# Techniques for Locally Adaptive Time Stepping Developed over the Last Two Decades

Martin J. Gander[1], Laurence Halpern[2]

[1] University of Geneva, CH-1211, Geneva, Switzerland martin.gander@unige.ch
[2] LAGA - Institut Galilée, Université Paris 13 halpern@math.univ-paris13.fr

Adaptive mesh refinement techniques are well established and widely used for space discretizations. In contrast, local time stepping is much less used, and the corresponding techniques are less mature, needing delicate synchronization steps, which involve interpolation, extrapolation or projection. These operations can have adverse effects on the stability, and can also destroy important geometric properties of the scheme, like for example the conservation of invariants. We give here a survey on the intensive research performed in this direction over the last two decades.

## 1 Methods from the ODE Community

Local time stepping started in the ODE community with the development of split Runge-Kutta methods with Rice [34]. Nowadays called multirate Runge-Kutta methods, these methods were first developed for naturally split systems of ordinary differential equations $y' = b(y,z,t)$ and $z' = c(y,z,t)$, in which the $z$ components need to be integrated on a finer time mesh than the $y$ components. One then uses a Runge-Kutta method for the fast, so called active components with a small time step, and another one for the slow, so called latent components, with a large time step, and uses either interpolation or extrapolation for the missing values, depending on which of the components are computed first, see [27].

Multirate time integration methods were also proposed for linear multistep methods in [22], with two main approaches: fastest-first and slowest-first. Suppose an implicit linear multistep method is used. In the fastest-first approach, one advances the $z$ components with small time steps $h$, and whenever one needs a component of the slow part $y$, one uses a predictor step for it. Once the fine stepping scheme arrives at a coarse step $H$, the slow solution component $y$ is also computed. The major disadvantage of this approach is that it is very difficult to do adaptive time stepping. This is easier in the slowest-first approach, where first the slow component is doing an adaptive integration step, until one is accepted with step size $H$. Then the adaptive fine integration is tried with small steps $h$, until one reaches with several accepted small steps the coarse level $H$. For the slow adaptive step $H$ however, one needs also

R. Bank et al. (eds.), *Domain Decomposition Methods in Science and Engineering XX*,
Lecture Notes in Computational Science and Engineering 91,
DOI 10.1007/978-3-642-35275-1_44, © Springer-Verlag Berlin Heidelberg 2013

an approximation of the fast component for coupled components, and the authors in [22] say: "There are several possible ways to control the fast extrapolation error, none of which is entirely satisfactory". The stability properties of such multirate schemes were analyzed in [35] for Backward Euler multirate schemes; see also [23].

In contrast to the multirate methods, multirate extrapolation methods aim at integrating systems of ODEs without a priori knowledge of which components need finer time integration steps than others. A method based on Richardson extrapolation was proposed in [13]: one computes approximations for all components for a time step sequence $\{h_1, h_2, h_3, \ldots\}$, e.g. $h_2 = \frac{h_1}{2}$, $h_3 = \frac{h_1}{3}, \ldots$, and then builds the Richardson extrapolation table. As soon as a component has reached the desired accuracy at step $h_k$ (an error estimate is available automatically in the Richardson table), extrapolation for this component is marked inactive, and only components needing further accuracy continue the extrapolation. Inactive components must then however be approximated in order for the extrapolation to continue. Using interpolation from the continuous approximation obtained from the Richardson extrapolation can completely destroy the extrapolation process, which is based on the same error expansion for all the components. The authors in [13] propose instead an elegant approximation from the asymptotic expansion assumption itself, and also introduce a defect control to avoid that inactivation fails in certain situations.

## 2 Methods from the PDE Community

Local time stepping schemes in the PDE community started with experimental work, see for example [28]. Such ad hoc solutions were quite different for parabolic and hyperbolic PDEs.

**Hyperbolic Problems:** a first complete mathematical analysis of two space-time adaptive schemes for the wave equation $u_t = u_x$, an interpolation based variant, and the so called coarse mesh approximation method were given by Berger [2] (see also [3], and an early analysis for a different technique based on finite volumes in [31]). Using for example a three point explicit scheme, the interpolation based approach starts with a coarse step at the interface, shown in red in Fig. 1 on the left, followed by an interpolation for the fine grid values, shown in blue. In the coarse mesh approximation, one uses the coarse spatial mesh to compute small time steps $\Delta t$, $2\Delta t$, $3\Delta t, \ldots$ at the interface, instead of interpolating these values, as indicated in Fig. 1 on the right for the second step $2\Delta t$ in red, where the blue value at $\Delta t$ has already been

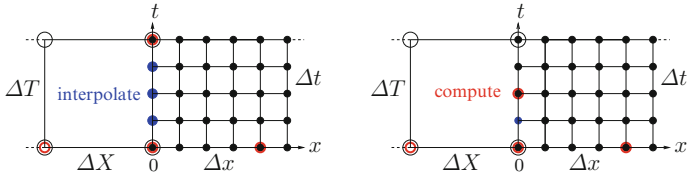


**Fig. 1.** Interpolation based approach on the *left*, and the coarse mesh method on the *right*

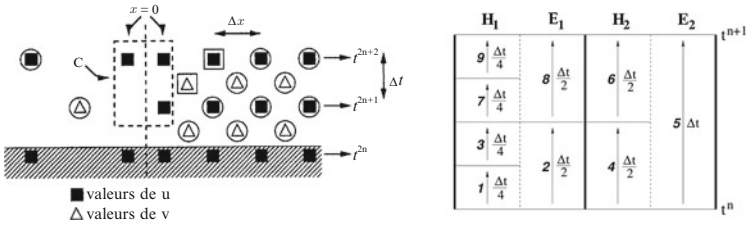


**Fig. 2.** First energy-preserving local time stepping for the wave equation on the *left*, and symplectic scheme for Maxwell's equation on the *right*

computed. The author proves for the hyperbolic model problem $u_t = u_x$ that both approaches are stable for the Lax-Wendroff scheme, but stability for the Leapfrog scheme can only be achieved with overlap. Elegant recursive versions of such algorithms are in [33].

A key new ingredient to obtain stability for a Leapfrog type scheme for the locally adaptive solution of the wave equation can be found in the seminal papers by Collino et al. in [7, 8]: the introduction of a discrete energy conservation. In presentations, this approach was always introduced with an impressive movie, where a wave passes a locally refined patch, and everything looks fine for quite a long time after the wave has passed, until suddenly an instability forms at the boundary of the patch, and the numerical solution explodes, if a simple interpolation based scheme is used. The method was first described for the 1d Maxwell system $u_t + v_x = 0$, $v_t + u_x = 0$, which is equivalent to the 1d second order wave equation $u_{tt} = u_{xx}$, and can best be described with the original picture from [7] shown in Fig. 2 on the left. Thinking just about the second order wave equation, discretized with a centered finite difference scheme both in space and time, we get the five point star, well visible with the black squares in Fig. 2 (the triangles would be for the unknowns $v$ we do not consider here). Now all points can be computed with this star at time levels $t^{2n+1}$ and $t^{2n+2}$, given the values at earlier time levels, except for the values in the dashed box. The key idea of the energy preserving scheme is now to permit two different values at $x = 0$ at even time levels $t^{2n}$, and to introduce as additional equation the discrete energy, which needs to be preserved. This leads naturally to a stable scheme, but it requires the solution of a small linear system at the interface. Energy conservation turned out to be a key tool for stability analysis, and is used now for other space-time adaptive methods, see for example [11], where the authors introduce an unusual energy, in order to analyze the stability of their space-time locally adaptive scheme.

A very elegant way of generalizing a symplectic integrator (which naturally preserves a nearby energy) for variable step size integration was presented in [26], and adapted to Maxwell's system in [32]. The Störmer-Verlet scheme is symplectic for these equations, and is shown in Fig. 2 on the right. Without refinement, the scheme is visible in the right part under $H_2, E_2$: we see that first a half step denoted by 4 is performed for the magnetic field $H$, followed by a full step denoted by 5 for the electric field $E$, and concluded by a second half step for $H$ denoted by 6. In each of these steps, the Störmer-Verlet scheme uses for $H$ the newest values available from

the other field $E$, and vice versa. It turns out that doing the same over the locally refined region shown in Fig. 2 on the right, and performing the steps in the given order, starting with 1 and ending with 9, and using each time the newest information available, is still symplectic! Since symplectic schemes preserve a nearby energy, this scheme has all the good stability properties needed.

In a finite volume or discontinuous Galerkin in the time domain setting (DGTD), on unstructured meshes in space, the scheme in each subdomain with given time step can be advanced until the new time value reaches that of its neighbor, according to the stability constraint, see [12] for elastodynamics computations in the context of ADER methods (Arbitrary high order, using high order DERivatives of polynomials).

**Parabolic Problems** are often integrated using implicit methods, which require the solution of large systems of equations. These systems are obtained using the same time step over the entire domain, and it is thus a priori not possible to use a local time step. The first ideas to change this are based on domain decomposition methods, where then interface values have to be predicted in some way, before the subdomain problems are advanced in time by an implicit method.

A first interesting way to explicitly predict the interface values appeared in [9], where a third spatial discretization size $H$ is introduced, in addition to $h_l$ and $h_r$, see Fig. 3 on the left. The method then first does an explicit prediction step over the big $\Delta t$, stable because the corresponding spatial step $H$ is big, as indicated in red. This is followed by interpolation (in blue) to obtain all needed values at the interface, and then on each side one can do implicit solves to advance the method. It is proved in [9] that this scheme is stable for the heat equation with a centered finite difference discretization in space, and forward/backward Euler in time, if $\Delta t \leq \frac{1}{2}H^2$, and the error satisfies the estimate $\max|\text{err}| \leq C(h_l^2 + h_r^2 + H^3 + \Delta t_l + \Delta t_r + H\Delta t)$, which shows impressively that the big prediction step $\Delta t$, $H$ only affects the accuracy in higher order terms!

A different approach was proposed by Blum et al. [4], as shown in Fig. 3 on the right. The authors do not consider local refinement in time and space, their main interest is to break up a large linear system from the implicit time integration into smaller ones, but their idea can also be used for local adaptation in time and space. The key idea is to use overlap, predict all values needed at the interfaces using a higher order extrapolation method, and then solve implicitly on the corresponding subdomains to advance the method. The authors prove for the heat equation without

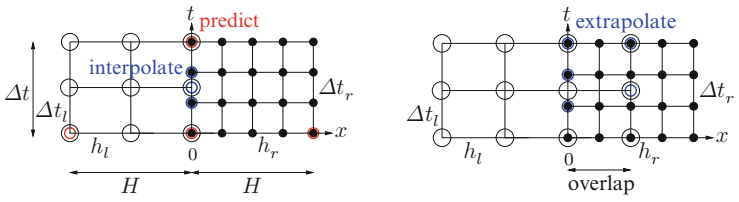


**Fig. 3.** Explicit prediction of the interface values on an intermediate spatial grid on the *left*, and by extrapolation with overlap on the *right*

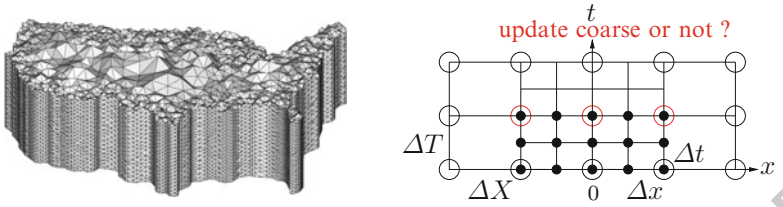


**Fig. 4.** A completely general space time mesh on the *left*, and the one-way and two way approaches on the *right*

local refinement, $h_l = h_r = h$ and $\Delta t_l = \Delta t_r = \Delta t$, that the Crank-Nicolson scheme is stable, provided that $\Delta t \leq C \left(\frac{L}{\log L}\right)^2 h^2$, where $Lh$ is the overlap, and an error estimate of the form $O(\Delta t^2 + h^2)$. So here increasing the overlap can lessen the stability constraint on the time step.

If one wants to avoid any time step constraints, one can perform the coupling fully implicitly, as proposed in [16]. Here, one simply writes the implicit scheme on the fine and coarse subdomain, and the interpolation conditions into one big system of linear equations, which is then solved. The authors show for a linear advection reaction diffusion equation that a standard centered scheme with backward Euler in time is unconditionally stable, and satisfies for $\Delta t = O(h)$ the error estimate $O(\Delta t + h^2)$ in 1d, but in 2d there is a loss of $|\log h|^{\frac{1}{2}}$, and in 3d a loss of $\frac{1}{\sqrt{h}}$ in accuracy.

A more general approach based on domain decomposition can be found in [17]. For the heat equation $u_t = u_{xx}$, and the decomposition of the domain $\Omega = (-1,1)$ into two subdomains $\Omega_1 = (-1,0)$ and $\Omega_2 = (0,1)$, the authors propose to discretize the coupling conditions $u_1(0) = u_2(0)$, $\partial_x u_1(0) = \partial_x u_2(0)$ using a conservative finite volume discretization over non-matching time grids. They also obtain, for each variant of the method, a very large system of equations to solve, but propose to solve it using one or several steps of an iterative Dirichlet-Neumann algorithm. They show that these schemes are conservative, provided one stops the iteration after a Neumann step, and satisfy an error estimate $O(\Delta t + h)$ under certain conditions. One can show that one of their methods corresponds to the approach in [16].

**Space-Time Finite Element Methods** consider the time direction like one of the spatial directions, and discretize the problem directly in space-time by a finite element method, which leads to a large discrete problem in space-time. These methods have their roots in the work of C. Johnson and co-authors, see for instance [15] for a review. Discontinuous Galerkin methods were used, and the adaptation was done through a posteriori estimates. In the first versions of the method, the space-time finite elements were still special, since they always had boundaries in time aligned with the time direction, for example prisms. Completely general triangular meshes in space time require special meshing techniques, since they need to satisfy certain angle constraints, in order to avoid total global coupling in space-time, see [36] for applications to Burger's equation and elastodynamics. An impressive example of such a mesh from [14] is reproduced in Fig. 4 on the left. A very recent contribution

using discontinuous Galerkin methods can be found in these proceedings, see [30].

**One-Way and Two-Way Methods** are in principle very different from all the methods we considered earlier, since they have both a coarse *and* a fine mesh in parts of the domain. They have their roots in weather and climate simulations, which often use a global model over a large region, for example the entire planet, and then refined models over a small region, for example a country. The question is then how to compute a refined solution based on the solution of the global coarse problem. In [10] and [6], the so called one-way (or "offline") and two-way (or "online") methods are proposed. In the one-way method, the coarse model is first solved once and for all, and stored. Then boundary data is extracted to be imposed on the boundary of the smaller refined region. The simplest approach is to use Dirichlet conditions, which can however lead to large errors. A more refined approach is to use so called open boundary conditions, which are related to absorbing boundary conditions, but different, see [6, 29]. Open boundary conditions lead in general to substantially more accurate fine models. In the two way approach, one only performs one or a few time steps of the coarse model, then solves the fine model in the refined region as before, but updates the coarse result whenever a more accurate fine result is available, before continuing the next coarse time step, see Fig. 4 on the right. If one simulates only one time step of the coarse model before solving the fine model and uses Dirichlet conditions, this approach is very much related to the first approach for hyperbolic problems described earlier.

**Schwarz waveform relaxation methods** are the most flexible methods for solving evolution problems locally adaptively in space time, since they permit not only refined time steps, but even different numerical methods, or different models in different regions. They were first described in [20] and are based on a decomposition in space of the domain over which the evolution problem is posed and a subdomain iteration in space-time: starting with an initial guess on each space-time interface between subdomains, on each subdomain the evolution problem is solved over an entire so called time window. Then information is exchanged between subdomains using transmission conditions, and the subdomain problems are solved again and again until a suitable matching is reached. So the price to pay for this flexibility and generality is the iteration. The method from [17] we have seen earlier is in this class of methods, but much faster convergence can be obtained when optimized transmission conditions are used, see [1, 18, 21, 24, 25], and references therein. Very general non-matching space-time grids can be coupled like this using a projection algorithm with optimal linear complexity from [19]. For recent realistic applications in a complex setting, see [5].

## Bibliography

[1] D. Bennequin, M. J. Gander, and L. Halpern. A homographic best approximation problem with application to optimized Schwarz waveform relaxation. *Math. of Comp.*, 78(265):185–232, 2009.

[2] M. Berger. Stability of interfaces with mesh refinement. *Math. of Comp.*, 45: 301–318, 1985.

[3] M. J. Berger and J. Oliger. Adaptive mesh refinement for hyperbolic partial differential equations. *J. Comp. Phys.*, 53:484–512, 1984.

[4] H. Blum, S. Lisky, and R. Rannacher. A domain splitting algorithm for parabolic problems. *Computing*, 49:11–23, 1992.

[5] M. Borrel, L. Halpern, and J. Ryan. Euler - Navier-Stokes coupling for aeroacoustics problems. In A. Kuzmin, editor, *Computational Fluid Dynamics 2010, ICCFD6*, pages 427–434. Springer-Verlag, 2010.

[6] S. Cailleau, V. Fedorenko, B. Barnier, E. Blayo, and L. Debreu. Comparison of different numerical methods used to handle the open boundary of a regional ocean circulation model of the bay of biscay. *Ocean Modelling*, 25:1–16, 2008.

[7] F. Collino, T. Fouquet, and P. Joly. A conservative space-time mesh refinement method for the 1d wave equation. part I: construction. *Numer. Math.*, 95:197–221, 2003.

[8] F. Collino, T. Fouquet, and P. Joly. A conservative space-time mesh refinement method for the 1d wave equation. part II: analysis. *Numer. Math.*, 95:223–251, 2003.

[9] C. Dawson, Q. Du, and T. Dupont. A finite difference domain decomposition algorithm for numerical solution of the heat equation. *Math. Comp.*, 57(195): 63–71, 1991.

[10] L. Debreu and E. Blayo. Two-way embedding algorithms: a review. *Ocean Dynamics*, 58:415–428, 2008.

[11] J. Diaz and M. J. Grote. Energy conserving explicit local time-stepping for second-order wave equations. *SIAM J. Scientific Computing*, 31:1985–2014, 2009.

[12] M. Dumbser, M. Käser, and E.F. Toro. An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes V: Local time stepping and p-adaptivity. *Geophysical Journal International*, 171(2):695–717, 2007.

[13] Ch. Engstler and Ch. Lubich. Multirate extrapolation methods for differential equations with different time scales. *Computing*, 58:173–185, 1997.

[14] J. Erickson, D. Guoy, J.M. Sullivan, and A. Üngör. Building spacetime meshes over arbitrary spatial domains. *Eng. with Comp.*, 20:342–353, 2005.

[15] K. Eriksson, C. Johnson, and A. Logg. Adaptive computational methods for parabolic problems. In *Encyclopedia of Computational Mechanics*, 2004.

[16] R. E. Ewing, R. D. Lazarov, and A. T. Vassilev. Finite difference scheme for parabolic problems on composite grids with refinement in time and space. *SIAM J. Numer. Anal.*, 31:1605–1622, 1994.

[17] I. Faille, F. Nataf, F. Willien, and S. Wolf. Two local time stepping schemes for parabolic problems. *ESAIM: proceedings*, 29:58–72, 2009.

[18] M. J. Gander and L. Halpern. Optimized Schwarz waveform relaxation methods for advection reaction diffusion problems. *SIAM J. Numer. Anal.*, 45(2): 666–697, 2007.

[19] M. J. Gander and C. Japhet. An algorithm for non-matching grid projections with linear complexity. In M. Bercovier, M.J. Gander, D. Keyes, and O.B. Widlund, editors, *Domain Decomposition Methods in Science and Engineering XVIII*, pages 185–192. Springer Verlag LNCSE, 2008.

[20] M. J. Gander and A. M. Stuart. Space-time continuous analysis of waveform relaxation for the heat equation. *SIAM J. Sci. Comput.*, 19(6):2014–2031, 1998.

[21] M. J. Gander, L. Halpern, and F. Nataf. Optimal Schwarz waveform relaxation for the one dimensional wave equation. *SIAM Journal of Numerical Analysis*, 41(5):1643–1681, 2003.

[22] C.W. Gear and D.R. Wells. Multirate linear multistep methods. *BIT*, 24:484–502, 1984.

[23] M. Günter, A. Kværnø, and P. Rentrop. Multirate partitioned Runge-Kutta methods. *BIT*, 38(2):101–112, 1998.

[24] L. Halpern. Non conforming space-time grids for the wave equation: a new approach. *Monografías del Seminario Matemático García de Galdeano*, 31: 479–495, 2004.

[25] L. Halpern. Local space-time refinement for the one dimensional wave equation. *J. of Comp. Acoustics*, 13(3):153–176, 2005.

[26] D.J. Hardy, D.I. Okunbor, and R.D. Skeel. Symplectic variable step size integration for N-body problems. *Appl. Numer. Math.*, 29(5):19–30, 1999.

[27] A. Kværno and P. Rentrop. Low order multirate Runge-Kutta methods in electric circuit simulation, 1999.

[28] R. Löhner, K. Morgan, and O. C. Zienkiewicz. The use of domain splitting with an explicit hyperbolic solver. *Computer Methods in Applied Mechanics and Engineering*, 45:313–329, 1984.

[29] V. Martin and E. Blayo. Revisiting the open boundary problem in computational fluid dynamics. In R. Bank, M. Holst, O.B. Widlund, and J. Xu, editors, *Domain Decomposition Methods in Science and Engineering XX*. Springer-Verlag, 2012.

[30] M. Neumüller. Space-time DG methods. In R. Bank, M. Holst, O.B. Widlund, and J. Xu, editors, *Domain Decomposition Methods in Science and Engineering XX*. Springer Verlag, 2012.

[31] S. Osher and R. Sanders. Numerical approximations to nonlinear conservation laws with locally varying time and space grids. *Math. of Comp.*, 41(164):321–336, 1983.

[32] S. Piperno. Symplectic local time-stepping in non-dissipative DGTD methods applied to wave propagation problems. *ESAIM: Mathematical Modelling and Numerical Analysis*, 40(5):815–841, 2006.

[33] F. Pretorius and L. Lehner. Adaptive mesh refinement for characteristic codes. *J. Comp. Phys.*, 198:10–34, 2004.

[34] R. C. Rice. Split Runge-Kutta methods for simulatneous equations. *J. Res. Natl. Bur. Standards*, 64B:151–170, 1960.

[35] S. Skelboe and P. U. Andersen. Stability properties of backward Euler multirate formulas. *SIAM J. Sci. Stat. Comp.*, 10:1000–1009, 1989.

[36] A. Üngör and A. Sheffer. Tent-pitcher: A meshing algorithm for space-time discontinuous galerkin methods. In *In proc. 9th int'l. meshing roundtable*, pages 111–122, 2000.