

Konrad-Zuse-Zentrum für Informationstechnik Berlin

Peter Deuffhard*

Jens Lang

Ulrich Nowak

Adaptive Algorithms in Dynamical Process Simulation

* Invited talk, ECMI meeting 1994, Kaiserslautern

Preprint SC 95-16 (May 1995)

Adaptive Algorithms in Dynamical Process Simulation

Peter Deuffhard*, Jens Lang and Ulrich Nowak

Konrad-Zuse-Zentrum (ZIB), Berlin, Germany

1 Introduction

Dynamical process simulation plays an increasingly important role in the design and control of chemical plants. Mathematically speaking, the simulation of such processes requires the numerical solution of systems of partial differential equations (PDEs) of *reaction-diffusion* type with possibly mild convection. In contrast to some other fields of application, *time dependence* of the process is of real interest. Moreover, due to additional constraints, *differential-algebraic* equations will naturally arise. As for the spatial geometry, radial or simply plane symmetry will result in 1-D problems, whereas more complex situations will lead to 2-D or even 3-D models, often only given in the form of some CAD input. Even though such problems have been around for quite a while, they still represent a class of hard problems. For this reason, the development of robust and fast algorithms has been a topic of continuing investigation during the last years. In particular, significant progress has been made by the development of *adaptive* algorithms, which aim at the control of time and space grids in such a way that on one hand the solution is as accurate as required by the user and on the other hand the necessary work to obtain such a solution is minimized. The present paper surveys some of the essential features of such adaptive methods, which have been developed recently by the authors.

In Section 2 below, the well-known method of lines approach based on *first space then time* discretization is revisited in view of adaptivity. After space discretization on a fixed grid, a block structured ODE system has to be solved numerically. Time integration with order and stepsize control represents a first element of adaptivity, which is already quite popular. In contrast to the widely used BDF integration formulas (GEAR, PETZOLD [14, 25]), which

*Invited talk, ECMI meeting 1994, Kaiserslautern

require the solution of a nonlinear system per time step, recent impressive progress has been made by the application of *linearly-implicit* time discretizations, which only require the solution of a linear system per time step. Especially when combined with techniques of *static regridding* the linearly-implicit *one-step* methods have proven to be the methods of choice both for theoretical reasons and for evidence of performance. However, spatial regridding techniques are typically restricted to 1-D situations or to associated tensor product extensions in more than one dimension (which implies rather simple geometries). Therefore, Section 3 treats the complementary type of method, the so-called *Rothe method*, which is based on *first time then space* discretization. In this approach, which in the described adaptive version has first been suggested by BORNEMANN [2, 3] for parabolic equations, the time dependent PDE is understood as an (ordinary) differential equation in an infinite dimensional space (Hilbert space). This permits one to apply the fully developed technology of ordinary differential equations (in finite dimensions) and to treat the necessary space discretization afterwards according to the accuracy requirements within each time layer. When combined with a multilevel of multigrid method in space, a special kind of *dynamic regridding* with time dependent number and distribution of nodes is realized, which at the same time also leads to rather fast algorithms. One technique of adaptive multilevel methods called *cascade principle*, which is due to DEUFLHARD, LEINEN, YSERENTANT [10], is explained in some detail – both for the selfadjoint (elliptic) case and for the non-selfadjoint case, which arises in process simulation. Three challenging numerical examples are inserted to illustrate the relative merits of the here discussed methods.

2 First Space then Time Discretization

Consider a system of partial differential equations (PDEs) of reaction–diffusion type with possibly mild convection

$$B(x, t, u, u_x)u_t = f(x, t, u, u_x, (D(x, t, u)u_x)_x) . \quad (1)$$

With additional initial and boundary conditions, we have a nonlinear parabolic initial boundary value problem for a system of PDEs.

In this section we treat the by know most popular technique, the *method of lines*, which approaches the discretization of the above PDE system by first

space then time discretization. In the standard method of lines, adaptivity only shows up in a variable step and possibly variable order time discretization. Variants of the method, where adaptivity in space is also achieved, thus arriving at fully adaptive methods, are, however, restricted essentially to 1D situations, as will be shown below.

Time Adaptive Method of Lines. Upon applying some kind of space discretization (e.g. finite differences, finite elements or spectral methods) to the above PDE system, we arrive at a system of ordinary differential equations (ODEs) or differential-algebraic equations (DAEs) of the kind

$$\mathbf{B}(\mathbf{u})\mathbf{u}' = \mathbf{f}(t, \mathbf{u}), \quad (2)$$

wherein the matrix \mathbf{B} may be singular. This system is nonlinear, stiff, block structured, and typically large and sparse in industrial applications. Its numerical solution can, in principle, be attacked by any stiff integrator, which in linear stability theory shows a vanishing root at infinity – compare, for instance, the textbooks HAIRER/WANNER [15] or DEUFLHARD/BORNEMANN [8]. Among the most efficient integrators of this type are the implicit multistep code DASSL due to PETZOLD [25], the implicit Runge-Kutta code RADAU5 due to HAIRER/WANNER [15], and the linearly implicit extrapolation code LIMEX due to DEUFLHARD/NOWAK [9]. The latter code has recently proved to be clearly preferable for large scale dynamical process simulation, since it requires – because of its linearly implicit structure – one iteration loop less than the other two codes, which are implicit. This greater simplicity is a clear structural advantage in really complex programming environments. The time discretization within LIMEX is based on the elementary linearly implicit Euler discretization of the type

$$(\mathbf{B}(\mathbf{u}) - \tau\mathbf{J})\Delta\mathbf{u} = \tau\mathbf{f}(t + \tau, \mathbf{u}), \quad (3)$$

where

$$\mathbf{J} \approx \frac{\partial}{\partial \mathbf{u}}(\mathbf{f} - \mathbf{B}\mathbf{u}')|_{t=t_0} . \quad (4)$$

This elementary discretization is then extended to variable order by means of extrapolation – cf. DEUFLHARD [6]. Note the special variant for time dependent right hand sides. Incidentally, the rather frequent occurrence that

boundary values arise via an ODE does obviously not create any difficulty in this setting. If the dynamical process under consideration has rich dynamics, then usually grid adaptation after each or few steps will be necessary. This technique, however, requires the generation of initial values at the new nodes by interpolation and may lead to changes of the dimension of the ODE or DAE to be solved; in such a situation one-step methods have a natural advantage – compare e.g. FLAHERTY ET.AL [12]. MOORE/FLAHERTY [20] recommend the application of (fixed order) singly-implicit Runge-Kutta methods, which however share the disadvantage of being implicit, so that some kind of Newton-like iteration is needed.

For an illustration of the comparative performance of LIMEX versus DASL we refer to the combustion problem simulations done by MAAS/WARNATZ [19], see also DEUFLHARD/NOWAK/WULKOW [11], wherein computational speed-up factors of 10 - 15 in 1D and (estimated) factors of more than 100 in 2D have been reported.

Space Adaptation. A systematic approach towards a fully adaptive method of lines treatment of nonlinear parabolic PDE systems has recently been published by NOWAK [21] in his thesis. An overview on the method and its application to some real life problems from chemical engineering can be found in NOWAK ET AL. [22]. The approach starts from second order finite difference discretizations, which allow for a quadratic consistency error; the use of central differences for the convection term (for symmetry reasons!) restricts this approach to mild convection (say $Re \leq 1000$). Under the assumption that the discretization error is also quadratic (hard to prove theoretically in the general case, but easy to monitor within the adaptive algorithm), a coupled extrapolation in both space and time is performed. This extrapolation procedure yields error estimates for both the error coming from space discretization and the error coming from time discretization. Based on these estimates the local approximation errors are controlled and new stepsize proposals can be derived. This fully adaptive scheme is worked out on a non-uniform grid using an implicit monotone grid function mapping it onto a uniform grid. The way of adapting the space grids is preferably done by *static rezoning*, which involves some careful consideration of the applied interpolation schemes. In [21] the interpolation is done by means of the piecewise monotone Hermite interpolation due to FRITSCH/BUTLAND [13]. The interpolation error is monitored by some heuristics. In addition to the grid adaptation by static rezoning a so-called *moving grid* technique

may be used. This dynamic rezoning approach is similar to the one used e.g. by PETZOLD [26] and pays off especially for problems where the solution is characterized by a single moving front. Both, the static and the dynamic rezoning approach are restricted to 1D problems or simple tensor product grid extensions in more than 1D (which means rather simple geometries).

Example 1: Automobile Catalytic Converter [22]. The mathematical model comprises the following system of PDEs

$$\begin{aligned}
\varepsilon^W \rho^W c_p^W \frac{\partial T^W}{\partial t} &= \varepsilon^W \lambda^W \frac{\partial^2 T^W}{\partial x^2} && + q_T^W \\
\varepsilon^G \rho^G c_p^G \frac{\partial T^G}{\partial t} &= \varepsilon^G \lambda^G \frac{\partial^2 T^G}{\partial x^2} && - \frac{\dot{m} c_p^G}{A} \frac{\partial T^G}{\partial x} + q_T^G \\
(1 - \varepsilon^G) \rho^K c_p^K \frac{\partial T^K}{\partial t} &= (1 - \varepsilon^G) \lambda^K \frac{\partial^2 T^K}{\partial x^2} && + q_T^K \\
S &= CO, C_3H_6, H_2, O_2 : \\
(1 - \varepsilon^G) \rho^G \frac{\partial g_S^K}{\partial t} &= (1 - \varepsilon^G) \rho^G D_{eff,S} \frac{\partial^2 g_S^K}{\partial x^2} && + q_S^K \\
\varepsilon^G \rho^G \frac{\partial g_S^G}{\partial t} &= \varepsilon^G \rho^G D_{eff,S} \frac{\partial^2 g_S^G}{\partial x^2} && - \frac{\dot{m}}{A} \frac{\partial g_S^G}{\partial x} + q_S^G,
\end{aligned}$$

wherein the 11 state variables are: 3 temperatures (gas, converter surface and converter hull) and 2×4 species concentrations (gas, converter surface).

In order to reduce air pollution the study of the startup phase is of great importance. The simulation presented in [22] assumes that the polluted air enters with a linearly increasing temperature (and with constant velocity) at the left boundary. In Fig. 1 the concentration of the pollutant C_3H_6 is shown in a (x, t) -plane. The numerical solution on the computational grid is plotted at all internally selected integration points – as obtained by the program PDEX1M. The displayed time interval is $[0, 100]$ (whereas the actual integration interval was $[0, 1000]$).

Initially the converter is rapidly filled with inflowing gas. This shows up in Fig. 1 for small times ($t \in [0, 0.2]$) as a front moving from left to right. In order to resolve this fast process, small timesteps and a grid with local refinements within the front arise automatically. After the initial filling process is completed, the temperature is not yet high enough to start the reaction so that larger time steps and a coarser grid appear to be appropriate. At $t = 25$, the reaction starts and reduces the concentration of C_3H_6 . Steep

gradients at the left boundary develop and require both small stepsizes in time and a very dense grid at the left boundary. At $t = 50$, the dynamics of the problem dies out and accordingly timesteps increase.

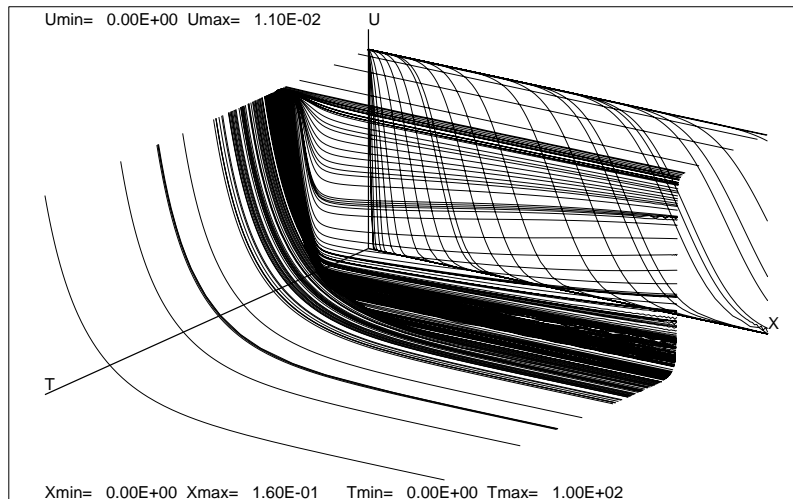


Fig. 1: Solution component C_3H_6 of example 1

Finally, the process becomes nearly stationary allowing for very large steps. The simulation from $t = 100$ to $t = 1000$ (not shown in Fig. 1) requires only 5 more steps.

3 First Time then Space Discretization

In the mathematical literature, the discretization first in time then in space is often referred to as Rothe method reminding of an early paper on the discretization of parabolic PDEs – see [28]. For *fixed* space and time grids, the sequence of the two discretizations is negligible from the numerical point of view. If, however, *adaptive* space grids as a function of time are to be applied, then the sequence of the discretizations matters indeed.

Adaptive Rothe Method. In his work on adaptive FEMs for parabolic PDEs, BORNEMANN[2, 3] introduced an algorithmic approach which is nowadays denoted by the term "adaptive Rothe method". For ease of presentation, we start with the simple scalar parabolic equation

$$u_t = (D(x)u_x)_x, \quad u(x, 0) = u_0(x), \text{ boundary conditions.} \quad (5)$$

If we incorporate the boundary conditions together with the diffusion operator into the operator \mathcal{A} , then we may write (5) as an ODE in Hilbert space of the form

$$U' = \mathcal{A}U, \quad U(0) = U_0. \quad (6)$$

In this formulation, we may apply all the well-developed ODE technology to solve this equation. For example, if we discretize this equation by the implicit Euler method, we arrive at

$$(I - \tau\mathcal{A})\Delta U = \tau\mathcal{A}U_0, \quad U_1 = U_0 + \Delta U. \quad (7)$$

This is a special *linear elliptic boundary value problem*, which may be attacked by a FEM or FDM, where the FEM should be preferred, if the geometry of the problem is sufficiently complicated. Extensions to higher order time discretizations lead to one linear BVP per stage. Error estimates required for the order and stepsize control may be approximated in the same framework. In his first paper, which dealt with 1D problems only, Bornemann had still used extrapolation methods, whereas in later papers, which included 2D and 3D as well, he had designed a more sophisticated time discretization of a recursive SDIRK type. In this approach higher order results U_j are computed from corrections η_j via a *multiplicative error correction* scheme of the form

$$\begin{aligned} &\text{Given } u_0, \eta_0. \\ &\text{Compute for } j = 1, \dots \\ &U_{j+1} = U_j + \eta_j, \quad \eta_{j+1} = R_j(\tau\mathcal{A}) \eta_j, \end{aligned} \quad (8)$$

wherein the rational functions R_j satisfy $R_j(\infty) \leq 1$. (The whole time discretization has the property $R(\infty) = 0$ and is also applicable to degenerate parabolic equations.) As seen above, the corrections are computed in a multiplicative way rather than in an additive way – so that small numbers are

computed by division rather than by subtraction. As a consequence, much less stringent spatial error requirements compared with time errors are possible as compared with extrapolation methods. Moreover, this variable order time discretization works on just one single space grid.

In LANG/WALTER[16], the adaptive Rothe method has been carried over to the situation of reaction–diffusion systems (1). Proceeding as in the simple parabolic case, we now arrive at an abstract ODE of the kind

$$U' = \mathcal{A}U + F(U), \quad U(0) = U_0. \quad (9)$$

with F representing the nonlinear reaction terms. One extension of the above implicit Euler discretization for the linear parabolic equation is the linearly-implicit Euler method applied to the equivalent formulation

$$U' - \mathcal{A}U - F_U(U_0)U = F(U) - F_U(U_0)U, \quad U(0) = U_0, \quad (10)$$

which leads to

$$(I - \tau(\mathcal{A} + F_U(U_0)))\Delta U = \tau(F(U_0) + \mathcal{A}U_0), \quad U_1 = U_0 + \Delta U. \quad (11)$$

This is a *non-selfadjoint* linear elliptic boundary value problem. As higher order extension, LANG[17, 18] selected a linearly-implicit differential-algebraic method due to ROCHE [27], which has only 3 stages for order 3 with embedded order 2 for stepsize control. As in the parabolic descendant, the discretization has $R(\infty) = 0$ and requires only one spatial mesh. The multiplicative error correction structure from above is not explicitly but essentially inherited. The method is of Rosenbrock type, which means that it would require an *exact* representation of the operator $\mathcal{A} + F_U(U_0)$ – which, however, is not available in the FE context here. The thus unavoidable Jacobian approximation errors may lead to restrictions of the time steps obtained from the stepsize control – an effect, which has been observed experimentally for low accuracy requirements. In principle, so-called W -methods (cf. the textbooks [15, 29, 8]) would be preferable to reflect the approximation property of the problem correctly, which, however, would require more stages.

Cascadic Finite Element Methods. Up to now, we did not specify the actual solution of the linear boundary value problems of the type (7) or (11). In both cases, an adaptive multilevel FEM has been applied. We explain the basic algorithmic approach for the simpler *selfadjoint* case first. In this case, the so-called *cascade principle* as developed by DEUFLHARD, LEINEN, YSERENTANT[10] has been used. It starts with the *direct* solution of the linear system obtained by the FE discretization on a comparatively coarse grid, which, however, is assumed to catch the essential features of the problem formulation (boundary and interface conditions). The obtained coarse grid solution is checked via FE error estimators in terms of the user required accuracy. If this requirement is not yet met, then local error indicators are used to obtain some refined grid. On the finer grid, the (interpolated) previous solution is used as starting point for a *preconditioned conjugate gradient* (PCG) iteration. This PCG iteration is terminated, as soon as an estimated algebraic error is sufficiently below the expected discretization error, so as to avoid unnecessarily accurate computations on the refinement levels. After termination of the PCG iteration, the FE error estimator is once more applied to check for the user prescribed tolerance – to be repeated recursively until this tolerance is met on some finest mesh, which in this way may come out to be highly non-uniform. As a preconditioner, the hierarchical basis preconditioner of YSERENTANT[32] may be used in 2D, which has nearly optimal formal computational complexity and is quite cheap to implement. In 3D, the more costly so-called BPX preconditioner of XU[5, 31] pays off, which has formally optimal computational complexity. Rather recently, even more effective so-called CCG methods have been developed in DEUFLHARD[7] and BORNEMANN/DEUFLHARD[4], which avoid preconditioning at all (apart from diagonal scaling) at the expense of a few more iterations on coarser grids, which are carefully controlled.

For the *non-selfadjoint* case of dynamical process simulation, the PCG iteration needs to be replaced by some unsymmetric iterative solver. Extensive numerical experiments clearly suggested the option BICG-STAB due to [30] in combination with SSOR preconditioning. In all the examples tested so far, this combination led to a nearly constant small number of iterations on all refinement levels. So, even in this non-selfadjoint case, optimal computational complexity is achieved – however, without any theoretical explanation yet. As for the FE estimator to monitor the discretization errors, the present version of the program

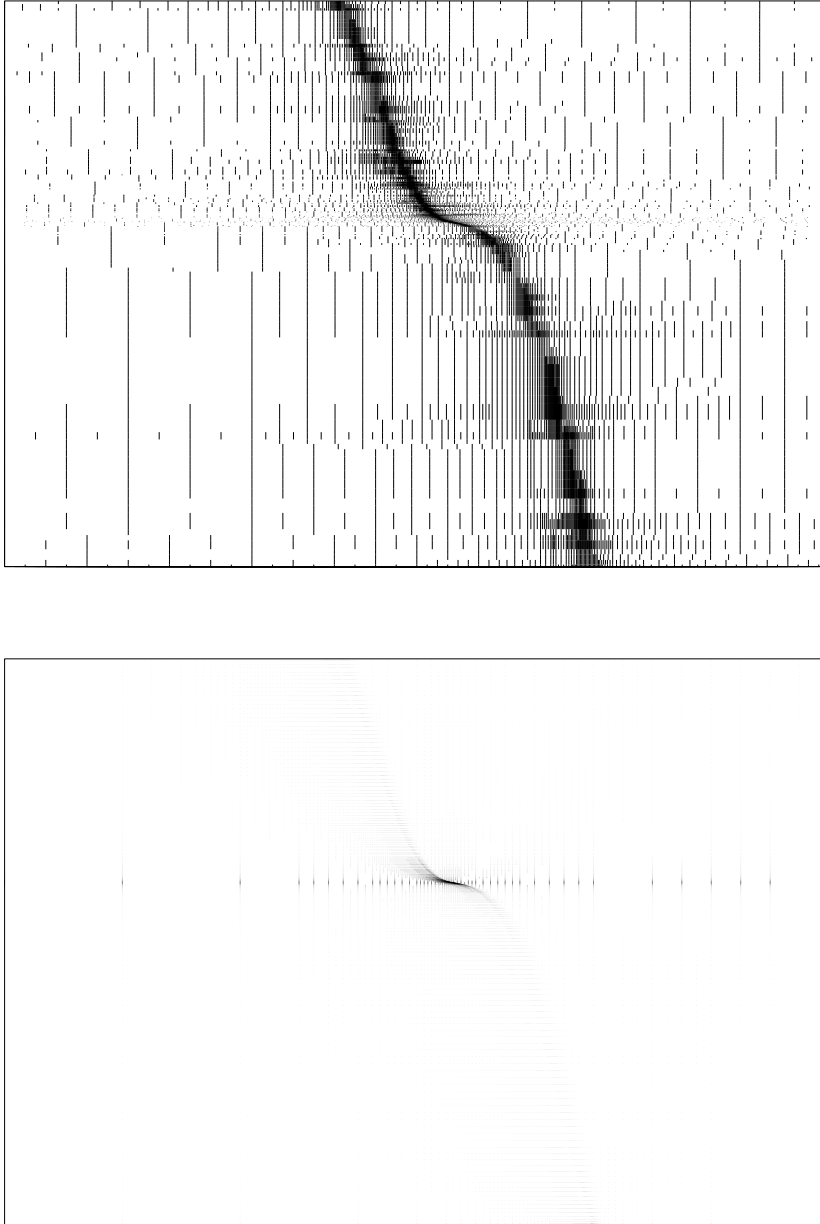


Fig. 2: Nodal flux of Example 2 from method of lines treatment (above) versus Rothe method (below).

KARDOS (mnemotechnically for KAskade Reaction DiffusiOn System) realizes a special interpolation error estimate – see [18].

Example 2: Comparative test problem [24]. This 1D problem describes a moving flame front, which changes its shape during the process. In the present context, this example is just taken to illustrate the two different adaptivity concepts described so far. In Fig. 2 the nodal flux (horizontal axis: space grids, vertical axis: time) obtained from the fully adaptive method of lines with static rezoning (above) is compared with the one obtained from the adaptive Rothe method with linear finite elements (below). Obviously, the plot below is a bit "smoother" than the plot above, requiring less spatial nodes but more time steps – an observation, which nicely goes with the underlying algorithmic concepts. In total, however, both pictures show a striking similarity. A comparison of the associated computing times, even though certainly desirable, is hard to give – at present, the two codes are just too different. On the other hand, the example is just 1D and therefore not at the center of interest of the present article.

Example 3: Thermally anchored flame [1]. This problem describes the propagation of a two-dimensional premixed flame in a gaseous mixture. For the numerical simulation the adaptive Rothe method (program KARDOS) has been applied. Some simplifications of the underlying physical processes lead to the so-called thermodiffusive model described by the reaction-diffusion equations

$$\begin{aligned} \partial_t T - \Delta T &= R(T, Y) + V \partial_x T \\ \partial_t Y - \frac{1}{Le} \Delta Y &= -R(T, Y) + V \partial_x Y \\ R(T, Y) &= \frac{\beta^2}{2 Le} Y \exp \left[\frac{-\beta(1-T)}{1-\alpha(1-T)} \right] \end{aligned} \quad (12)$$

where T is a normalized temperature variable, Y is the reduced mass fraction of the reactant. The initial data are chosen to represent a planar steady flame in the limit $\beta \rightarrow \infty$. Furthermore, homogeneous Neumann conditions are imposed at the whole tube wall except at some special part, where the temperature is forced to be equal to the adiabatic flame temperature $T_A = 1$. This additional condition inhibits the run-off of the flame through the reactor thus representing some thermal "anchoring". The problem was solved for the parameters $Le = 1$, $\beta = 10$, $\alpha = 0.84$, $V = -5$.

A correct simulation of the flame propagation requires a dense computational mesh within the thin flame region, and especially in the boundary layer caused by the time-fixed Dirichlet condition. First experiences have

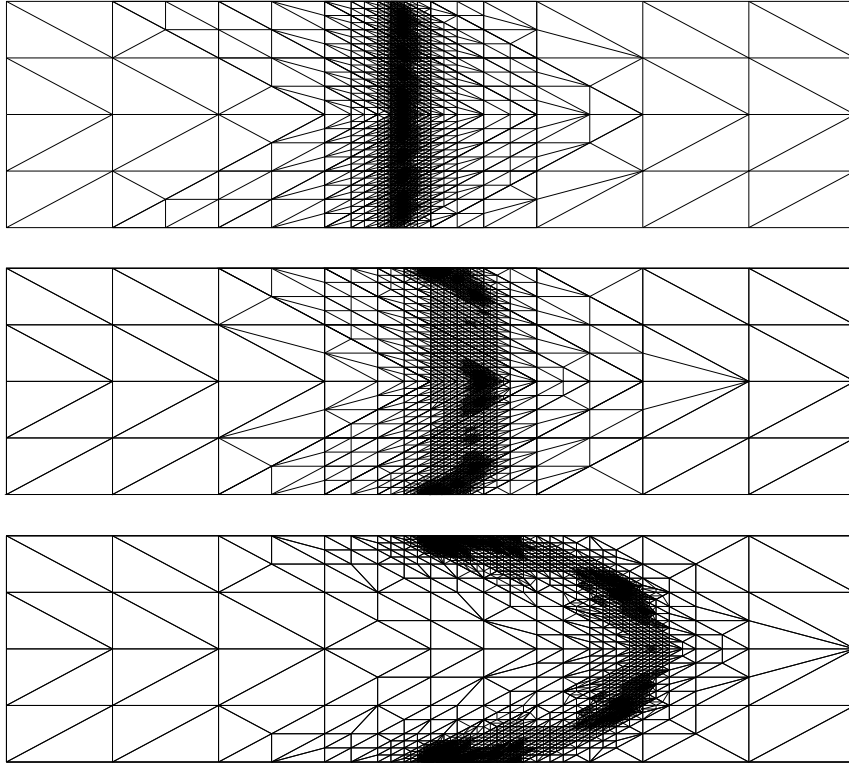


Fig. 3: Adaptive grids at $t=0.0$, 2100 nodes (above); $t=1.35$, 2900 nodes (middle); at $t=4.29$, 14000 nodes (below)

shown that at least a tolerance $TOL = 1.0e - 4$ is needed to reflect the dynamics of the process adequately. Such a quite stringent tolerance requires 2100 nodes at the beginning up to 14000 nodes at the end of the process. Note, however, that estimated 10^{10} triangles would be needed to guarantee the same accuracy on a uniform mesh. In Fig. 3 the dynamics of the grid development is shown, whereas in Fig. 4 the level surfaces of the solution are given for comparison. The whole computation is rather time consuming; speed-ups would certainly be possible replacing the linear finite elements by some $h - p$ -strategy [33] for higher order elements – which is the subject of further work.

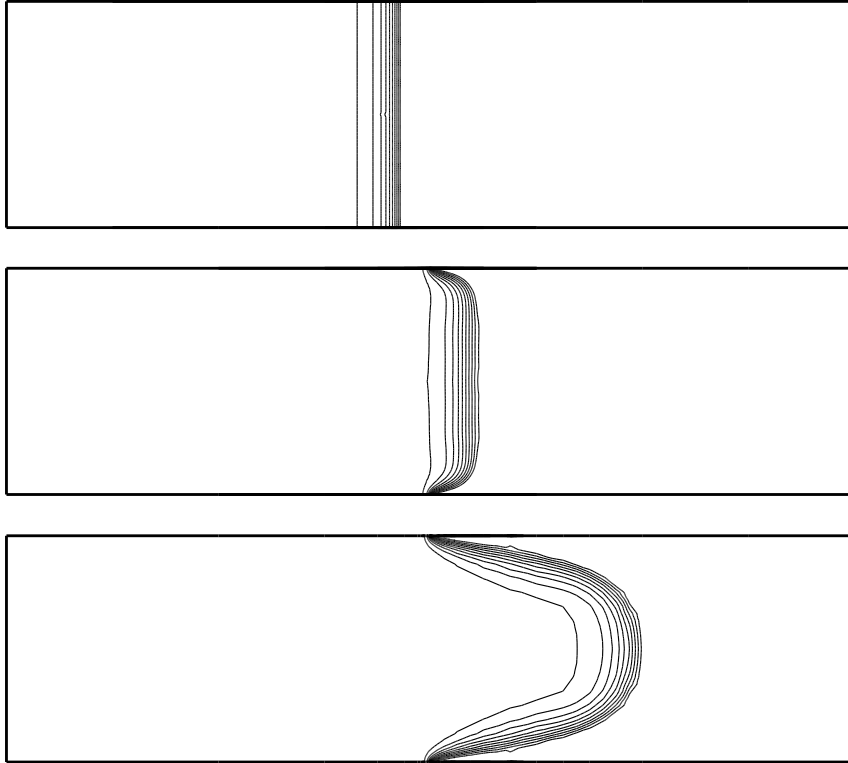


Fig. 4: Adaptive solutions at $t=0.0$ (above), $t=1.35$ (middle), $t=4.29$ (below)

Conclusion

Dynamical simulation of industrially relevant processes strongly advises the use of algorithms, which are *adaptive* both in time and in space discretization. The paper presented two alternatives: (a) a fully adaptive method of lines approach, which is based on finite difference methods and essentially applicable to 1D problems; (b) a fully adaptive Rothe method, which is based on a fast multilevel finite element method and applicable to 1D up to 3D.

Bibliography

- [1] F. Benkhaldoun, B. Larrouturou: *Explicit adaptive calculations of wrinkled flame propagation*. Int. J. Numer. Meth. in Fluids 7, p. 1147–1158 (1987)
- [2] F.A. Bornemann: *An Adaptive Multilevel Approach to Parabolic Equations. I: General Theory and 1D-Implementation*. IMPACT Comput. Sci. Engrg. 2, p. 279–317 (1990)
- [3] F.A. Bornemann: *An Adaptive Multilevel Approach to Parabolic Equations. II: Variable Order Time Discretization Based on Multiplicative Error Correction*. IMPACT Comput. Sci. Engrg. 3, p. 93–122 (1991)
- [4] F. Bornemann, P. Deuffhard: *Cascadic multigrid methods for elliptic problems*. Submitted to Numer. Math. (1994)
- [5] J.H. Bramble, J.E. Pasciak, J. Xu: *Convergence estimate for multigrid algorithms without regularity assumptions*. Mat. Comp. 57, p. 23–45 (1991)
- [6] P. Deuffhard: *Recent Progress in Extrapolation Methods for ODE's*. SIAM Review 27, p. 505–535 (1985)
- [7] P. Deuffhard: *Cascadic conjugate gradient methods for elliptic partial differential equations: algorithm and numerical results*. D.E Keyes, J. Xu (eds.): AMS, Contemporary Mathematics Vol. 180, p. 29–42 (1994)
- [8] P. Deuffhard, F. Bornemann: *Numerische Mathematik II. – Integration von gewöhnlichen Differentialgleichungen*. de Gruyter Lehrbuch, Berlin, New York (1991)
- [9] P. Deuffhard, U. Nowak: *Extrapolation Integrators for Quasilinear Implicit ODE's*. In: P. Deuffhard, B. Engquist (eds.): Large Scale Scientific Computing. Progress in Scientific Computing, Birkhaeuser 7, p. 37–50 (1987)
- [10] P. Deuffhard, P. Leinen, H. Yserentant: *Concepts of an Adaptive Hierarchical Finite Element Code*. IMPACT Comput. Sci. Engrg. 1, p. 3–35 (1989)
- [11] P. Deuffhard, U. Nowak, M. Wulkow: *Recent Developments in Chemical Computing*. Computers Chem. Engrg. 14, p. 1249–1258 (1990)
- [12] J.E. Flaherty, P.K. Moore, 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 , p. 176–193 (1989)
- [13] F.N. Fritsch, J. Butland: *A Method for Constructing Local Monotone Piecewise Cubic Interpolants*. SIAM J. Sci. Stat. Comput. 5, p. 300–304 (1984)

- [14] C.W. Gear: *Numerical Initial Value Problems in Ordinary Differential Equations*. Prentice Hall (1971)
- [15] E. Hairer, G. Wanner: *Solving Ordinary Differential Equations II. – Stiff and Differential–Algebraic Problems*. Springer Series in Computational Mathematics 14, Springer Verlag, Berlin–Heidelberg (1991)
- [16] J. Lang, A. Walter: *A finite element method adaptive in space and time for nonlinear reaction–diffusion systems*. IMPACT Comput. Sci. Engrg. 4, p. 269–314 (1992)
- [17] J. Lang: *High–resolution selfadaptive computations on chemical reaction–diffusion problems with internal boundaries*. Submitted to Chem. Engrg. Sci.
- [18] J. Lang: *Two–dimensional fully adaptive solutions of reaction–diffusion equations*. Accepted for publication in Appl. Numer. Math. (1995)
- [19] U. Maas, J. Warnatz: *Simulation of chemically reacting Flows in two–dimensional geometries*. IMPACT Comput. Sci. Engrg. 1, p. 394–420 (1989)
- [20] P.K. Moore, J.E. Flaherty: *High–order adaptive finite element–singly implicit Runge–Kutta methods for parabolic differential equations*. BIT 33, p. 309–331 (1993)
- [21] U. Nowak: *Adaptive Linienmethoden für nichtlineare parabolische Systeme in einer Raumdimension*. Technical Report TR 93–14, Konrad–Zuse–Zentrum Berlin (1993)
- [22] U. Nowak, J. Frauhammer, U. Nieken, G. Eigenberger: *A fully adaptive algorithm for parabolic partial differential equations in one space dimension*. accepted for publication in Comput. & Chem. Engrg. (1995)
- [23] A.C. Hindmarsh: *ODEPACK, A Systematized Collection of ODE Solvers*. In Scientific Computing, R. S. Stepleman et al. (eds.), North–Holland, Amsterdam, p. 55–64 (1983)
- [24] N. Peters, J. Warnatz (Eds): *Numerical Methods in Laminar Flame Propagation*. Notes in Numerical Fluid Dynamics 6, p. 232–260, Vieweg (1982)
- [25] L.R. Petzold: *A Description of DASSL: a differential–algebraic system solver*. Proc. IMACS World Congress, Montreal, Canada (1982)
- [26] L.R. Petzold: *An adaptive Moving Grid Method for One–Dimensional Systems of Partial Differential Equations and its Numerical Solution*. Proc. Workshop on Adaptive Methods for Partial Differential Equations, Renselaer Polytechnic Institute (1988)

- [27] M. Roche: *Rosenbrock methods for differential algebraic equations*. Numer. Mat. **52**, p. 45–63 (1988)
- [28] E. Rothe: *Zweidimensionale parabolische Randwertaufgaben als Grenzfall eindimensionaler Anwendungen*. Math. Ann. **102**, p. 650–670 (1930)
- [29] K. Strehmel, R. Weiner: *Linear-implizite Runge-Kutta-Methoden und ihre Anwendung*. Teubner-Texte zur Mathematik, Band 127, B.G. Teubner Verlagsgesellschaft, Stuttgart–Leipzig (1992)
- [30] H.A. van der Vorst: *BI-CGSTAB: A fast and smoothly converging variant of BI-CG for the solution of nonsymmetric linear systems*. SIAM J. Sci. Stat. Comp. **13**, No. 2, p. 631–644 (1992)
- [31] J. Xu: *Theory of Multilevel Methods*. Department of Mathematics, Pennsylvania State University, University Park, Report No. AM 48, Thesis (1989)
- [32] H. Yserentant: *On the multilevel splitting of finite element spaces*. Numer. Math. **49**, p. 379–412 (1986)
- [33] G.W. Zumbusch: *Simultaneous h-p Adaptation in Multilevel Finite Elements*, Freie Universität Berlin, Fachbereich Mathematik, Thesis (1995)