

APNUM 408

Local uniform grid refinement and systems of coupled partial differential equations

Ron Trompert

CWI, P.O. Box 4079, 1009 AB Amsterdam, Netherlands

Abstract

Trompert, R., Local uniform grid refinement and systems of coupled partial differential equations, Applied Numerical Mathematics 12 (1993) 331–355.

In this paper we consider an adaptive grid method based on local uniform grid refinement applied to systems of coupled time-dependent PDEs. Local uniform grid refinement means that the PDEs are solved on a series of nested, uniform, increasingly finer subgrids which cover only a part of the domain. These subgrids are created up to a level of refinement where sufficient spatial accuracy is obtained and their location and shape is adjusted after each time step in order to follow the moving steep fronts. When a system of coupled PDEs is solved, the behavior of the local and global error associated with each separate PDE can be very different from one PDE to another. A refinement strategy based on a global error analysis has been developed which takes these differences into account. This refinement strategy aims at the domination of the global space error by the space discretization error at the finest subgrid.

Keywords. Partial differential equations; numerical mathematics; time-dependent problems; adaptive grid methods; error analysis.

1. Introduction

The local uniform grid refinement method is an adaptive grid method used to solve time-dependent partial differential equations (PDEs) with locally steep solutions. For such problems, a uniform space grid can be computationally very inefficient, since, to obtain an accurate approximation, such a grid would easily have to contain an excessive number of nodes, particularly in two and three space dimensions.

The main feature of local uniform grid refinement is that the PDEs are solved on a series of nested, uniform, cartesian, increasingly finer subgrids covering only that part of the domain where the spatial error is high. The PDEs are solved on each separate subgrid in a consecutive manner, from coarse to fine. The location and size of the subgrids are automatically adjusted at discrete times in order to follow the movement of the steep fronts. The generation of subgrids is continued until sufficient spatial accuracy is reached.

So far, local uniform grid refinement methods were proposed in a number of different varieties and applied to different kinds of PDEs. Here, we will not attempt to give a complete overview of the field. We will only briefly sketch some varieties of the local uniform grid

Correspondence to: R. Trompert, CWI, P.O. Box 4079, 1009 AB Amsterdam, Netherlands.

refinement method and provide some references for interested readers. The methods contained in [1–4] are applied to hyperbolic PDEs and use explicit time stepping techniques. The method proposed by Berger and Oliger in [3] employs subgrids which are rectangles which may be skewed with respect to the coordinate axes in order to align with the steep region of the solution. Subgrids having the same cell sizes can partially overlap each other in this method. In [1], Arney and Flaherty developed a method very similar to the one in [3] except that the subgrids here are created by cellular refinement, meaning that the fine grid cells are properly nested within coarser grid cells. Hence, these subgrids have a piecewise polygonal shape.

Local uniform grid refinement is combined with grid movement in [2,4]. In [4], a method proposed by Gropp uses subgrids which are rectangles having sides parallel to the coordinate axes and which are able to move as a whole with the moving steep fronts. In this method the subgrids are also allowed to overlap each other. In [2], Arney and Flaherty added grid movement to their method described in [1]. The grid nodes of the coarsest grid are able to move and the fine grid movement is induced by the movement of the coarsest grid. Local uniform grid refinement methods are also used to solve parabolic and elliptic PDEs in [5,6] and involve the implicit solution of systems in equations. The subgrids in [6] are piecewise polygonal and those in [5] are rectangles. In both [5,6] domain decomposition is applied to improve the performance on parallel computers.

Our previous work on this type of adaptive grid method is contained in [7–11]. The subgrids in our method have a piecewise polygonal shape and do not overlap. Our method is a static-regridding method which means that no grid movement is applied during a time step. The refinement strategy controlling the generation of subgrids in [9] is based on heuristic criteria while in [7,8,10,11] it is underlied by a comprehensive error analysis which has resulted in a so-called refinement condition. This condition has been designed so that when this condition is satisfied during the grid refinement process and the number of subgrids is fixed in time, then the spatial accuracy of the solution obtained with the adaptive grid method should be comparable to the spatial accuracy obtained using one uniform grid covering the entire spatial domain when the cell sizes of this uniform grid are identical to those of the finest subgrid in use in the adaptive grid method. The refinement strategy is designed to fulfill the refinement condition. Due to the refinement condition a convergence result as if a single uniform grid was used could be proved in certain model situations. The error analysis was carried out for the local uniform grid refinement method applied to time-dependent PDEs which after spatial discretization yield a system of ordinary differential equations (ODEs). However, when a system of coupled PDEs is solved, this need not be the case. It is known that the global and local error components associated with each separate PDE belonging to such a system can behave differently from one PDE to another. This means that, for example, the global error corresponding with one PDE can propagate in a different way to future time levels than the one associated with another PDE. With respect to the local error, this difference in behavior means that the local errors connected with different PDEs do not always behave in the same way when the time step size tends to zero. For this reason the refinement strategy has to be adapted to this more general class of PDEs. Moreover, in most of our previous work, the refinement strategy is aimed at controlling the spatial accuracy or global space error by, in some sense, controlling the local space error. This strategy performs satisfactorily but can be very restrictive, especially when the number of subgrids is large or the time step size very small. In this paper, the error analysis is redone for systems of coupled PDEs. From this, a more

general and much less restrictive refinement strategy is obtained aiming at controlling the spatial accuracy by estimating the global space error itself.

In Section 2 a brief outline will be given of our version of the local uniform grid refinement method. Section 3 deals with the mathematical formulation of the method needed for the error analysis. The results of the error analysis are given in Section 4. This section also considers the influence of a system of coupled PDEs on the behavior of the global and local error. The refinement strategy is discussed in Section 5. Three example problems were used to illustrate the performance of the method. The results of these tests are given in Section 6. Although the example problems involve two space dimensions, the error analysis, refinement condition, and refinement strategy apply to any number of space dimensions. The final Section 7 contains the summary and concluding remarks.

2. Outline of the adaptive grid method

Although its elaboration readily becomes complicated, the idea behind local uniform grid refinement is simple. Starting from a coarse grid, finer and finer uniform subgrids are created locally in a nested manner in regions where the solution is steep. Here, a set of interconnected grid cells, all having the same size, is called a *subgrid*. A set of subgrids having the same cell size is called a *grid level* or just *grid*. Hence, a grid level consists of a single subgrid or several disjoint non-overlapping subgrids. A new (initial) boundary value problem is solved at each grid level in a consecutive order, from coarse to fine using the same time step size for all grid levels. This means that the refinement in time is global, i.e. the step size is adapted in time but is the same for all grid levels in use. Note that the PDEs are solved on a grid level as a whole, in spite of the fact that the grid level can consist of several disjoint subgrids. The required initial values for the finer subgrids are defined by interpolation from the coarser subgrid or taken from a finer subgrid from the previous time step when available. Internal boundaries, i.e. subgrid boundaries lying in the interior of the domain, are treated as Dirichlet boundaries and values are also interpolated from the next coarser grid level. Where the boundary of a fine subgrid coincides with the boundary of the domain, the prescribed boundary conditions are used. Except for the necessary initial and boundary conditions, all subgrids are independent of each other. Therefore, the subgrids are not patched into the coarser grids but are actually overlaying them. The generation of subgrids is continued until the spatial phenomena are described accurately enough by the finest grid. The fine grid cells are created by bisecting the sides of the cells of the next coarser grid; so the refinement is cellular. The subgrids created this way have a piecewise polygonal shape. Furthermore, the unknowns are defined at cell vertices, which implies that in the region where the coarse grid is overlapped by the fine grid, the coarse grid nodes coincide with the fine grid nodes.

During each time step the following operations are performed:

- (1) Solve the PDEs on the coarsest grid level.
- (2) If the desired accuracy in space or the maximum number of grid levels is reached then go to (8).
- (3) Determine a new finer uniform grid level at a forward time.
- (4) Interpolate internal boundary values at forward time.

- (5) Provide new initial values are backward time.
- (6) Solve the PDEs on the new grid level, using the same step length.
- (7) Go to (2).
- (8) Assign fine grid values to the corresponding coarser grid points.

Thus, for each time step the computation starts at the coarse base grid using the most accurate solution available, since coarse grid solution values are always replaced by fine grid values at coarse grid nodes coinciding with fine grid nodes and all subgrids are kept in storage for step continuation.

3. Mathematical formulation

A mathematical formulation will be given which is needed for the error analysis of the local uniform grid refinement method. The following system of PDEs is considered, together with the initial and boundary conditions, defined on a domain Ω in \mathbb{R} , \mathbb{R}^2 , or \mathbb{R}^3 with boundary $\partial\Omega$ and sides parallel to the coordinate axes,

$$\mathcal{A}_\Omega(\mathbf{x}, t, u)u_t = \mathcal{F}_\Omega(\mathbf{x}, t, u), \quad t > t_0, \quad \mathbf{x} \in \Omega, \quad (3.1a)$$

$$\mathcal{A}_{\partial\Omega}(\mathbf{x}, t, u)u_t = \mathcal{F}_{\partial\Omega}(\mathbf{x}, t, u), \quad t > t_0, \quad \mathbf{x} \in \partial\Omega, \quad (3.1b)$$

$$u(\mathbf{x}, t_0) = u_0(\mathbf{x}), \quad \mathbf{x} \in \Omega \cup \partial\Omega. \quad (3.1c)$$

This system of PDEs is assumed to possess a unique solution $u(\mathbf{x}, t)$, which is as often differentiable as the numerical analysis requires. The matrices \mathcal{A}_Ω and $\mathcal{A}_{\partial\Omega}$ are possibly singular matrices; \mathcal{F}_Ω and $\mathcal{F}_{\partial\Omega}$ are functions containing spatial partial differential operators. The matrices \mathcal{A}_Ω and $\mathcal{A}_{\partial\Omega}$ do not contain space or time derivatives of $u(\mathbf{x}, t)$. The space discretization of (3.1) (method of lines) yields

$$A(t, U)\dot{U} = F(t, U), \quad t > t_0, \quad U(t_0) = U_0, \quad (3.2)$$

where $U(t)$ is the numerical approximation of $u(\mathbf{x}, t)$ on a space grid. If $A(t, U)$ is singular, then (3.2) will be a system of differential-algebraic equations (DAEs); (3.2) will be a system of ordinary differential equations (ODEs), otherwise. In case we have, for example, Neumann or Dirichlet boundary conditions, $A(t, U)$ will possess rows containing only zeros, which implies that (3.2) is a DAE system.

Local uniform grid refinement methods use local subgrids of changing size in time and thus generate solution vectors with a variable dimension. This complicates the analysis. In order to circumvent this problem, the fine local subgrids are expanded over the whole of $\Omega \cup \partial\Omega$. The solution to (3.1) is computed only within the original perimeter of the subgrid and interpolated from the next coarser subgrid outside this region. This is only done in the mathematical formulation of the method to make the error analysis easier. It does *not* take place in the implementation of the method.

Let Ω_k , $1 \leq k \leq l$, be uniform space grids covering $\Omega \cup \partial\Omega$ with l denoting the maximum number of grid levels needed to advance the solution from t_{n-1} to t_n . The grid refinement is cellular so Ω_k is obtained from Ω_{k-1} by bisecting sides of cells of Ω_{k-1} . Note that nodes of Ω_{k-1} coincide with nodes on Ω_k . Let S_k be the vector space of all grid functions on Ω_k and let

$U_k^n \in S_k$ be the approximation to $u(x, t_n)$ at Ω_k . Suppose that (3.2) is defined on Ω_k , then using an s -step backward differentiation formula (BDF) for time stepping results in the following system of equations,

$$\begin{aligned} \frac{1}{\theta_s \tau} A_k^n(U_k^n) U_k^n &= \frac{1}{\theta_s \tau} A_k^n(U_k^n) V_k^{n-1} + F_k^n(U_k^n), & k = 1, \dots, l, \\ V_k^{n-1} &= a_1 U_k^{n-1} + \dots + a_s U_k^{n-s}, \end{aligned} \quad (3.3)$$

where $\tau = t_n - t_{n-1}$, V_k^{n-1} is the history vector collecting values computed at backward time points and $\theta_s, a_1, \dots, a_s$ are coefficients depending on current and previous time step sizes.

Our formulation of the grid refinement method uses the following matrix operators:

- the identity matrix $I_k : S_k \rightarrow S_k$;
- a diagonal matrix $D_k^n : S_k \rightarrow S_k$, with diagonal entries equal to unity or zero, $D_1^n = I_1$;
- the restriction operator $R_{lk} : S_l \rightarrow S_k$, $R_{ll} = I_l$;
- the interpolation operator $P_{k-1k} : S_{k-1} \rightarrow S_k$.

The matrix D_k^n determines whether the solution at a particular grid node is obtained by solving (3.1) or by interpolation from Ω_{k-1} . The diagonal entries of D_k^n associated with this node are equal to unity when (3.1) is solved and equal to zero otherwise. The number of the diagonal entries associated with each grid node is equal to the number of PDEs. Note that on the coarsest grid, system (3.1) is solved on the whole of Ω_1 , meaning that $D_1^n = I_1$. For example, the components of the vector $D_k^n \delta_k^n$, where δ_k^n is an arbitrary vector in S_k , are nonzero when their corresponding nodes lie inside the region where (3.1) is solved, and, if interpolation takes place, then these components are zero. With the vector $(I_k - D_k^n) \delta_k^n$ it is just the other way around. The injection of fine grid solution values in the coarser grid solution is denoted by the operator R_{lk} and the interpolation by the operator P_{k-1k} . Since all nodes of Ω_k are also contained in Ω_l , injection takes place at each node of Ω_k .

One time step of the grid refinement method consists of l consecutive interpolation and solution steps on grids Ω_k . Those are defined by

$$(I_k - D_k^n) U_k^n = (I_k - D_k^n) P_{k-1k} U_{k-1}^n, \quad (3.4a)$$

$$\frac{1}{\theta_s \tau} D_k^n A_k^n(U_k^n) U_k^n = \frac{1}{\theta_s \tau} D_k^n A_k^n(U_k^n) R_{lk} V_l^{n-1} + D_k^n F_k^n(U_k^n), \quad k = 1, \dots, l. \quad (3.4b)$$

Formula (3.4a) represents the interpolation step and (3.4b) the BDF solution step. The subgrids in the local uniform grid refinement method are properly nested. This means that the region of the domain covered by that part of grid level k where (3.1) is solved is covered completely by its counterpart associated with grid level $k - 1$. Hence, the set of nodes contained by that part of grid level k where interpolation takes place will also belong to the set of nodes where interpolation takes place at the grid levels $k + 1, k + 2, \dots, l$. In other words, the solution at the part of grid level k where interpolation takes place will be repeatedly interpolated until grid level l is reached. Finally, we emphasize that this occurs only in the formulation of the method to make the analysis easier. In practice, interpolation only takes place where it is really needed.

4. Error analysis

In this section the results of the error analysis are presented. First, the truncation errors of the interpolation and the space and time discretization will be introduced. Then, using the mathematical formulation (3.4), relations are derived for the local and global error. Finally, we give an example of the behavior of the local and global error when a coupled system of PDEs is solved.

4.1. Error relations

For u_k^n , the pointwise restriction to Ω_k of the exact solution $u(\mathbf{x}, t_n)$ of (3.1), we have the following error relations

$$(I_k - D_k^n)u_k^n = (I_k - D_k^n)P_{k-1k}u_{k-1}^n + (I_k - D_k^n)\gamma_k^n, \quad (4.1a)$$

$$\frac{1}{\theta_s \tau} D_k^n A_k^n(u_k^n)u_k^n = \frac{1}{\theta_s \tau} D_k^n A_k^n(u_k^n)R_{lk}v_l^{n-1} + D_k^n F_k^n(u_k^n) + D_k^n(\alpha_k^n - A_k^n(u_k^n)\beta_k^n),$$

$$k = 1, \dots, l, \quad (4.1b)$$

where $v_l^{n-1} = a_1 u_l^{n-1} + \dots + a_s u_l^{n-s}$ and α_k^n , β_k^n , and γ_k^n are the truncation errors of the space discretization, the time discretization, and the interpolation, respectively. They are defined by

$$\alpha_k^n = A_k^n(u_k^n)(u_t)_k^n - F_k^n(u_k^n), \quad (4.2a)$$

$$\beta_k^n = (u_t)_k^n - \frac{1}{\theta_s \tau} [u_k^n - a_1 u_k^{n-1} - \dots - a_s u_k^{n-s}], \quad (4.2b)$$

$$\gamma_k^n = u_k^n - P_{k-1k}u_{k-1}^n, \quad (4.2c)$$

where $(u_t)_k^n$ is the restriction of $u_t(\mathbf{x}, t_n)$ to Ω_k . Since the restriction operator R_{lk} involves only the replacement of coarse grid values by fine grid values at coinciding nodes after a time step has been performed on all grid levels, no additional errors are introduced here. Therefore, we have $u_k^{n-1} = R_{lk}u_l^{n-1}$. The global error at t_n at Ω_k is defined by

$$e_k^n = u_k^n - U_k^n. \quad (4.3)$$

Subtracting (3.4) from (4.1) we get

$$(I_k - D_k^n)e_k^n = (I_k - D_k^n)P_{k-1k}e_{k-1}^n + (I_k - D_k^n)\gamma_k^n, \quad (4.4a)$$

$$\begin{aligned} \frac{1}{\theta_s \tau} D_k^n [K_k^n + W_k^n]e_k^n &= \frac{1}{\theta_s \tau} D_k^n [L_k^n e_k^n + W_k^n R_{lk} f_l^{n-1}] \\ &\quad + D_k^n M_k^n e_k^n + D_k^n (\alpha_k^n - A_k^n(u_k^n)\beta_k^n), \end{aligned}$$

$$k = 1, \dots, l, \quad (4.4b)$$

where

$$f_l^{n-1} = a_1 e_l^{n-1} + \dots + a_s e_l^{n-s}, \quad (4.5)$$

and

$$[K_k^n + W_k^n] e_k^n = \int_0^1 \frac{d}{d\theta} [A_k^n(\xi)\xi] d\theta = A_k^n(u_k^n)u_k^n - A_k^n(U_k^n)U_k^n, \tag{4.6a}$$

$$\begin{aligned} L_k^n e_k^n + W_k^n R_{lk} f_l^{n-1} &= \int_0^1 \frac{d}{d\theta} [A_k^n(\xi)\eta] d\theta \\ &= A_k^n(u_k^n)R_{lk}v_l^{n-1} - A_k^n(U_k^n)R_{lk}V_l^{n-1}, \end{aligned} \tag{4.6b}$$

$$M_k^n e_k^n = \int_0^1 \frac{d}{d\theta} [F_k^n(\xi)] d\theta = F_k^n(u_k^n) - F_k^n(U_k^n), \tag{4.6c}$$

$$W_k^n = \int_0^1 A_k^n(\xi) d\theta, \tag{4.6d}$$

$$K_k^n = \int_0^1 \frac{d}{d\tilde{\xi}} [A_k^n(\tilde{\xi})\xi] d\tilde{\xi}, \quad L_k^n = \int_0^1 \frac{d}{d\tilde{\xi}} [A_k^n(\tilde{\xi})\eta] d\tilde{\xi},$$

$$\xi = \tilde{\xi} = \theta u_k^n + (1 - \theta)U_k^n, \quad \eta = \theta R_{lk}v_l^{n-1} + (1 - \theta)R_{lk}V_l^{n-1}, \tag{4.6e}$$

which are obtained by applying the mean value theorem for vector functions. Combining (4.4a) and (4.4b) yields a recurrence relation for the global error

$$e_k^n = \Gamma_k^n f_l^{n-1} + X_k^n e_{k-1}^n + \phi_k^n, \quad k = 1, \dots, l, \tag{4.7}$$

where

$$\Gamma_k^n = (Z_k^n)^{-1} \frac{1}{\theta_{sT}} D_k^n W_k^n R_{lk}, \tag{4.8a}$$

$$X_k^n = (Z_k^n)^{-1} (I_k - D_k^n) P_{k-1k}, \tag{4.8b}$$

$$\phi_k^n = (Z_k^n)^{-1} \{ D_k^n \alpha_k^n - D_k^n A_k^n(u_k^n) \beta_k^n + (I_k - D_k^n) \gamma_k^n \}, \tag{4.8c}$$

$$Z_k^n = I_k - D_k^n + D_k^n \left(\frac{1}{\theta_{sT}} [W_k^n + K_k^n - L_k^n] - M_k^n \right). \tag{4.8d}$$

The vector ϕ_k^n is the local level error which is the contribution associated with a single time step of grid level k to the global error e_k^n , and Z_k^n is the integrated Jacobian of the system of equations. Using the specific form of Z_k^n , we note that Z_k^n can be written as $I_k - D_k^n + D_k^n Z_k^n$. When $(Z_k^n)^{-1}$ is written as $(I_k - D_k^n)(Z_k^n)^{-1} + D_k^n(Z_k^n)^{-1}$, it can—by pre-multiplying $(Z_k^n)^{-1}$ with Z_k^n —very easily be shown that

$$(Z_k^n)^{-1} = I_k - D_k^n + D_k^n (Z_k^n)^{-1}, \quad k = 1, \dots, l. \tag{4.9}$$

Relation (4.7) is similar to the one obtained in [7,8,10,11] and leads to the following expressions for the global and local error:

$$e_k^n = G_k^n f_l^{n-1} + \psi_k^n, \quad n = 1, 2, \dots; \quad k = 1, \dots, l, \tag{4.10}$$

where G_k^n is the amplification matrix and ψ_k^n the local error, which is the contribution associated with one time step of k grid levels to the global error e_k^n . They are given by

$$G_1^n = \Gamma_1^n, \quad G_k^n = X_k^n G_{k-1}^n + \Gamma_k^n, \quad k = 2, \dots, l, \tag{4.11a}$$

$$\psi_1^n = \phi_1^n, \quad \psi_k^n = X_k^n \psi_{k-1}^n + \phi_k^n, \tag{4.11b}$$

which result in

$$G_k^n = \sum_{j=1}^{k-1} \left(\prod_{i=k}^{j+1} X_i^n \right) \Gamma_j^n + \Gamma_k^n, \quad (4.12a)$$

$$\psi_k^n = \sum_{j=1}^{k-1} \left(\prod_{i=k}^{j+1} X_i^n \right) \phi_j^n + \phi_k^n, \quad k = 2, \dots, l. \quad (4.12b)$$

Now the local level error ϕ_k^n in (4.8c), can be split up in a spatial part $\phi_{k,s}^n$ and a temporal part $\phi_{k,t}^n$ where $\phi_k^n = \phi_{k,s}^n + \phi_{k,t}^n$. These parts are given by

$$\phi_{k,s}^n = (Z_k^n)^{-1} [D_k^n \alpha_k^n + (I_k - D_k^n) \gamma_k^n], \quad (4.13a)$$

$$\phi_{k,t}^n = -(Z_k^n)^{-1} D_k^n A_k^n(u_k^n) \beta_k^n. \quad (4.13b)$$

This yields two distinct relations for the local space error $\psi_{k,s}^n$ and the local time error $\psi_{k,t}^n$. With (4.9), (4.11b), and (4.13a), the relations for $\psi_{k,s}^n$ read

$$D_k^n \psi_{k,s}^n = D_k^n (Z_k^n)^{-1} \{D_k^n \alpha_k^n + (I_k - D_k^n) [P_{k-1k} \psi_{k-1,s}^n + \gamma_k^n]\}, \quad (4.14a)$$

$$(I_k - D_k^n) \psi_{k,s}^n = (I_k - D_k^n) [P_{k-1k} \psi_{k-1,s}^n + \gamma_k^n]. \quad (4.14b)$$

Here $D_k^n \psi_{k,s}^n$ denotes the local space error inside the region of grid level k where (3.1) is solved and $(I_k - D_k^n) \psi_{k,s}^n$ the local space error outside this region. In a similar way, the local time error can be written as

$$D_k^n \psi_{k,t}^n = D_k^n (Z_k^n)^{-1} \{-D_k^n A_k^n(u_k^n) \beta_k^n + (I_k - D_k^n) P_{k-1k} \psi_{k-1,t}^n\}, \quad (4.15a)$$

$$(I_k - D_k^n) \psi_{k,t}^n = (I_k - D_k^n) P_{k-1k} \psi_{k-1,t}^n. \quad (4.15b)$$

Since the local error can be split up in a spatial and temporal part, the same can be done with the global error. Using (4.7)–(4.9) and (4.13) we get similar relations for the global space error

$$D_k^n e_{k,s}^n = D_k^n (Z_k^n)^{-1} \left\{ D_k^n \left[\frac{1}{\theta_s \tau} W_k^n R_{lk} f_{l,s}^{n-1} + \alpha_k^n \right] + (I_k - D_k^n) [P_{k-1k} e_{k-1,s}^n + \gamma_k^n] \right\}, \quad (4.16a)$$

$$(I_k - D_k^n) e_{k,s}^n = (I_k - D_k^n) [P_{k-1k} e_{k-1,s}^n + \gamma_k^n], \quad (4.16b)$$

and for the global time error

$$D_k^n e_{k,t}^n = D_k^n (Z_k^n)^{-1} \left\{ D_k^n \left[\frac{1}{\theta_s \tau} W_k^n R_{lk} f_{l,t}^{n-1} - A_k^n(u_k^n) \beta_k^n \right] + (I_k - D_k^n) P_{k-1k} e_{k-1,t}^n \right\}, \quad (4.17a)$$

$$(I_k - D_k^n) e_{k,t}^n = (I_k - D_k^n) P_{k-1k} e_{k-1,t}^n. \quad (4.17b)$$

When we consider, for example, the global space error given by (4.16), we observe that the global space error in the region of grid level k where interpolation takes place (4.16b) is determined by the global space error at grid level $k-1$ and the interpolation error. The global space error in the region of grid level k where (3.1) is solved (4.16a) is determined by the

inverse of the Jacobian Z_k^n operating on a vector which consists of a part due to the spatial discretization error and the global space error at previous time points, only nonzero inside this region, and a part due to the global space error outside this region.

4.2. Error behavior

In this section an example will be given of the behavior of the global and local errors. We consider a system of two coupled PDEs which is solved on a single grid. Therefore, we will drop the subscripts denoting grid levels, k and l , in the remainder of this section. This system of PDEs leads to the following system of differential-algebraic equations after spatial discretization

$$\begin{pmatrix} I & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \dot{U}_1 \\ \dot{U}_2 \end{pmatrix} = \begin{pmatrix} F_1(t, U_1, U_2) \\ F_2(t, U_1, U_2) \end{pmatrix}, \quad (4.18)$$

which is written in the form (3.2). The BDF method (3.3) is applied to solve (4.18) and some notation will be introduced that is needed for the examination of the global and local errors. Relations for the local and global errors are derived, using (4.7) and (4.8). Due to (4.6c), (4.6d), and (4.18), we have

$$W^n = \begin{pmatrix} I & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}, \quad K^n = L^n = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}, \quad M^n = \begin{pmatrix} M_{11}^n & M_{12}^n \\ M_{21}^n & M_{22}^n \end{pmatrix}. \quad (4.19)$$

The matrix D_k^n appearing in the error relation (4.7) will be equal to the identity matrix I_k in this case which implies that the $(I_k - D_k^n)$ -terms vanish from (4.7) and (4.8). The matrix Z^n is now given by

$$Z^n = \begin{pmatrix} \frac{1}{\theta_s \tau} I - M_{11}^n & -M_{12}^n \\ -M_{21}^n & -M_{22}^n \end{pmatrix}. \quad (4.20)$$

Furthermore, the matrix R_{lk} will be equal to the identity matrix. Using the notation above, the relation (4.7) for the global error is now given by

$$\begin{pmatrix} e_1^n \\ e_2^n \end{pmatrix} = \begin{pmatrix} \frac{1}{\theta_s \tau} I - M_{11}^n & -M_{12}^n \\ -M_{21}^n & -M_{22}^n \end{pmatrix}^{-1} \left\{ \frac{1}{\theta_s \tau} \begin{pmatrix} I & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} f_1^{n-1} \\ f_2^{n-1} \end{pmatrix} + \begin{pmatrix} \alpha_1^n \\ \alpha_2^n \end{pmatrix} - \begin{pmatrix} I & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \beta_1^n \\ \beta_2^n \end{pmatrix} \right\}. \quad (4.21)$$

The global error e^n at the current time t_n is given by $(e_1^n, e_2^n)^T$ and the vector f^{n-1} in which global errors at backward times are collected, by $(f_1^{n-1}, f_2^{n-1})^T$ (cf. (4.5)). The vectors e_1^n and e_2^n represent the global error belonging to U_1^n and U_2^n respectively. Now examining (4.21) leads to the conclusion that only f_1^{n-1} contributes to e^n . This means that in this case only e_1^n carries over to future time points while e_2^n does not.

For the local error ψ^n we have, according to (4.21):

$$\begin{pmatrix} \psi_1^n \\ \psi_2^n \end{pmatrix} = \begin{pmatrix} \frac{1}{\theta_s \tau} I - M_{11}^n & -M_{12}^n \\ -M_{21}^n & -M_{22}^n \end{pmatrix}^{-1} \left\{ \begin{pmatrix} \alpha_1^n \\ \alpha_2^n \end{pmatrix} - \begin{pmatrix} I & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \beta_1^n \\ \beta_2^n \end{pmatrix} \right\}. \quad (4.22)$$

The behavior of the local error for small τ is determined by the operator $(Z^n)^{-1}$. The behavior of this operator when it operates on a vector will be investigated for this case where it is assumed that the cell sizes of the space grid remain constant. This operator can be written as

$$(Z^n)^{-1} = \begin{pmatrix} \frac{1}{\theta_s \tau} I - M_{11}^n & -M_{12}^n \\ -M_{21}^n & -M_{22}^n \end{pmatrix}^{-1} = \begin{pmatrix} Y_{11}^n & Y_{12}^n \\ Y_{21}^n & Y_{22}^n \end{pmatrix}. \quad (4.23)$$

When we assume that the diagonal blocks of Z^n are nonsingular, we obtain

$$Y_{11}^n = \left[I + \left(\frac{1}{\theta_s \tau} I - M_{11}^n \right)^{-1} M_{12}^n (M_{22}^n)^{-1} M_{21}^n \right]^{-1} \left(\frac{1}{\theta_s \tau} I - M_{11}^n \right)^{-1}, \quad (4.24a)$$

$$Y_{21}^n = -(M_{22}^n)^{-1} M_{21}^n Y_{11}^n, \quad (4.24b)$$

$$Y_{22}^n = - \left[I + (M_{22}^n)^{-1} M_{21}^n \left(\frac{1}{\theta_s \tau} I - M_{11}^n \right)^{-1} M_{12}^n \right]^{-1} (M_{22}^n)^{-1}, \quad (4.24c)$$

$$Y_{12}^n = \left(\frac{1}{\theta_s \tau} I - M_{11}^n \right)^{-1} M_{12}^n Y_{22}^n. \quad (4.24d)$$

This leads to the following approximations for small τ ,

$$\begin{aligned} Y_{11}^n &\approx \theta_s \tau I, & Y_{21}^n &\approx -\theta_s \tau (M_{22}^n)^{-1} M_{21}^n, \\ Y_{22}^n &\approx -(M_{22}^n)^{-1}, & Y_{12}^n &\approx -\theta_s \tau M_{12}^n (M_{22}^n)^{-1}, \end{aligned} \quad (4.25)$$

from which we obtain the following approximations for the local error components:

$$\psi_1^n \approx \theta_s \tau I (\alpha_1^n - \beta_1^n) - \theta_s \tau M_{12}^n (M_{22}^n)^{-1} \alpha_2^n, \quad (4.26a)$$

$$\psi_2^n \approx -\theta_s \tau (M_{22}^n)^{-1} M_{21}^n (\alpha_1^n - \beta_1^n) - (M_{22}^n)^{-1} \alpha_2^n. \quad (4.26b)$$

We see from (4.26) that the first component of the local error ψ_1^n consists of the truncation errors of the space and time discretizations multiplied by an operator which behaves like $O(\tau)$ for $\tau \rightarrow 0$, meaning that this component of the local error will vanish in this case. Further we see that, unlike ψ_1^n , the component ψ_2^n does not disappear completely when $\tau \rightarrow 0$.

From this example we conclude that, when a system of coupled PDEs is solved, the local and global error components can exhibit a very different behavior. Therefore, a refinement strategy of an adaptive grid method based on error estimation will have to take such differences in behavior into account. This means that in case of a system of coupled PDEs, distinction must

be made between errors associated with each separate PDE when we want to develop a refinement strategy based on error estimations. This will be discussed in the next section.

5. Refinement strategy

In [7], PDEs were considered which, upon discretization of the space derivatives, lead to a system of explicit ODEs in which the boundary conditions were incorporated. This system of ODEs was solved using the implicit Euler method. The idea was to control the spatial accuracy of the solution by controlling the local space error. Moreover, in case the number of grid levels is constant in time, the local space error at the finest local subgrid should be comparable to the local space error on a single uniform grid having the same cell sizes as the finest subgrid. A refinement strategy was developed aiming to fulfill the following inequality, which is called *the refinement condition*,

$$\|(Z_l^n)^{-1}(I_l - D_l^n)[P_{l-1} \psi_{l-1,s}^n + \gamma_l^n]\|_\infty < c \|(Z_l^n)^{-1} \tau D_l^n \alpha_l^n\|_\infty, \quad c > 0, \quad (5.1)$$

where Z_l^n is defined as

$$Z_l^n = I_l - \tau D_l^n M_l^n. \quad (5.2)$$

The matrices I_l and D_l^n are defined similarly as in Section 3. The matrix M_l^n arises after discretizing the space derivatives of a time-dependent PDE and is comparable to the matrix defined by (4.6c). When $\|(Z_l^n)^{-1}\|_\infty \leq 1$, condition (5.1) results in the following bound for the local space error at the finest subgrid:

$$\|\psi_{l,s}^n\|_\infty < (1 + c) \|(Z_l^n)^{-1} \tau D_l^n \alpha_l^n\|_\infty. \quad (5.3)$$

Apart from the constant c , this error bound is similar to the error bound we would get using a single uniform grid. This indicates that by satisfying the refinement condition it is possible to get more or less the same spatial accuracy as if a single uniform grid was used. Further, it was proved that (5.1) holds when the inequality

$$\max_{2 \leq j \leq l} \|(I_j - D_j^n) \lambda_j^n\|_\infty \leq \frac{c}{l-1} \|(Z_l^n)^{-1} \tau D_l^n \alpha_l^n\|_\infty, \quad (5.4)$$

is satisfied while creating finer local uniform subgrids, where

$$\lambda_j^n = \gamma_j^n + P_{j-1j} (Z_{j-1}^n)^{-1} \tau D_{j-1}^n \alpha_{j-1}^n. \quad (5.5)$$

In [8], a similar refinement condition and error bound were derived based on a general Range-Kutta time stepping scheme and, in [11], a refinement condition was derived for elliptic PDEs.

Although the refinement strategy based on fulfilling (5.4) worked satisfactorily in practice, it stems from rather conservative estimates of norms and can therefore be restrictive, especially when the number of grid levels is large. Further, in order to fulfill (5.4) when $\tau \rightarrow 0$, the matrix D_k^n inevitably has to approach the identity matrix I_k . This means that when τ is decreased, the local subgrids will cover an increasingly larger part of the domain and will in the limit of $\tau = 0$ cover the entire domain. In this respect, (5.4) is also restrictive and one can argue whether it is really necessary to let the local subgrids cover a larger part of the domain with decreasing τ in

order to retain a high spatial accuracy. Finally, from the previous section we have concluded that components of the global and local error can exhibit a different behavior. This is due to the fact that these different components are associated with different PDEs. The inequalities (5.1) and (5.4) and the error bound (5.3) are based on estimates of matrix norms where it is assumed that the system of PDEs at hand leads to a system of explicit ODEs after spatial discretization. Hence, it is assumed that all local and global error components behave in a similar manner. This implies that (5.1) and (5.4) are not sufficiently accurate in case a coupled system of general PDEs is solved. This might also be the case with a system of coupled PDEs where, upon semi-discretization, one or more PDEs lead to a much stiffer system of ODEs than the other ones. For this reason, a new, more general refinement strategy is developed for a general system of coupled PDEs. In contrast to most of our previous work, this new strategy will be based on controlling the global space error which is less restrictive than (5.1). It should satisfy two demands. First, it must make a distinction between vector components associated with different PDEs and, second, it must computationally not be too expensive.

Basically a refinement strategy has to answer two questions. The first one is, *when should a new finer grid level be created*, and the second one is, *which grid cells need to be refined*. In order to answer these questions we will now introduce some notation. Suppose (3.1) consists of q different PDEs in which the boundary conditions are included, i.e. boundary conditions are regarded as separate PDEs defined on boundaries only. Note that a single PDE with Neumann or Dirichlet boundary conditions can also be regarded this way.

An arbitrary vector $\delta_k^n \in S_k$ is generically denoted by $(\delta_{k,1}^n, \delta_{k,2}^n, \dots, \delta_{k,q}^n)^T$, where the component $\delta_{k,j}^n$ is associated with the j th PDE. The matrices I_k , D_k^n , P_{k-1k} , and R_{lk} are block diagonal and can be written as

$$I_k = \text{diag}(I_{k,1}, I_{k,2}, \dots, I_{k,q}), \quad (5.6a)$$

$$D_k^n = \text{diag}(D_{k,1}^n, D_{k,2}^n, \dots, D_{k,q}^n), \quad (5.6b)$$

$$P_{k-1k} = \text{diag}(P_{k-1k,1}, P_{k-1k,2}, \dots, P_{k-1k,q}), \quad (5.6c)$$

$$R_{lk} = \text{diag}(R_{lk,1}, R_{lk,2}, \dots, R_{lk,q}). \quad (5.6d)$$

The matrices W_k^n , K_k^n , L_k^n , M_k^n , and Z_k^n are written as block matrices with blocks $W_{k,ij}^n$, $K_{k,ij}^n$, $L_{k,ij}^n$, $M_{k,ij}^n$, and $Z_{k,ij}^n$, where $i, j = 1, \dots, q$. The blocks of Z_k^n are given by

$$\begin{aligned} Z_{k,ii}^n &= I_{k,i} - D_{k,i}^n + D_{k,i}^n \left(\frac{1}{\theta_{s\tau}} [W_{k,ii}^n + K_{k,ii}^n - L_{k,ii}^n] - M_{k,ii}^n \right), \\ Z_{k,ij}^n &= D_{k,i}^n \left(\frac{1}{\theta_{s\tau}} [W_{k,ij}^n + K_{k,ij}^n - L_{k,ij}^n] - M_{k,ij}^n \right), \quad i \neq j. \end{aligned} \quad (5.7)$$

This leads to the following relation for the global space error:

$$\begin{aligned} e_{k,s,i}^n &= \sum_{j=1}^q Y_{k,ij}^n \left\{ D_{k,ij}^n \left[\frac{1}{\theta_{s\tau}} \sum_{m=1}^q W_{k,jm}^n R_{lk,m} f_{l,s,m}^{n-1} + \alpha_{k,j}^n \right] \right. \\ &\quad \left. + (I_{k,j} - D_{k,j}^n) [P_{k-1k,j} e_{k-1,s,j}^n + \gamma_{k,j}^n] \right\}, \\ i &= 1, \dots, q, \quad k = 1, \dots, l, \end{aligned} \quad (5.8)$$

where $Y_{k,ij}^n$ is a block of $(Z_k^n)^{-1}$. We have established that the various components of the global and local space errors can behave very differently. Consequently, a criterion like, *refine the grid when the local space error ψ_k^n or the global space error e_k^n exceeds some tolerance*, is simply too crude. Although the operators $(Z_k^n)^{-1}$ and W_k^n determine how the truncation error of the space discretization α_k^n affects the spatial accuracy, one can say that the source of the space error is α_k^n . This is certainly true when a single uniform grid is used. Therefore, the following criterion can be used. A new grid level $k + 1$ is created if there exists a component i for which

$$\|\alpha_{k,i}^n\|_\infty \geq TOL, \quad i = 1, \dots, q, \tag{5.9}$$

holds, where TOL is a user-defined tolerance. It is assumed here that the PDE problem at hand is properly scaled. Otherwise the scaling of the various PDEs has to be taken into account in criterion (5.9). We have also built in an extra condition in our research code to smoothen its behavior. Suppose that the maximum number of grid levels during the previous time step is l and that at grid level $k < l$, $\|\alpha_{k,i}^n\|_\infty < TOL$ for all i . Although this means that a new finer grid level $k + 1$ is actually not necessary, it will still be created when $\|\alpha_{k,i}^n\|_\infty > 0.9 \times TOL$, to avoid fluctuations of the maximum number of grid levels from one time point to the next.

Further, it should be pointed out that when the number of grid levels is increasing in time additional interpolation errors are introduced, because new initial values have to be interpolated over the whole newly created grid. It is possible that these extra interpolation errors diminish the spatial accuracy. For example, this can be the case when solving reaction-diffusion problems with a small diffusion coefficient. In such case, the interpolation error, which will be committed when an extra grid level is introduced, can be very large due to a steep solution while the global space error is small. For this reason it may be necessary to use an extra criterion to create a new finer grid level based on controlling this potential interpolation error. This means that the situation may occur that grid refinement takes place not to reduce the global space error but to reduce the potential interpolation error. However, in most cases when these potential interpolation errors are large, the global space error will also be large and, therefore, a criterion like (5.9) will be sufficient to control the global spatial accuracy. In the refinement strategy we have implemented, only (5.9) is used as a refinement criterion.

Having devised a criterion to generate new finer grid levels, we now have to find a criterion to determine which grid cells need to be refined. In order to do this we use (4.9) to rewrite (5.8) as

$$D_{k,i}^n e_{k,s,i}^n = D_{k,i}^n \sum_{j=1}^q Y_{k,ij}^n \left\{ D_{k,j}^n \left[\frac{1}{\theta_s \tau} \sum_{m=1}^q W_{k,jm}^n R_{lk,m} f_{l,s,m}^{n-1} + \alpha_{k,j}^n \right] + (I_{k,j} - D_{k,j}^n) [P_{k-1,k,j} e_{k-1,s,j}^n + \gamma_{k,j}^n] \right\}, \tag{5.10a}$$

$$(I_{k,i} - D_{k,i}^n) e_{k,s,i}^n = (I_{k,j} - D_{k,j}^n) [P_{k-1,k,j} e_{k-1,s,j}^n + \gamma_{k,j}^n], \tag{5.10b}$$

$$i = 1, \dots, q, \quad k = 1, \dots, l.$$

When we for a moment abandon the idea of expanded grids and think in terms of local subgrids, then this new criterion will be based on the notion that after the coarser grid values have been replaced by the finer grid values at coinciding nodes, the largest absolute nodal value of the global space error should be at the finest grid level. If this is not the case, then the

maximum norm of the global space error over all grids will not be reduced by creating the finest grid level, which means that this finest grid level is of no use.

When a grid is locally refined, the nodal values at the refined part of the grid are eventually replaced by finer grid values. However, the nodal values outside this part of the domain remain unchanged. This means that, if the nodal value of the maximum space error should be at the finest grid level, we have to make sure that the values of the global space error at nodes outside the part of a grid which is going to be refined are smaller than the maximum global space error at the finest grid level. This means that, returning to the expanded grids, the global space error at the part of a grid level where interpolation takes place is smaller than the maximum global space error in the region of the finest grid level where (3.1) is solved. In other words, we have to demand that

$$\begin{aligned} \|(I_{k,i} - D_{k,i}^n)[P_{k-1,k,i}e_{k-1,s,i}^n + \gamma_{k,i}^n]\|_\infty &\leq c \|D_{l,i}^n e_{l,s,i}\|_\infty, \\ i = 1, \dots, q, \quad k = 2, \dots, l, \quad 0 < c \leq 1, \end{aligned} \tag{5.11}$$

where c is a user-defined constant and l the finest grid level which is going to be used during this time step. It a grid level $k - 1$ (5.9) holds for PDE component i and if at node j the associated component of the vector $(I_{k,i} - D_{k,i}^n)e_{k,s,i}^n$ satisfies

$$|((I_{k,i} - D_{k,i}^n)[P_{k-1,k,i}e_{k-1,s,i}^n + \gamma_{k,i}^n])_j| > c \|D_{l,i}^n e_{l,s,i}\|_\infty, \tag{5.12}$$

then the sixteen cells surrounding j will be refined. This implies that the refinement is only controlled by the error components connected with the space discretization error components for which (5.9) holds. In practice the right-hand side of (5.12) is estimated at grid level $k - 1$. Not all grid level $k - 1$ nodes are scanned but only the nodes within the region of $k - 1$ where the PDEs are solved. The reason for this is that outside the region where the PDEs are solved only repeated interpolation takes place until grid level l is reached. The interpolation error committed by repeated interpolation will be bounded since repeated interpolation implies that only more intermediate points will be computed on the same interpolation polynomial. Hence, the estimate of the right-hand side of (5.12) at $k - 2$ can be regarded as a first-order approximation to the estimate at $k - 1$ for the nodes lying outside the region of $k - 1$ where the PDEs are solved. This means that, when in the region of $k - 1$ where interpolation takes place, (5.12) did not hold at a node belonging to $k - 2$, we can assume that it will neither hold at its corresponding node plus its nearest neighbors at $k - 1$.

We use (4.16) to compute the global space error which implies that on top of solving (3.4) for the solution we solve an extra equation for the global space error. The spatial discretization error α_k^n is estimated by computing $F_k^n(U_k^n)$ in (3.3) with a higher and a lower order discretization and subtracting the two. The interpolation error γ_k^n is computed by numerically estimating the truncation error. For both estimates the numerically computed solution is used. To estimate the right-hand side of (5.11) we use the asymptotic behavior of the global space error. In case the space discretization is of order p we have

$$\|D_{l,i}^n e_{l,s,i}\|_\infty \approx 2^{-p(l-k+1)} \|D_{k-1,i}^n e_{k-1,s,i}\|_\infty. \tag{5.13}$$

If $\|\alpha_{k-1,i}^n\|_\infty$ is computed and (5.9) holds, then using the asymptotics we can estimate how many grid levels are needed to achieve that (5.9) does not hold any longer. The maximum

number of

This means
 l -value is
 can have
 times whe
 criterion (
 the accur
 overcome
 taking the
 Finally,
 does not
 is comput
 single un
 method. I
 get a bou
 these req
 posteriori
 refinement
 refined a
 tions is li
 possible
 respectiv
 restrictiv
 according
 (5.4) whe
 is also av

6. Exampl

Three
 strategy.

Table 1
 Problem I

Number of
 grid levels

- 1
- 2
- 3
- 4

number of grid levels l which are necessary during this time step is then estimated as

$$l = k + \text{entier} \left\{ \frac{\log(\|\alpha_{k-1,i}^n\|_\infty) - \log(TOL)}{p \log(2)} \right\}, \quad i = 1, \dots, q. \quad (5.14)$$

This means that for component i we need l grid levels in order to fulfill $\|\alpha_{l,i}^n\|_\infty < TOL$. This l -value is used in (5.13) to estimate $\|D_{l,i}^n e_{l,s,i}\|_\infty$. Note that for different PDE components we can have different l -values. The estimates above might not always be accurate, especially at times when the number of grid levels in use is about to change. This may lead to a refinement criterion (5.12) which is too restrictive or not restrictive enough. Only the latter can influence the accuracy in a negative manner. However, choosing c in (5.14) sufficiently small can overcome this problem. It is also possible to improve the estimates of $\|D_{l,i}^n e_{l,s,i}\|_\infty$ and l by taking these values at the previous time step into consideration.

Finally, we conclude this section with the following remarks. The strategy based on (5.13) does not guarantee that, in case l is fixed over the entire time interval over which the solution is computed, the global space error is comparable to the global space error obtained with a single uniform grid having the same cell sizes as the finest grid level in the adaptive grid method. If such a guarantee is desired, then extra requirements have to be fulfilled in order to get a bound for $\|e_{l,s,i}\|_\infty$ which is similar to the bound using a single uniform grid. However, these requirements are difficult to satisfy in practice and they can only be satisfied in an a posteriori manner. For this reason we have not incorporated these requirements in the refinement strategy. Nevertheless, when the constant c decreases then more grid cells are refined and the gap between the spatial error of the adaptive grid and uniform grid computations is likely to become smaller. Further, the strategy described in this section is not the only possible strategy. The relations for the local and global space errors, given by (4.14) and (4.16), respectively, leave room for other strategies as well. The new strategy based on (5.11) is less restrictive than the previous one based on (5.4). When the time step size tends to zero, then, according to (5.11), the finer subgrids do not necessarily need to grow. This is in contrast to (5.4) where the subgrids eventually will cover the entire domain. Moreover, the $(l-1)^{-1}$ term is also avoided in (5.11).

6. Example problems

Three example problems are used to illustrate the method and to test the refinement strategy. For time integration, implicit Euler for the first time step and BDF2 with variable

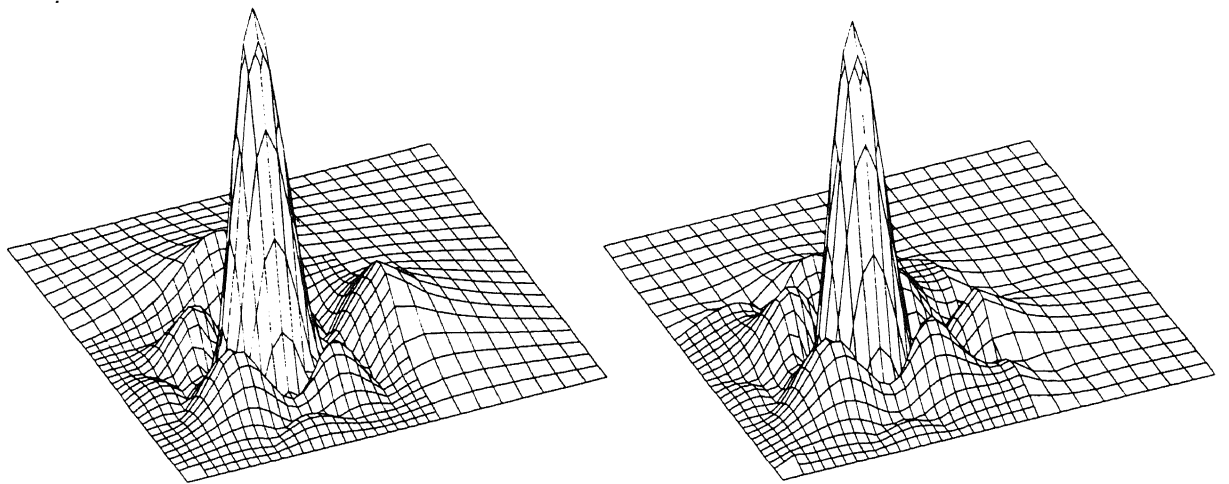
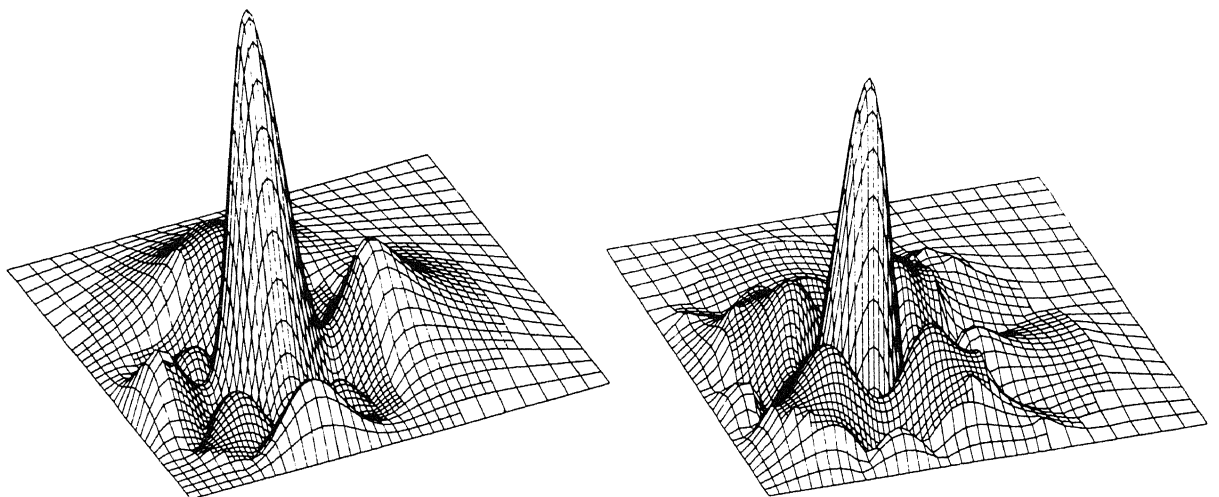
Table 1
Problem I: maxima of the exact global space error restricted to the finest grid

Number of grid levels	Global space error	
	u	v
1	0.043973	0.044304
2	0.013465	0.012874
3	0.003594	0.002991
4	0.000757	0.000631

Table 2

Problem I: maxima of the numerically estimated global space error restricted to the finest grid

Number of grid levels	Global space error	
	u	v
1	0.043441	0.044379
2	0.013277	0.013304
3	0.003543	0.003135
4	0.000748	0.000663

Fig. 1. Problem I: the scaled absolute values of the exact global space error in u and v obtained with two grid levels at $t = 0.25$.Fig. 2. Problem I: the scaled absolute values of the exact global space error in u and v obtained with three grid levels at $t = 0.25$.

coefficients for the following time steps are used. Standard second-order finite differences are used for space discretization and the interpolation is fourth-order Lagrangian.

6.1. Problem I

This test example is hypothetical and is given by a coupled parabolic and elliptic equation, both linear:

$$u_t = u_{xx} + u_{yy} - v + g(x, y, t), \quad 0 < x, y < 1, \quad t > 0. \quad (6.1a)$$

$$0 = v_{xx} + v_{yy} + u + h(x, y, t), \quad (6.1b)$$

The initial function, the Dirichlet boundary conditions, and the source terms g and h are selected so that the exact solution is given by

$$u(x, y, t) = v(x, y, t) = \exp\left[-80\left((x - r(t))^2 + (y - s(t))^2\right)\right], \quad (6.2)$$

where $r(t) = \frac{1}{4}[2 + \sin(\pi t)]$ and $s(t) = \frac{1}{4}[2 + \cos(\pi t)]$. This solution is a cone that is initially centered at $(\frac{1}{2}, \frac{3}{4})$ and that rotates around $(\frac{1}{2}, \frac{1}{2})$ in a clockwise direction with a constant speed. We have used this problem to subdue the method to a convergence test. The solution was computed from $t = 0$ to $t = 0.25$. Starting from a coarse 20×20 grid, one, two, and three additional grid levels were used. The number of grid levels was kept constant throughout the entire time interval. The associated TOL -values were 20, 5 and 1. These tolerance values appear to be large compared to the tolerance values one is used to. The reason for this is that the $\|\alpha_{k,i}^n\|_\infty$ values can be large. However the accuracy does not deteriorate severely by this, because the inverse of the Jacobian operates on the vector α_k^n (cf. (4.14a) and (4.16a)) which reduces the values of the components of this vector considerably. This is due to the large high frequency components of the grid function α_k^n and the fact that the Jacobian stems from an elliptic/parabolic operator of which the inverse strongly damps such components. The constant

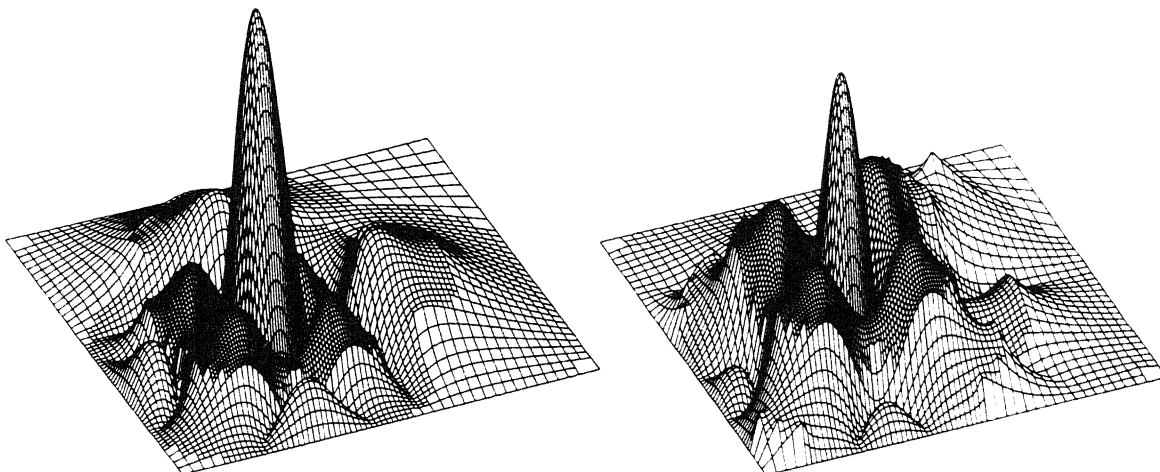


Fig. 3. Problem I: the scaled absolute values of the exact global space error in u and v obtained with four grid levels at $t = 0.25$.

time step size was chosen to be equal to 0.005 and the constant c from (5.17) equal to 0.5. The results at the final time, given in Table 1, show that the obtained global space errors decrease roughly with a factor of four indicating the normal second-order convergence behavior which would also be obtained with a single uniform grid. Since the success of the refinement strategy, described in the previous section, depends on the accuracy of the error estimates, we have also compared the numerical estimates of the global space error with the exact values. In Table 2 the numerical estimates of the global space errors are given and it appears that the estimates are quite accurate. The scaled absolute values of the exact global space error in u and v at $t = 0.25$ for all computations are shown in Figs. 1–3. The positioning of the finer subgrids in

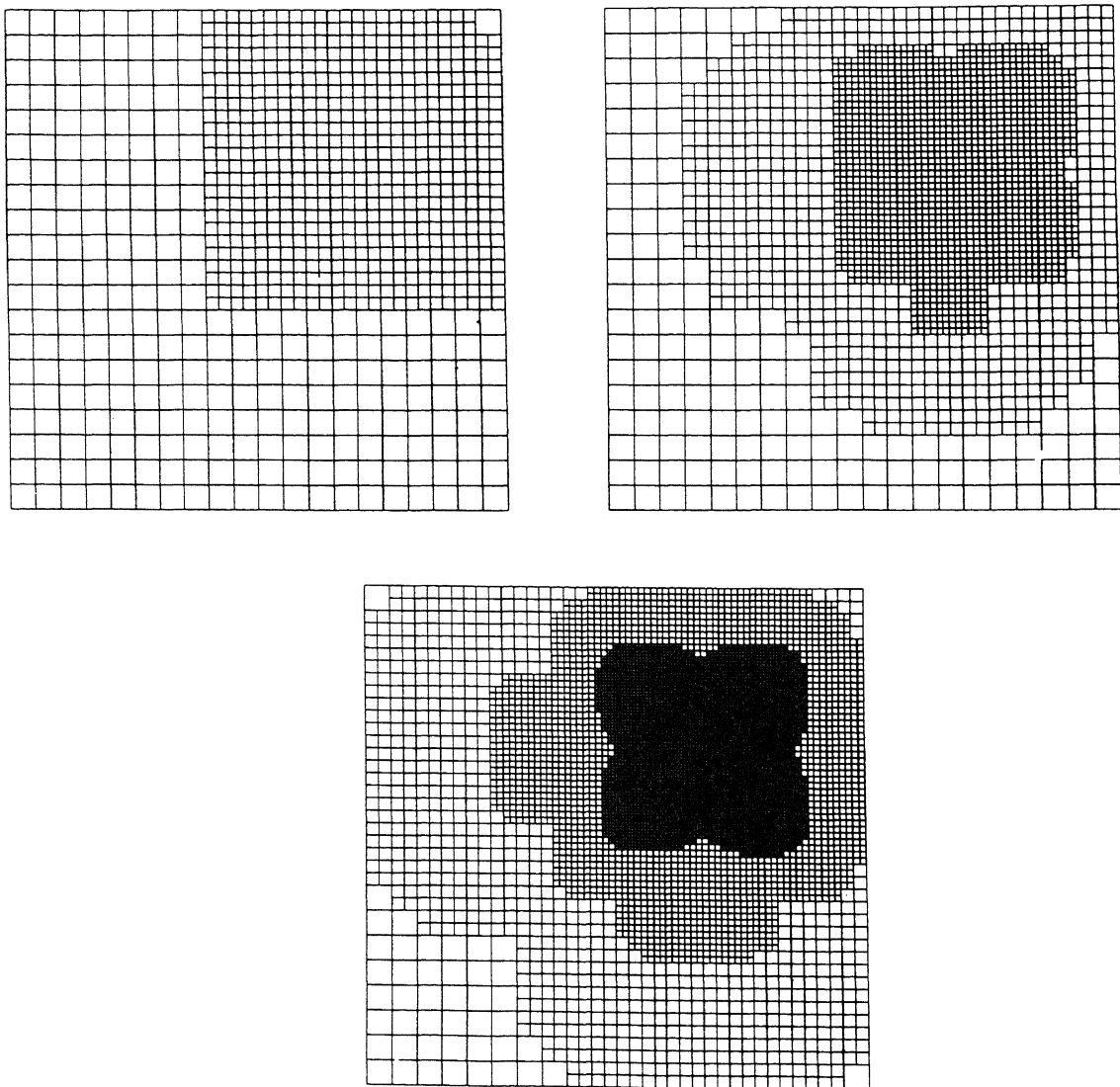


Fig. 4. Problem I: the grids of the two-level (top left), three-level (top right), and four-level (bottom) computations at $t = 0.25$.

Figs. 1–3 appears to be good. The maximum global space error for both components is located at the finest subgrid in use. Moreover, the refinements are fairly efficient, meaning that not many grid cells are unnecessarily refined. Figure 4 shows the grids at $t = 0.25$.

6.2. Problem II

The second test problem is a problem with a steady state solution. Again the system of PDEs given by (6.1) is solved but this time the sources g and h and the initial function and Dirichlet boundary conditions are chosen such that the exact solution is given by

$$u(x, y, t) = v(x, y, t) = \exp\left[-80\left((x - r(t))^2 + \left(y - \frac{1}{2}\right)^2\right)\right], \quad (6.3)$$

where $r(t) = \frac{1}{2} - \frac{1}{4} \exp(-1000t)$. This represents a cone which is centered at $(\frac{1}{2}, \frac{1}{4})$ at $t = 0$ and moves towards the center of the domain $(\frac{1}{2}, \frac{1}{2})$ with a continuously decreasing speed. In the steady state situation the cone will have reached the center of the domain.

Just like in example problem I, (6.1) was solved using one, two, and three extra grid levels after starting from a 20×20 uniform grid. Variable time step sizes were used. On the 20×20 grid at t_n , τ is predicted for the next time step so that $\tau U_t^n = 0.1$, where U_t^n is a numerical approximation of $u_t(x, t_n)$. These computed τ -values were kept in storage and used as time step sizes for all computations. The constant c from (5.17) is chosen equal to 0.5 for all computations.

This problem was used to illustrate the differences in behavior of the local and global space error belonging to (6.1a) and (6.1b) and to compare the maxima of the global space errors in u and v of the adaptive grid solution obtained with a different number of grid levels throughout the entire time interval. The solution was computed to $t = 1.0$. The *TOL*-values were equal to 20, 5, and 1. The number of grid levels was kept constant throughout the entire time interval. The results of Section 4.2 apply to this case, since (6.1) fits in the format (4.18). It is to be expected that the local space error in u behaves like $O(\tau)$ and the local space error in v like $O(1)$ when $\tau \rightarrow 0$. Figure 5 shows the behavior of the maximum local space error in u and in v

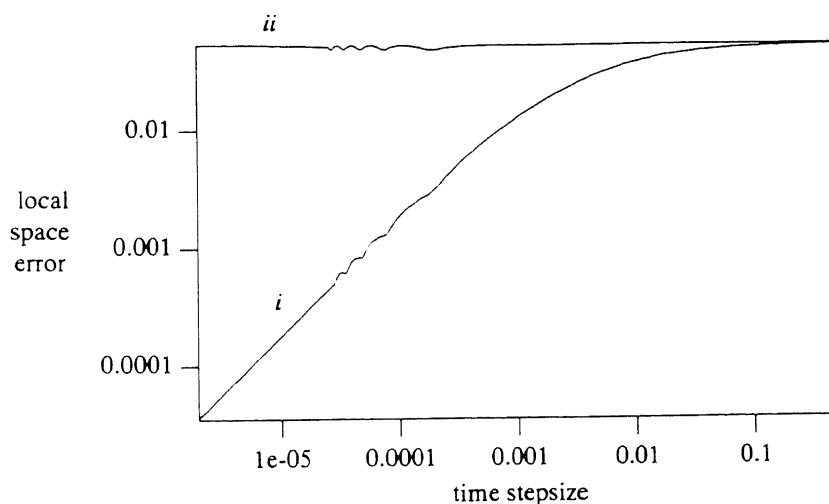


Fig. 5. Problem II: the maximum local space error in u (i) and in v (ii).

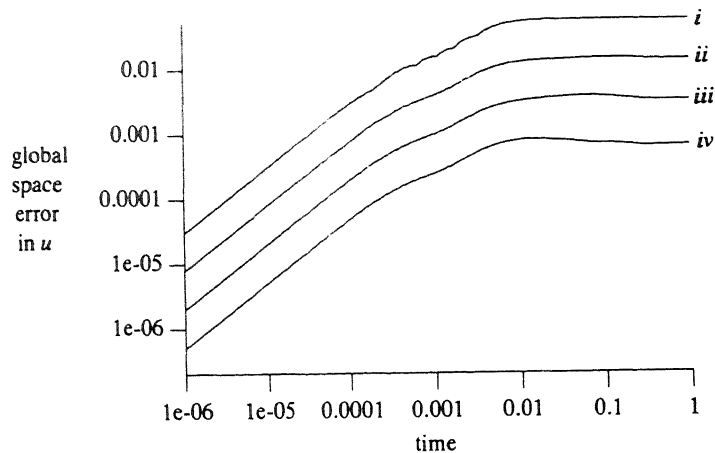


Fig. 6. Problem II: the maximum global space error in u obtained with one, two, three, and four grid levels, indicated by i , ii , iii and iv , respectively.

over the interior of a 20×20 grid as a function of the time step sizes. A double logarithmic scale was used in this figure and the slope of the local space error in u is almost equal to unity for small time step sizes indicating a linear behavior in τ . Further, the local space error in v appears to be almost constant. This means that the local space errors in u and v behave indeed like predicted by (4.26) for small τ . Figures 6 and 7 compare the maximum global space errors in u and v , respectively, obtained with the adaptive grid method on one, two, three, and four grid levels. These figures clearly reveal that the global space error in u , belonging to the PDE (6.1a) gradually increases in time until a certain maximum is reached while the global space error in v , connected with the PDE (6.1b) remains at an almost constant level over the entire time interval. The distances between the lines in Figs. 6 and 7 reveal a second-order convergence behavior which would also be obtained with a single uniform grid. The scaled

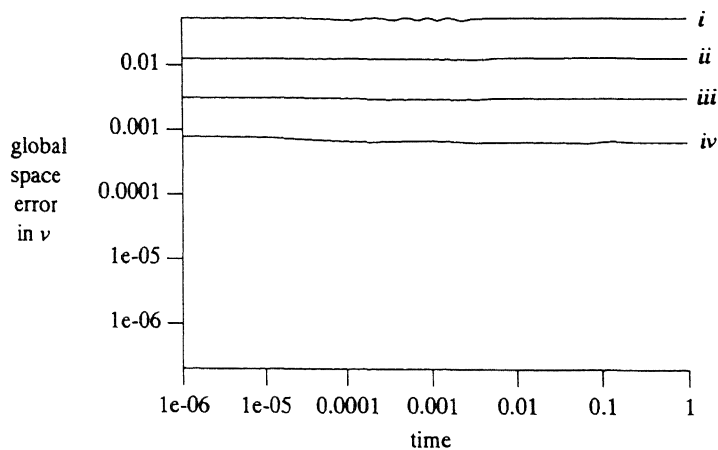


Fig. 7. Problem II: the maximum global space error in v obtained with one, two, three, and four grid levels, indicated by i , ii , iii , and iv , respectively.

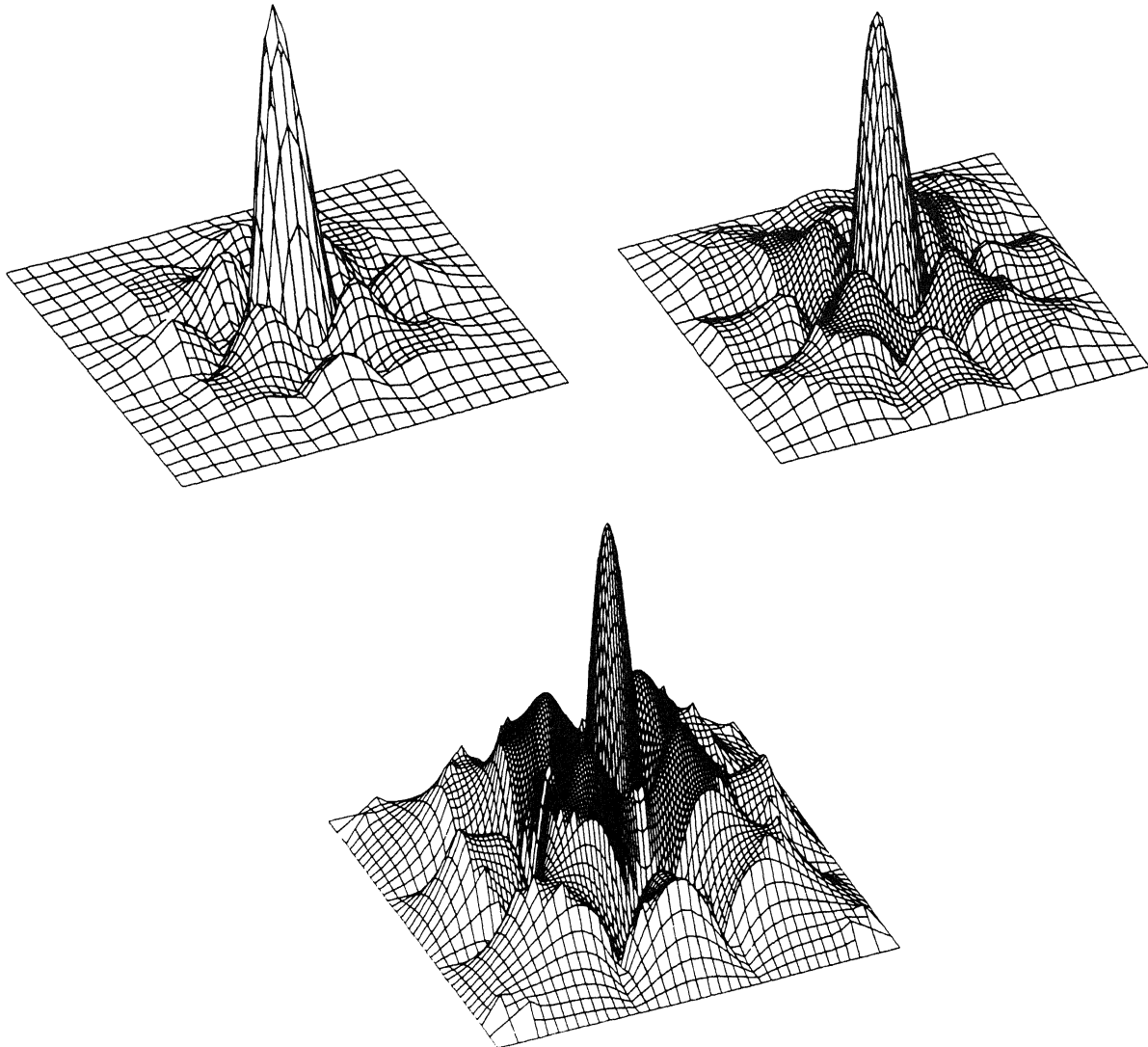


Fig. 8. Problem II: the scaled absolute values of the exact global space error in u obtained with two (top left), three (top right), and four (bottom) grid levels at $t = 1.0$.

absolute values of the global error in u at $t = 1.0$ are shown in Fig. 8. The global space error in v is not shown here, because it is very similar to the one in u for this case. Again, the grids are reasonably efficient and the maximum global space error is located at the finest subgrid in use. The grids at the final time are shown in Fig. 9.

6.3. Problem III

The third test problem is a problem with an oscillatory solution. The system of PDEs given by (6.1) is solved once more but this time the sources g and h and the initial function and

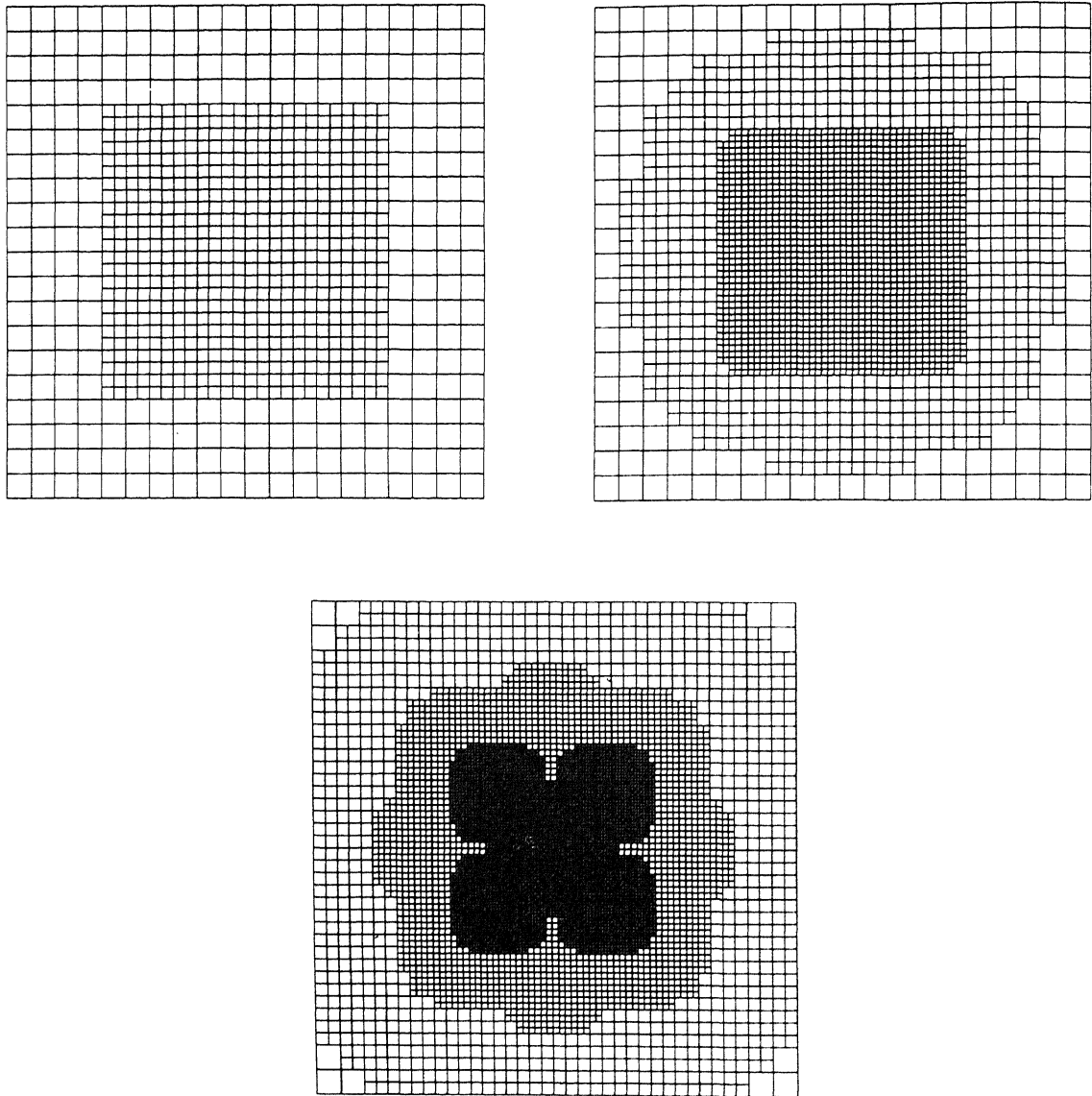


Fig. 9. Problem II: the grids of the two-level (top left), three-level (top right), and four-level computations at $t = 0.1$.

Dirichlet boundary conditions are chosen such that the exact solution is given by

$$u(x, y, t) = v(x, y, t) = \sin(\pi t) \exp\left[-320\left(\left(x - \frac{1}{2}\right)^2 + \left(y - \frac{1}{2}\right)^2\right)\right]. \quad (6.4)$$

This represents an oscillating cone which is centered at $(\frac{1}{2}, \frac{1}{2})$. At $t = 0$ the solution is zero everywhere. Then a steep pulse emerges at the center of the domain which reaches its maximum at $t = 0.5$. After this it will decay until the solution is equal to zero again at $t = 1.0$.

This problem was solved to test the performance of the method when a variable number of grid levels is used. The solution was computed four times from $t = 0$ to $t = 1.0$ using a maximum number of grid levels of two, three, four, and five. The corresponding *TOL*-values

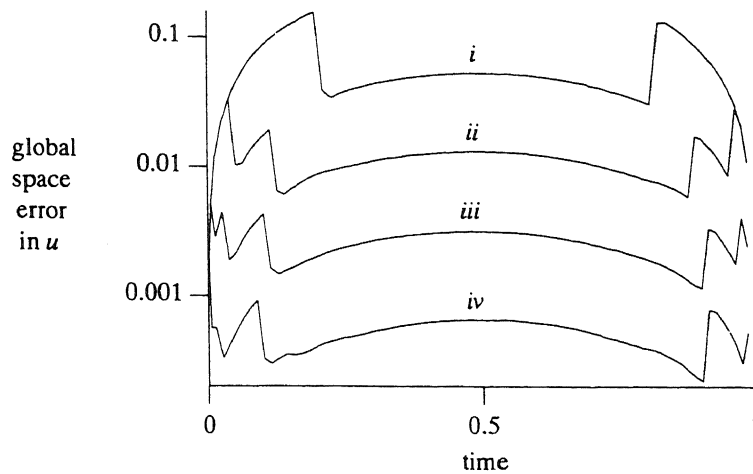


Fig. 10. Problem III: the maximum global space error in u as a function of time. The maximum number of grid levels is two (*i*), three (*ii*), four (*iii*), and five (*iv*).

were 160, 40, 10, and 2.5, respectively, and the constant c was chosen to be 0.5. Variable time steps were also used here which were determined in exactly the same manner as in problem II and also kept in storage to be used for all computations. The maximum global space error in u as a function of time is shown in Fig. 10. Here, the behavior of the global space error in v is very similar to the one in u . The kinks in this figure indicate that, at that time, a new finer grid level is created or discarded. It appears that the global space error decreases with a certain factor when the TOL -value is divided by four. Inspection of the data revealed that this factor is larger than four in both the infinity and the L1 norm. These norms of the maximum global space error were taken over the values at all time levels. This implies that, when the TOL -value is decreased by a factor of four, the spatial accuracy is increased by a factor of at least four for this example problem. The grids at $t = 0.5$ are shown in Fig. 11.

7. Summary and concluding remarks

In this paper we have discussed the application of a local uniform grid refinement method to systems of coupled PDEs. The main feature of local uniform grid refinement is that the PDEs are solved on a series of nested, uniform, cartesian, increasingly finer subgrids covering only a part of the domain where the spatial error is high. The PDEs are solved on these subgrids in a consecutive manner, from coarse to fine. The location and size of the subgrids are automatically adjusted at discrete times in order to follow the movement of the steep fronts. The generation of subgrids is continued until sufficient spatial accuracy is reached.

An error analysis was performed for the local uniform grid refinement method applied to systems of coupled PDEs. It was shown that the global and local error components associated with each separate PDE can exhibit an entirely different behavior. With respect to the global error, this means that the global error components can carry over to future time points in a very different way from one PDE to another. The local space error components can show a different behavior for small time step sizes. A refinement strategy controlling the generation of finer

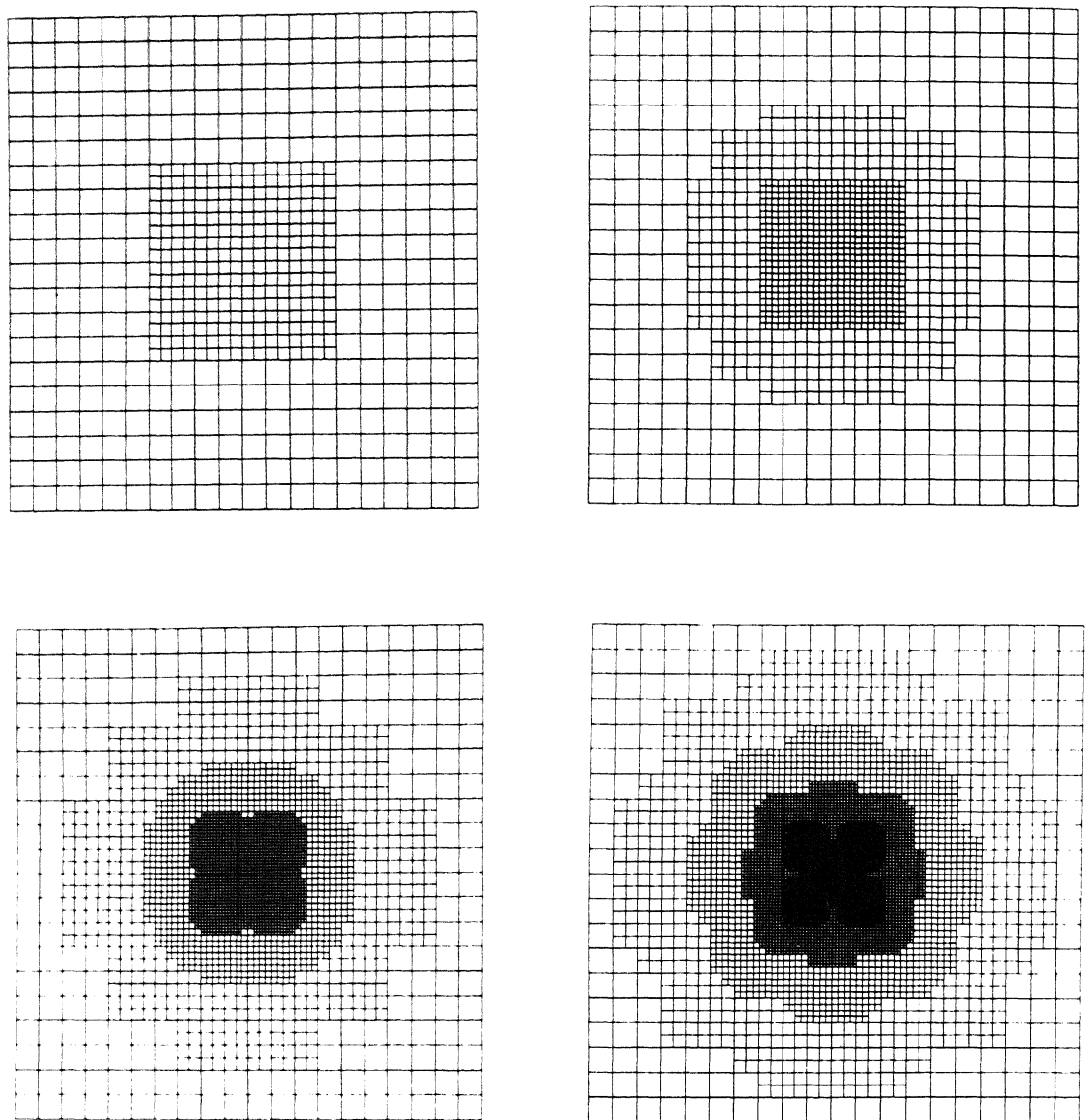


Fig. 11. Problem III: the grids of the two-level (top left), three-level (top right), four-level (bottom left), and five-level (bottom right) computations at $t = 0.5$.

subgrids was developed from the results of the error analysis. This strategy takes these differences in behavior into account and is based on estimating and controlling the global space error. We have applied the method to three example problems, all involving a system containing a parabolic and an elliptic equation to test the refinement strategy. The observed convergence behavior of the global space error is comparable to uniform grid computations. We have also seen the predicted differences in behavior of the components of the global and local space error. Further, the global error estimates are fairly accurate and not many grid cells appear to be unnecessarily refined. Using both a fixed and a variable number of grid levels in

time and a second-order space discretization, we have observed that when the tolerance value is decreased with a factor of four, the spatial accuracy also appears to improve with a factor of four.

We consider these results to be very satisfactory. However, we feel that testing on more difficult (nonlinear) problems needs to be done in order to fully appreciate this refinement strategy. Further, in the example problems where variable time step sizes were used, these step sizes were adapted in order to equidistribute a heuristic monitor. It would be desirable to implement a time step strategy based on (4.15) or (4.17). However, such a strategy only works well when the time error estimates are sufficiently accurate. In [9], we have already reported that when using the local uniform grid refinement method these time error estimates do not resemble the actual time error at all. Nevertheless, perhaps there is a remedy for this so that a time step strategy can be developed which is just as successful as the refinement strategy in space.

References

- [1] D.C. Arney and J.E. Flaherty, An adaptive local mesh refinement method for time-dependent partial differential equations, *Appl. Numer. Math.* 5 (1989) 257–274.
- [2] D.C. Arney and J.E. Flaherty, An adaptive mesh-moving and local refinement method for time-dependent partial differential equations, *ACM Trans. Math. Software* 16 (1990) 48–71.
- [3] M.J. Berger and J. Olinger, Adaptive mesh refinement for hyperbolic partial differential equations, *J. Comput. Phys.* 53 (1984) 484–512.
- [4] W.D. Gropp, Local uniform mesh refinement with moving grids, *SIAM J. Sci. Statist. Comput.* 8 (1987) 292–304.
- [5] W.D. Gropp and D.E. Keyes, Domain decomposition with local mesh refinement, *SIAM J. Sci. Statist. Comput.* 13 (1992) 967–993.
- [6] J.E. Flaherty, P.K. Moore and C. Ozturan, Adaptive overlapping grid methods for parabolic systems, in: J.E. Flaherty, P.J. Paslow, M.S. Shephard, J.D. Vasilakis, eds., *Adaptive Methods for Partial Differential Equations* (SIAM, Philadelphia, PA, 1989) 176–193.
- [7] R.A. Trompert and J.G. Verwer, Analysis of the implicit Euler local uniform grid refinement method, *SIAM J. Sci. Comput.* 14 (1993) 259–278.
- [8] R.A. Trompert and J.G. Verwer, Runge–Kutta methods and local uniform grid refinement, Report NM-R9022, Centre for Mathematics and Computer Science, Amsterdam (1990); also: *Math. Comp.* (to appear).
- [9] R.A. Trompert and J.G. Verwer, A static-regridding method for two-dimensional parabolic partial differential equations, *Appl. Numer. Math.* 8 (1991) 65–90.
- [10] J.G. Verwer and R.A. Trompert, An adaptive-grid finite-difference method for time-dependent partial differential equations, in: D.F. Griffiths and G.A. Watson, eds., *Proceedings 14th Biennial Conference on Numerical Analysis*, Pitman Research Notes in Mathematics Series 260 (Pitman, London, 1991) 267–284.
- [11] J.G. Verwer and R.A. Trompert, Analysis of local uniform grid refinement, Report NM-R9211, Centre for Mathematics and Computer Science, Amsterdam (1992); also: *Appl. Numer. Math.* 13 (1993).

these
space
system
erved
tions.
l and
l cells
els in