

Detonation Structure Simulation with AMROC

Ralf Deiterding

California Institute of Technology,
1200 East California Blvd., Mail-Code 158-78, Pasadena, CA 91125
ralf@cacr.caltech.edu

Abstract. Numerical simulations can be the key to the thorough understanding of the multi-dimensional nature of transient detonation waves. But the accurate approximation of realistic detonations is extremely demanding, because a wide range of different scales needs to be resolved. In this paper, we summarize our successful efforts in simulating multi-dimensional detonations with detailed and highly stiff chemical kinetics on recent parallel machines with distributed memory, especially on clusters of standard personal computers. We explain the design of AMROC, a freely available dimension-independent mesh adaptation framework for time-explicit Cartesian finite volume methods on distributed memory machines, and discuss the locality-preserving rigorous domain decomposition technique it employs. The framework provides a generic implementation of the blockstructured adaptive mesh refinement algorithm after Berger and Collela designed especially for the solution of hyperbolic fluid flow problems on logically rectangular grids. The ghost fluid approach is integrated into the refinement algorithm to allow for embedded non-Cartesian boundaries represented implicitly by additional level-set variables. Two- and three-dimensional simulations of regular cellular detonation structure in purely Cartesian geometry and a two-dimensional detonation propagating through a smooth 60 degree pipe bend are presented. Briefly, the employed upwind scheme and the treatment of the non-equilibrium reaction terms are sketched.

1 Introduction

Reacting flows have been a topic of on-going research since more than hundred years. The interaction between hydrodynamic flow and chemical kinetics can be extremely complex and even today many phenomena are not very well understood. One of these phenomena is the propagation of detonation waves in gaseous media. Detonations are shock-induced combustion waves that internally consist of a discontinuous hydrodynamic shock wave followed by a smooth region of decaying combustion. In a self-sustaining detonation, shock and reaction zone propagate essentially with an identical supersonic speed between 1000 and 2000 m/s that is approximated to good accuracy by the classical Chapman-Jouguet (CJ) theory, cf. [26]. But up to now, no theory exists that describes the internal flow structure satisfactory. The Zel'dovich-von Neumann-Döring (ZND) theory is widely believed to reproduce the one-dimensional detonation structure correctly, but experiments [21] uncovered that the reduction to one space

dimension is not even justified in long combustion devices. It was found that detonation waves usually exhibit non-neglectable instationary multi-dimensional sub-structures in the millimeter range and do not remain exactly planar. The multi-dimensional instability manifests itself in instationary shock waves propagating perpendicular to the detonation front. A complex flow pattern is formed around each *triple point*, where the detonation front is intersected by a transverse shock. Pressure and temperature are increased enormously leading to a drastic enhancement of the chemical reaction. Hence, the accurate representation of triple points is essential for safety analysis, but also in technical applications, e.g. in the pulse detonation engine. Some particular mixtures, e.g. low-pressure hydrogen-oxygen with high argon diluent, are known to produce very regular triple point movements. The triple point trajectories form regular “fish-scale” patterns, so called detonation cells, with a characteristic length L and width λ (compare left sketch of Fig. 1).

Figure 1 displays the hydrodynamic flow pattern of a detonation with regular cellular structure as it is known since the early 1970s, cf. [21]. The right sketch shows the periodic wave configuration around a triple point in detail. It consists of a Mach reflection, a flow pattern well-known from non-reactive supersonic hydrodynamics [3]. The undisturbed detonation front is called the incident shock, while the transverse wave takes the role of the reflected shock. The triple point is driven forward by a strong shock wave, called Mach stem. Mach stem and reflected shock enclose the slip line, the contact discontinuity. The shock front inside the detonation cell travels as two Mach stems from point A to the line BC, see left graphic of Fig. 1. In the points B and C the triple point configuration is inverted nearly instantaneously and the front in the cell becomes the incident shock. Along the symmetry line AD the change is smooth and the shock strength decreases continuously. In D the two triple points merge exactly in a single point. The incident shock vanishes completely and the slip line, which was necessary for a stable triple point configuration between Mach stem and incident shock, is torn off and remains behind. Two new triple points with two new slip lines develop immediately after.

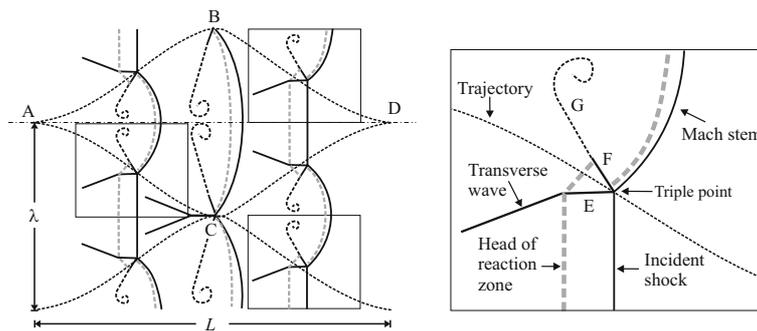


Fig. 1. Left: regular detonation structure at three different time steps on triple point trajectories, right: enlargement of a periodical triple point configuration. E: reflected shock, F: slip line, G: diffusive extension of slip line with flow vortex.

2 Governing Equations

The appropriate model for detonation propagation in premixed gases with realistic chemistry are the inviscid Euler equations for multiple thermally perfect species with reactive source terms [26]. These equations form a system of inhomogeneous hyperbolic conservation laws that reads

$$\begin{aligned} \partial_t \rho_i + \nabla \cdot (\rho_i \mathbf{u}) &= W_i \dot{\omega}_i, \quad i = 1, \dots, K, \\ \partial_t (\rho \mathbf{u}) + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) + \nabla p &= 0, \\ \partial_t (\rho E) + \nabla \cdot ((\rho E + p) \mathbf{u}) &= 0. \end{aligned} \quad (1)$$

Herein, ρ_i denotes the partial density of the i th species and $\rho = \sum_{i=1}^K \rho_i$ is the total density. The ratios $Y_i = \rho_i/\rho$ are called mass fractions. We denote the velocity vector by \mathbf{u} and E is the specific total energy. We assume that all species are ideal gases in thermal equilibrium and the hydrostatic pressure p is given as the *sum* of the partial pressures $p_i = \mathcal{R}T\rho_i/W_i$ with \mathcal{R} denoting the universal gas constant and W_i the molecular weight, respectively. The evaluation of the last equation necessitates the previous calculation of the temperature T . As detailed chemical kinetics typically require species with *temperature-dependent* material properties, each evaluation of T involves the approximative solution of an implicit equation by Newton iteration [4]. The chemical production rate for each species is derived from a reaction mechanism of J chemical reactions as

$$\dot{\omega}_i = \sum_{j=1}^J (\nu_{ji}^r - \nu_{ji}^f) \left[k_j^f \prod_{l=1}^K \left(\frac{\rho_l}{W_l} \right)^{\nu_{jl}^f} - k_j^r \prod_{l=1}^K \left(\frac{\rho_l}{W_l} \right)^{\nu_{jl}^r} \right], \quad i = 1, \dots, K, \quad (2)$$

with $\nu_{ji}^{f/r}$ denoting the forward and backward stoichiometric coefficients of the i th species in the j th reaction. The rate expressions $k_j^{f/r}(T)$ are calculated by an Arrhenius law, cf. [26].

3 Numerical Methods

We use the time-operator splitting approach or method of fractional steps to decouple hydrodynamic transport and chemical reaction numerically. This technique is most frequently used for time-dependent reactive flow computations. The *homogeneous* Euler equations and the usually stiff system of ordinary differential equations

$$\partial_t \rho_i = W_i \dot{\omega}_i(\rho_1, \dots, \rho_K, T), \quad i = 1, \dots, K \quad (3)$$

are integrated successively with the data from the preceding step as initial condition. The advantage of this approach is that a globally coupled implicit problem is avoided and a time-implicit discretization, which accounts for the stiffness of the reaction terms, needs to be applied only *local* in each finite volume cell. We use a semi-implicit Rosenbrock-Wanner method [10] to integrate

(3). Temperature-dependent material properties are derived from look-up tables that are constructed during start-up of the computational code. The expensive reaction rate expressions (2) are evaluated by a mechanism-specific Fortran-77 function, which is produced by a source code generator on top of the Chemkin-II library [11] in advance. The code generator implements the reaction rate formulas without any loops and inserts constants like $\nu_{ji}^{f/r}$ directly into the code.

As detonations involve supersonic shock waves we use a finite volume discretization that achieves a proper upwinding in all characteristic fields. The scheme utilizes a quasi-one-dimensional approximate Riemann solver of Roe-type [8] and is extended to multiple space dimensions via the method of fractional steps, cf. [22]. To circumvent the intrinsic problem of unphysical total densities and internal energies near vacuum due to the Roe linearization, cf. [6], the scheme has the possibility to switch to the simple, but extremely robust Harten-Lax-Van Leer (HLL) Riemann solver. Negative mass fraction values are avoided by a numerical flux modification proposed by Larrouturou [12]. Finally, the occurrence of the disastrous carbuncle phenomena, a multi-dimensional numerical cross-flow instability that destroys every simulation of strong grid-aligned shocks or detonation waves completely [18], is prevented by introducing a small amount of additional numerical viscosity in a multi-dimensional way [20]. A detailed derivation of the entire Roe-HLL scheme including all necessary modifications can be found in [4]. This hybrid Riemann solver is extended to a second-order accurate method with the MUSCL-Hancock variable extrapolation technique by Van Leer [22].

Higher order shock-capturing finite volume schemes are most efficient on rectangular Cartesian grids. In order to consider complex moving boundaries within the scheme outlined above we use some of the finite volume cells as ghost cells to enforce immersed boundary conditions [7]. Their values are set immediately before the original numerical update to model moving embedded walls. The boundary geometry is mapped onto the Cartesian mesh by employing a scalar level set function φ that stores the signed distance to the boundary surface and allows the efficient evaluation of the boundary outer normal in every mesh point as $\mathbf{n} = \nabla\varphi/|\nabla\varphi|$ [15]. A cell is considered to be a valid fluid cell in the interior, if the distance in the cell *midpoint* is positive and is treated as exterior otherwise. The numerical stencil by itself is not modified, which causes a slight diffusion of the boundary location throughout the method and results in an overall non-conservative scheme. We alleviate such errors and the unavoidable staircase approximation of the boundary with this approach by using the dynamic mesh adaptation technique described in Sec. 4 to also refine the Cartesian mesh appropriately along the boundary.

For the inviscid Euler equations (1) the boundary condition at a rigid wall moving with velocity \mathbf{w} is $\mathbf{u} \cdot \mathbf{n} = \mathbf{w} \cdot \mathbf{n}$. Enforcing the latter with ghost cells, in which the discrete values are located in the cell centers, involves the mirroring of the primitive values ρ_i , \mathbf{u} , p across the embedded boundary. The normal velocity in the ghost cells is set to $(2\mathbf{w} \cdot \mathbf{n} - \mathbf{u} \cdot \mathbf{n})\mathbf{n}$, while the mirrored tangential velocity remains unmodified. The construction of the velocity vector within the ghost cells

therefore reads $\mathbf{u}' = (2\mathbf{w} \cdot \mathbf{n} - \mathbf{u} \cdot \mathbf{n})\mathbf{n} + (\mathbf{u} \cdot \mathbf{t})\mathbf{t} = 2((\mathbf{w} - \mathbf{u}) \cdot \mathbf{n})\mathbf{n} + \mathbf{u}$ with \mathbf{t} denoting the boundary tangential. The utilization of mirrored cell values in a ghost cell center \mathbf{x} requires the calculation of spatially interpolated values in the point

$$\tilde{\mathbf{x}} = \mathbf{x} + 2\varphi\mathbf{n} \quad (4)$$

from neighboring interior cells. For instance in two space dimensions, we employ a bilinear interpolation between usually four adjacent cell values, but directly near the boundary the number of interpolants needs to be decreased, cf. Fig. 2. It has to be underlined that an extrapolation in such situations is inappropriate for hyperbolic problems with discontinuities like detonation waves that necessarily require the monotonicity preservation of the numerical solution. Figure 2 highlights the reduction of the interpolation stencil for some exemplary cases close to the embedded boundary. The interpolation location according to (4) are indicated by the origins of the red arrows.

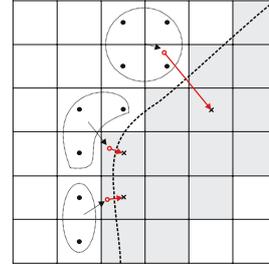


Fig. 2. Construction of values from interior cells used in internal ghost cells (gray)

4 An Adaptive Mesh Refinement Framework

Numerical simulations of detonation waves require computational meshes that are able to represent the strong local flow changes due to the reaction correctly. In particular, the induction zone between leading shock and head of reaction zone needs a high local resolution. The shock of a self-sustained detonation is very sensitive to changes in the energy release from the reaction behind and the inability to resolve all reaction details usually causes a considerable error in approximating the correct speed of propagation. In order to supply the necessary temporal and spatial resolution efficiently, we employ the blockstructured adaptive mesh refinement (AMR) method after Berger and Colella [2], which is tailored especially for hyperbolic conservation laws on logically rectangular finite volume grids. We have implemented the AMR method 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 [5]. An effective parallelization strategy for distributed memory machines has been found and the codes can be executed on all systems that provide the MPI library.

Instead of replacing single cells by finer ones, as it is done in cell-oriented refinement techniques, the Berger-Colella AMR method follows a patch-oriented approach. Cells being flagged by various error indicators (shaded in Fig. 3) are clustered with a special algorithm [1] into non-overlapping rectangular grids. Refinement grids are derived recursively from coarser ones and a hierarchy of successively embedded levels is thereby constructed, cf. Fig. 3. All mesh widths on level l are r_l -times finer than on level $l - 1$, i.e. $\Delta t_l := \Delta t_{l-1}/r_l$ and $\Delta x_{n,l} := \Delta x_{n,l-1}/r_l$ with $r_l \geq 2$ for $l > 0$ and $r_0 = 1$, and a time-explicit finite

volume scheme (in principle) remains stable on all levels of the hierarchy. The recursive temporal integration order is an important difference to usual unstructured adaptive strategies and is one of the main reasons for the high efficiency of the approach.

The numerical scheme is applied on level l by calling a single-grid routine in a loop over all subgrids. The subgrids are computationally decoupled by employing additional ghost cells around each computational grid. Three types of different ghost cells have to be considered in the sequential case: Cells outside of the root domain are used to implement physical boundary conditions. Ghost cells overlaid by a grid on level l have a unique interior cell analogue and are set by copying the data value from the grid, where the interior cell is contained (synchronization). On the root level no further boundary conditions need to be considered, but for $l > 0$ also internal boundaries can occur. They are set by a conservative time-space interpolation from two previously calculated time steps of level $l - 1$.

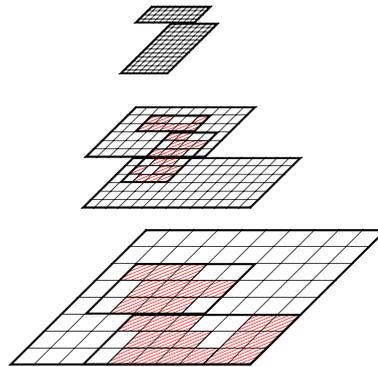


Fig. 3. AMR hierarchy

Beside a general data tree that stores the topology of the hierarchy, the AMR method utilizes at most two regular arrays assigned to each subgrid. They contain the discrete vector of state for the actual and updated time step. The regularity of the data allows high performance on vector and super-scalar processors and cache optimizations. Small data arrays are effectively avoided by leaving coarse level data structures untouched, when higher level grids are created. Values of cells covered by finer subgrids are overwritten by averaged fine grid values subsequently. This operation leads to a modification of the numerical stencil on the coarse mesh and requires a special flux correction in cells abutting a fine grid. The correction replaces the coarse grid flux along the fine grid boundary by a *sum* of fine fluxes and ensures the discrete conservation property of the hierarchical method at least for purely Cartesian problems without embedded boundaries. See [2] or [4] for details.

Up to now, various reliable implementations of the AMR method for single processor computers have been developed. Even the usage of parallel computers with shared memory is straight-forward, because a time-explicit scheme allows the parallel calculation of the grid-wise numerical update [1]. But the question for an efficient parallelization strategy becomes more delicate for distributed memory architectures, because on such machines the costs for communication can not be neglected. Due to the technical difficulties in implementing dynamical adaptive methods in distributed memory environments only few parallelization strategies have been considered in practice yet, cf. [19,17].

In the AMROC framework, we follow a rigorous domain decomposition approach and partition the AMR hierarchy from the root level on. The key idea is

that all higher level domains are required to follow this “floor-plan”. A careful analysis of the AMR algorithm uncovers that the only parallel operations under this paradigm are ghost cell synchronization, redistribution of the AMR hierarchy and the application of the previously mentioned flux correction terms. Interpolation and averaging, but in particular the calculation of the flux corrections remain strictly local [4]. Currently, we employ a generalization of Hilbert’s space-filling curve [16] to derive load-balanced root level distributions at runtime. The entire AMR hierarchy is considered by projecting the accumulated work from higher levels onto the root level cells. Although rigorous domain decomposition does not lead to a perfect balance of workload on single levels, good scale-up is usually achieved for moderate CPU counts. Figure 4 shows a representative scalability test for a three-dimensional spherical shock wave problem for the computationally inexpensive Euler equations for a single polytropic gas without chemical reaction. Roe’s approximate Riemann solver within the multi-dimensional Wave Propagation Method [13] is used as efficient single-grid scheme. The test was run on the ASC Linux cluster (ALC) at Lawrence Livermore National Laboratories that connects Pentium-4-2.4 GHz dual processor nodes with Quadrics Interconnect. The base grid has 32^3 cells and two additional levels with refinement factors 2 and 4. The adaptive calculation uses approx. 7.0 M cells in each time step instead of 16.8 M cells in the uniform case. The calculation on 256 CPUs employs between 1,500 and 1,700 subgrids on each level. Displayed are the average costs for each root level time step, which involve two time steps on the middle level and eight on the highest. All components of the dynamically adaptive algorithm, especially regridding and parallel redistribution are activated to obtain realistic results. Although we utilize a single-grid update routine in Fortran 77 in a C++ framework with full compiler optimization, the fraction of the time spent in this Fortran routine are 90.5 % on four and still 74.9 % on 16 CPUs. Hence, Fig. 4 shows a satisfying scale-up for at least up to 64 CPUs.

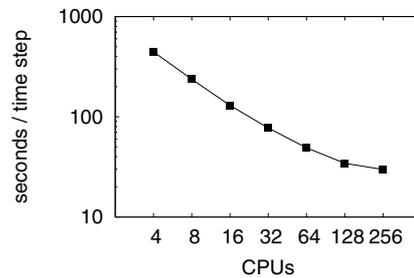


Fig. 4. Representative AMROC scale-up test for fixed problem size

5 Numerical Results

An ideal candidate for fundamental detonation structure simulations is the self-sustaining $\text{H}_2 : \text{O}_2 : \text{Ar}$ CJ detonation with molar ratios 2 : 1 : 7 at $T_0 = 298$ K and $p_0 = 6.67$ kPa that is known to produce extremely regular detonation cell patterns [21]. The analytical solution according to the one-dimensional ZND theory is extended to multiple space dimensions and transverse disturbances are initiated by placing a small rectangular unreacted pocket behind the detonation front, cf. [14] or [4]. Throughout this paper, only the hydrogen-oxygen reac-

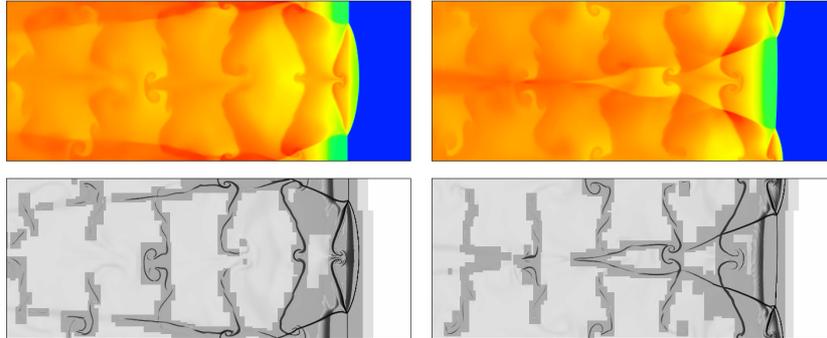


Fig. 5. Color plots of the temperature and schlieren plots of the density on refinement regions in the first (left) and second half (right) of a detonation cell

tion mechanism extracted from the larger hydrocarbon mechanism assembled by Westbrook is used [24]. The mechanism consists of 34 elementary reactions and considers the 9 species H, O, OH, H₂, O₂, H₂O, HO₂, H₂O₂ and Ar. According to the ZND theory, the induction length, the distance between leading shock and head of reaction zone in one space dimension, is $l_{ig} = 1.404$ mm for this mechanism in above configuration. The detonation velocity is 1626.9 m/s.

The application of the numerical methods of Sec. 3 within the parallel AMROC framework allowed a two-dimensional cellular structure simulation that is four-times higher resolved ($44.8 \text{ Pts}/l_{ig}$) than earlier calculations [14]. Only recently Hu et al. presented a similarly resolved calculation for the same CJ detonation on a uniform mesh [9]. Unfortunately, no technical details are reported for this simulation. In our case, the calculation was run on a small Beowulf-cluster of 7 Pentium 3-850 MHz-CPU's connected with a 1 Gb-Myrinet network and required 2150 h CPU-time. The calculation is in a frame of reference attached to the detonation. Because of the regularity of the oscillation only one cell is simulated. The adaptive run uses a root level grid of 200×40 cells and two refinement levels with $r_{1,2} = 4$. A physically motivated combination of scaled gradients and heuristically estimated relative errors is applied as adaptation criteria. See [4] for details. Two typical snapshots with the corresponding refinement are displayed in Fig. 5.

The high resolution of our simulation admits a remarkable refinement of the triple point pattern introduced in Sec. 1. Figure 6 displays the flow situation around the primary triple point A that is mostly preserved before the next collision. An analysis of the flow field uncovers the existence of two minor triple points B and C along the transverse wave downstream of A. While B can be clearly identified by a characteristic inflection, the triple point C is much weaker and very diffused. B is caused by the interaction of the strong shock wave BD with the transverse wave. The slip line emanating from B to K is clearly present. C seems to be caused by the reaction front and generates the very weak shock wave CI. A detailed discussion of the transient flow field is given in [4].

On 24 Athlon-1.4 GHz double-processor nodes (2 Gb-Myrinet interconnect) of the HEidelberg LInux Cluster System (Helics) our approach allowed a

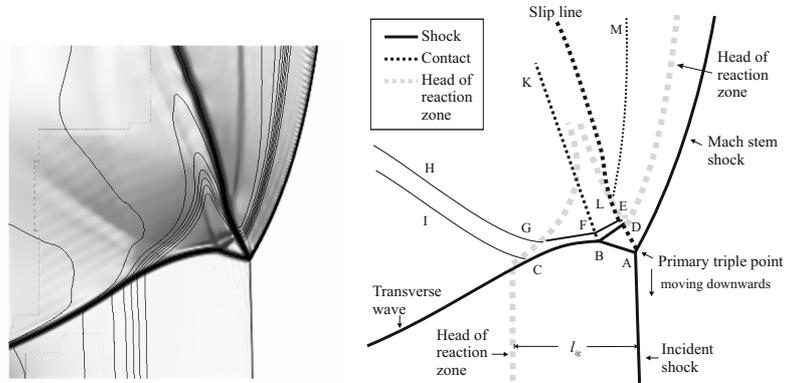


Fig. 6. Flow structure around a triple point before the next collision. Left: isolines of Y_{OH} (black) on schlieren plot of velocity component u_2 (gray).

sufficiently resolved computation of the three-dimensional cellular structure of a hydrogen-oxygen detonation. The maximal effective resolution of this calculation is $16.8 \text{ Pts}/l_{\text{ig}}$ and the run required 3800 h CPU-time. Our adaptive results are in perfect agreement with the calculations by Tsuboi et al. for the same configuration obtained on a uniform mesh on a super-scalar vector machine [23]. A snapshot of the regular two-dimensional solution of the preceding section is used to initialize a three-dimensional oscillation in the x_2 -direction and disturbed with an unreacted pocket in the orthogonal direction. We use a computational domain that exploits the symmetry of the initial data, but allows the development of a full detonation cell in the x_3 -direction. The AMROC computation uses a two-level refinement with $r_1 = 2$ and $r_2 = 3$ on a base grid of $140 \times 12 \times 24$ cells and utilizes between 1.3 M and 1.5 M cells, instead of 8.7 M cells like a uniformly refined grid.

After a settling time of approx. 20 periods a regular cellular oscillation with identical strength in x_2 - and x_3 -direction can be observed. In both transverse directions the strong two-dimensional oscillations is present and forces the creation of rectangular detonation cells with the same width as in two dimensions, but the transverse waves now form triple point lines in three space dimensions. During a complete detonation cell the four lines remain mostly parallel to the boundary and hardly disturb each other. The characteristic triple point pattern can therefore be observed in Fig. 7 in all planes perpendicular to a triple point line. Unlike Williams et al. [25] who presented a similar calculation for an overdriven detonation with simplified one-step reaction model, we notice no phase-shift between both transverse directions. In all our computations for the hydrogen-oxygen CJ detonation only this regular three-dimensional mode, called “rectangular-mode-in-phase”, or a purely two-dimensional mode with triple point lines just in x_2 - or x_3 -direction did occur.

In order to demonstrate the enormous potential of the entire approach even for non-Cartesian problems we finally show an example that combines highly efficient dynamic mesh adaptation with the embedded boundary method sketched

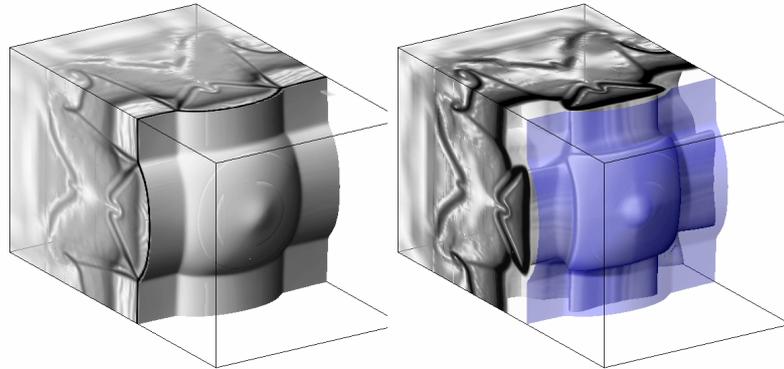


Fig. 7. Schlieren plots of ρ (left) and Y_{OH} (right) in the first half of a detonation cell (computational domain mirrored for visualization at lower boundary). The plot of Y_{OH} is overlaid by a translucent blue isosurface of ρ at the leading shock wave that visualizes the variation of the induction length l_{ig} in three space dimensions.

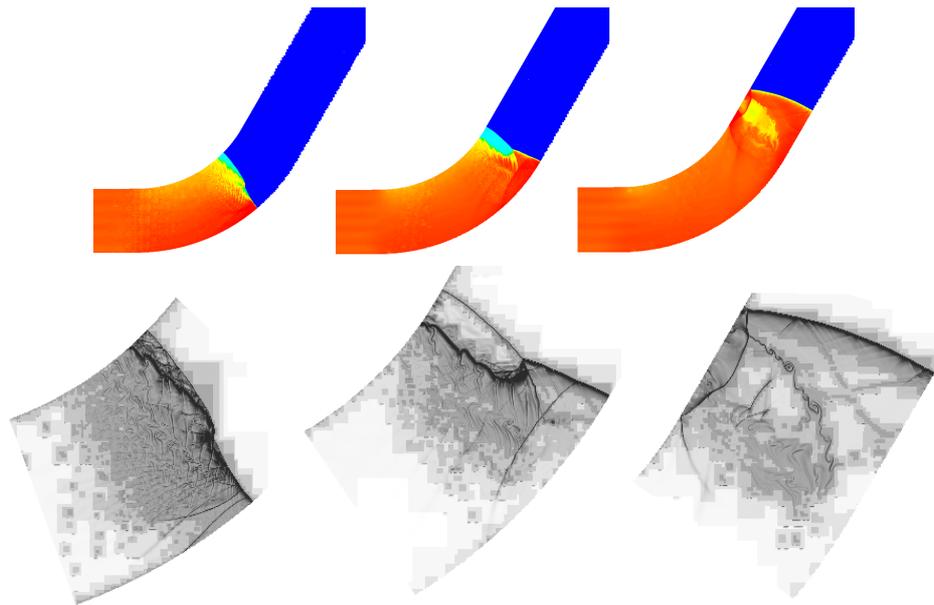


Fig. 8. Color plots of the temperature (upper row) and corresponding enlarged schlieren plots of the density on refinement regions (lower row) for a regular oscillating hydrogen-oxygen detonation propagating upwards a pipe bend.

in Sec. 3. A two-dimensional regular oscillating detonation is placed into a pipe of width 5λ . The pipe bends at an angle of 60 degree and with inner radius 9.375λ . When the detonation propagates through the bend it gets compressed and consequently overdriven near the outer wall, but a continuous shock wave

diffraction occurs near the inner wall. This diffraction causes a pressure decrease below the limit of detonability that leads to a continuous decoupling of shock and reaction front. This effect is clearly visible in the earliest graphics of Fig. 8. The detonation exits the bend before the decay to a flame occurs across the entire tube width. A re-ignition wave arises from the successfully transmitted region and reinitiates the detonation in the decoupled area, cf. middle graphics of Fig. 8. It propagates in the direction normal to the pipe middle axis and causes a strong shock wave reflection as it hits the inner wall, compare last graphics of Fig. 8. This simulation uses a base grid of 300×248 cells, four levels of refinement with $r_{1,2,3} = 2$, $r_4 = 4$, and has an effective resolution of 16.9 Pts/l_{ig}. Approximately 1.0 M to 1.5 M cells are necessary on all levels instead ≈ 76 M in the uniform case. See lower row of Fig. 8 for some snapshots of the dynamic mesh evolution. The simulation used approximately 3000 CPU hours on 64 CPUs of the ASC Linux cluster.

6 Conclusions

We have described an efficient solution strategy for the numerical simulation of gaseous detonations with detailed chemical reaction. All temporal and spatial scales relevant for the complex process of detonation propagation were successfully resolved. Beside the application of the time-operator splitting technique and the construction of a robust high-resolution shock capturing scheme, the key to the high efficiency of the presented simulations is the generic implementation of the blockstructured AMR method after Berger and Collella [2] in our AMROC framework [5]. AMROC provides the required high local resolution dynamically and follows a parallelization strategy tailored especially for the emerging generation of distributed memory architectures. An embedded boundary method utilizing internal ghost cells extends the framework effectively to non-Cartesian problems. All presented results have been achieved on Linux-Beowulf-clusters of moderate size in a few days real time which confirms the practical relevancy of the approach.

References

1. Bell, J., Berger, M., Saltzman, J., Welcome, M.: Three-dimensional adaptive mesh refinement for hyp. conservation laws. *SIAM J. Sci. Comp.* **15** (1994) (1):127–138
2. Berger, M., Colella, P.: Local adaptive mesh refinement for shock hydrodynamics. *J. Comput. Phys.* **82** (1988) 64–84
3. Courant, R., Friedrichs, K. O.: *Supersonic flow and shock waves*. Applied mathematical sciences **21** (Springer, New York, Berlin, 1976)
4. Deiterding, R.: *Parallel adaptive simulation of multi-dimensional detonation structures* (PhD thesis, Brandenburgische Technische Universität Cottbus, 2003)
5. Deiterding, R.: AMROC - Blockstructured Adaptive Mesh Refinement in Object-oriented C++. Available at <http://amroc.sourceforge.net> (2005)
6. Einfeldt, B., Munz, C. D., Roe, P. L., Sjögren, B.: On Godunov-type methods near low densities. *J. Comput. Phys.* **92** (1991) 273–295

7. Fedkiw, R. P., Aslam, T., Merriman, B., Osher, S.: A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method). *J. Comput. Phys.* **152** (1999) 457–492
8. Grossmann, B., Cinella, P.: Flux-split algorithms for flows with non-equilibrium chemistry and vibrational relaxation. *J. Comput. Phys.* **88** (1990) 131–168
9. Hu, X. Y., Khoo, B. C., Zhang, D. L., Jiang, Z. L.: The cellular structure of a two-dimensional H₂/O₂/Ar detonation wave. *Combustion Theory and Modelling* **8** (2004) 339–359
10. Kaps, P., Rentrop, P.: Generalized Runge-Kutta methods of order four with step-size control for stiff ordinary differential equations. *Num. Math.* **33** (1979) 55–68
11. Kee, R. J., Rupley, F. M., Miller, J. A.: *Chemkin-II: A Fortran chemical kinetics package for the analysis of gas-phase chemical kinetics*. (SAND89-8009, Sandia National Laboratories, Livermore, 1989)
12. Larrouturou, B.: How to preserve the mass fractions positivity when computing compressible multi-component flows. *J. Comput. Phys.* **95** (1991) 59–84
13. LeVeque, R. J.: Wave propagation algorithms for multidimensional hyperbolic systems. *J. Comput. Phys.* **131** (1997) (2):327–353
14. Oran, E. S., Weber, J. W., Stefaniw, E. I., Lefebvre, M. H., Anderson, J. D.: A numerical study of a two-dimensional H₂-O₂-Ar detonation using a detailed chemical reaction model. *J. Combust. Flame* **113** (1998) 147–163
15. Osher, S., Fedkiw, R.: *Level set methods and dynamic implicit surfaces*. Applied Mathematical Science **153** (Springer, New York, 2003)
16. Parashar, M., Browne, J. C.: On partitioning dynamic adaptive grid hierarchies. In Proc. of 29th Annual Hawaii Int. Conf. on System Sciences (1996)
17. Parashar, M., Browne, J. C.: System engineering for high performance computing software: The HDDA/DAGH infrastructure for implementation of parallel structured adaptive mesh refinement. In *Structured Adaptive Mesh Refinement Grid Methods*, IMA Volumes in Mathematics and its Applications (Springer, 1997)
18. Quirk, J. J.: Godunov-type schemes applied to detonation flows. In J. Buckmaster, editor, *Combustion in high-speed flows*, Proc. Workshop on Combustion, Oct 12-14, 1992, Hampton (Kluwer Acad. Publ., Dordrecht, 1994) 575–596
19. Rendleman, C. A., Beckner, V. E., Lijewski, M., Crutchfield, W., Bell, J. B.: Parallelization of structured, hierarchical adaptive mesh refinement algorithms. *Computing and Visualization in Science* **3** (2000)
20. Sanders, R., Morano, E., Druguet, M.-C.: Multidimensional dissipation for upwind schemes: Stability and applications to gas dynamics. *J. Comput. Phys.* **145** (1998) 511–537
21. Strehlow, R. A.: Gas phase detonations: Recent developments. *J. Combust. Flame* **12** (1968) (2):81–101
22. Toro, E. F.: *Riemann solvers and numerical methods for fluid dynamics* (Springer, Berlin, Heidelberg, 1999)
23. Tsuboi, N., Katoh, S., Hayashi, A. K.: Three-dimensional numerical simulation for hydrogen/air detonation: Rectangular and diagonal structures. *Proc. Combustion Institute* **29** (2003) 2783–2788
24. Westbrook, C. K.: Chemical kinetics of hydrocarbon oxidation in gaseous detonations. *J. Combust. Flame* **46** (1982) 191–210
25. Williams, D. N., Bauwens, L., Oran, E. S.: Detailed structure and propagation of three-dimensional detonations. *Proc. Combustion Institute* **26** (1997) 2991–2998
26. Williams, F. A.: *Combustion theory* (Addison-Wesley, Reading, 1985)