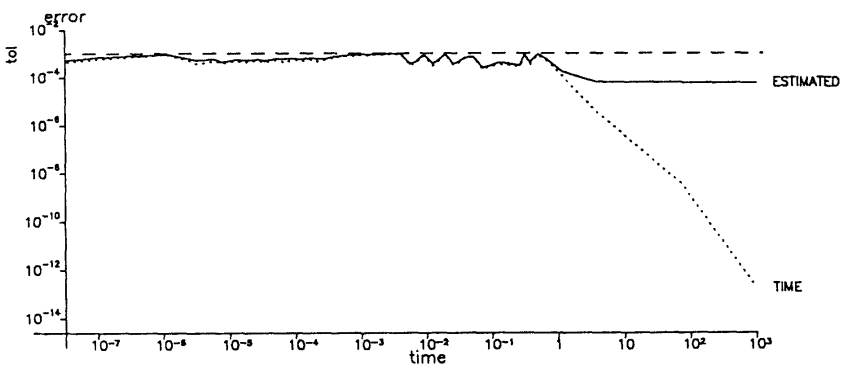


FIG. 10. Error behavior of KASTIO1 for TOL = 10^{-3} (Example 8.2).

Finally the complexity index κ shows for KASTIO1 *multigrid complexity* in contrast to KASTIX1.

Fig. 10, where the estimated time–error component is plotted in addition to the estimated total error, nicely shows how KASTIO1 is able to detect stationary phases.

8.2. EXAMPLES IN TWO SPACE DIMENSIONS

The program KASTIO2 is the 2D version of the 1D program KASTIO1 of Section 8.1. It uses for the elliptic subproblems the adaptive multilevel 2D elliptic solver KASKADE [25, 26, 44, 45] with the multilevel nodal basis preconditioner described in Section 6.

NOTATION. In the tables of this section we make use of the following — beside the notation introduced earlier in this paper:

$$
\begin{aligned}
n_{\text{step}} &= \text{no. of time steps,} \\
[N] &= \sum_{l=1}^{n_{\text{step}}} \text{no. of nodal points of the final triangulation } \mathcal{T}_{\text{fin},l}/n_{\text{step}}, \\
N_{\max} &= \max_{1\le l\le n_{\text{step}}} \text{no. of nodal points of a triangulation } \mathcal{T}_{\text{fin},l}, \\
N_{\text{tot}} &= \sum_{l=1}^{n_{\text{step}}} \text{no. of nodal points of the triangulation } \mathcal{T}_{\text{fin},l},
\end{aligned}
$$

$$
\begin{aligned}
\epsilon_{(\mathrm{rel})} &= \max_{1\le l\le n_{\mathrm{step}}} \text{true (relative) } L^2\text{–error of the solution at time step } l,\\
\mathrm{CPU} &= \text{computing time in seconds on a SPARC–station1+},\\
\varpi &= \sum_{\text{all CG–iterations}} \text{no. of nodal points of the actual triangulation}/N_{\mathrm{tot}},\\
\kappa_1 &= \frac{\mathrm{CPU}}{\varpi N_{tot}}\cdot 1000,\\
\kappa_2 &= N_{\mathrm{tot}}\,\mathsf{TOL}^2/100,\\
\kappa_3 &= n_{\mathrm{step}}\sqrt{\mathsf{TOL}}/10.
\end{aligned}
$$

Thus $[N]$ is the average number of nodal points, ϖ the average number of CG–iteration per nodal point, κ_1 the average cpu–time per nodal point CG–iteration.

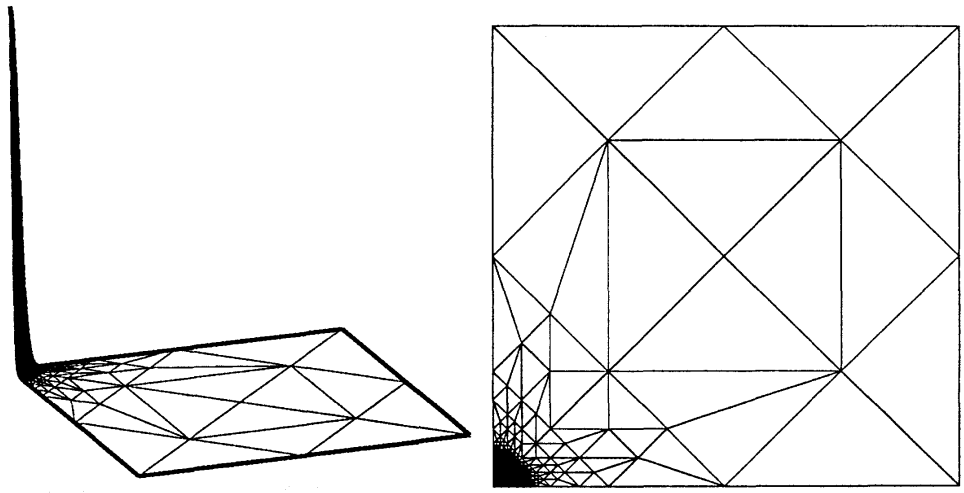


FIG. 11. Solution and triangulation at $t_1 = 10^{-4}$ (Example 8.3).

EXAMPLE 8.3. *Evolution of δ–function* (Due to ERIKSSON/JOHNSON [27]). This model problem is is a very challenging test for the time–stepping procedure, cf. Example 8.1. We solve the homogeneous heat equation with the following approximate δ–function as initial data:

$$u_0(x) = 250\exp(-250\|x\|^2).$$

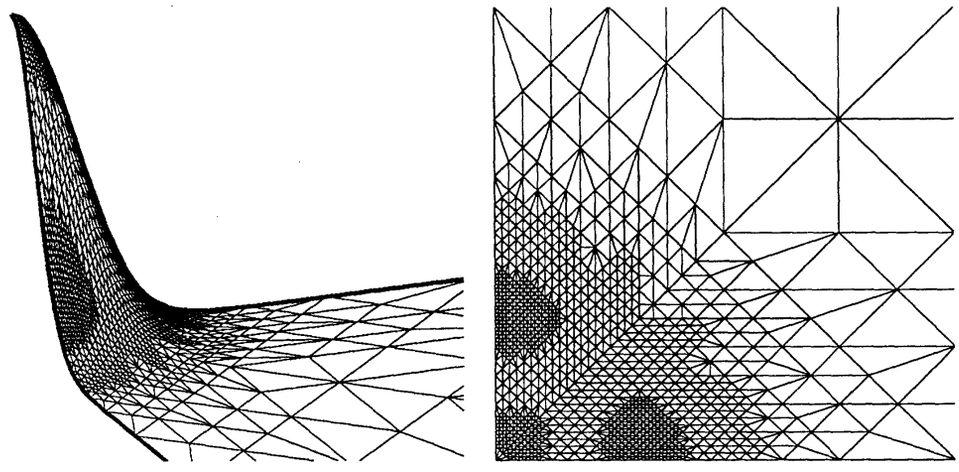


FIG. 12. Amplification of parameter plane in Fig. 11 by a factor of 8.

The Dirichlet boundary conditions are chosen to model on $\Omega = [0,2] \times [0,2]$ the evolution of u_0 on the whole plane. Thus the exact solution is given by the *Gauss-kernel* (modulo a factor of π)

$$u(t,x) = \frac{1}{4t + 1/250} \exp\left(-\frac{\|x\|^2}{4t + 1/250}\right).$$

The program was started at $t = 10^{-4}$ with a required tolerance TOL $= 10^{-1}$; stop time was $t = 2$. The computed solutions and the corresponding triangulations are shown in Fig. 11 at the starting time $t_1 = 1.0_{10} - 4$; those at the time $t_{24} = 1.01_{10} - 2$ (i.e., time step 24) are shown in Fig. 13. For the first time step the subdomain $[0, 0.25] \times [0, 0.25]$ is shown in Fig. 12 with a amplification by a factor of 8. The maximum of u at t_1 is $\|u(t_1, \cdot)\|_{L^\infty} = 227.3$ and at t_{24} it is $\|u(t_{24}, \cdot)\|_{L^\infty} = 22.5$. The figures are scaled with respect to 227.3 in the u-direction. For $t_1(t_{24})$ the number of nodal points is 941(170) whereas the number of triangles is 1812(312).

The Sobolev embedding lemma states that $\delta \in \dot{H}^{-1-\epsilon}$ for every $\epsilon > 0$. If we now take estimate (1.5) for the local error, i.e. $n = 1$, of the implicit Euler step, $p = 1$, for which the error is estimated, we observe that the error estimation works optimal if

$$\tau^2 \|u(t)\|_{\dot{H}^4} = \text{const}.$$

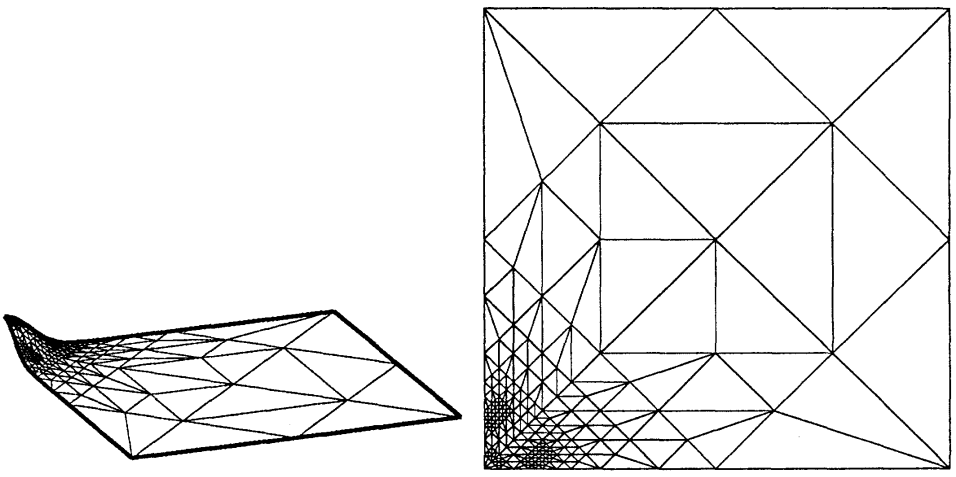


FIG. 13. Solution and triangulation at $t_{24} = 1.01_{10} - 2$ (Example 8.3).

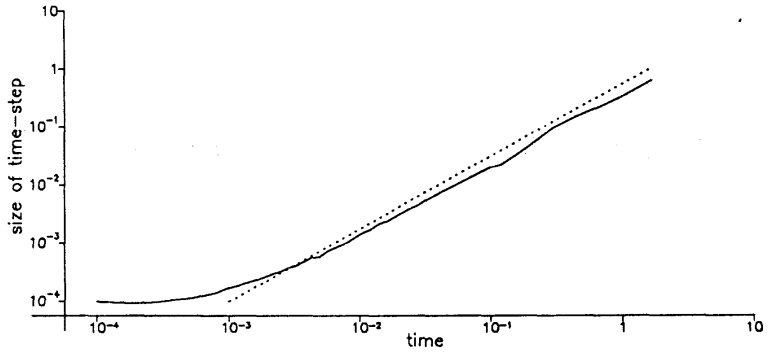


FIG. 14. Automatic increase of the time step (Example 8.3).

By semigroup results we obtain $\|u(t)\|_{\dot{H}^4} \leq Ct^{-5/2-\epsilon/2}\|\delta\|_{\dot{H}^{-1-\epsilon}}$ for $t \geq t_0 > 0$, since u_0 is a smooth approximation of δ. Thus we expect

$$\tau \propto t^{5/4},$$

and this behavior would prove the quality of the time error estimation. The automatic increase of the time step of KASTIO2 as shown in Fig. 14 really resembles this theoretical expectation: the dotted line has slope 5/4. The flat start of the curve is due to the $t_0 > 0$, i.e., the in fact smooth but large in norm starting function u_0.

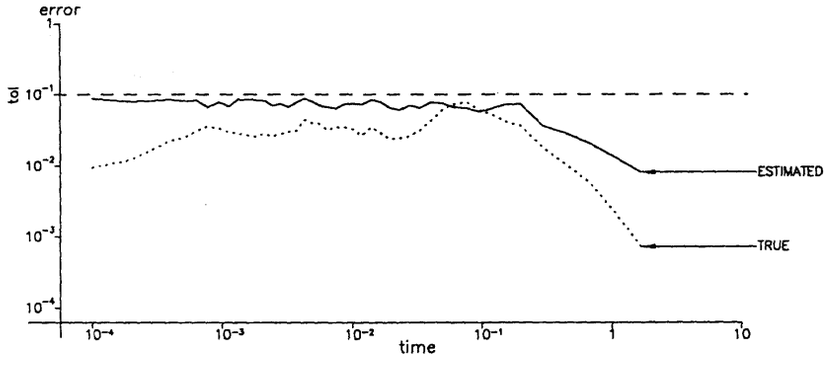


FIG. 15. Estimated and true error (Example 8.3).

The very reliable behavior of the error estimation is shown in Fig. 15 as well. The estimated error consists of the time error estimate $\hat{\epsilon}_1$(TIME) and the spatial error estimate $[\delta_2]$(SPATIAL) (when time approximations u^1, u^2 are computed only). Fig. 16 shows the reliability of these components and the fact that the time error has to be 4 times more accurate than the spatial error, cf. Sec. 7.2.1.

The automatic decrease of nodal points due to the smoothing of the solution is shown in Fig. 17. An optimal approximation on a family of nonuniform triangulations would give result to the estimate

$$\inf_{u_{j_{\text{fin}}} \in \mathcal{S}_{j_{\text{fin}}}} \|u(t) - u_{j_{\text{fin}}}\| \leq Cn_{j_{\text{fin}}}^{-1}\|u(t)\|_0.$$

We obtain the estimate $\|u(t)\|_0 \leq Ct^{-1/2-\epsilon/2}\|\delta\|_{\dot{H}^{-1-\epsilon}}$ for $t \geq t_0 > 0$ by again using semigroup results. Thus assuming a constant spatial error yields

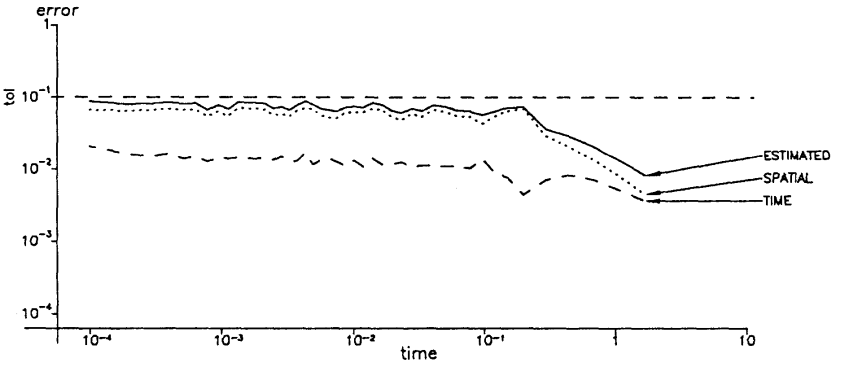


FIG. 16. The estimators $\hat{\epsilon}_1, [\delta_2]$ (Example 8.3).

$n_{j_{\text{fin}}} \propto t^{-1/2}$ for $t \geq t_0$. The dotted line in Fig. 17 is the fitting line with slope $-1/2$ in the double logarithmically scale. It shows that besides some oscillation due to the nonsmooth refinement process we achieved the optimal behavior of nodal points.

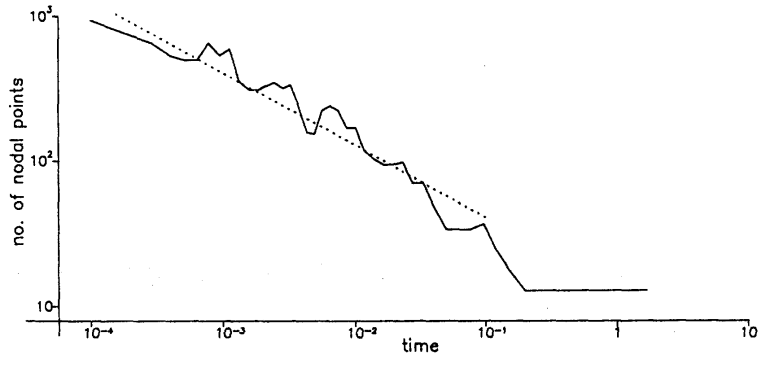


FIG. 17. Automatic decrease of nodal points (Example 8.3).

The maximal depths of the triangulations at each time step are shown in Fig. 18. Since the peak smoothes out we need fewer local refinement while time progresses. The highly satisfactory behavior of the preconditioned CG–iterations are also shown in this figure, it shows that the effective number of iterations per nodal point is between 2 and 3 as long as the problem is

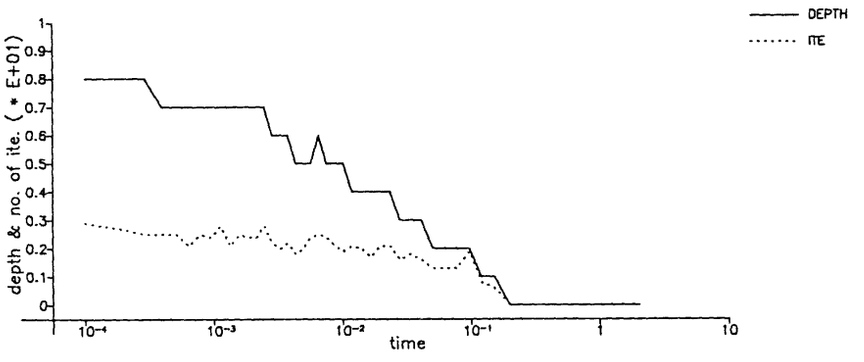


FIG. 18. Max. depth of triangulation and effective number of iterations (Example 8.3).

nontrivial; it drops to zero as the solution is the zero solution with respect to the tolerance $\mathrm{TOL} = 10^{-1}$.

TABLE X.
PERFORMANCE OF KASTIO2 FOR DIFFERENT TOLERANCES (EXAMPLE 8.3)

TOL	n_{step}	$[N]$	$N_{\max}$	N_{tot}	ϵ	CPU	ϖ	κ_1	κ_2	κ_3
$5.0_{10}-1$	23	34	118	768	$4.72_{10}-1$	20	4.5	5.8	1.9	1.6
$2.5_{10}-1$	29	77	295	2232	$1.32_{10}-1$	80	7.4	4.8	1.4	1.5
* $1.0_{10}-1$	45	236	941	10588	$7.85_{10}-2$	580	11.1	4.9	1.1	1.4
$7.5_{10}-2$	52	379	1519	19674	$4.31_{10}-2$	1170	11.7	5.1	1.1	1.4
$5.0_{10}-2$	65	648	2150	42072	$3.08_{10}-2$	2850	13.6	5.0	1.1	1.5

* run represented in Figs. 11–18.

The behavior of KASTIO2 for different tolerances is shown in Table X. Besides the nice behavior of the error estimation it shows three effects, each indicated by one of the κ_ι:

1. The constancy of κ_1 backs the result that the number of operations while applying the preconditioner is proportional to the number of unknowns, i.e., Theorem 6.5.

2. The constancy of κ_2 backs the assertion that our algorithm produces adequate triangulation, i.e., those for which estimate (6.24) is fulfilled. Thus all arguments which rely on this estimate are strengthened.

3. The constancy of κ_3 shows that the local order of approximation of $\mathcal{O}(\tau^3)$ in time is exploited such efficiently, that we gain *globally* an approximation of order $\mathcal{O}(n_{\text{step}}^{-2})$.

In particular we see that the factor of Section 7.2.1 is justified even globally and that the amount of work formula of Section 7.2.2 has been chosen correctly.

Finally we discuss the influence of the kind of progression of the number of nodal points during refinement of the triangulations. As we have seen in Section 4.4 we may loose the complexity estimate $\mathcal{O}(n \log n)$ if we do not force the progression to be geometrically. In this example the progression is even *arithmetically* for most of the time steps, which results in a far lower number of nodal points. Moreover the global complexity, which really interests, *computing time in dependence of the required accuracy* is optimal: CPU $\propto$ TOL^{-2}. This is shown in Fig. 19, where the slope of the curve of our example (NON) is really -2. For comparison we have computed the same example with forcing the progression to be at least geometrically with an increase of a factor 2, giving curve (GEOM) as shown in Fig. 19. In terms of computing time per required accuracy the geometrical progression does not pay off. Thus it seems not to be reasonable to ask for a complexity bound in the number of unknowns unless their progression is no matter of any restrictions.

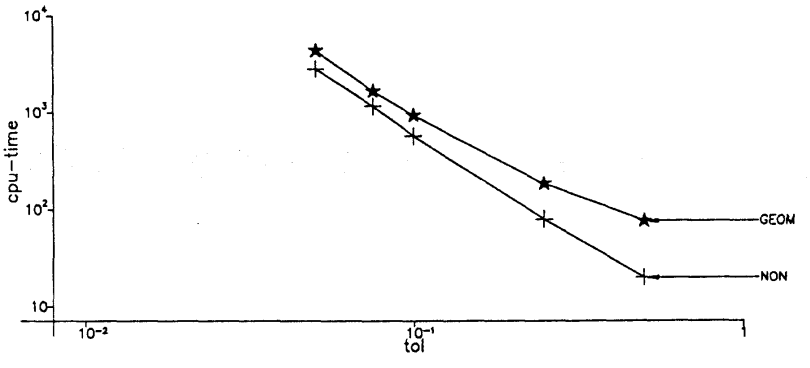


FIG. 19. Computing time vs. TOL, different progression of the n_k (Example 8.3).

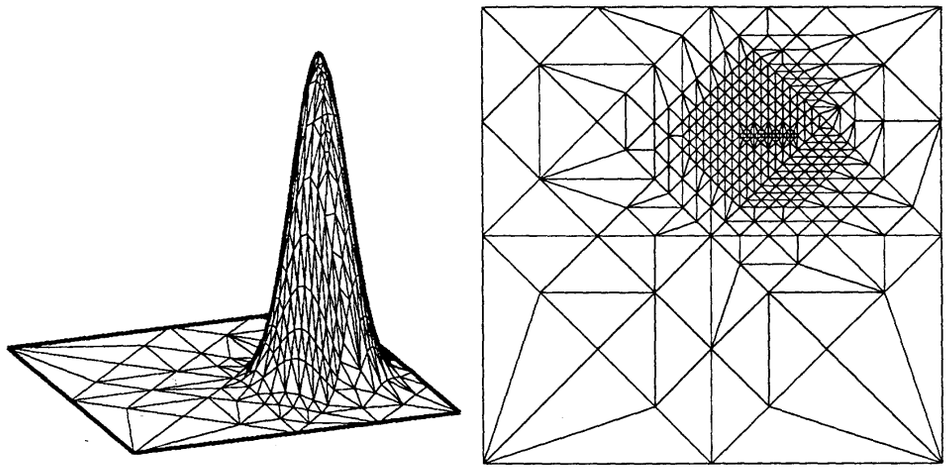


FIG. 20. Solution and triangulation at $t_2 = 1.55_{10} - 1$ (Example 8.4).

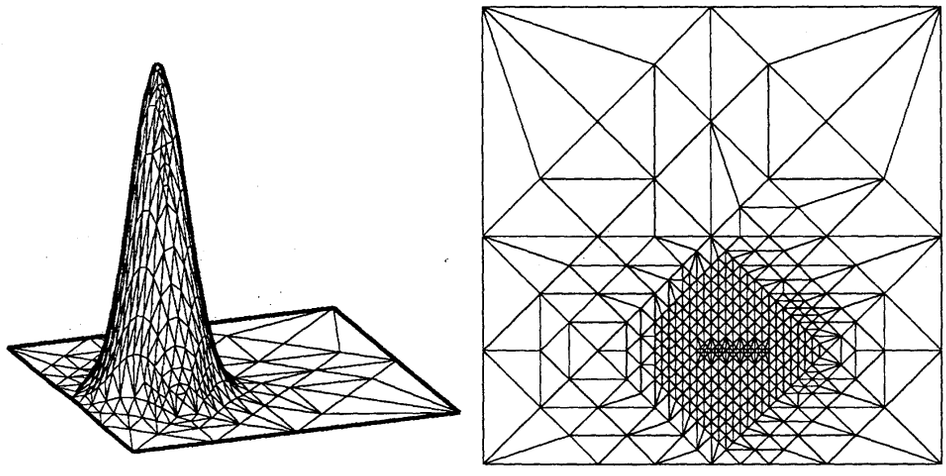


FIG. 21. Solution and triangulation at $t_{12} = 9.42_{10} - 1$ (Example 8.4).

EXAMPLE 8.4. *Rotating parabolic pulse.* (Due to ADJERID/FLAHERTY [2]). This model problem is a test for the adaptive choice of the triangulation. The equation is

$$\frac{\partial}{\partial t}u(t,x) = \Delta_x u(t,x) + f(t,x),$$

on the domain $\Omega = [0,1] \times [0,1]$, with

$$\begin{aligned} u(0,\cdot) &= u_{\text{exact}}(0,\cdot) \\ u(t,\cdot)|_{\partial\Omega} &= u_{\text{exact}}(t,\cdot)|_{\partial\Omega}, \qquad t > 0. \end{aligned}$$

The source f is chosen so that the solution is

$$u_{\text{exact}}(t,x) = 0.8 \exp\left(-80\left((x_1 - r_1(t))^2 + (x_2 - r_2(t))^2\right)\right),$$

where the midpoint of the pulse rotates as

$$r_1(t) = (2 + \sin(\pi t))/4, \quad r_2(t) = (2 + \cos(\pi t))/4.$$

The program was started at $t = 0.08$ with a required tolerance TOL $= 5.0_{10} - 2$ of the *relative* L^2–error; stop time was $t = 2$, i.e., the time of one rotation of the pulse. The computed solutions and the corresponding triangulations are shown in Fig. 20 at the time $t_2 = 1.55_{10} - 1$; in Fig. 21 at the time $t_{12} = 9.42_{10} - 1$. For $t_2(t_{12})$ the number of nodal points is 295(346), the number of triangles 575(675).

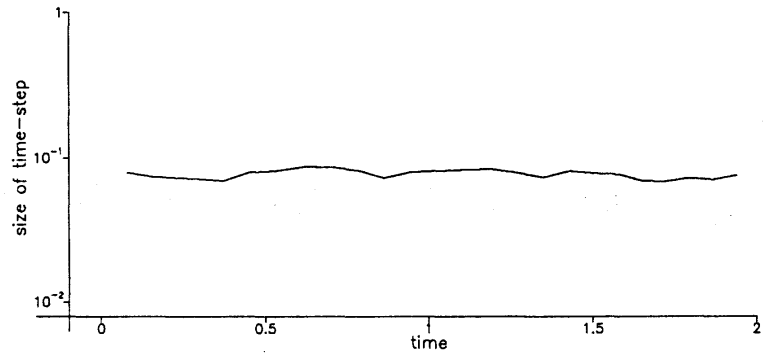


FIG. 22. Time step vs. time (Example 8.4).

Because of $\|u(t,\cdot)\|_{H^4} = \text{const.}$ in t one expects a nearly constant time step (cf. the considerations in Example 8.3). This actually occurs as shown in Fig. 22, $\tau \approx 0.08$.

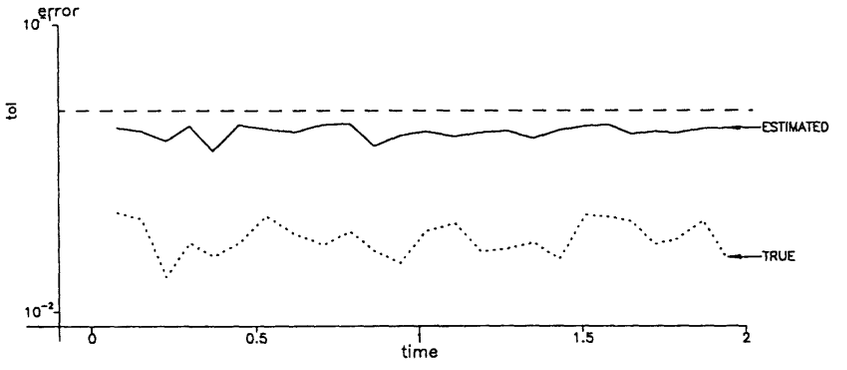


FIG. 23. Estimated and true error (Example 8.4).

The very reliable behavior of the error estimation is shown in Fig. 23. The estimated error consists of the time error estimate $\hat{\epsilon}_1$(TIME) and the spatial error estimate $[\delta_2]$(SPATIAL) (when time approximations u^1, u^2 are computed only). Fig. 24 shows the reliability of these components and the fact that the time error has to be 4 times more accurate than the spatial error, cf. Sec. 7.2.1. Moreover all estimates are nearly constant as is adequate for this problem.

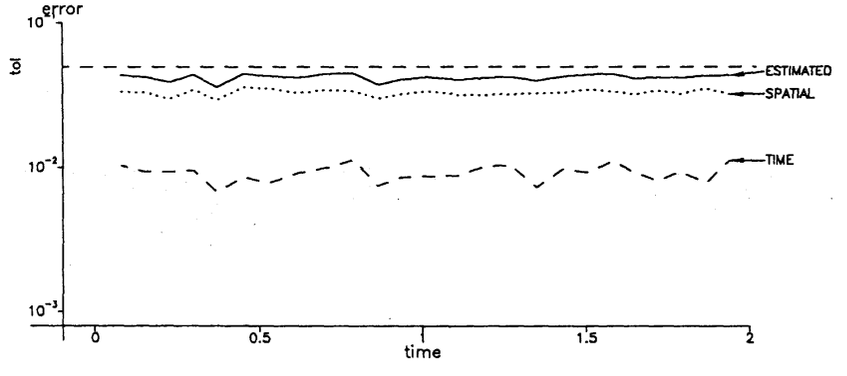


FIG. 24. The estimators $\hat{\epsilon}_1, [\delta_2]$ (Example 8.4).

The behavior of the adaptive triangulations is shown besides the examples of Figs. 20 and 21 in Fig. 25: The number of nodal points remains nearly

constant, only small fluctuations due to the changing geometric situation occur.

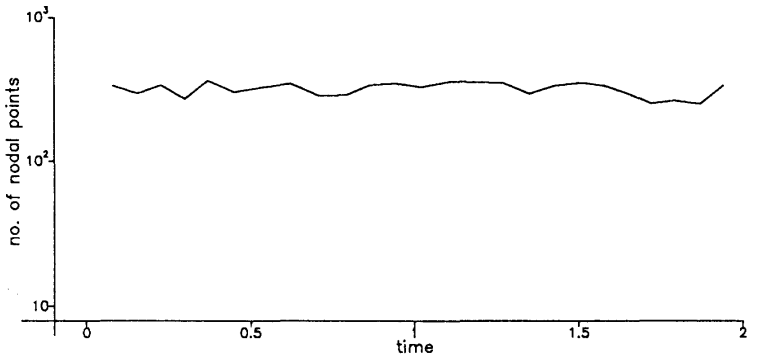


FIG. 25. Nodal points vs. time (Example 8.4).

Finally Fig. 26 shows the constancy of the depth of triangulation as well as the satisfactory behavior of the preconditioned CG–iteration process.

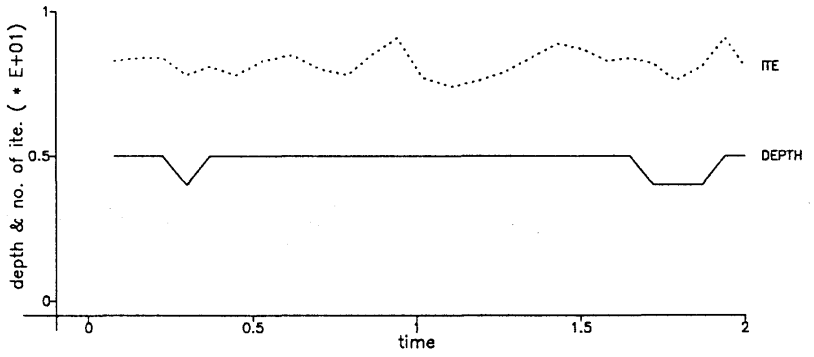


FIG. 26. Max. depth of triangulation and effective number of iterations (Example 8.4).

We again include a list of the behavior of KASTIO2 for different tolerances (Table XI). As in Example 8.3 the constancy of κ_1, κ_2 and κ_3 is observed. A discussion may be found in Example 8.3.

TABLE XI.
PERFORMANCE OF KASTIO2 FOR DIFFERENT TOLERANCES (EXAMPLE 8.4)

TOL	n_{step}	$[N]$	$N_{\max}$	N_{tot}	ϵ_{rel}	CPU	ϖ	κ_1	κ_2	κ_3
$1.0_{10}-1$	18	136	178	2431	$5.23_{10}-2$	153	22.4	2.8	0.2	0.6
$7.5_{10}-2$	22	181	221	3981	$4.17_{10}-2$	274	23.9	2.9	0.2	0.6
$*\ 5.0_{10}-2$	26	318	362	8265	$2.21_{10}-2$	664	27.2	3.0	0.2	0.6
$2.5_{10}-2$	40	878	1068	35095	$1.74_{10}-2$	3141	29.1	3.1	0.2	0.6

* run represented in Figs. 20–26.

EXAMPLE 8.5. *Inconsistent initial data and elliptic singularity.* This is the 2D version of Example 8.2. Two difficulties have to be dealt with: Inconsistency of the initial data u_0 with the homogeneous Dirichlet boundary conditions and an elliptic singularity in the stationary limit due to a reentrant corner.

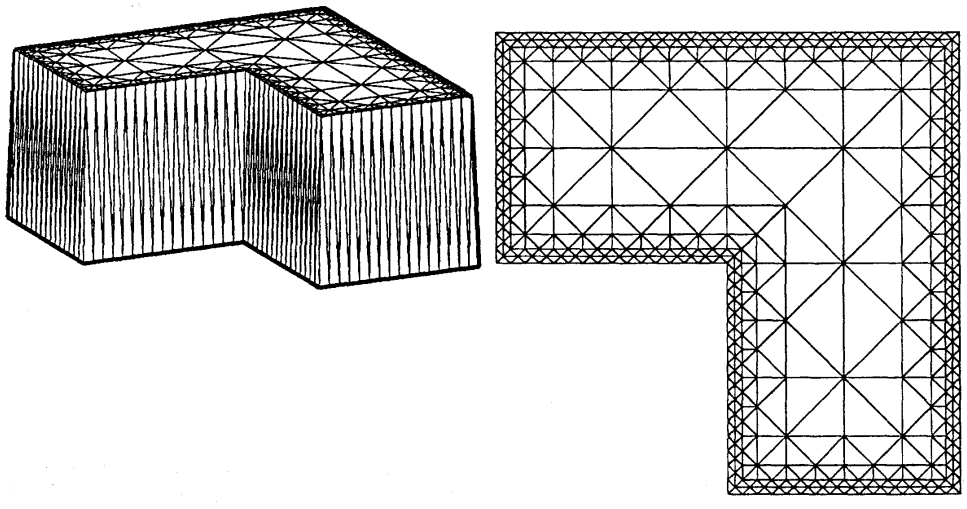


FIG. 27. Solution (reflected) and triangulation at $t_1 = 1.0_{10} - 6$ (Example 8.5).

The problem is

$$\frac{\partial}{\partial t}u(t,x) = \Delta_x u(t,x) + 30.0,$$

on the L-shaped domain $\Omega = [-1,1] \times [-1,1] \setminus [-1,0] \times [-1,0]$ with

$$\begin{aligned} u(0,\cdot) &\equiv -1.0 \\ u(t,\cdot)|_{\partial\Omega} &\equiv 0, \qquad t > 0. \end{aligned}$$

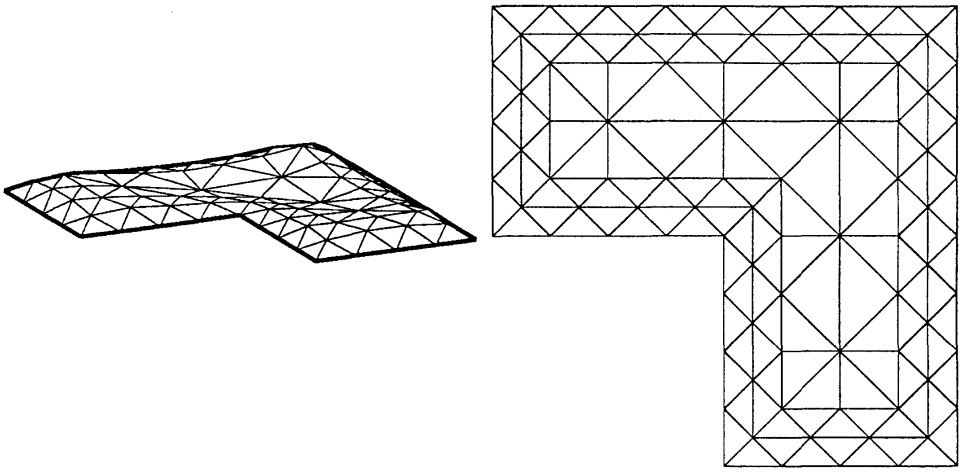


FIG. 28. Solution and triangulation at $t_{16} = 2.38_{10} - 2$ (Example 8.5).

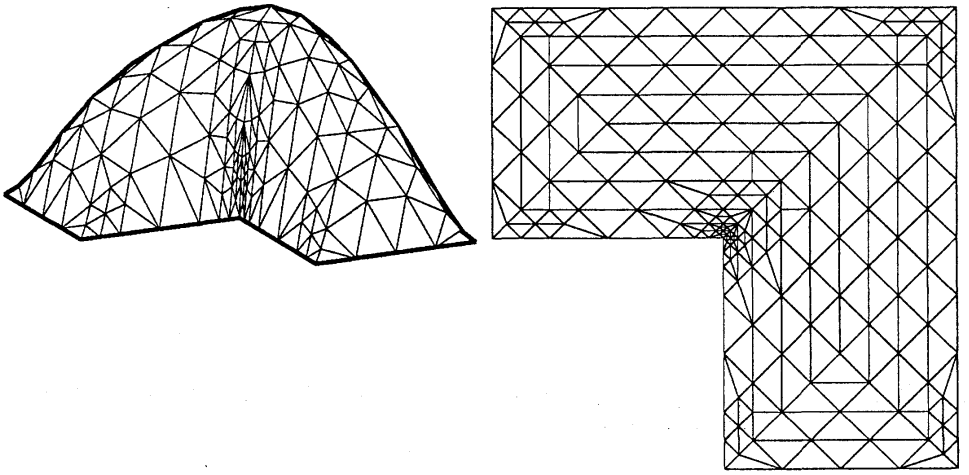


FIG. 29. Solution and triangulation at $t_{52} = 1.0_{10} + 6$ (Example 8.5).

The program was started at $t = 10^{-6}$ with a required tolerance $\mathsf{TOL} = 10^{-1}$; stop time was $t = 10^{+6}$. The computed solutions and the corresponding triangulations are shown in Fig. 27 at the starting time $t_1 = 1.0_{10} - 6$ (the solution has been reflected at the plane $u = 0$); those at the time $t_{16} = 2.38_{10} - 2$ in Fig. 28, the stationary solution at the stop time $t_{52} = 1.0_{10} + 6$

in Fig. 29. All have been scaled with respect to 1 in the u–direction. For $t_1(t_{16}, t_{52})$ the number of nodal points is 457(89, 180), the number of triangles 784(144, 316).

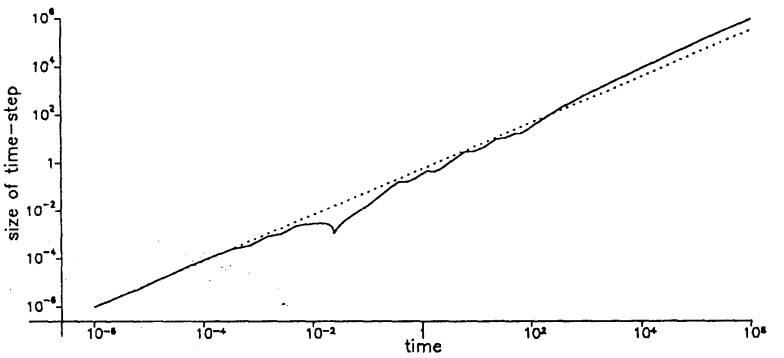


FIG. 30. Automatic increase of the time step (Example 8.5).

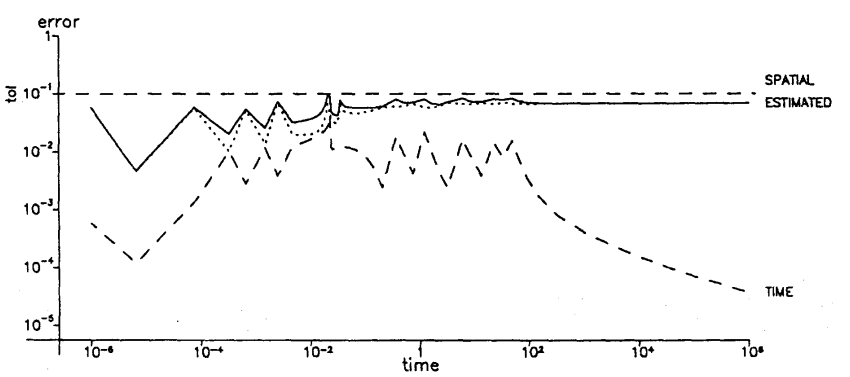


FIG. 31. The estimators $\hat{\epsilon}_1, [\delta_2]$ (Example 8.5).

Because of the inconsistency of u_0 (for which the L^2 projection is stabilized according to Section 7.2.5), quantitatively given as $u_0 \in \dot{H}^{1/2-\epsilon}$ with any $\epsilon > 0$, we get $\|u(t,\cdot)\|_{\dot{H}^4} \propto t^{1/4-2-\epsilon/2} = t^{-7/4-\epsilon/2}$. Thus by the same argumentation as in Example 8.3 the time error estimation would work optimal if we obtain an increase of the time step as $\tau \propto t^{7/8}$. The automatic increase

of the time step computed by KASTIO2 as shown in Fig. 30 resembles this optimal increase: the dotted line has slope 7/8.

The estimated error is shown in Fig. 31, the time–error component (TIME) nicely shows that KASTIO2 is able to detect stationary phases.

The development of nodal points is shown in Fig. 32. It shows that moving mesh techniques are not suited for this example since the number of nodal points are subject of serious changes during the computation, *if one computes with respect to a given accuracy.* The automatic decrease of nodal points during the transient before the source comes into play can be predicted a priori; one expects an optimal behavior of $n_{j_{\text{fin}}} \propto t^{-1/4}$. This is really achieved, since the dotted line in Fig. 32 is the fitting line with slope $-1/4$.

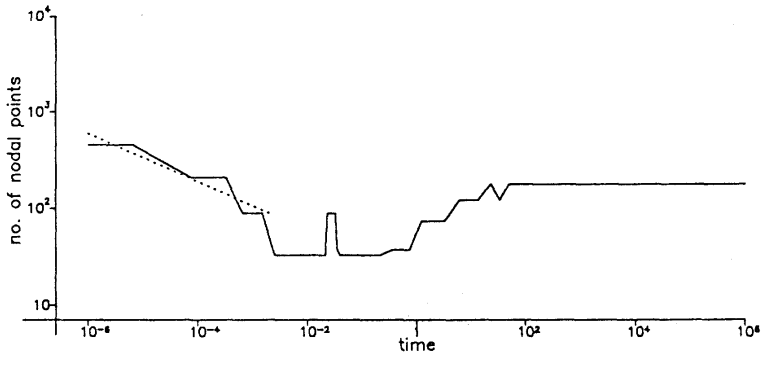


FIG. 32. Nodal points vs. time (Example 8.5).

The maximal depths of the triangulations at each time step are shown in Fig. 33. The elliptic singularity gets influence approximately from $t = 1.0_{10} - 1$. From then the effective number of CG–iterations is no longer bounded independently of the depth. This is explained by Remark 6.7. The elliptic singularity given by the reentrant corner with inner angle of $\beta = \frac{3}{2}\pi$ gives $H^{1+\alpha}$–regularity with $\alpha < \pi/\beta = 2/3$. Thus the worst possible behavior of the condition number is $\kappa \propto j_{\text{depth}}^{3/2}$ in view of Remark 6.7; the average number of effective iterations should grow at most like $j_{\text{depth}}^{3/4}$.

This theoretically bound is actually attained in this example as shown in Fig. 34 (double logarithmical scale) for the average number of effective iterations from $t = 1.0_{10} - 1$ on: the dotted line has just the slope 3/4.

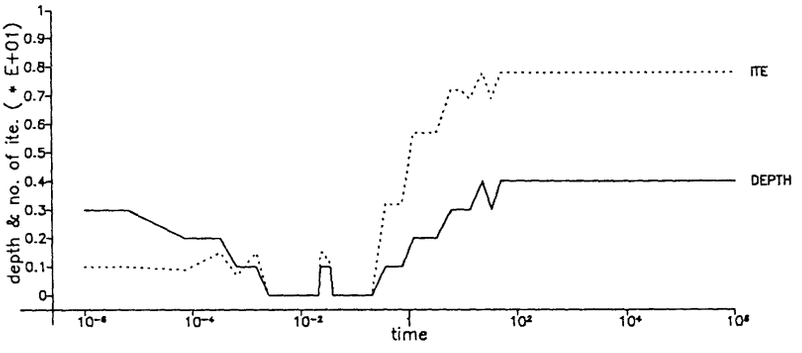


FIG. 33. Max. depth of triangulation and effective number of iterations (Example 8.5).

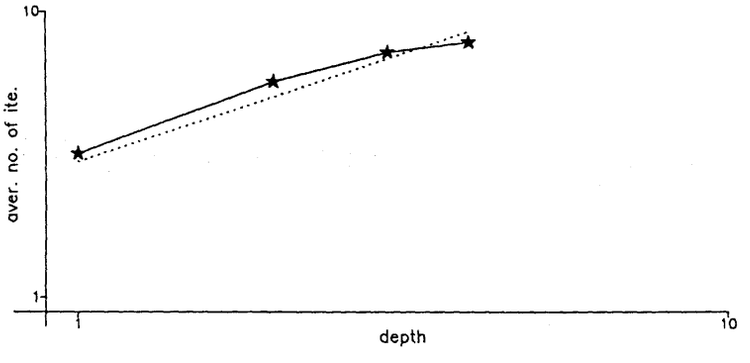


FIG. 34. Average no. of effect. iterations vs. depth ($t > 1.0_{10} - 1$) (Example 8.5).

We close this example with a list of the behavior of KASTIO2 for different tolerances (Table XII).

TABLE XII.
PERFORMANCE OF KASTIO2 FOR DIFFERENT TOLERANCES (EXAMPLE 8.5)

TOL	n_{step}	$[N]$	$N_{\max}$	N_{tot}	CPU	ϖ	κ_1	κ_2	κ_3
$*\ 1.0_{10}-1$	52	104	457	5378	208	11.0	3.5	0.5	1.6
$7.5_{10}-2$	67	152	961	10173	443	12.8	3.4	0.6	1.8
$5.0_{10}-2$	72	277	1977	19873	1210	18.0	3.4	0.5	1.6
$2.5_{10}-2$	93	696	4017	64969	6265	29.7	3.2	0.4	1.5

* run represented in Figs. 27–34.

9. A Real Life Application: Hyperthermia

In order to prove the applicability of our method to real life problems, which *combine* the difficulties of complex problem geometry, discontinuous coefficients etc., we will present the solution of the so–called *Bio–Heat–Transfer equation* (BHT equation) in the framework of *hyperthermia.*

Hyperthermia, i.e., the heating of tissue to temperatures approximately above 42 °C, is a recently developed clinical method for cancer therapy. It allows *in combination with radiotherapy* an improvement of the local control of the tumor. The deep heating of tissue is obtained by an electric field (E–field), which is generated by the radio waves of four antenna pairs. Their parameters (frequency 60 – 120 MHz, phase and amplitude) have to be selected appropriately. To allow the clinician to ask the somehow simplified question:

> "What is the probability of an effective hyperthermia for *my patient* and what treatment approach, which parameters would be optimal ?"

one has to be able to simulate the treatment in a planning phase. This planning phase should include:

- optimization of the individual treatment plan
- thermal tomography (estimation of the temperature distribution from some simple point measurements)
- thermal dosimetry

(cf. Wust et al. [50]). Thus it is essential to solve effectively and robust the BHT equation, which models the temperature distribution for a given E–field. For future applications it could be even desirable to solve the BHT equation so effectively, that the computation could be done within clinical tolerances *on line.* This would allow the clinician to control the treatment *interactively* in combination with the above mentioned thermal tomography, which itself requires the solution of the BHT equation.

We will show for a set of real life data, that our method would in principle allow such an on line computation on a workstation. We will present computations for 2D cross sections generated by computer tomography (CT) data.

9.1. The Bio–Heat–Transfer Equation

The BHT equation was developed 1948 by Pennes [43] to model the heat transport in live tissue. A characteristic is a local, isotropic blood flow term.

In the case of hyperthermia the BHT equation reads as (cf. [47]):

$$\text{i)} \qquad \varrho(x)c(x)\frac{\partial T(t,x)}{\partial t} = \operatorname{div}\left(\kappa(x)\ \operatorname{grad} T(t,x)\right) - \\ - \varrho_b c_b \varrho(x)\omega(x)\left(T(t,x) - T_a\right) + \\ + \frac{1}{2}\sigma(x)|E(t,x)|^2$$

(BHT)

$$\text{where } t \in [0, T_{\text{fin}}],\ \ x \in \Omega$$

$$\text{ii)} \quad -\kappa(x)\frac{\partial}{\partial n}T(t,x)\Big|_{x\in\partial\Omega} = h\left(T(t,x) - T_{\text{bolus}}\right)\Big|_{x\in\partial\Omega}$$

$$\text{iii)} \qquad T(0,x) = T_a.$$

Here we denote, using SI–units:

$\varrho(x)$, ϱ_b	$\left[\frac{\text{kg}}{\text{m}^3}\right]$	:	density of tissue, resp. blood.
$c(x)$, c_b	$\left[\frac{\text{J}}{\text{kg °C}}\right]$	:	specific heat of tissue, resp. blood.
$T(x,t)$, T_a, T_{bolus}	[°C]	:	temperature of tissue, of arterial blood, resp. of bolus.
$\kappa(x)$	$\left[\frac{\text{W}}{\text{m °C}}\right]$	:	thermal conductivity of tissue.
$\omega(x)$	$\left[\frac{\text{m}^3}{\text{kg s}}\right]$	:	blood flow in tissue (perfusion).
$\sigma(x)$	$\left[\frac{1}{\text{m}\,\Omega}\right]$	:	electric conductivity of tissue.
$\lvert E(t,x)\rvert$	$\left[\frac{\text{V}}{\text{m}}\right]$	:	magnitude of the electric field.
h	$\left[\frac{\text{W}}{\text{m}^2\ \text{°C}}\right]$	:	heat flow at the boundary (body surface).

Figure 35 shows the CT–data of a rectum malignancy of a patient, who has been treated at the Klinikum Rudolf Virchow, Freie Universität Berlin.

The data of the involved tissues are given in Table XIII. The heat flow h for the Cauchy boundary condition (BHT.ii) is assumed to be

$$h = 45\frac{\text{W}}{\text{m}^2\ \text{°C}}$$

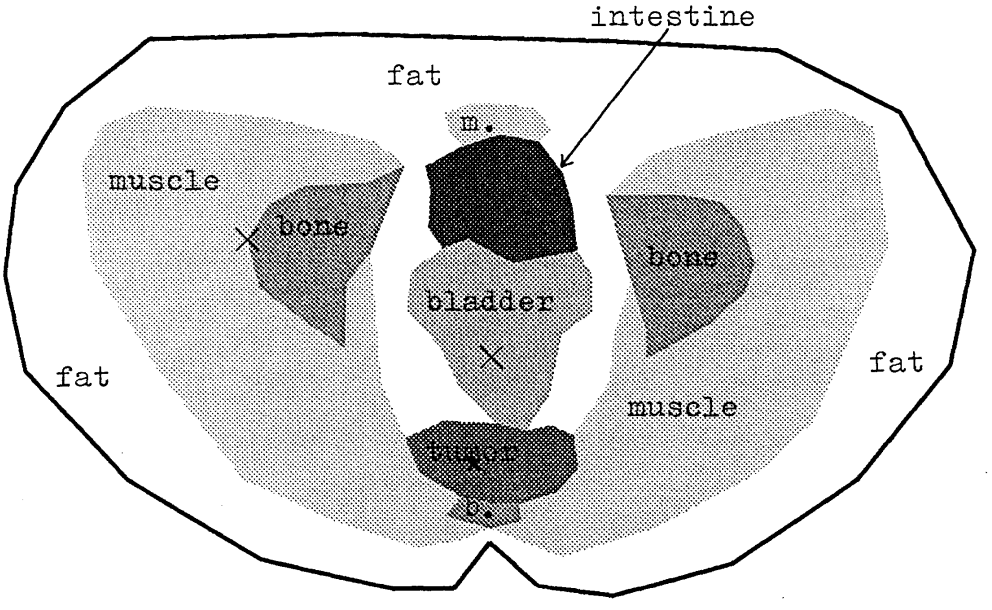


FIG. 35. CT-data of a rectum malignancy.

TABLE XIII.
DATA OF TISSUES

tissue	$\varrho[10^3\,\mathrm{kg/m^3}]$	$c[10^3\,\mathrm{J/kg\,°C}]$	$\kappa[\mathrm{W/m\,°C}]$	ω[ml/100g per min]	$\sigma[1/\mathrm{m}\,\Omega]$
blood	1.0	3.72	—	—	—
fat	0.9	2.36	0.210	5	0.21
muscle	1.0	3.72	0.642	20	0.80
bone	1.6	1.41	0.436	5	0.02
intestine	1.0	3.81	0.550	30	0.60
bladder	1.0	3.98	0.561	30	0.20
tumor	1.0	3.72	0.642	20	0.80

and the bolus temperature (i.e. the temperature of the water bolus, which is cooling the patient) is assumed to be

$$T_{\text{bolus}} = 25\,^{\circ}\text{C}.$$

As temperature of the arterial blood we take

$$T_a = 37\,^{\circ}\text{C}.$$

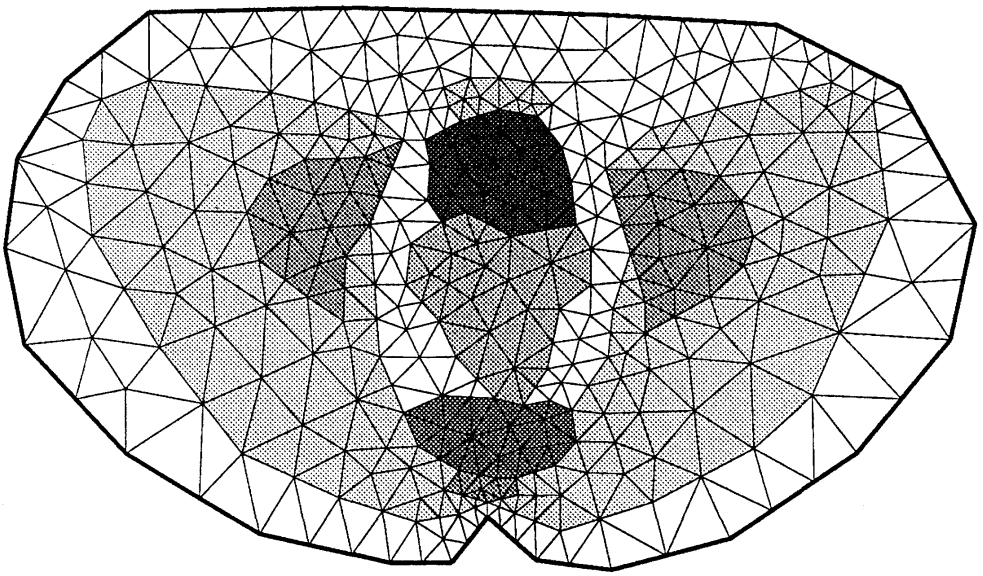


FIG. 36. Initial triangulation $\mathcal{T}_0$ of the CT cross section.

The initial triangulation $\mathcal{T}_0$ (Figure 36) (351 nodal points, 642 triangles) of the CT cross section was created by TRIGEN from the PLTMG–package of BANK [6]. On this triangulation an optimal E–field was computed by the second author of [51], which is shown in Figure 37 and which we will use for our example. The magnitude of the E–field ranges from 217.9 V/m in the left and right muscle/fat regions up to 628.2 V/m in the tumor/bladder region.

9.2. COMPUTATIONAL DETAILS FOR THE BHT EQUATION

9.2.1. Continuity Conditions at Tissue Boundaries

Since the coefficient $\kappa(x)$ has a jump discontinuity at the tissue boundary (e.g., muscle/bone), one can find in the literature — in order to give the

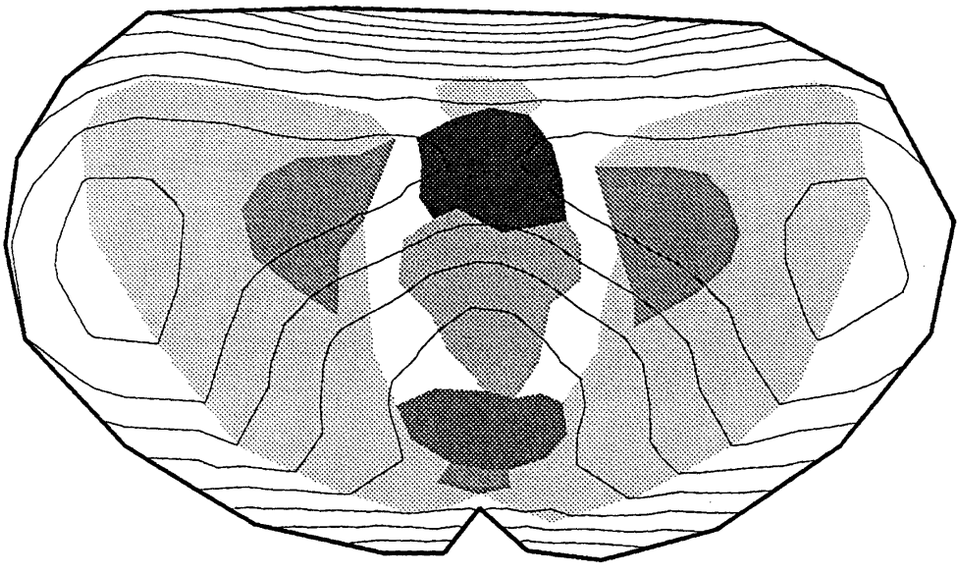


FIG. 37. Level lines of the E–field.

BHT equation a classical meaning – the additional conditions

$$\begin{aligned} T_1 &= T_2 \\ \kappa_1 \frac{\partial T_1}{\partial n} &= \kappa_2 \frac{\partial T_2}{\partial n}\Big|_\Gamma , \end{aligned}$$

where Γ is the boundary between tissue no. 1 and tissue no. 2 with conductivities κ_1 resp. κ_2.

However, our elliptic operator description of Section 1.1 together with Theorem 1.1 and in turn our algorithm *does not need* those conditions. This relies on the weak formulation, which is also the base for the FEM method — this method therefore *implicitly* realizes the continuity conditions, another feature, which distinguishes the FEM approach.

9.2.2. Time Discretization and Preconditioning

The abstract setting of the BHT equation is

$$\phi(x)\frac{\partial u(t,x)}{\partial t} + A(x,\partial)u(t,x) = f(x). \tag{9.1}$$

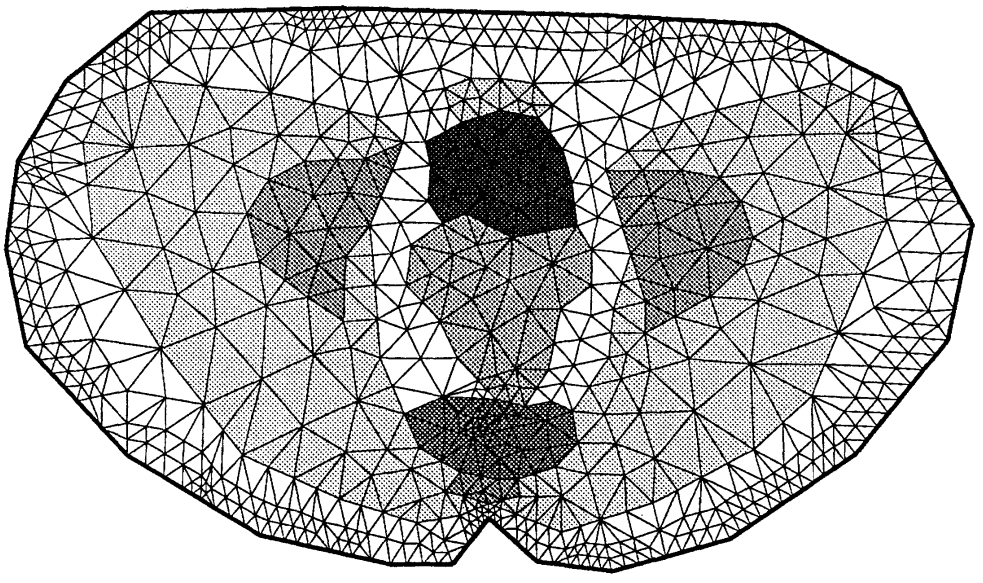


FIG. 38. Triangulation at time step 10, t = 13 min 11 s.

We briefly discuss the effect of $\phi \not\equiv 1$. Since assumptions 2 & 3 of Section 1.1 are fulfilled the semigroup setting of (9.1) is given by

$$\Phi u' + Au = f \tag{9.2}$$

with the bounded positive selfadjoint operator

$$\begin{aligned} \Phi : L^2(\Omega) &\rightarrow L^2(\Omega) \\ u &\mapsto \phi u. \end{aligned}$$

Since Φ^{-1} is bounded positive selfadjoint as well we obtain the equivalence of (9.2) and

$$\hat{u}' + \Phi^{-1/2} A \Phi^{-1/2} \hat{u} = \Phi^{-1/2} f,$$

with the transformed $\hat{u} = \Phi^{1/2} u$. Now the operator $\Phi^{-1/2} A \Phi^{-1/2}$ has the same properties as A. Note that a likely transformation by dividing equation (9.1) through $\sqrt{\phi}$ is impossible, since the principal part of $A(x, \partial)$ would loose its divergence form. Taking the time discretization (2.18) for the transformed $\hat{u}$

we get after back-transformation:

$$\begin{aligned} \text{i)} \quad u^1 &= (\Phi + \tau A)^{-1}(\Phi u_0 + \tau f), \\ \text{ii)} \quad \eta_1 &= \frac{1}{2}\tau A(\Phi + \tau A)^{-1}(u^1 - u^0), \\ \text{iii)} \quad u^2 &= u^1 + \eta_1. \end{aligned}$$

All results are valid as if $\phi \equiv 1$.

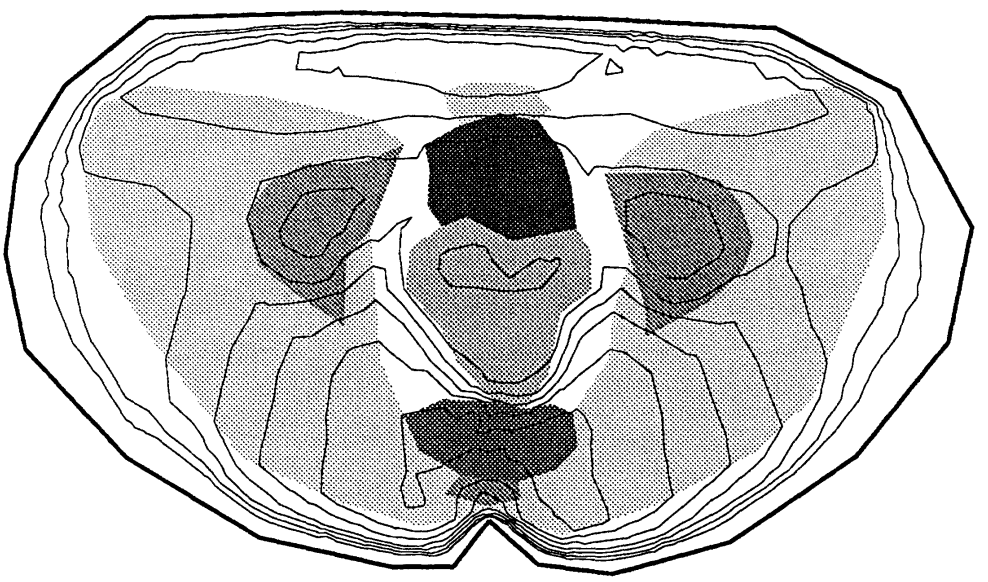


FIG. 39. Isothermals (37 - 43°C) at time step 10, t = 13 min 11 s.

We now have to find a preconditioner for the operator

$$\Lambda_\Phi = \frac{1}{1+\tau}\Phi + \frac{\tau}{1+\tau}A.$$

We estimate with Λ of Section 6.2.4

$$\min(1, \phi_{\min})(\Lambda u, u) \leq (\Lambda_\Phi u, u) \leq \max(1, \phi_{\max})(\Lambda u, u)$$

for $u \in \dot{H}^2$. Thus we can take the same preconditioner $(\Theta_H)_j$ for $(\Lambda_\Phi)_j$ as for Λ_j. The conditioner number will grow at most like

$$\kappa((\Theta_H)_j(\Lambda_\Phi)_j) \leq \frac{\max(1, \phi_{\max})}{\min(1, \phi_{\min})}\kappa((\Theta_H)_j\Lambda_j). \tag{9.3}$$

9.2.3. Time Scaling and Choice of $\bar{q}$

The grow factor of the condition number in (9.3) has for the above example the value

$$\frac{\max(1, 3.98 \cdot 10^6)}{\min(1, 2.12 \cdot 10^6)} = 3.98 \cdot 10^6,$$

which is *not feasible*. The reason for this big value is the comparison with the value 1 in the denominator. Now we make use of the possibility of a *time-scaling*: Introducing $t = t_{\text{scal}} \cdot \hat{t}$ we get

$$\varrho(x)c(x)\frac{\partial T(t,x)}{\partial t} = \frac{\varrho(x)c(x)}{t_{\text{scal}}}\frac{\partial T(\hat{t},x)}{\partial \hat{t}}.$$

In our example we choose

$$t_{\text{scal}} = 3.0 \cdot 10^6$$

and obtain the grow factor

$$\frac{\max(1, 1.33)}{\min(1, 0.71)} = 1.88.$$

Physically this scaling means that we take $3.0 \cdot 10^6$ s as time unit.

For the correct choice of $\bar{q}$, i.e., the version of the Helmholtz preconditioner of Section 6.2, we observe that exactly Case II of Section 6.2 is the case, because of $\Gamma_D = \emptyset$ and $q_{\min} > 0$. We thus have to use Version II of Section 6.2. In our example

$$q_{\min} = 2790$$

and

$$q_{\max} = 18600,$$

hence $\bar{q} = \sqrt{q_{\min} q_{\max}} = 7200$ is the correct value.

REMARK 9.1. Even in the case of Dirichlet boundary conditions this choice of $\bar{q}$ would be preferable compared to the choice $\bar{q} = 0$. In this case $\Gamma_D = \partial\Omega$, $\Gamma_C = \emptyset$ would yield just the case of doubt mentioned in Section 6.2.3. Hence we would have to refer to the decision criterion (6.17) of Section 6.2.3: With

$$d_\Omega = 0.24[\text{m}]$$

for the width of the vertical strip, i.e., the depth of the body, and

$$\delta = 0.21 \left[\frac{\text{W}}{\text{m } ^\circ\text{C}}\right],$$

we would get

$$q_{\min} = 2790 \left[\frac{\mathrm{W}}{\mathrm{m}^3\,^\circ\mathrm{C}}\right] \gg 7.29 \left[\frac{\mathrm{W}}{\mathrm{m}^3\,^\circ\mathrm{C}}\right] = \frac{2\delta}{d_\Omega^2}.$$

Hence we would obtain a reduction of the condition number by a factor of $3.8 \cdot 10^2$ by using $\bar{q} = 7200$ instead of $\bar{q} = 0$. This reduction can be observed in numerical examples.

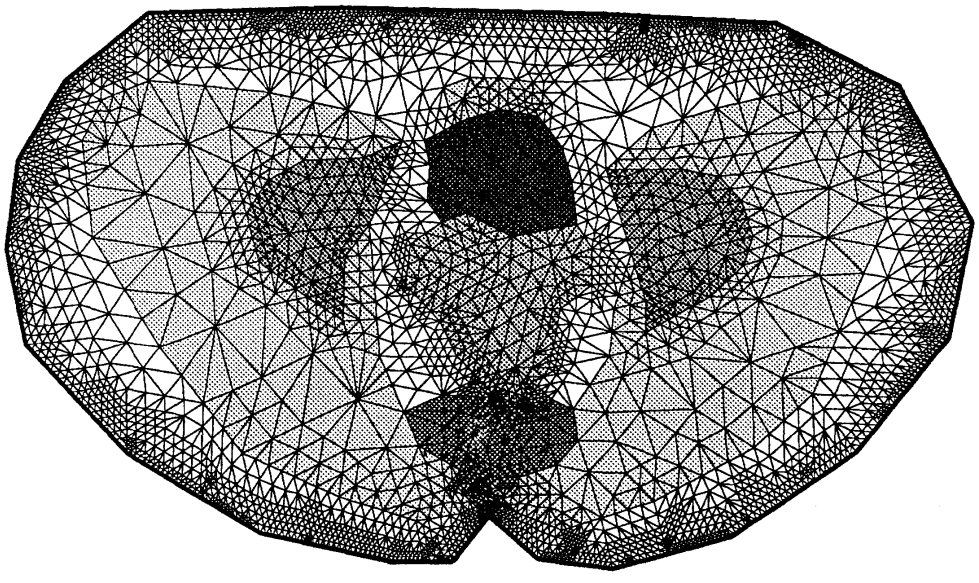


FIG. 40. Triangulation at time step 16, t = 53 min 14 s.

9.2.4. A Simple Comparison Approximation in the Interior of Tissues

In the interior of tissues we obtain a quite good approximation of the BHT equation considering the effect of the diffusion as another cooling term proportional to the heating:

$$\operatorname{div}(\kappa \operatorname{grad} T) \approx -\frac{T - T_a}{T_\infty - T_a} \triangle Q,$$

where T_∞ denotes the temperature of the stationary solution and $\triangle Q$ the magnitude of power loss through diffusion, thus

$$\triangle Q = \frac{1}{2}\sigma |E|^2 - \varrho_b c_b \varrho T_\infty.$$

Our approximation now reads as

$$T(t) = T_\infty - (T_\infty - T_a)\exp\left(-\frac{\sigma|E|^2}{2\varrho c(T_\infty - T_a)}t\right). \tag{9.4}$$

We stress that this approximation is valid in the interior of tissue only — due to the comparatively high specific heat and low thermal conductivity. It requires the solution of the *stationary* problem. The validity will be backed by our FEM approximation of the time dependent BHT equation.

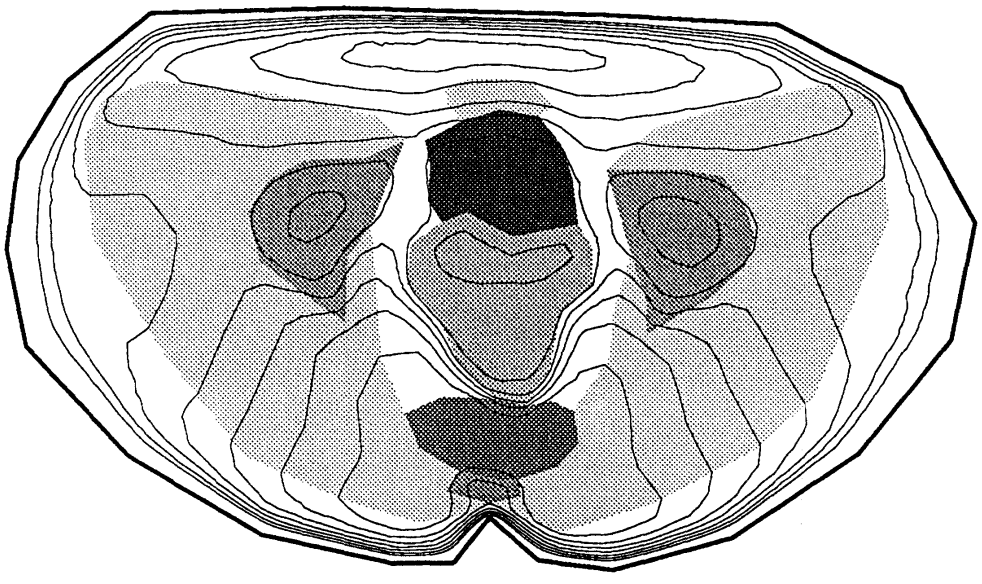


FIG. 41. Isothermals (37 – 43°C) at time step 16, t = 53 min 14 s.

9.3. COMPUTATIONAL RESULTS

Here we present the computational results of the program KASTIO2 for the BHT equation with the above described data. We have chosen the accuracy

$$\mathrm{TOL} = 7.5 \cdot 10^{-2},$$

which corresponds to an accuracy of the temperature of ±0.25 °C, *assuming an equidistributed error.*

We did the computations until a treatment time of 1 h = 3600 s.

The triangulation computed at the problem time $t = 13$ min 11 s (time step 10) is show in Figure 38. This triangulation contains 724 points and 1304 triangles. The refinement occurred mainly at the boundary where the steepest temperature differences can be found. The corresponding solution is show in Figure 39, where the isothermals are plotted for 37 – 43 °C in 1 °C steps.

The triangulation computed at the problem time $t = 53$ min 14 s (time step 16) is show in Figure 40. This triangulation contains 3688 points and 6922 triangles. The refinement occurred here also at the critical tissue boundaries like muscle/bone, fat/bladder and bladder/tumor. The corresponding solution is show in Figure 41, where the isothermals are plotted for 37 – 43 °C in 1 °C steps. We observe that the region above 43 °C at the tumor has spread out, now surrounding it nearly. Also the top fat region, which is a region of high electric field (cf. Figure 39), did get an temperature increase of about 2 °C.

Figure 42 shows the development of the time step τ during the 16 time steps, Figure 43 the increase of the number of nodal points.

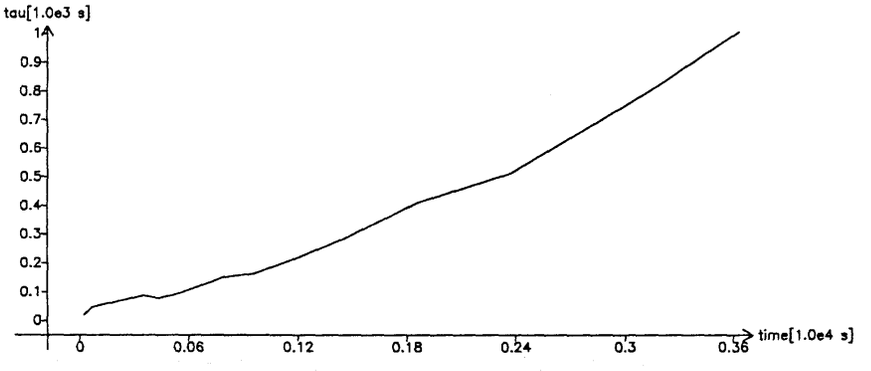


FIG. 42. Development of time step.

Figure 44 shows the behavior of the error estimator.

We now discuss the possibility of thermal tomography. In Figure 35 we have marked three measurement points. The computed heating of the tumor and the bladder point are shown in Figure 45. We observe that after approximately 16 min treatment time the temperature has reached its stationary value for this point of the tumor. The dotted comparison lines show the result of our simple approximation of Section 9.2.4, which should be valid

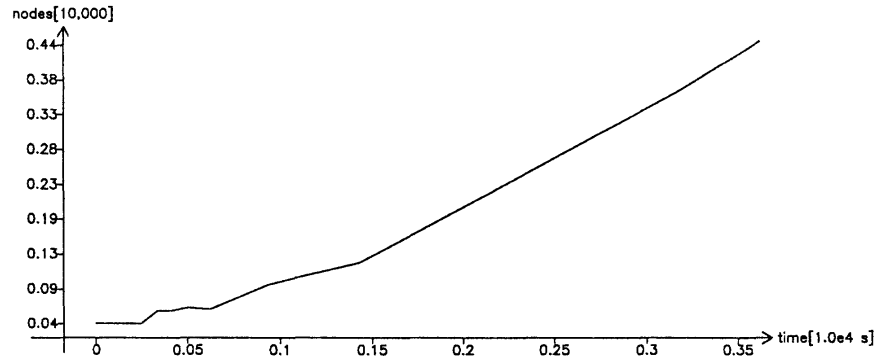


FIG. 43. Increase of number of nodal points.

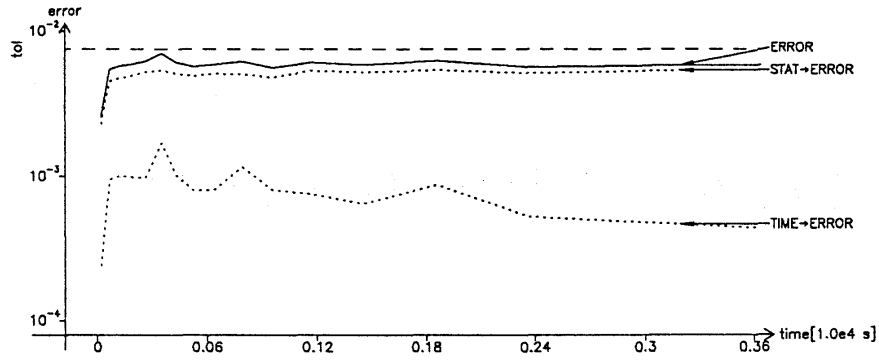


FIG. 44. Behavior of error estimation

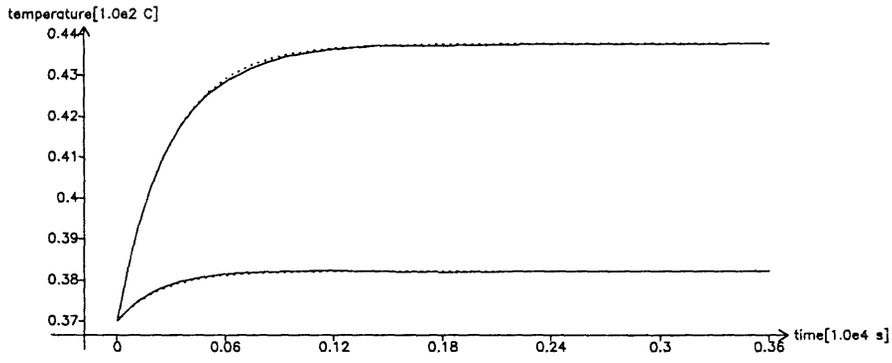


FIG. 45. Heating of tumor (top) and bladder (bottom); (—) KASTIO2 (···) Sec. 9.2.4

in the interior of tissues. The validity of this approximation shows that the temperature increases the first couple of minutes nearly linear, thus allowing an accurate measurement of the power deposit through $|E|$. This is an essential feature for the clinician, since the magnitude of $|E|$ can be only computed in advance by the antenna data modulo an unknown factor.

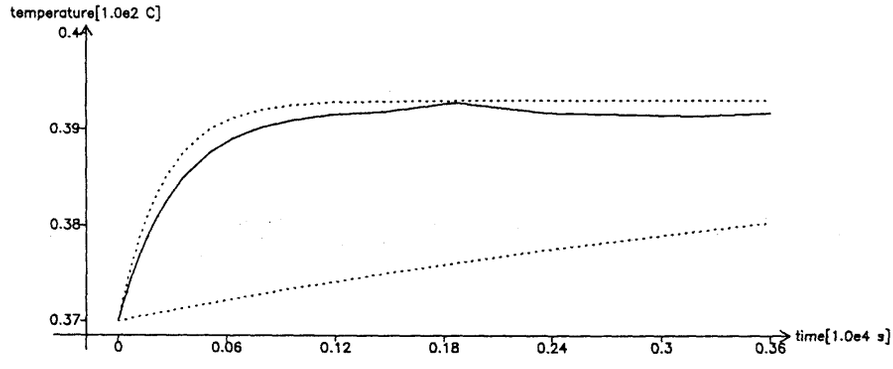


FIG. 46. Heating at the boundary muscle/bone ; (—) KASTIO2 (···) Sec. 9.2.4

Figure 46 shows as expected that the comparison model is not valid at tissue boundaries, here a point at the muscle/bone boundary. The two dotted lines are extreme cases of the comparison model. Thus a computation of the

time dependent BHT equation is *indispensable* if one is interested in the whole temperature distribution.

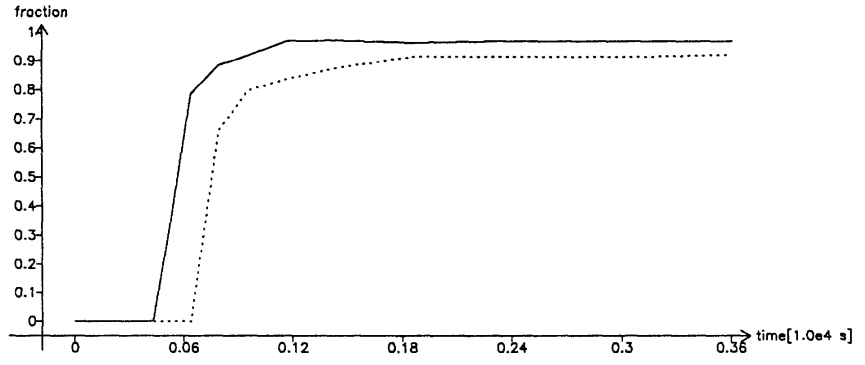


FIG. 47. Clinical success: Fraction of tumor above 42.5°C (—) and above 43°C (···).

The clinician defines the success of the heating in terms of the fraction of the tumor which is heated above 43 °C resp. 42.5 °C. These two fractions are shown in time development in Figure 47. It can be seen that after 16 min treatment time more than 90% of the tumor are heated above 42.5 °C and after 30 min treatment time more than 90% are heated above 43 °C.

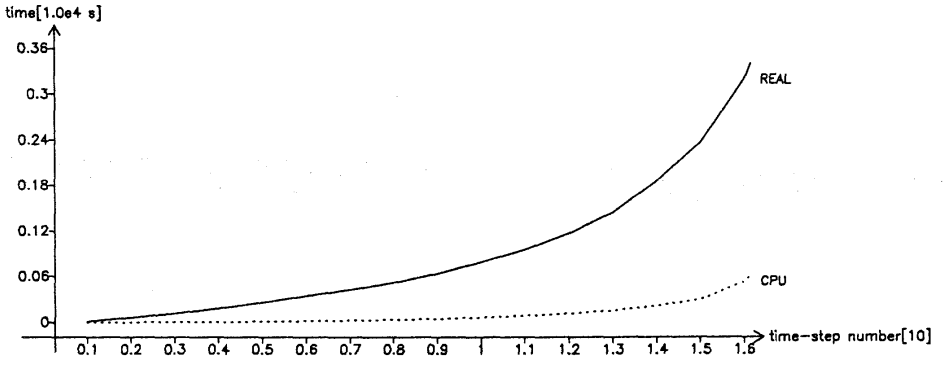


FIG. 48. Development of real treatment time (—) and cpu-time (···).

We close this section and the paper by showing that the computation could be done in principle *on line* on the workstation used: Figure 47 contains the

treatment time and the cpu time versus the time step number, we observe an increasing gain of time to react: For 53 min 14 s treatment time the computation on the workstation (SPARC–station 1+) ended after 8 min 53.6 s, hence we have gained 44 min 20 s.

REFERENCES

[1] Abramowitz, M., Stegun, I.A.: *Handbook of Mathematical Functions.* National Bureau of Standards (1964).

[2] Adjerid, S., Flaherty, J.E.: *A Local Refinement Finite-Element Method for Two-Dimensional Parabolic Systems.* SIAM J. Sci. Stat. Comput. 9, 792–811 (1988).

[3] Babuška, I., Kellogg, R.B., Pitkäranta, J.: *Direct and Inverse Estimates for Finite Elements with Mesh Refinements.* Numer. Math. 15, 447–471 (1979).

[4] Babuška, I., Rheinboldt, W.C.: *Error Estimates for Adaptive Finite Element Computations.* SIAM J. Numer. Anal. 15, 736–754 (1978).

[5] Bank, R.E.: *Analysis of a Local a posteriori Error Estimate for Elliptic Equations.* In: Accuracy Estimates and Adaptive Refinements in Finite Element Computations (I. Babuška et al. eds.), John Wiley & Sons, Chichester, New York, Brisbane, Toronto, Singapore (1986).

[6] Bank, R.E.: *PLTMG User's Guide.* Edition 5.0, Tech. Rep., Department of Mathematics, University of California at San Diego (1988).

[7] Bank, R.E.: *PLTMG: A Software Package for Solving Elliptic Partial Differential Equations.* Philadelphia, SIAM (1990).

[8] Bank, R.E., Dupont, T., Yserentant, H.: *The Hierarchical Basis Multigrid Method.* Numer. Math. 52, 427-458 (1988).

[9] Bank, R.E., Scott, L.R.: *On the Conditioning of Finite Element Equations with Highly Refined Meshes.* SIAM J. Numer. Anal. 6, 1383–1394 (1989).

[10] Bank, R.E., Sherman, A.H.: *Algorithmic Aspects of the Multi-Level Solution of Finite Element Equations.* Report CNA–144, Center for Numerical Analysis, The University of Texas at Austin (1978).

[11] Bank, R.E., Sherman, A.H., Weiser, A.: *Refinement Algorithms and Data Structures for Regular Local Mesh Refinement.* In: Scientific Computing (eds.: R. Stepleman et al.), Amsterdam: IMACS/North Holland, 3–17 (1983).

[12] Bank, R.E., Weiser, A.: *Some A Posteriori Error Estimators for Elliptic Partial Differential Equations.* Math. Comp. 44, 283–301 (1985).

[13] Bieterman, M., Babuška, I.: *The Finite Element Method for Parabolic Equations. I. A Posteriori Error Estimation.* Numer. Math. 40, 339–371 (1982).

[14] Bieterman, M., Babuška, I.: *The Finite Element Method for Parabolic Equations. II. A Posteriori Error Estimation and Adaptive Approach.* Numer. Math. 40, 373–406 (1982).

[15] Bieterman, M., Babuška, I.: *An Adaptive Method of Lines with Error Control for Parabolic Equations of the Reaction-Diffusion Type.* J. Comp. Phys. 63, 33–66 (1986).

[16] Bornemann, F.A.: *Adaptive Multilevel Discretization in Time and Space for Parabolic Partial Differential Equations.* Technical Report TR 89-7, ZIB (1989).

[17] Bornemann, F.A.: *An Adaptive Multilevel Approach to Parabolic Equations I. General Theory and 1D-Implementation.* IMPACT Comput. Sci. Engrg. 2, 279–317 (1990).

[18] Bornemann, F.A.: *An Adaptive Multilevel Approach to Parabolic Equations II. Variable-Order Time Discretization Based on a Multiplicative Error Correction.* IMPACT Comput. Sci. Engrg. 3, 93–122 (1991).

[19] Bramble, J.H., Pasciak, J.E., Xu, J.: *Parallel Multilevel Preconditioners.* Math. Comp. 55, 1–22 (1990).

[20] Ciarlet, P.G.: *The Finite Element Method for Elliptic Problems.* North–Holland Publ. Comp., Amsterdam–New York–Oxford (1978).

[21] Dautray, R., Lions, J.L.: *Mathematical Analysis and Numerical Methods for Science and Technology. Vol.2, Functional and Variational Methods.* Springer, Berlin–Heidelberg–New York (1988).

[22] Davis, S.F., Flaherty, J.E.: *An Adaptive Finite Element Method for Initial-Boundary Value Problems for Partial Differential Equations.* SIAM j. Sci. Stat. Comput. 3, 6–27 (1982).

[23] Deuflhard, P.: *Order and Stepsize Control in Extrapolation Methods.* Numer. Math. 41, 399–422 (1983).

[24] Deuflhard, P.: *Recent Progress in Extrapolation Methods for Ordinary Differential Equations.* SIAM Rev. 27, 505–535 (1985).

[25] Deuflhard, P., Leinen, P., Yserentant, H.: *Concepts of an Adaptive Hierarchical Finite Element Code.* IMPACT Comput. Sci. Engrg. 1, 3–35 (1989).

[26] Erdmann, B., Roitzsch, R., Bornemann, F.A.: *KASKADE, Numerical Experiments.* Technical Report TR 91–1, ZIB (1991).

[27] Eriksson, K., Johnson, C.: *Adaptive Finite Element Method for Parabolic Problems I: A Linear Model Problem.* Preprint 31, Department of Mathematics, University of Göteborg (1988).

[28] George, A., Liu, J.W.H.: *Computer Solution of Large Sparse Positive Definite Systems.* Prentice Hall, New Jersey (1981).

[29] Hille, E.: *Functional Analysis and Semi-Groups.* Amer. Math. Soc. Colloq. Publ. 31, New York (1948).

[30] Kačur, J.: *Application of Rothe's Method to Nonlinear Evolution Equations.* Mat. Čas. 25, 63–81 (1975).

[31] Kadlec, J.: *O reguljarnosti rešenija sadači Puassona na oblasti s granicej, lokol'no podobnoj granice vypukloj oblasti.* Czechoslovak Math. J. 14, 386–393 (1964).

[32] Kato, T.: *Perturbation Theory for Linear Operators.* Second Corrected Printing of the second Edition. Springer, Berlin–Heidelberg–New York (1984).

[33] Leinen, P.: *Ein schneller adaptiver Löser für elliptische Randwertprobleme auf Seriell- und Parallelrechnern.* Thesis, University of Dortmund (1990).

[34] Lubich, C.: *Discretized Operational Calculus Part I: Theory.* Institut für Mathematik und Geometrie, Universität Innsbruck, Institutsnotiz Nr. 4 (1984).

[35] Maubach, J.: *Iterative Methods for Non-Linear Partial Differential Equations.* Thesis, Catholic University of Nijmegen (1991).

[36] Miller, K., Miller R.N.: *Moving Finite Elements I.* SIAM J. Numer. Anal. 18, 1019–1032 (1981).

[37] Miller, K., Miller R.N.: *Moving Finite Elements II.* SIAM J. Numer. Anal. 18, 1033–1057 (1981).

[38] Nečas, J.: *Sur la coercivité des formes sesqui-linéaires elliptiques.* Rev. Roumaine Math. Pures Appl. 9, 47–69 (1964).

[39] Nečas, J.: *Application of Rothe's Method to Abstract Parabolic Equations.* Czech. Math. J. 24, 496–500 (1974).

[40] Nikiforov, A.F., Uvarov, V.B.: *Special Functions of Mathematical Physics.* Birkhäuser, Basel, Boston (1988).

[41] Nørsett, S.P.: *Restricted Padé Approximations to the Exponential Function.* SIAM J. Numer. Anal. 15, 1008–1029 (1978).

[42] Pazy, A.: *Semigroups of Linear Operators and Applications to Partial Differential Equations.* Springer, Berlin–Heidelberg–New York (1983).

[43] Pennes, H.H.: *Analysis of Tissue and Arterial Blood Temperatures in the Resting Human Forearm.* J. Appl. Physiol. 1, 93–122 (1948).

[44] Roitzsch, R.: *KASKADE User's Manual.* Technical Report TR 89–4, ZIB (1989).

[45] Roitzsch, R.: *KASKADE Programmer's Manual.* Technical Report TR 89–5, ZIB (1989).

[46] Rothe, E.: *Zweidimensionale parabolische Randwertaufgabe als Grenzfall eindimensionaler Randwertaufgaben.* Math. Ann. 102, 650–670 (1930).

[47] Seebaß, M.: *3D-Computersimulation der interstitiellen Mikrowellen-Hyperthermie von Hirntumoren.* Bericht Nr. CVR 1/90, Deutsches Krebsforschungszentrum Heidelberg (1990).

[48] Triebel, H.: *Höhere Analysis.* Verlag Harry Deutsch, Thun, Frankfurt a. M. (1980).

[49] Wathen, A.J.: *Realistic Eigenvalue Bounds for the Galerkin Mass Matrix.* IMA J. Numer. Anal. 7, 449–457 (1987).

[50] Wust, P., Nadobny, J., Felix, R., Deuflhard, P., John, W., Louis, A.: *Numerical Approaches to Treatment Planning in Deep RF-Hyperthermia.* Strahlenther. Onkol. 165, 751–757 (1989).

[51] Wust, P., Nadobny, J., Felix, R., Deuflhard, P., Louis, A., John, W.: *Strategies for Optimized Application of Annular-Phased-Array Systems in Clinical Hyperthermia.* Int. J. Hyperthermia 7, 157–173 (1991).

[52] Xu, J.: *Theory of Multilevel Methods.* Report No. AM 48, Department of Mathematics, Pennsylvania State University (1989).

[53] Xu, J.: *Iterative Methods by Space Decomposition and Subspace Correction: A Unifying Approach.* Report No. AM 67, Department of Mathematics, Pennsylvania State University (1990).

[54] Yserentant, H.: *On the Multi-Level Splitting of Finite Element Spaces.* Numer. Math. 49, 379–412 (1986).

[55] Yserentant, H.: *Hierarchical Bases in the Numerical Solution of Parabolic Problems.* Large Scale Scientific Computing, Deuflhard, P., Engquist, B. (eds.), Birkhäuser, Boston, 22–37 (1987).

[56] Yserentant, H.: *Two Preconditioners Based on the Multi-Level Splitting of Finite Element Spaces.* Numer. Math. 58, 163–184 (1990).

[57] Zegeling, P.A., Blom, J.G.: *A Note on the Grid Movement Induced by MFE.* Centrum voor Wiskunde en Informatica, Report NM–R9019, Amsterdam (1990)