Proceedings of ALGORITMY 2002
Conference on Scientific Computing, pp. 94–101

EFFICIENT SIMULATION OF MULTI-DIMENSIONAL DETONATION PHENOMENA*

RALF DEITERDING†

Abstract. The paper presents a solution strategy for Euler equations for multiple thermally perfect gaseous species with detailed chemical reaction. Via operator-splitting a high-resolution finite-volume scheme and a stiff ODE solver are coupled. A parallel blockstructured adaptive mesh refinement algorithm is utilized to achieve the required local resolution. A highly resolved computation of regular detonation-cell patterns of the hydrogen-oxygen-argon system demonstrates the efficiency of the entire approach.

Key words. unstable detonations, numerical methods, adaptive mesh refinement, parallelization

AMS subject classifications. 80A32, 65M50, 68W10

1. Introduction. The incorporation of reactive source terms into hydrodynamic gaseous flow introduces a wide range of new non-neglectable temporal and spatial scales. In Fig. 1.1 a typical self-sustaining Chapman-Jouguet detonation wave is displayed. It travels at a constant super-sonic speed of 1627 m/s, but 99 % of the fuel is burned 13 μs after being ignited by the leading shock wave. Fig. 1.1 visualizes the characteristic substructure of a detonation with a hydrodynamic shock followed by a region of decaying continuous burning. The shock initiates the reaction, but it takes an induction time of about 0.6 μs until the reactants start to burn rapidly to the constant equilibrium state. Therefore shock and flame in fact are separated by a distance of $\approx$ 1.4 mm and numerical simulations usually do hard to resolve this induction zone sufficiently. Especially multi-dimensional simulations are very challenging and require integrated approaches that combine accurate discretizations with sophisticated technics from computer science.

In this paper we present an efficient solution strategy for the Euler equations for mixtures of thermally perfect gases with detailed reactive source terms. Hydrodynamic transport and chemical reaction are treated successively with an operator-splitting approach. The description of the physical model in Section 2 is followed by the presentation of the different numerical schemes in Section 3. A parallel variant of the blockstructured adaptive mesh refinement algorithm by Berger and Oliger (AMR) is described in Section 4. The choice of a robust high-resolution upwind scheme, properly adjusted adaption criteria and an efficient parallelization strategy tailored for AMR allow simulations with remarkable high resolution. As a large-scale example a simulation of regular detonation-cell patterns of the H_2:O_2:Ar-system in two space dimensions is presented in Section 5. The results are of highest quality and can serve as a reference. Analogous simulations in three space dimensions are currently carried out by the author, but are not included into this paper. The results will be presented elsewhere.

Successful simulations on uniform cartesian meshes of the example of Section 5 with detailed chemical reaction were presented for the first time by E. S. Oran et at in [9]. To reduce the extraordinary costs of uniform computations T. Geßner applied an unstructured adaption strategy [5]. C. A. Eckett employed a serial version of the blockstructured AMR-algorithm of Berger and Oliger, but used only significantly reduced chemical reaction models [4].

*This work was supported by DFG Grant No. Ba 840/3-3

†Institut für Mathematik, Technische Universität Cottbus, Postfach 10 13 44, 03013 Cottbus, Germany, (deiterding@math.tu-cottbus.de).

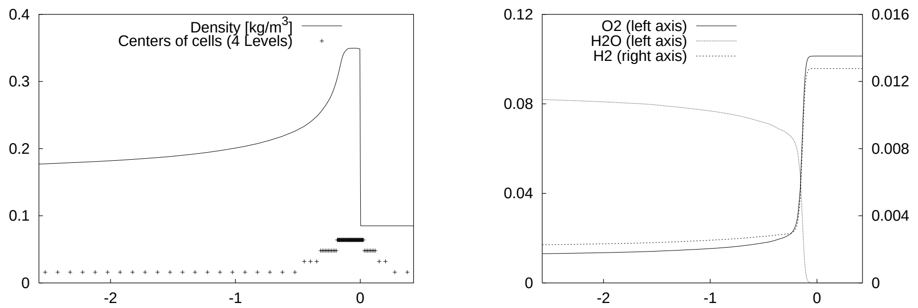


FIGURE 1.1. *Chapman-Jouguet detonation for a stoichiometric* H_2:O_2*-mixture (molar ratio* $2:1$*) diluted with* $70\,\%$ Ar *at* 298 K. *At the* x*-axis the distance in the downstream direction in* cm *to the detonation front is displayed. Left: Density distribution and dynamically adapted computational grid utilizing the adaption criteria of Section 5.3. Right: Mass fractions* Y_i *of reactants* H_2, O_2 *and of main reaction product* H_2O.

2. Governing equations. We consider the Euler equations in cartesian coordinates in two space dimensions with chemical reactive source terms [13].

2.1. Generalized Euler equations. Conservation of mass of K different gaseous species, conservation of momentum and of total energy gives

$$\begin{array}{lclclcll} \partial_t \rho_i & + & \partial_x(\rho_i u) & + & \partial_y(\rho_i v) & = & W_i\,\dot\omega_i & i=1,\dots,K \\ \partial_t(\rho u) & + & \partial_x(\rho u^2+p) & + & \partial_y(\rho u v) & = & 0 & \\ \partial_t(\rho v) & + & \partial_x(\rho u v) & + & \partial_y(\rho v^2+p) & = & 0 & \\ \partial_t(\rho E) & + & \partial_x\left[u(\rho E+p)\right] & + & \partial_y\left[v(\rho E+p)\right] & = & 0 & . \end{array} \tag{2.1}$$

For the total density $\rho = \sum \rho_i$ holds, where ρ_i are the partial densities. The ratios $Y_i = \rho_i/\rho$ are called the mass fractions. Note, that $\sum Y_i = 1$ holds true. Chemical production rate and molecular weight of each species are denoted by $\dot\omega_i$ and by W_i, respectively. The velocities in x- and y-direction are u and v, p is the total hydrodynamic pressure and E is the total energy per unit mass.

Under the assumption of thermal equilibrium[1] the total pressure p is given by Dalton's law as $p = \sum p_i = \mathcal{R}T\sum \rho_i/W_i$. Each species is assumed to be thermally perfect with a temperature-dependent specific heat $c_{pi}(T)$.[2] The enthalpies per unit mass are written as

$$h_i(T) = h_i^0 + \int_0^T c_{pi}(s)ds \quad .$$

For the enthalpy of the mixture $h(T) = \sum Y_i\; h_i(T)$ holds. Inserting this into the thermodynamic relation $\rho h - p - \rho e = 0$ and applying the above equation for p reads

$$\sum_{i=1}^{K} \rho_i\, h_i(T) - \mathcal{R}T \sum_{i=1}^{K} \frac{\rho_i}{W_i} - \rho E + \frac{\rho}{2}\left(u^2+v^2\right) = 0 \quad . \tag{2.2}$$

Computation of the temperature T from the conserved quantities utilizing the implicit relation (2.2) is unavoidable whenever the pressure p has to be evaluated. Like in the standard case of a single polytropic gas the local *frozen* speed of sound is given by $c^2 = \gamma p/\rho$. It may be calculated by applying the basic relations $Y_i = \rho_i/\rho$, $c_p = \sum Y_i\, c_{pi}$, $W = (\sum Y_i/W_i)^{-1}$ and $\gamma = \dfrac{c_p}{c_p - \mathcal{R}/W}$.

[1]The same temperature T can be used for all K species.

[2]The functions $c_{pi}(T)$ are usually approximated by polynoms of degree 4 and can be taken from thermodynamic data bases.

2.2. Reaction mechanisms. The chemical production rates $\dot\omega_i(\rho_1,\ldots,\rho_K,T)$ are derived from a reaction mechanism that consists of M chemical reactions

$$\sum_{i=1}^{K} \nu_{ji}^f S_i \rightleftharpoons \sum_{i=1}^{K} \nu_{ji}^r S_i \qquad\qquad j=1,\ldots,M \quad ,$$

where ν_{ji}^f and ν_{ji}^r are the stoichiometric coefficients of species S_i appearing as a reactant and as a product. The forward reaction rate $k_j^f(T)$ of each reaction is calculated by the Arrhenius law

$$k_j^f(T) = A_j T^{\beta_j} \exp(-E_j/\mathcal{R}T) \tag{2.3}$$

and the corresponding backward reaction rate may be derived from the equilibrium constant $K_j^c(T)$ by $k_j^r(T) = k_j^f(T)/K_j^c(T)$ [13]. The production rate of species S_i is now given by

$$\dot\omega_i = \sum_{j=1}^{M}(\nu_{ji}^r - \nu_{ji}^f)\left[k_j^f \prod_{n=1}^{K}\left(\frac{\rho_n}{W_n}\right)^{\nu_{jn}^f} - k_j^r \prod_{n=1}^{K}\left(\frac{\rho_n}{W_n}\right)^{\nu_{jn}^r}\right] \quad i=1,\ldots,K \quad . \tag{2.4}$$

A chemical kinetics package (e.g. Chemkin) is usually utilized to compute (2.3)-(2.4) according to the particular reaction mechanism and given thermodynamic data.

3. Numerical methods. The generalized Euler equations (2.1) define a system of conservation laws of the form $q_t + f(q)_x + g(q)_y = \psi(q)$. We apply fractional step methods for the solution of the multi-dimensional homogeneous transport equations and for incorporation of reactive source terms. The one-dimensional transport equations

$$\begin{aligned} q_t + f(q)_x &= 0\ , \qquad \text{IC: } Q^n \quad \stackrel{\Delta t}{\Longrightarrow} \quad \tilde Q^{1/2} \\ q_t + g(q)_y &= 0\ , \qquad \text{IC: } \tilde Q^{1/2} \quad \stackrel{\Delta t}{\Longrightarrow} \quad \tilde Q \end{aligned} \tag{3.1}$$

and the usually stiff system of ordinary differential equations

$$q_t = \psi(q)\ , \quad \text{IC: } \tilde Q \stackrel{\Delta t}{\Longrightarrow} Q^{n+1} \tag{3.2}$$

are solved successively. Intermediate steps use the result from the preceding step as initial condition (IC).

The advantage of this successive application of splitting methods is that the entire method only requires two analogue one-dimensional transport schemes and a standard stiff ODE solver. The extension to three dimensions is obvious. Although the method (3.1)-(3.2) is formally only first-order accurate it usually gives very satisfactory results, if high-resolution upwind schemes are employed. Second-order accurate extensions with additional intermediate steps are possible [11], but they usually lead to similar results in most practical cases. Due to the analogy of the update steps in the coordinate directions the following description of van Leer's upwind scheme is restricted to the first step of (3.1).

3.1. Van Leer scheme. A standard conservative finite-volume discretization

$$Q_j^{n+1} = Q_j^n - \Delta t/\Delta x\left(F(Q_{j+1}^n, Q_j^n) - F(Q_j^n, Q_{j-1}^n)\right)$$

is utilized to integrate the first equation of (3.1). A straight-forward calculation shows that the eigenvalues of the Jacobians of the flux function $f(q)$ are $u-c$, u and $u+c$. These are the same eigenvalues as in the standard case of a single polytropic gas. Therefore the homogeneous transport equations are hyperbolic. Various upwind schemes that have been originally developed for standard Euler equations have been extended successfully to the generalized case of thermally perfect gas-mixtures [8].

In our numerical investigations we have found that especially van Leer's flux vector splitting combined with second-order MUSCL reconstruction is a good choice. It is comparably robust, easy to implement and avoids negative partial densities ρ_i by definition [7].

The flux vector splitting approach uses $F(Q_{j+1}, Q_j) = F(Q_j)^+ + F(Q_{j+1})^-$ to approximate the inter-cell flux. In case of van Leer's scheme for mixtures of thermally perfect gases $F(Q)^\pm$ is calculated as follows [10, 8]:

$$F(Q)^\pm = \pm\frac{\rho}{4c}(u \pm c)^2 \begin{bmatrix} Y_i \\ u - (u \mp 2c)/\gamma \\ v \\ H - \zeta(u \mp c)^2 \end{bmatrix} \qquad \begin{array}{l} H = h + \dfrac{1}{2}(u^2 + v^2) \\[2ex] \zeta = \dfrac{h/c^2}{1 + 2h/c^2} \end{array} \qquad \text{if } |u| < c$$

$$F(Q)^+ = F(Q),\ F(Q)^- = 0 \ \text{ if } u \ge c \ \text{ and } \ F(Q)^- = F(Q),\ F(Q)^+ = 0 \ \text{ if } u \le -c$$

Note that the stability condition of this scheme is

$$C_{cfl} \equiv \frac{\Delta t}{\Delta x}(|u| + c) \le \begin{cases} \dfrac{2\gamma + u(3-\gamma)/c}{\gamma + 3} & \text{if } |u| < c \\ 1 & \text{otherwise} \end{cases} .$$

We apply the MUSCL-Hancock variable reconstruction technique (see [11] for a detailed description) to the *primitive* variables Y_i, ρ, u, v, p to achieve second-order accuracy in smooth solution regions.

3.2. ODE integration. After each transport step the ODE system

$$\partial_t \rho_i = W_i \, \dot\omega_i(\rho_1, \ldots, \rho_K, T) \qquad i = 1, \ldots, K$$

is integrated within each grid cell to incorporate the reactive source terms. ODE systems that arise in chemical kinetics are usually stiff and we employ a semi-implicit Rosenbrock-Wanner method by Kaps and Rentrop of fourth order with automatic step-size adjustment [6]. Note that ρ, e, u, v are assumed to be constant during ODE integration and in each grid cell the chemical kinetics are computed like in the zero-dimensional constant volume adiabatic case.

3.3. Computation of the temperature. Whenever the temperature T has to be calculated from the conserved variables the implicit equation (2.2) is solved utilizing Newton's method. The iteration is initialized with the temperature value of the preceding time step.[3]

4. Adaptive mesh refinement. In order to achieve the required temporal and spatial resolution the blockstructured adaptive mesh refinement algorithm of Berger and Oliger is applied [2]. We have implemented this algorithm in a generic, dimension-independent object-oriented framework in C++. It is called AMROC (Adaptive Mesh Refinement in object-oriented C++) and is free of charge for scientific use. An efficient parallelization strategy especially tailored for distributed memory machines has been found and the codes can be executed on all high-performance computers that provide the MPI-library.

[3] Note that the solution of (2.2) is only unique within the valid temperature-range of the approximating polynoms for $c_{pi}(T)$.

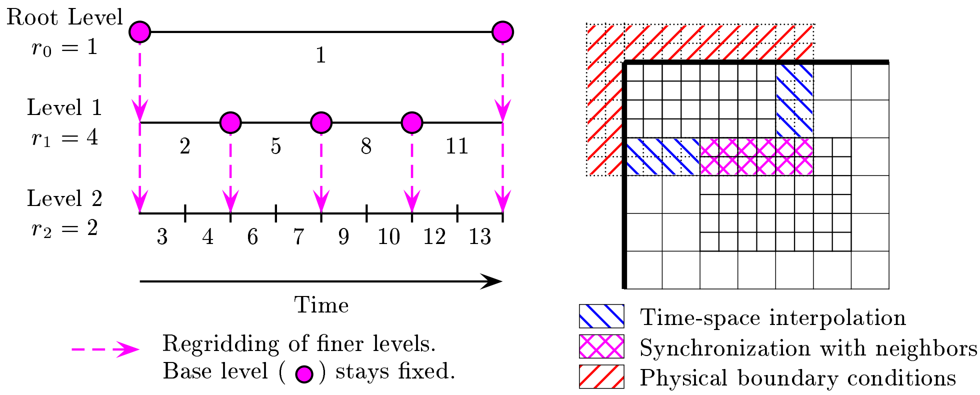


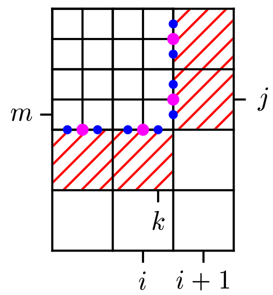
FIGURE 4.1. *Left: Recursive integration order of AMR. Right: Sources of ghost cell values.*

The adaptive algorithm has been realized completely decoupled from a particular finite-volume method. All what the algorithm requires is an equation-specific time-explicit scheme for hyperbolic conservation laws for a single rectangular grid. For the example of Section 5 this single-grid scheme is implemented exactly as described in Section 3.

4.1. Berger-Oliger AMR-method. Instead of replacing single cells by finer ones the AMR-method follows a patch-wise refinement strategy. Cells being flagged by various criteria, e.g. error estimators, are clustered into rectangular boxes of appropriate size. They describe refinement regions geometrically and subgrids with refined mesh spacing in space and time are generated according to them. Refined grids are derived recursively from coarser ones and an entire multi-level hierarchy of successively embedded grid patches is therefore constructed. All grid patches are logically rectangular and the adaptive algorithm calls the equation-dependent single-grid routine during a loop over all subgrids of the entire level (see function IntegrateLevel(l) in Fig. 4.2).

It is important to note, that refined grids overlay the coarser subgrids from which they have been derived. Values of cells covered by refined subgrids are overwritten by averaged fine grid values subsequently. The resulting extra work is usually negligible compared to the computational costs for integrating the superimposed refinement grids.

Replacing coarse cell values by averaged fine grid values modifies the numerical stencil on the coarse grid. In general the important property of conservation is lost. A flux correction replacing the coarse grid flux at the affected side of a neighboring cell by accumulated fine grid fluxes is necessary to ensure conservation. In two and three space dimensions hanging nodes additionally have to be considered.



Example: Flux correction in cell $i+1, j$ of level l

$$Q_{i+1,j}^{n+1} := Q_{i+1,j}^{n+1} - \frac{\Delta t_l}{\Delta x_l} \left(F_{i+1/2,j}^{l} - \frac{1}{\Delta y_l \, r_{l+1}} \sum_{\{\Delta t_{l+1}\}}^{\Delta t_l} \sum_{p=0}^{r_{l+1}-1} F_{k+1/2,m+p}^{l+1} \, \Delta y_{l+1} \right)$$

```
IntegrateLevel(l) - Integrate all grids on level l
  Repeat r_l times
      Set ghost cells of level l at time t
      If (time to regrid?) Then
          Regrid(l)
      Step Δt_l on all grids at level l
      If (level l+1 exists?) Then
          Set level l ghost cells at t + Δt_l
          IntegrateLevel(l+1)
          Average level l+1 grids onto level l

Regrid(l_fix) - Regrid all levels l > l_fix
  For l=l_max Downto l_fix Do
      Flag cells for refinement on level l
      Find proper new grids on level l
  Reorganize whole hierarchical data

Start - Start integration on root level
  l = 0, r_0 = 1
  IntegrateLevel(l)
```

FIGURE 4.2. *Pseudo-code of the AMR-algorithm. Refinement factor on level l is $r_l = \Delta t_{l-1}/\Delta t_l$.*

The application of refined time-steps on finer subgrids as displayed in Fig. 4.1 is an important difference of the AMR approach in comparison to usual unstructured adaptive strategies. By applying the same refinement factor in time as in space C_{cfl} in principle remains unchanged on refined subgrids.

4.2. Parallelization. A natural parallelization strategy of the AMR-method relies on the use of auxiliary (ghost) cells around a subgrid for the setting of boundary conditions. The use of ghost cells allows a successive treatment of all kind of boundaries and eliminates the handling of boundary conditions from the single-grid solution routine. The ghost cells of all subgrids on level l are set in the following order (see right picture of Fig. 4.1):

1. Time-space interpolation at internal boundaries from level $l-1$.
2. Synchronization with neighboring subgrids of level l transparently even over processor borders.
3. Application of physical boundary conditions.

Distribution of subgrids to processors is carried out under the restriction that higher level data must reside on the same computing node as the coarsest level data. The coarsest level defines the "floor-plan" of the entire hierarchy. This distribution strategy ensures that almost all computational operations of the parallel AMR-algorithm, like time-space interpolation and averaging onto the coarser levels, remain strictly local.

A further advantage of this approach is that partitioning based on the accumulated workload of all levels only has to take place on cells of the coarsest level. Note that the partitioning algorithm has to be extraordinarily fast, because it is executed at the end of each Regrid(l)-operation whenever the hierarchy has to be reorganized in a load-balanced manner (see Fig. 4.2). For the following computational example a generalization of the Peano-Hilbert space-filling curve has been used as an efficient partitioner [1].

5. Simulation of a planar detonation with transverse waves. Experiments have shown that self-sustaining detonation waves are locally multi-dimensional and nonsteady. Triple-points may form, which enhance the local chemical reaction significantly. Equilibrium-configurations with regular detonation cells have been observed in particular cases. We employ a relatively simple hydrogen-oxygen mechanism from [12]. It consists of 34 elementary reactions for the 9 species H, O, OH, H_2, O_2, H_2O, HO_2, H_2O_2, Ar.

5.1. Initial and boundary conditions. The one-dimensional Chapman-Jouguet detonation of Section 1 is calculated with very high accuracy and is extended to a pla-

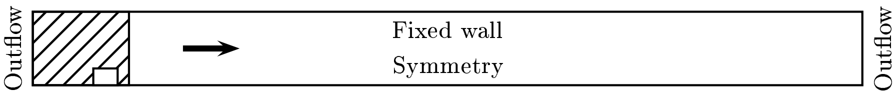


FIGURE 5.1. *The computational domain has the size* 62 cm × 3 cm. *The leading shock of the CJ-detonation is placed at* $x = 4.6$ cm. *The center of the unreacted pocket is at* $x = 3.8$ cm.

FIGURE 5.2. *Time history of released chemical energy at* $t = 260\,\mu s$. *Regular detonation-cells with a width of* ≈ 5.3 cm *are visible.*

nar detonation in the x-direction on a two-dimensional grid.[4] Computational domain and boundary conditions are shown in Fig. 5.1. Following [9] transverse disturbances are initiated by placing an unreacted pocket of 10 mm × 7 mm at 2086 K at a distance of 3 mm behind the detonation front. The computation is stopped at $t_{end} = 350\,\mu$s when the detonation approaches the right boundary.

5.2. Physical explanation. A shock emanating from the burning of the unreacted pocket behind the detonation initiates a triple-point when it hits the heading shock front. The triple-point heats the unreacted gas to a temperature about 500 K higher than the incident shock of the CJ-detonation. The induction time is significantly shortened and reduces the distance between flame (or fire) and leading shock (see Fig. 5.3). The local energy release near the triple-point is considerably higher than behind the undisturbed incident shock. The accumulated chemical energy release over the whole simulation time can therefore be utilized in Fig. 5.2 to visualize all triple-point movements.[5] The triple-point takes energy from the detonation wave and reduces its propagation velocity. It disturbs the perfect balance between released chemical and kinetic energy. After about 100 μs the increasing mismatch leads to the formation of new triple-point. With this additional triple-point our particular configuration quickly reaches a nonsteady equilibrium configuration with regular detonation-cells.

5.3. Adaptive simulation. A grid of 620 × 20 cells is selected as the base grid. 1044 time-steps on this base grid are calculated to reach $t_{end} = 350\,\mu$s. Three refinement levels with refinement factors 2, 4 and 4 are employed. This corresponds to a resolution of ≈ 44 cells within the induction length of the CJ-detonation. A similar uniformly refined simulation would require 19840 × 640 = 12.700.000 cells. Note that the transverse waves would fade away quickly, if no sufficient resolution in the induction zone would be supplied.

While an appropriate adaption to the discontinuous shock waves can easily be achieved, for instance by scaled gradients $|Q^n_{i+1} - Q^n_i|$ of pressure and density, an efficient adaption within the smooth flow regions with significant chemical reaction is harder. We have successfully applied a criterium that estimates the error of the mass fractions Y_i heuristically. It is based on Richardson extrapolation and is implemented with the aid of a coarser mesh. The actual solution Q^n is advanced one time-step

[4]A detonation in a combustible mixture may be initiated by a spark that can easily be modeled by a heat source in the energy equation or by a sufficiently strong shock wave [3]. Every detonation of real chemistry will invariably converge against the Chapman-Jouguet limit, which is determined by the state of the unburned gas [13].

[5]The released chemical energy ϵ is calculated cell-wise as the sum of enthalpy changes during the entire computation as $\epsilon = \sum_{\{\Delta t\}} (\rho h)^{n-1} - (\rho h)^n$.

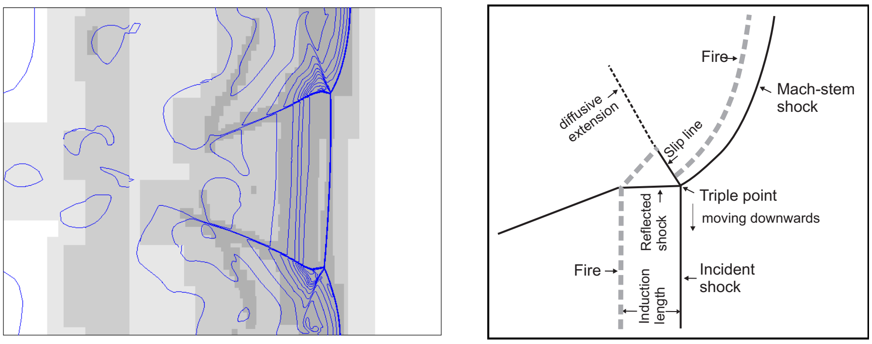


FIGURE 5.3. *Left: Isolines of density on refinement grids show the dynamic adaption of the detonation wave ($t = 260\,\mu s$). Right: Schematic diagram of the flow around the upper triple-point.*

utilizing the method (3.1)-(3.2). It is compared with an intermediate solution of (3.1)-(3.2) that has been computed from averaged data of the preceding step Q^{n-1} on an auxiliary mesh coarsened by a factor of 2. After computing

$$Q^n \overset{\Delta t,\Delta x}{\Longrightarrow} \tilde{Q} \quad \text{and} \quad Q^{n-1} \overset{2\Delta t,2\Delta x}{\Longrightarrow} \hat{Q} \qquad\qquad \tau = \frac{|\tilde{Q}-\hat{Q}|}{2^{d+1}-2}$$

is applied to estimate the absolute error τ. d denotes the order of the entire scheme.

We have adjusted these two different types of adaption criteria very thoroughly to the specific problem and have been able to reduce the number of cells in the entire hierarchy to a value between 150.000 and 200.000. Nevertheless, the calculation needed 121 h real time on 7 computing nodes of a typical PC-cluster with Pentium III-750 MHz processors.

REFERENCES

[1] G. Bader and R. Deiterding. A distributed memory adaptive mesh refinement package for inviscid flow simulations. In P. Jonas and V. Uruba, editors, *Proc. of Colloquium on Fluid Dynamics*, pages 9–14. Institute of Thermodynamics, Academy of Sciences of Czech Republic, October 19-20 1999.
[2] M. Berger and J. Oliger. Adaptive mesh refinement for hyperbolic partial differential equations. *J. Comput. Phys.*, 53:484–512, 1984.
[3] R. Deiterding. Simulation of a shock tube experiment with non-equilibirum chemistry. Technical Report NMWR-00-3, Brandenburgische Technische Universität Cottbus, Oct 2000.
[4] C. A. Ecket. *Numerical and analytical studies of the dynamics of gaseous detonations.* PhD thesis, California Institute of Technology, Pasadena, California, 2001.
[5] T. Geßner. *Time-dependent adaption for supersonic combustion waves modeled with detailed reaction mechanisms.* PhD thesis, Universität Freiburg im Breisgau, 2001.
[6] P. Kaps and P. Rentrop. –. *Numerische Mathematik*, 33:55–68, 1979.
[7] B. Larrouturou. How to preserve the mass fractions positivity when computing compressible multi-component flows. *J. of Comput. Phys.*, 95:59–84, 1991.
[8] B. Larrouturou and L. Fezoui. On the equations of multi-component perfect or real gas inviscid flow. In Carasso et al., editor, *Proc. of Second Int. Conf. on Nonlinear Hyperbolic Equations - Theory, Numerical Methods, and Applications, Aachen 1988*, volume Lecture Notes in Mathematics 1402, pages 69–98. Springer-Verlag Berlin, 1989.
[9] E. S. Oran, J. W. Weber, E. I. Stefaniw, M. H. Lefebvre, and J. D. Anderson. A numerical study of a two-dimensional h$_2$-o$_2$-ar detonation using a detailed chemical reaction model. *J. Combustion and Flame*, 113:147–163, 1998.
[10] J.-S. Shuen, M.-S. Liou, and B. van Leer. Inviscid flux-splitting algorithms for real gases with non-equilibrium chemistry. *J. of Comput. Phys.*, 90:371–395, 1990.
[11] E. F. Toro. *Riemann solvers and numerical methods for fluid dynamics.* Springer-Verlag, Berlin, Heidelberg, 2nd edition, 1999.
[12] C. K. Westbrook. Chemical kinetics of hydrocarbon oxidation in gaseous detonations. *J. Combustion and Flame*, 46:191–210, 1982.
[13] F. A. Williams. *Combustion theory.* Addison-Wesley, Reading, Massachusetts, 1985.