Multiscale Concept for Conservation Laws

Birgit Gottschlich-Müller,[1] Siegfried Müller

Institut für Geometrie und Praktische Mathematik
RWTH Aachen, 52056 Aachen

E-Mail: mueller@igpm.rwth-aachen.de or
birgit@igpm.rwth-aachen.de

Abstract

The physical modeling of fluid flows neglecting effects due to diffusion, viscosity and heat conduction results in hyperbolic differential equations of first order describing the change of state in space and time. These equations are the so-called conservation laws. The solution is characterized by the finite speed of propagation and the developing of discontinuities. Stable and robust numerical schemes are based on upwind techniques in order to handle shocks and contact dicontinuities. In general, schemes of this type are very expensive. In 1993 Harten derived a concept to reduce the computational costs. It is based on a multiscale decomposition of the approximation. This decomposition enables him to locate regions with singularities. Near discontinuities Harten applies an expensive upwind scheme and otherwise a much cheaper finite difference scheme of high order.

Harten's multiscale decomposition is based on the primitive function of the solution. Thus, his ansatz is inherently restricted to structured grids. In this paper we present an alternative multiscale decomposition using general reconstruction methods. These techniques can be extended to multidimensional problems even for unstructured grids. Here, we only consider the onedimensional case and we verify that our ansatz coincides with those of Harten's decomposition method. Furthermore, we apply our algorithm to dimensionally reduced hypersonic stagnation point flows around spheres.

Key Words: Multiresolution, conservation laws, finite volume methods, hybrid schemes.

AMS subject classification: 65M12, 76Axx, 35L65

[1]The author is supported by the Deutsche Forschungsgemeinschaft in the Graduiertenkolleg "Transportvorgänge in Hyperschallströmungen" at the RWTH Aachen.

1 Introduction

In this work we consider partial differential equations of the form

$$\frac{\partial\, u(t,x)}{\partial\, t} + \frac{\partial\, f(u(t,x))}{\partial\, x} = s(x, u(t,x)), \qquad 0 < t < T,\ a < x < b \tag{1}$$

with $T > 0$ and $a, b \in \mathrm{IR}$ fixed, so-called conservation laws. Herein, $u : [a,b] \to \mathrm{IR}$ is the conservative quantity, $f : \mathrm{IR} \to \mathrm{IR}$ the flux and $s : \mathrm{IR} \to \mathrm{IR}$ the source term. In order to ensure a unique solution, an initial condition

$$u(0,x) \ = \ u_0(x), \qquad a < x < b \tag{2}$$

has to be imposed. In practical applications, boundary conditions of the form

$$\begin{aligned} g_a(u(t,a), u_x(t,a)) &= 0, \qquad 0 \le t \le T \\ g_b(u(t,b), u_x(t,b)) &= 0, \qquad 0 \le t \le T \end{aligned} \tag{3}$$

are also needed.

It is well known that a smooth solution u of (1) in general only exists for small times T even if u_0 is smooth. Due to the nonlinearity of the flux discontinuities might develop which result in the break down of the differentiable solution. From a physical point of view those discontinuities are reasonable and should be admissible as solution of (1). Therefore, an extended class of solutions is introduced, the so-called weak solution. In one space dimension the weak solution is characterized by

$$\begin{aligned} &\int_I u(t+\tau, x)\, dx - \int_I u(t,x)\, dx = \\ &\int_t^{t+\tau} f(u(s,x_a))\, ds - \int_t^{t+\tau} f(u(s,x_b))\, ds + \int_I \int_t^{t+\tau} s(x, u(s,x))\, ds\, dx \end{aligned} \tag{4}$$

for all $I = [x_a, x_b] \subset \mathrm{IR}$ and $[t, t+\tau] \subset \mathrm{IR}_0^=$ (cf. [12]).

From equation (4) the class of finite volume schemes can be derived. For this purpose we discretize the continuum $[a,b] \times [0,T]$ in space and time by a grid, which is not necessarily uniform,

$$a = x_0 < \ldots < x_N = b, \qquad N \in \mathrm{IN}$$

and a uniform temporal grid

$$t_n = \tau\, n,\ \ \tau := T/\overline{N} \qquad \overline{N} \in \mathrm{IN}.$$

Then, equation (4) takes the form

$$\hat{u}_j^{n+1} = \hat{u}_j^n - \lambda_j \left[\hat{f}_j^n - \hat{f}_{j-1}^n\right] + \tau \hat{s}_j^n \tag{5}$$

where $\hat{u}_j^n$ is the average of the solution in the cell $I_j = [x_{j-1}, x_j]$ at time level t_n, $\hat{s}_j^n$ the spatial and temporal average of the source, $\hat{f}_j^n$ the temporal average at the interface x_j and λ_j the ratio of the discretization defined by

$$\begin{aligned}
\hat{u}_j^n &:= \frac{1}{|I_j|}\int\limits_{I_j} u(t_n, x)\, dx, && (6)\\
\hat{s}_j^n &:= \frac{1}{|I_j|}\int\limits_{I_j} \frac{1}{\tau}\int\limits_{t_n}^{t_{n+1}} s(x, u(t_n, x))\, dx, && (7)\\
\hat{f}_j^n &:= \frac{1}{\tau}\int\limits_{t_n}^{t_{n+1}} f(u(t, x_j))\, dt, && (8)\\
\lambda_j &:= \frac{\tau}{|I_j|}, \quad |I_j| := x_j - x_{j-1}, && (9)
\end{aligned}$$

This means that the average of the solution on the new time level can be calculated from the one on the old time level, the flow over the cell interfaces and the average of the source term. In order to evaluate the flux and the source the solution u itself has to be known explicitly. But except for the linear case, we do not have this knowledge. Therefore, the averages $\hat{f}_j^n$ and $\hat{s}_j^n$ have to be approximated which results in an approximation $\hat{v}_j^n$ of $\hat{u}_j^n$, i.e. equation (5) takes the form

$$\hat{v}_j^{n+1} = \hat{v}_j^n - \lambda_j \left[\hat{f}_j^n - \hat{f}_{j-1}^n\right] + \tau \hat{s}_j^n, \qquad j = 1, \ldots, N \tag{10}$$

with

$$\begin{aligned}
\hat{f}_j^n &:= \hat{f}_j^n(\hat{v}_{j-p+1}^n, \ldots, \hat{v}_{j+p}^n), \quad p \in \mathrm{IN} && (11)\\
\hat{s}_j^n &:= s(\hat{v}_j^n). && (12)
\end{aligned}$$

We notice that the numerical flux $\hat{f}_j^n$ can be calculated by (11) only for $p \leq j \leq N-p$. In the case of $j < p$ or $N - p < j$ we need either further information for $\hat{v}_j^n$ or numerical fluxes based on onesided stencils, i.e.

$$\begin{aligned}
\hat{f}_j^n &= \hat{f}_j^n(\hat{v}_1^n, \ldots, \hat{v}_{2p+1}^n), \quad 1 \leq j < p && (13)\\
\hat{f}_j^n &= \hat{f}_j^n(\hat{v}_{N-2p}^n, \ldots, \hat{v}_N^n), \quad N - p < j < N
\end{aligned}$$

In both cases we have to make sure that the boundary conditions (3) are fulfilled. In view of nonlinear boundary conditions we prefer the definition of auxiliary data. For the numerical fluxes $\hat{f}_0^n$ and $\hat{f}_N^n$ we have to apply the boundary conditions (3) themselves and the definition of the averaged flux (8).

Schemes that can be written in the form (10) are called finite volume schemes. They are of great importance from an analytical as well as from a practical point of view. During the last decade they have been applied to numerous practical applications in complex flow fields (cf. [18]). Two characteristic features most of the finite volume schemes have in common are solving Riemann problems at the cell interfaces and the reconstruction of a function by it's discrete cell averages to ensure upwinding and higher order (cf. [7]). Thus, the evaluation of the numerical flux is in general very expensive. In order to reduce the costs Harten developed the strategy of hybrid schemes (cf. [8]), i.e. in smooth flow regions a cheap finite difference scheme is used and in regions with singularities an expensive upwind scheme. But how to distinguish between those regions? For this purpose, Harten developed a multiresolution technique similar to multigrid techniques (cf. [9]). It is based on a sequence of grids

$$G_k := \{x_{k,j}\}_{j=0,\ldots,N_k} \subset [a,b], \quad k = 0,\ldots,L$$

where L is the number of refinement levels and N_k the number of cells on each level increasing with k. The cells are given by

$$I_{k,j} := [x_{k,j-1}, x_{k,j}], \quad j = 1,\ldots,N_k$$

with the interval length

$$h_{k,j} := x_{k,j} - x_{k,j-1}.$$

The grids are assumed to be nested, i.e.

$$G_{k+1} \subset G_k, \quad k = 0,\ldots,L-1.$$

Without loss of generality we assume throughout this work that the discretization is uniform on every level, i.e.

$$h_k = \frac{b-a}{N_k}, \quad k = 0,\ldots,L-1 \tag{14}$$

and the factor of refinement between two levels is just 0.5, i.e.

$$N_{k+1} = 2\,N_k, \quad k = 0,\ldots,L-1. \tag{15}$$

From this we conclude

$$\begin{aligned}
&h_{k+1} = h_k/2,\\
&x_{k+1,2j} = x_{k,j} = a + j\,h_k,\ j = 0,\ldots,N_k,\\
&G_{k+1} = G_k \cup \{x_{k+1,2j-1}\}_{j=1,\ldots,N_k},\\
&I_{k,j} = I_{k+1,2j-1} \cup I_{k+1,2j},\\
&\lambda_{k+1} = 2\,\lambda_k.
\end{aligned} \tag{16}$$

On every level a finite volume scheme is now defined analogously to (10) by

$$\hat{v}_{k,j}^{n+1} = \hat{v}_{k,j}^{n} - \lambda_k \left[\hat{f}_{k,j}^{n} - \hat{f}_{k,j-1}^{n}\right] + \tau \hat{s}_{k,j}^{n}, \qquad j = 1, \ldots, N_k \tag{17}$$

with

$$\hat{f}_{k,j}^{n} := \hat{f}_{k,j}^{n}(\hat{v}_{L,j_0-p+1}^{n}, \ldots, \hat{v}_{L,j_0+p}^{n}), \qquad j_0 = 2^{L-k}\, j \tag{18}$$

$$\hat{s}_{k,j}^{n} := \frac{h_L}{h_k} \sum_{i=1}^{2^{L-k}} s(\hat{v}_{L,2^{L-k}(j-1)+i}^{n}). \tag{19}$$

These read in vector representation as follows

$$\hat{\mathbf{v}}_k^{n+1} = \hat{\mathbf{v}}_k^{n} \;-\; \lambda_k \left\{\hat{\mathbf{f}}_{k,+}^{n} - \hat{\mathbf{f}}_{k,-}^{n}\right\} \;+\; \tau\, \hat{\mathbf{s}}_k^{n} \tag{20}$$

with

$$\hat{\mathbf{v}}_k^{n} = \left(\hat{v}_{k,j}^{n}\right)_{j=1,\ldots,N_K}, \; \hat{\mathbf{s}}_k^{n} = \left(\hat{s}_{k,j}^{n}\right)_{j=1,\ldots,N_K}, \; \hat{\mathbf{f}}_{k,\pm}^{n} = \left(\hat{f}_{k,j-\frac{1}{2}\pm\frac{1}{2}}^{n}\right)_{j=1,\ldots,N_K}.$$

We notice that (17) coincides with (10) for $k = L$. For $0 \leq k < L$ the numerical fluxes and the source terms are calculated from the data of the finest grid, i.e. level L. This is essential for the stability as numerical calculations have shown.

The main idea of Harten's concept is to determine a linear relation between the approximations $\mathbf{v}_k^n \in \mathrm{IR}^{N_k}$ and $\mathbf{v}_k^{n+1} \in \mathrm{IR}^{N_{k+1}}$ of two neighbouring levels. For this purpose the space IR^{N_k} has to be completed to $\mathrm{IR}^{N_{k+1}}$ by a vector $\mathbf{d}_k^n \in \mathrm{IR}^{N_{k+1}-N_k}$. The quantities $d_{k,j}^n$ can be interpreted as difference information between the coarse grid k and the fine grid $k+1$. On level $k+1$ we have N_{k+1} datas, whereas there are only N_k datas available on level k. Hence, we need $N_{k+1} - N_k$ additional information in order to reconstruct the information on the finer level from the data of the coarse level. One possible reconstruction technique is the interpolation of the coarse grid data and the evaluation at the additional points of the fine grid. This value can be interpreted as a prediction. Of course, an error is introduced by this value, the so-called predicition error. This is the difference between the interpolation and the exact solution on the fine level denoted by $d_{k,j}$.

Succesively the single scale representation $\hat{v}_L^n$ can be replaced by the multiscale representation

$$\hat{\mathbf{v}}_{MR,L}^{n} = \left((\mathbf{v}_0^n)^T, (\mathbf{d}_0^n)^T, \ldots, (\mathbf{d}_{L-1}^n)^T\right)^T. \tag{21}$$

Together with (17) a formula can be derived how to calculate the coefficients $d_{k,j}^{n+1}$ of the new time level t_{n+1}. If $|d_{k,j}^n|$ is sufficiently small, then this coefficient is truncated. In this case the numerical flux is computed by an interpolation formula instead of the original one, i.e. the computational costs are reduced and thus the computation itself is accelerated. We will see later that there must be a minimal number of truncated coefficients in order to gain a speed up since there is a certain overhead due to the multiscale transformation.

2 Multiscale Transformation

In the following we will derive the multiscale decomposition for the averages $\hat{u}_{k,j}$ of an arbitrary function $u : [a, b] \to \mathrm{IR}$ defined on every level $k = 0, \ldots, L$ by

$$\hat{u}_j^n := \frac{1}{|I_{k,j}|} \int\limits_{I_{k,j}} u(x)\, dx, \quad j = 1, \ldots, N_k. \tag{22}$$

The main task is to determine a completion of IR^{N_k} to $\mathrm{IR}^{N_{k+1}}$ such that

$$\hat{u}_{k+1,j} = \sum_{r=1}^{N_k} m_{j,r}^k \; \hat{u}_{k,r} + \sum_{r=1}^{N_k} \tilde{m}_{j,r}^k \; d_{k,r}, \quad j = 1, \ldots, N_{k+1} \tag{23}$$

holds and that can be reversed, i.e. there exist coefficients $g_{r,j}^k$ and $\tilde{g}_{r,j}^k$ such that

$$\hat{u}_{k,j} = \sum_{r=1}^{N_{k+1}} g_{j,r}^k \; \hat{u}_{k+1,r}, \tag{24}$$

$$d_{k,j} = \sum_{r=1}^{N_{k+1}} \tilde{g}_{j,r}^k \; \hat{u}_{k+1,r}. \tag{25}$$

Here, the coefficients $d_{k,j}$ are difference information between the data on level k and $k+1$. Those can be interpreted as a prediction error. Equation (23) is referred to as the decomposition and equations (24, 25) as the refinement. In matrix-vector representation they read

$$\hat{\mathbf{u}}_{k+1} = \hat{\mathbf{M}}_{k,0}\, \hat{\mathbf{u}}_k + \hat{\mathbf{M}}_{k,1}\, \mathbf{d}_k \tag{26}$$

respectively

$$\hat{\mathbf{u}}_k = \hat{\mathbf{G}}_{k,0}\, \hat{\mathbf{u}}_{k+1}, \tag{27}$$

$$\mathbf{d}_k = \hat{\mathbf{G}}_{k,1}\, \hat{\mathbf{u}}_{k+1} \tag{28}$$

with the vectors

$$\hat{\mathbf{u}}_k = (\hat{u}_{k,j})_{j=1,\ldots,N_k}, \quad \mathbf{d}_k = (d_{k,j})_{j=1,\ldots,N_k}$$

and the prolongation and restriction -matrices

$$\begin{aligned} \hat{\mathbf{M}}_{k,0} &= \left(m_{j,r}^k\right)_{\substack{j=1,\ldots,N_{k+1}\\ r=1,\ldots,N_k}}, & \hat{\mathbf{M}}_{k,1} &= \left(\tilde{m}_{j,r}^k\right)_{\substack{j=1,\ldots,N_{k+1}\\ r=1,\ldots,N_k}} \\ \hat{\mathbf{G}}_{k,0} &= \left(g_{j,r}^k\right)_{\substack{j=1,\ldots,N_k\\ r=1,\ldots,N_{k+1}}}, & \hat{\mathbf{G}}_{k,1} &= \left(\tilde{g}_{j,r}^k\right)_{\substack{j=1,\ldots,N_k\\ r=1,\ldots,N_{k+1}}}. \end{aligned} \tag{29}$$

With the help of (26, 27, 28) the single scale representation of $\hat{\mathbf{u}}_L$ can be transformed into the multiscale representation

$$\hat{\mathbf{u}}_{MR,L} = \left(\hat{\mathbf{u}}_0^T, \mathbf{d}_0^T, \ldots, \mathbf{d}_{L-1}^T\right)^T$$

by introducing the multiscale transformation $\mathbf{T}_L : \mathrm{IR}^{N_L} \to \mathrm{IR}^{N_L}$ and the inverse multiscale transformation $\mathbf{T}_L^{-1} : \mathrm{IR}^{N_L} \to \mathrm{IR}^{N_L}$ defined by

$$\mathbf{T}_L := \hat{\mathbf{G}}_0 \cdot \ldots \cdot \hat{\mathbf{G}}_{L-1} \quad \text{resp.} \quad \mathbf{T}_L^{-1} := \hat{\mathbf{M}}_{L-1} \cdot \ldots \cdot \hat{\mathbf{M}}_0$$

where the matrices $\hat{\mathbf{M}}_k$ and $\hat{\mathbf{G}}_k$ are given by

$$\begin{aligned}
\hat{\mathbf{G}}_k &:= \begin{pmatrix} \begin{pmatrix} \hat{\mathbf{G}}_{k,0} \\ \hat{\mathbf{G}}_{k,1} \end{pmatrix} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_k \end{pmatrix}, \qquad k = 0 \ldots, L-1, \\
\hat{\mathbf{M}}_k &:= \begin{pmatrix} \left(\hat{\mathbf{M}}_{k,0}, \hat{\mathbf{M}}_{k,1}\right) & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_k \end{pmatrix}, \qquad k = 0 \ldots, L-1.
\end{aligned}$$

By $\mathbf{I}_k$ we denote the identity matrix in $\mathrm{IR}^{(N_L - N_{k+1}) \times (N_L - N_{k+1})}$.

Then the following identities between $\hat{\mathbf{u}}_L$ and $\hat{\mathbf{u}}_{MR,L}$ hold:

$$\hat{\mathbf{u}}_{MR,L} = \mathbf{T}_L \, \hat{\mathbf{u}}_L \quad \text{resp.} \quad \hat{\mathbf{u}}_L = \mathbf{T}_L^{-1} \, \hat{\mathbf{u}}_{MR,L}.$$

3 Prolongation and Restriction

In the previous chapter a general multiscale concept has been presented. It is essentially based on the prolongation and restriction matrices which are derived in the following.

One possible method — examined by Gottschlich [6] — is the introduction of refinable basis functions, where the discrete values $\hat{u}_{k,j}$ and $d_{k,j}$ are identified with the coefficients of a serial expansion with respect to these functions. In this context the multiscale transformation can be interpreted as a change of basis. The latter is performed by certain matrices that can be identified with prolongation and restriction matrices. The principal advantage inherently lies in the construction of appropriate basis functions by means of functional analysis that lead to a stable multiscale transformation (see [4]). This analytical tool can only be applied to reflexive spaces. However, it is well known that the solution of scalar conservation laws is related to the non-reflexive space L^1 (see [11]).

In the following we will present a different concept for the prolongation and restriction matrices which first has been derived by Harten [9]. He introduced discretization and reconstruction mappings between a functional space and a sequence of vector spaces related to different grids. Then he connects two discretization levels by concatenating the discretization and reconstruction operators appropriately. Since the multiresolution transformation is only based on the relation between two discretizations the functional space as well as the discretization and reconstruction operators are not really needed. Therefore, they will not be introduced here.

From the definition (22) of the cell average and the assumption of nested and uniform grids (14, 15, 16) we get the relation between the averages defined on two nested grids:

$$\begin{aligned}
\hat{u}_{k,j} &= \frac{1}{h_k} \int\limits_{I_{k,j}} u(x)\, dx \\
&= \frac{1}{2\ h_{k+1}} \left\{ \int\limits_{I_{k+1,2j-1}} u(x)\, dx \ + \int\limits_{I_{k+1,2j}} u(x)\, dx \right\} \\
&= \frac{1}{2} \left\{ \hat{u}_{k+1,2j-1} \ + \ \hat{u}_{k+1,2j} \right\} .
\end{aligned} \tag{30}$$

Thus, the coefficients of $\hat{\mathbf{G}}_{k,0}$ are uniquely determined by

$$g^k_{j,r} = \begin{cases} \frac{1}{2} & , \quad r = 2j-1 \text{ or } r = 2j \\ 0 & , \quad \text{else} \end{cases} .$$

In order to determine the matrices $\hat{\mathbf{G}}_{k,1}$ and $\hat{\mathbf{M}}_{k,0}$, $\hat{\mathbf{M}}_{k,1}$ respectively we now have to determine a predicition of the data on level $k+1$ by those of level k for $k = 0, \ldots, L-1$ and hence the matrices (29). We notice that from equation (30) either $\hat{u}_{k+1,2j-1}$ or $\hat{u}_{k+1,2j}$ can be written in terms of $\hat{u}_{k,j}$ and the other cell average on the level $k+1$. Thus, we only have to predict one of those two values. In our analysis we calculate a prediction for $\hat{u}_{k+1,2j-1}$, i.e. we want to determine coefficients $a^k_{r,j}$ such that

$$\tilde{u}_{k+1,2j-1} := \sum_{r=1}^{N_k} a^k_{r,j}\ \hat{u}_{k,r} \tag{31}$$

is an approximation of $\hat{u}_{k+1,2j-1}$. Hence, the predicition error is given by

$$d_{k,j} := \hat{u}_{k+1,2j-1} - \tilde{u}_{k+1,2j-1}. \tag{32}$$

From equations (30, 31, 32) we now can calculate the matrices $\hat{\mathbf{M}}_{k,0}$ and $\hat{\mathbf{M}}_{k,1}$:

$$\hat{u}_{k+1,2j-1} = \tilde{u}_{k+1,2j-1} \ + \ d_{k,j} = \sum_{r=1}^{N_k} a^k_{r,j}\ \hat{u}_{k,r} \ + \ d_{k,j} \tag{33}$$

$$\hat{u}_{k+1,2j} = 2\ \hat{u}_{k,j} \ - \ \hat{u}_{k+1,2j-1} = 2\ \hat{u}_{k,j} \ - \ \sum_{r=1}^{N_k} a^k_{r,j}\ \hat{u}_{k,r} \ - \ d_{k,j} \tag{34}$$

and thus

$$m^k_{j,r} = \begin{cases} a^k_{r,l} & , \quad j = 2l-1 \\ 2\delta_{r,l} - a^k_{r,l} & , \quad j = 2l \end{cases} \tag{35}$$

$$\tilde{m}^k_{j,r} = \begin{cases} \delta_{r,l} & , \quad j = 2l-1 \\ -\delta_{r,l} & , \quad j = 2l \end{cases} \tag{36}$$

We notice that the matrix $\hat{\mathbf{M}}_{k,1}$ is uniquely determined by the definition of the cell average (6) and of the prediction error (32) as well as by the assumption of nested grids (16). The matrix $\hat{\mathbf{M}}_{k,0}$ is not uniquely determined yet. Here, the mask coefficients $a^k_{r,j}$ of the prediction (31) can be choosen by the user. Thus, we derived a whole class of multiresolution representations. In applications sparse matrices are preferable. In order to achieve these, the prediction should access only to a small number of local values, i.e. most of the mask coefficients should be zero.

From equations (30, 31, 32) we now conlcude

$$d_{k,j} = \hat{u}_{k+1,2j-1} - \frac{1}{2} \sum_{r=1}^{N_k} a^k_{r,j} \left\{ \hat{u}_{k+1,2r-1} \; + \; \hat{u}_{k+1,2r} \right\} \tag{37}$$

and thus

$$\tilde{g}^k_{j,r} = \frac{1}{2} \begin{cases} -a^k_{l,j} & , \quad r = 2l \\ 2\,\delta_{l,j} - a^k_{l,j} & , \quad r = 2l-1 \end{cases} .$$

4 Prediction via Reconstruction

In order to determine the coefficients $a^k_{r,j}$ of the prediction (31) we introduce the reconstruction function $R^N_{k,j} : I_{k,j} \to \mathrm{I\!R}$ defined by

$$R^N_{k,j}(x, \hat{\mathbf{u}}_k) = \sum_{i=0}^{N} \alpha^i_{k,j}(\hat{\mathbf{u}}_k)\, \omega^i_{k,j}(x), \qquad x \in I_{k,j} \tag{38}$$

of the discrete averages $\hat{\mathbf{u}}_k$. The functions $\omega^i_{k,j}$ have to be chosen judiciously. The coefficients $\alpha^i_{k,j}$ are determined from the recovery conditions

$$\frac{1}{|I_{k,l}|} \int_{I_{k,l}} R^N_{k,j}(x, \hat{\mathbf{u}}_k)\, dx \stackrel{!}{=} \hat{u}_{k,l}, \qquad l \in S_{k,j}. \tag{39}$$

Herein, $S_{k,j}$ is a set of stencils that is chosen locally for every cell j on level k.

$$S_{k,j} = \{ i_{k,j,l} \mid l = 0, \ldots, N \} . \tag{40}$$

An example for the choice of the set of stencil $S_{k,j}$ and the functions $\omega^i_{k,j}$ will be given later.

In order to ensure the uniqueness of the solution of (39) there must be exactly $N+1$ elements in this set. In general, the cell j will be a member of $S_{k,j}$. Otherwise the recovery conditions result in an extrapolation.

Substituting (38) into (39) renders a system of linear equations for the coefficients $\alpha^i_{k,j}$:

$$\mathbf{A}_{k,j}\ \boldsymbol{\alpha}_{k,j} = \mathbf{b}_{k,j} \tag{41}$$

with

$$\begin{aligned}
(\mathbf{A}_{k,j})_{l,r} &= \frac{1}{|I_{k,i_{k,j,l}}|} \int\limits_{I_{k,i_{k,j,l}}} \omega_{k,j}^{i_{k,j,r}}(x)\, dx, \qquad l,r = 0,\dots,N, \\
(\mathbf{b}_{k,j})_l &= \hat{u}_{k,i_{k,j,l}}, \quad (\boldsymbol{\alpha}_{k,j})_l = \alpha_{k,j}^l, \qquad l = 0,\dots,N.
\end{aligned} \tag{42}$$

In connection with ENO schemes special reconstruction polynomials were constructed by Harten et al. [7] based on primitive functions or deconvolution techniques in one space dimension. Later, Harten and Chakravarthy [8] derived general polynomial reconstruction methods for the multidimensional space on unstructered grids. In recent years Sonar [16] developed so-called optimal reconstruction techniques based on splines. They all have in common that they can be written in the form (38), where only the functions $\omega_{k,j}^i$ have to be altered.

We notice that the reconstruction functions $R_{k,j}^N$ are approximations of the function u itself, which is not known in general. Thus, a quite natural choice for the prediction $\tilde{u}_{k+1,2j-1}$ is

$$\tilde{u}_{k+1,2j-1} := \frac{1}{|I_{k+1,2j-1}|} \int\limits_{I_{k+1,2j-1}} R_{k,j}^N(x,\hat{\mathbf{u}}_k)\, dx = \sum_{i=0}^{N} \alpha_{k,j}^i\ \hat{\omega}_{k,j}^i = \sum_{r=1}^{N_k} a_{r,j}^k\ \hat{u}_{k,r} \tag{43}$$

with

$$\hat{\omega}_{k,j}^i := \frac{1}{|I_{k+1,2j-1}|} \int\limits_{I_{k+1,2j-1}} \omega_{k,j}^i(x)\, dx \tag{44}$$

The coefficients $\alpha_{r,j}^k$ are just the components of the solution of a system of linear equations (41). These depend linearily on the right hand side, here on $\hat{\mathbf{u}}_k$. Thus, the prediction also depends linearily on $\hat{\mathbf{u}}_k$, i.e. there exist coefficients $a_{r,j}^k,\ j = 1,\dots,N_k$ and the predicition can be written in the representation (31).

In the end, we want to remark that Pander and Sonar [17] in 1995 already published this ansatz in the context of Harten's framework.

5 Update

We now apply the multiscale transformation to the approximations $\hat{\mathbf{v}}_L^n$ of the finite volume scheme (17). According to the previous section we can calculate the multiscale representation $\hat{\mathbf{v}}_{MR,L}^n$ in analogy to (21). Up to now, the multiresolution representation has been derived for a fixed time level. The question now arising is, how to calculate the multiresolution representation of the new time level by that from the old time level. Therefore, we make use of the linearity of the representation and the finite volume scheme on level k.

On the coarsest level we have by (17),

$$v_{0,j}^{n+1} = \hat{v}_{0,j}^n - \lambda_0 \left[\hat{f}_{0,j}^n - \hat{f}_{0,j-1}^n\right] + \tau \hat{s}_{0,j}^n, \qquad j = 1, \ldots, N_0. \tag{45}$$

By the definiton (32) of the prediction error and equations (17, 20, 38, 39, 43) we get for the multiscale coefficients:

$$\begin{aligned}
d_{k,j}^{n+1} &= \hat{v}_{k+1,2j-1}^{n+1} - \tilde{v}_{k+1,2j-1}^{n+1} \\
&= \hat{v}_{k+1,2j-1}^{n} - \lambda_{k+1} \left\{\hat{f}_{k+1,2j-1}^n - \hat{f}_{k+1,2j-2}^n\right\} + \tau\, \hat{s}_{k+1,2j-1}^n - \\
&\quad \frac{1}{h_{k+1}} \int\limits_{I_{k+1,2j-1}} R_{k,j}^N \left(x, \hat{\mathbf{v}}_k^n - \lambda_k \left\{\hat{\mathbf{f}}_{k,+}^n - \hat{\mathbf{f}}_{k,-}^n\right\} + \tau\, \hat{\mathbf{s}}_k^n\right) dx \\
&= d_{k,j}^n - \lambda_{k+1}\, d_{k,j}^n \left(\hat{\mathbf{f}}_k^n\right) + \tau\, d_{k,j}^n(\hat{\mathbf{s}}_k^n), \qquad j = 1, \ldots, N_k.
\end{aligned} \tag{46}$$

Here, $\tilde{f}_{k+1,2j-1}^n$ is the prediction of the flux, $\tilde{s}_{k+1,2j-1}^n$ the prediction of the source and $d_{k,j}^n(\hat{\mathbf{s}}_k^n)$ the prediction error of the source term, $d_{k,j}^n\left(\hat{\mathbf{f}}_k^n\right)$ the prediction error of the numerical flux in point $x_{k+1,2j-1}$ defined by

$$d_{k,j}^n(\hat{\mathbf{s}}_k^n) := \hat{s}_{k+1,2j-1}^{n+1} - \tilde{s}_{k+1,2j-1}^n, \tag{47}$$

$$\tilde{s}_{k+1,2j-1}^n := \frac{1}{h_{k+1}} \int\limits_{I_{k+1,2j-1}} R_{k,j}^N \left(x, \hat{\mathbf{s}}_k^n\right) dx, \tag{48}$$

$$d_{k,j}^n \left(\hat{\mathbf{f}}_k^n\right) := \hat{f}_{k+1,2j-1}^n - \tilde{f}_{k+1,2j-1}^n, \tag{49}$$

$$\tilde{f}_{k+1,2j-1}^n := \hat{f}_{k,j-1}^n + \frac{1}{h_k} \int\limits_{I_{k+1,2j-1}} R_{k,j}^N \left(x, \hat{\mathbf{f}}_{k,+}^n - \hat{\mathbf{f}}_{k,-}^n\right) dx. \tag{50}$$

According to (43), the interpolation of the flux (50) and the source (48) can be written as

$$\tilde{f}_{k+1,2j-1}^n = \hat{f}_{k,j-1}^n + \frac{1}{2} \sum_{r=1}^{N_k} a_{r,j}^k \left(\hat{f}_{k,r}^n - \hat{f}_{k,r-1}^n\right) \tag{51}$$

$$\tilde{s}_{k+1,2j-1}^n = \sum_{r=1}^{N_k} a_{r,j}^k\, \hat{s}_{k,r}^n \tag{52}$$

Hence, the prediction error of the the numerical flux takes the form

$$d_{k,j}^n \left(\hat{\mathbf{f}}_k^n\right) = \hat{f}_{k+1,2j-1}^n - \hat{f}_{k,j-1}^n - \frac{1}{2} \sum_{r=1}^{N_k} a_{r,j}^k \left(\hat{f}_{k,r}^n - \hat{f}_{k,r-1}^n\right). \tag{53}$$

The values for the flux and the source term on the level $k+1$ in the even grid points are given according to (18, 30) by

$$\hat{f}_{k+1,2j}^n = \hat{f}_{k,j}^n, \tag{54}$$

$$\hat{s}_{k+1,2j}^n = 2\, \hat{s}_{k,j}^n - \hat{s}_{k+1,2j-1}^n. \tag{55}$$

In vector representation (45, 46) read:

$$\hat{\mathbf{v}}_0^{n+1} = \hat{\mathbf{v}}_0^n - \lambda_0 \left\{ \hat{\mathbf{f}}_{0,+}^n - \hat{\mathbf{f}}_{0,-}^n \right\} + \tau\, \hat{\mathbf{s}}_0^n, \tag{56}$$

$$\hat{\mathbf{d}}_k^{n+1} = \hat{\mathbf{d}}_k^n - \lambda_{k+1}\, \mathbf{d}_k^n \left(\hat{\mathbf{f}}_k^n \right) + \tau\, \mathbf{d}_k^n(\hat{\mathbf{s}}_k^n), \qquad k = 1, \ldots, L. \tag{57}$$

By the coefficients (35) and (36) as well as by (53) the identity

$$\frac{1}{2}\, \hat{\mathbf{M}}_{k,0} \left[\hat{\mathbf{f}}_{k,+}^n - \hat{\mathbf{f}}_{k,-}^n \right] + \hat{\mathbf{M}}_{k,1}\, \mathbf{d}_k^n \left(\hat{\mathbf{f}}_k^n \right) = \hat{\mathbf{f}}_{k+1,+}^n - \hat{\mathbf{f}}_{k+1,-}^n \tag{58}$$

holds and thus we can prove by induction for $k = 0, \ldots, L-1$ applying equations (57, 26, 58, 20):

$$\begin{aligned}
\hat{\mathbf{M}}_{k,0}\, \hat{\mathbf{v}}_k^{n+1} + \hat{\mathbf{M}}_{k,1}\, \hat{\mathbf{d}}_k^{n+1} &= \hat{\mathbf{M}}_{k,0} \left\{ \hat{\mathbf{v}}_k^n - \lambda_k \left[\hat{\mathbf{f}}_{k,+}^n - \hat{\mathbf{f}}_{k,-}^n \right] + \tau\, \hat{\mathbf{s}}_k^n \right\} + \\
&\quad\ \hat{\mathbf{M}}_{k,1} \left\{ \hat{\mathbf{d}}_k^n - \lambda_{k+1}\, \mathbf{d}_k^n \left(\hat{\mathbf{f}}_k^n \right) + \tau\, \mathbf{d}_k^n(\hat{\mathbf{s}}_k^n) \right\} \\
&= \hat{\mathbf{v}}_{k+1}^n + \tau\, \hat{\mathbf{s}}_{k+1}^n - \\
&\quad\ \lambda_{k+1} \left\{ \frac{1}{2}\, \hat{\mathbf{M}}_{k,0} \left[\hat{\mathbf{f}}_{k,+}^n - \hat{\mathbf{f}}_{k,-}^n \right] + \hat{\mathbf{M}}_{k,1}\, \mathbf{d}_k^n \left(\hat{\mathbf{f}}_k^n \right) \right\} \\
&= \hat{\mathbf{v}}_{k+1}^n - \lambda_{k+1} \left[\hat{\mathbf{f}}_{k+1,+}^n - \hat{\mathbf{f}}_{k+1,-}^n \right] + \tau\, \hat{\mathbf{s}}_{k+1}^n \\
&= \hat{\mathbf{v}}_{k+1}^{n+1}.
\end{aligned} \tag{59}$$

This means that the update of the coefficients $d_{k,j}$ of the multiscale representation is done in a way that does not change the finite volume scheme on the finest level, i.e. the inverse multiscale transformation of the updated coefficients $\hat{\mathbf{v}}_0^{n+1}$, $\mathbf{d}_k^{n+1}$ results in the average values $\hat{\mathbf{v}}_L^{n+1}$. This fact is essentially based on the evaluation of the numerical flux (18) and the source term (19) to the data of the finest level.

6 Thresh-Holding due to Harten

The main purpose of the multiscale transformation applied to conservation laws is to accelerate the numerical calculation. Harten's Ansatz is based on the reduction of numerical flux evaluations. Instead of an expensive flux based on upwind strategies and Riemann problems a finite difference formula is used in those parts of the flow field where the solution is smooth. This leads to a hybrid scheme.

In order to determine the smooth part the multiscale coefficients are helpful. If $d_{k,j}^n$ is sufficiently small, we will truncate this coefficient, i.e.

$$d_{k,j}^n := 0. \tag{60}$$

Therefore, we define a set of significant points

$$\mathcal{D}^{n+1} \subset \{(k,j) \mid k = 0, \ldots, L-1,\ j = 1, \ldots, N_k\}.$$

If $(k,j) \notin \mathcal{D}^{n+1}$, then we replace the numerical flux by the flux prediction. In this case the equations (49, 46) reduce to

$$
\begin{aligned}
d^n_{k,j}\left(\hat{\mathbf{f}}^n_k\right) &= 0, && (61)\\
\hat{f}^n_{k+1,2j-1} &= \tilde{f}^n_{k+1,2j-1}, && (62)\\
d^{n+1}_{k,j} &= d^n_{k,j} + \tau\, d^n_{k,j}(\hat{\mathbf{s}}^n_k). && (63)
\end{aligned}
$$

In the smooth part of the solution the coefficients $d^n_{k,j}$ are of magnitude $O\left(h^p_k\right)$ because they are defined as the prediction error. This error becomes smaller when the grid becomes finer. In order to guarantee a prescribed tolerance when truncating the coefficients on every level we have to define tolerances ϵ_k on each level. The level specific tolerances have to become smaller with coarser grid, i.e.

$$\epsilon_k := \epsilon_L/2^{L-k}. \qquad k = 0,\ldots,L$$

If the coefficient $d^n_{k,j}$ is not truncated, i.e. it is significant, the information can be propagated according to the finite speed of propagation — which is characteristic for hyperbolic problems — only into the neighbouring cells by the time step (cf. Fig. 1). This may result in significant coefficients in the neighbourhood, i.e.

$$|d^n_{k,j-l}| > \epsilon_k,\ -\overline{K} \le l \le \overline{K},\ 1 \le \overline{K} \le k. \tag{64}$$

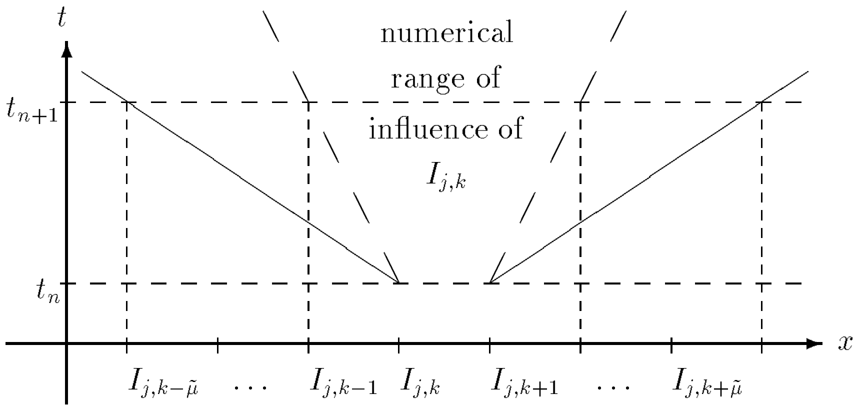

Figure 1: Propagation of information

It is characteristic for nonlinear conservation laws that discontinuities can develop with increasing time. It might happen that all coefficients on one level are truncated. With increasing gradients the coefficients will become larger than the prescribed specific tolerance. In this case there must be the possibility that the coarse level influences the finer one, i.e.

$$|d^n_{k,j}| > \epsilon_k \ \Rightarrow\ (k+1, 2j-1),\ (k+1, 2j) \in \mathcal{D}^{n+1},\ k < L-1. \tag{65}$$

We want to emphasize that there is no possibility to reduce the number of source evaluations, because of (19).

7 Computational Costs

The multiscale decomposition results in a certain overhead of floating point operations for every time level t_n. In order to accelerate the computation this must be compensated by reducing the number of expensive numerical flux evaluations below a certain number. In the following we determine the minimum number of non-significant points. For this purpose we assume that the matrices $\hat{\mathbf{G}}_{k,1}$ and $\hat{\mathbf{M}}_{k,0}$ in (23, 25) are uniformly banded, i.e. the number of non-vanishing coefficients $m^k_{j,r}$ respectively $\tilde{g}^k_{j,r}$ is less than K for all (k,j). The number of floating point operations for decoding C^{MR}_D, updating the multiscale coefficients on the new time level C^{MR}_U (cf. Appendix 11.1, Algs. 11.2, 11.4) and evaluating the numerical flux C^{MR}_f and source term C^{MR}_s are

$$\begin{aligned}
C^{MR}_D &= 2\,(2\;K-1)\;N_T,\\
C^{MR}_U &= 5\;N_0+(5\;K+7)\;N_T+3\;|\mathcal{D}^{n+1}|,\\
C^{MR}_f &= \left(N_0+1+|\mathcal{D}^{n+1}|\right)\;F_f,\\
C^{MR}_s &= N_L\;F_s
\end{aligned}$$

where F_f and F_s are the number of floating point operations for the evaluation of the numerical flux and the source and N_T is given by (16)

$$N_T := \sum_{k=0}^{L-1} N_k = N_L - N_0.$$

For the original scheme without multiscale decomposition and thresh-holding (10) we analogously get

$$\begin{aligned}
C_U &= 5\;N_L,\\
C_f &= (N_L+1)\;F_f,\\
C_s &= N_L\;F_s.
\end{aligned}$$

In C^{MR}_U there are $2\,N_0{+}2\,(3{+}K)\,N_T$ floating point operations included caused by the source term in (1), however in C_U there are only $2\,N_L$ operations. This results in an additional overhead of $2\,(2+K)\,N_T$ operations we have to pay for applying the multiresolution technique to inhomogeneous problems. Up to now, it can not be reduced by the thresh-holding technique.

The total amount of floating point operations in both cases is

$$C^{MR} := C^{MR}_D + C^{MR}_U + C^{MR}_f + C^{MR}_s, \tag{66}$$

$$C := C_U + C_f + C_s. \tag{67}$$

In order to achieve a speed up of the computation we need

$$C^{MR} \stackrel{!}{<} C.$$

From this condition the number of significant points is just

$$|\mathcal{D}^{n+1}| \stackrel{!}{<} \frac{F_f - 9\ K - 4}{F_f + 3}\ N_T.$$

Hence, the minimal reduction factor $R_{sp,min}$ is

$$\frac{N_T - |\mathcal{D}^{n+1}|}{N_T} \stackrel{!}{>} \frac{9\ K + 7}{F_f + 3} =: R_{sp,min} \tag{68}$$

We notice that for $F_f \leq 9\ K + 4$ no speed up is possible, but with increasing F_f the factor $R_{sp,min}$ becomes smaller. This means that for a more expensive numerical flux a smaller minimal reduction factor is sufficient to accelerate the computation.

But how is the reduction factor of the significant points

$$R_{sp} := \frac{N_T - |\mathcal{D}^{n+1}|}{N_T}$$

related to the reduction rate of the computational time

$$R_C := \frac{C - C^{MR}}{C}\ ?$$

From (66, 67) we get

$$R_C = \alpha\, R_{sp} - \beta \tag{69}$$

with

$$\alpha = \frac{F_f + 3}{C/N_T}, \quad \beta = \frac{7 + 9\, K}{C/N_T}\,. \tag{70}$$

From (15) the factor α can be estimated by

$$0.3 \leq \alpha \leq 1\,.$$

In section 9 we will see that also β can significantly reduce the computational speed up.

Finally, we want to remark that the costs for the thresh-holding procedure has been omitted in C^{MR}. This is not quite correct.

8 Harten's Multiscale Transformation

We now want to verify that the multiscale transformation presented by Harten (cf. [9]) coincides with our transformation for nested and uniform grids, if we choose $\omega_{k,j}^i$ and $S_{k,j}$ in an appropriate way. In the following we make use of (cf. [8])

$$\omega_{k,j}^i := (x - \hat{x}_{k,j})^i\,, \qquad i = 0, \ldots, N$$

where $\hat{x}_{k,j}$ is the center of mass of cell $I_{k,j}$:

$$\hat{x}_{k,j} := \frac{1}{h_k} \int\limits_{I_{k,j}} x \, dx = a + \left(j - \frac{1}{2}\right) \, h_k$$

and the set of stencil

$$S_{k,j} := \{j - s + l \mid l = 0, \ldots, N\} = \{j - s, \ldots, j + s\}, \quad N = 2s.$$

Thus, the reconstruction function $R^N_{k,j}$ defined by (38) is just the taylor expansion of u in the point $\hat{x}_{k,j}$. From this fact as well as equations (16, 42) we immediately conclude:

$$(A_{k,j})_{l,r} = \frac{h_k^r}{r+1} \left\{ \left(l - s + \frac{1}{2}\right)^{r+1} - \left(l - s - \frac{1}{2}\right)^{r+1} \right\}, \; l, r = 0, \ldots, N \tag{71}$$

$$\alpha^l_{k,j} = \frac{\partial^l \, u}{\partial \, x^l} + O\left(h_k^{N+1-l}\right), \tag{72}$$

$$R^N_{k,j}(x, \hat{\mathbf{u}}_k) = u(x) + O\left(h_k^{N+1}\right), \qquad x \in I_{k,j}. \tag{73}$$

The coefficients $\alpha^l_{k,j}$ are presented in Tab. 1 for $N = 0, 2, 4$.

	N = 0	N = 2	N = 4
$\alpha^0_{k,j}$	1	$-\frac{\hat{u}_{k,j+1} - 26\,\hat{u}_{k,j} + \hat{u}_{k,j-1}}{24}$	$\frac{9\,\hat{u}_{k,j+2} - 116\,\hat{u}_{k,j+1} + 2134\,\hat{u}_{k,j} - 116\,\hat{u}_{k,j-1} + 9\,\hat{u}_{k,j-2}}{1920}$
$\alpha^1_{k,j}$		$\frac{\hat{u}_{k,j+1} - \hat{u}_{k,j-1}}{2\,h_k}$	$-\frac{5\,\hat{u}_{k,j+2} - 34\,\hat{u}_{k,j+1} + 34\,\hat{u}_{k,j-1} - 5\,\hat{u}_{k,j-2}}{48\,h_k}$
$\alpha^2_{k,j}$		$-\frac{\hat{u}_{k,j+1} - 2\,\hat{u}_{k,j} + \hat{u}_{k,j-1}}{2\,h_k^2}$	$-\frac{\hat{u}_{k,j+2} - 12\,\hat{u}_{k,j+1} + 22\,\hat{u}_{k,j} - 12\,\hat{u}_{k,j-1} + \hat{u}_{k,j-2}}{16\,h_k^2}$
$\alpha^3_{k,j}$			$\frac{\hat{u}_{k,j+2} - 2\,\hat{u}_{k,j+1} + 2\,\hat{u}_{k,j-1} - \hat{u}_{k,j-2}}{42\,h_k^3}$
$\alpha^4_{k,j}$			$\frac{\hat{u}_{k,j+2} - 4\,\hat{u}_{k,j+1} + 6\,\hat{u}_{k,j} - 4\,\hat{u}_{k,j-1} + \hat{u}_{k,j-2}}{24\,h_k^4}$

Table 1: Coefficients $\alpha^l_{k,j}$ to stencil $S_{k,j} = \{j - s + l \mid l = 0, \ldots, N\}$

Instead of the above ansatz Harten uses the reconstruction via primitive function (cf [7]), i.e. he interpolates the primitive function of u

$$U(x) := \int\limits_{x_{k,0}}^{x} u(x) \, dx$$

at the points

$$U_{k,j} := U(x_{k,j}) = h_k \sum_{i=1}^{j} \hat{u}_{k,i}, \quad l = j - s - 1, \ldots, j + s, \tag{74}$$

using the Lagrange interpolation polynomial $L_{k,j}^{N+1}$ of degree $N+1$

$$L_{k,j}^{N+1}(x, \mathbf{U}_k) := \sum_{i=0}^{N+1} l_{i,j}^k(x)\, U_{k,j-s+i-1}, \qquad x \in I_{k,j} \tag{75}$$

where the Lagrange coefficients are defined by

$$l_{i,j}^k(x) := \prod_{\substack{l=0 \\ l \neq i}}^{N+1} \frac{x - x_{k,j-s+l-1}}{x_{k,j-s+i-1} - x_{k,j-s+l-1}}. \tag{76}$$

The latter satisfy

$$l_{i,j}^k(x_{k,j-s+l-1}) = \delta_{i,l}, \qquad i, l \in \{0, \ldots, N+1\}. \tag{77}$$

Because of the following relation between the cell averages and the primitive function on level k resulting from equations (16, 74, 77)

$$\begin{aligned}
\hat{u}_{k,l} &= \frac{1}{h_k}\, \{U_{k,l} - U_{k,l-1}\} \\
&= \frac{1}{h_k}\, \left\{L_{k,j}^{N+1}(x_{k,l}, \mathbf{U}_k) - L_{k,j}^{N+1}(x_{k,l-1}, \mathbf{U}_k)\right\} \\
&= \frac{1}{h_k} \int\limits_{I_{k,l}} \frac{d}{dx} L_{k,j}^{N+1}(x, \mathbf{U}_k)\, dx
\end{aligned} \tag{78}$$

the reconstruction polynomial via primitive function is defined by

$$R_{k,j}^{N,RP}(x, \hat{\mathbf{u}}_k) := \frac{d}{dx} L_{k,j}^{N+1}(x, \mathbf{U}_k), \qquad x \in I_{k,j}. \tag{79}$$

From the comparison of equation (78) and (39) and the fundamental theorem of polynomials we can conclude:

$$R_{k,j}^{N,RP}(x, \hat{\mathbf{u}}_k) \;=\; R_{k,j}^{N}(x, \hat{\mathbf{u}}_k), \qquad x \in I_{k,j}. \tag{80}$$

The prediction (31) now can be written as

$$\begin{aligned}
\tilde{u}_{k+1,2j-1} &= \frac{1}{h_{k+1}} \int\limits_{I_{k+1,2j-1}} R_{k,j}^{N,RP}(x, \hat{\mathbf{u}}_k)\, dx \\
&= \frac{1}{h_{k+1}}\, \left\{L_{k,j}^{N+1}(x_{k+1,2j-1}, \mathbf{U}_k) - L_{k,j}^{N+1}(x_{k+1,2j-2}, \mathbf{U}_k)\right\} \\
&= \frac{1}{h_{k+1}}\, \left\{L_{k,j}^{N+1}(x_{k+1,2j-1}, \mathbf{U}_k) - U_{k,j-1}\right\}
\end{aligned} \tag{81}$$

and the coefficients $a_{r,j}^k$ immediately result from

$$
\begin{aligned}
\tilde{u}_{k+1,2j-1} &= \frac{h_k}{h_{k+1}} \left\{ \sum_{i=0}^{N+1} l_{i,j}^k \sum_{l=1}^{j-s+i-1} \hat{u}_{k,l} - \sum_{l=1}^{j-1} \hat{u}_{k,l} \right\} \\
&= 2 \left\{ \sum_{i=1}^{N+1} l_{i,j}^k \sum_{l=0}^{i-1} \hat{u}_{k,j-s+l} - \sum_{l=0}^{s-1} \hat{u}_{k,j-s+l} \right\} \\
&= 2 \left\{ \sum_{l=0}^{N} \left(\sum_{i=l+1}^{N+1} l_{i,j}^k \right) \hat{u}_{k,j-s+l} - \sum_{l=0}^{s-1} \hat{u}_{k,j-s+l} \right\} \\
&= \sum_{r=1}^{N_k} a_{r,j}^k \, \hat{u}_{k,r}
\end{aligned}
\tag{82}
$$

with

$$
\begin{aligned}
a_{j-s+l,j}^k &:= -2 \sum_{i=0}^{l} l_{i,j}^k, \qquad l = 0, \ldots, s-1, \\
a_{j+s-l,j}^k &:= -2 \sum_{i=l+1}^{N} l_{i,j}^k, \qquad l = 0, \ldots, s.
\end{aligned}
$$

For $N = 0, 2, 4$ the coefficients are presented in Tab. 2 (cf. $\mu = 0$). We notice the antisymmetric property

$$
\begin{aligned}
a_{j,j}^k &= 1, \\
a_{j-s+l,j}^k &= -a_{j+s-l,j}^k, \qquad l = 0, \ldots, s-1.
\end{aligned}
$$

In the special case considered here the prediction error is given by the interpolation error of the interpolation polynomial $L_{k,j}^{N+1}(x, \mathbf{U}_k)$ in the point $x_{k+1,2j-1}$:

$$
\begin{aligned}
d_{k,j} &= \hat{u}_{k+1,2j-1} - \tilde{u}_{k+1,2j-1} \\
&= \frac{1}{h_{k+1}} \left\{ U_{k+1,2j-1} - L_{k,j}^{N+1}(x_{k+1,2j-1}, \mathbf{U}_k) \right\} \\
&= \frac{1}{h_{k+1}} \frac{\omega(x_{k+1,2j-1}) \, U^{(N+2)}(\xi)}{(N+2)!}
\end{aligned}
\tag{83}
$$

with

$$
\begin{aligned}
\omega(x) &:= \prod_{l=0}^{N+1} (x - x_{k,j-s+l-1}), \\
\omega(x_{k+1,2j-1}) &= h_{k+1}^{N+2} \prod_{l=0}^{N+1} (N - 2l + 1).
\end{aligned}
$$

In order to determine the flux prediction $\tilde{f}_{k+1,2j-1}^n$ for the update of the multiscale coefficients we first remark the property

$$R^N_{k,j}(x,\hat{\mathbf{f}}^n_{k,\pm}) \;=\; \frac{d}{dx} L^{N+1}_{k,j}(x,\hat{\mathbf{F}}^n_{k,\pm}), \qquad x \in I_{k,j} \tag{84}$$

with

$$\hat{\mathbf{F}}^n_{k,\pm} := \left(h_k \sum_{i=1}^{j} \hat{\mathbf{f}}^n_{k,i-\frac{1}{2}\pm\frac{1}{2}}. \right)_{j=1,\ldots,N_k} \tag{85}$$

From this follows

$$\begin{aligned} \int_{I_{k+1,2j-1}} R^N_{k,j}\left(x,\hat{\mathbf{f}}^n_{k,\pm}\right)\,dx \;&=\; L^{N+1}_{k,j}\left(x_{k+1,2j-1},\hat{\mathbf{F}}^n_{k,\pm}\right) \;-\; L^{N+1}_{k,j}\left(x_{k+1,2j-2},\hat{\mathbf{F}}^n_{k,\pm}\right) \\ &=\; L^{N+1}_{k,j}\left(x_{k+1,2j-1},\hat{\mathbf{F}}^n_{k,\pm}\right) \;-\; \hat{F}^n_{k,j-\frac{3}{2}\pm\frac{1}{2}} \end{aligned} \tag{86}$$

and

$$\frac{1}{h_k} \int_{I_{k+1,2j-1}} R^N_{k,j}\left(x,\hat{\mathbf{f}}^n_{k,+} \;-\; \hat{\mathbf{f}}^n_{k,-}\right)\,dx = L^{N+1}_{k,j}\left(x_{k+1,2j-1},\hat{\mathbf{f}}^n_k\right) \;-\; \hat{f}^n_{k,j-1}. \tag{87}$$

Thus, in the special case considered here the flux prediction $\tilde{f}^n_{k+1,2j-1}$ is just the interpolation of the flux:

$$\tilde{f}^n_{k+1,2j-1} = L^{N+1}_{k,j}\left(x_{k+1,2j-1},\hat{\mathbf{f}}^n_k\right).$$

Comparing this result with Harten's multiresolution representation, we notice that the general ansatz derived here coincides with those of Harten.

There still remains one problem. How to calculate $\hat{v}^n_{k,j},\ j \in I_a$ and $\hat{v}^n_{k,N_k+j},\ j \in I_b$ for $I_a := \{1-s,\ldots,0\},\ I_b := \{1,\ldots,s\}$ at the boundary of the computational domain? In general, we have two alternatives. We either can introduce artificial quantities outside of the computational domain, i.e.

a) Nonlinear boundary conditions:

$$\begin{aligned} \hat{v}_{k,j} &= g_{a,j}(\hat{\mathbf{v}}_k), \quad j \in I_a, \\ \hat{v}_{k,N_k+j} &= g_{b,j}(\hat{\mathbf{v}}_k), \quad j \in I_b; \end{aligned}$$

b) Linear boundary conditions:

$$\hat{v}_{k,j} = \sum_{r=1}^{N_k} \tilde{a}^k_{r,j}\ \hat{v}_{k,r}, \quad j \in I_a,$$

$$\hat{v}_{k,N_k+j} = \sum_{r=1}^{N_k} \tilde{a}^k_{r,j}\ \hat{v}_{k,r}, \quad j \in I_b;$$

or we resort to onesided interpolation stencils, i.e.

$$S_{k,j} = \{1, \dots, N\}, \quad j \in \{1, \dots, s\},$$
$$S_{k,j} = \{N_k - N, \dots, N_k\}, \quad j \in \{N_k - s + 1, \dots, N_k\}.$$

In the case of artificial quantities we have to make sure that

$$\begin{aligned} \hat{v}_{k,j}^{n+1} &= \hat{v}_{k,j}^{n} - \lambda_k \left[\hat{f}_{k,j}^{n} - \hat{f}_{k,j-1}^{n} \right] + \tau \hat{s}_{k,j}^{n}, \qquad j \in I_a, \\ \hat{v}_{k,N_k+j}^{n+1} &= \hat{v}_{k,N_k+j}^{n} - \lambda_k \left[\hat{f}_{k,N_k+j}^{n} - \hat{f}_{k,N_k+j-1}^{n} \right] + \tau \hat{s}_{k,N_k+j}^{n}, \qquad j \in I_b, \end{aligned} \tag{88}$$

holds in view of (59). This is satisfied for linear boundary conditions, since inside the computational domain the update of the multiscale coefficients (45, 46) renders (59). In order to guarantee (88) for nonlinear boundary conditions, we have to put

$$\hat{f}_{k,j}^{n} := \hat{f}_{k,0}^{n}, \qquad j \in I_a,$$
$$\hat{f}_{k,N_k+j}^{n} := \hat{f}_{k,N_k}^{n}, \qquad j \in I_b.$$

s	N	μ	$\dots$	$a_{j-4,j}^k$	$a_{j-3,j}^k$	$a_{j-2,j}^k$	$a_{j-1,j}^k$	$a_{j,j}^k$	$a_{j+1,j}^k$	$a_{j+2,j}^k$	$a_{j+3,j}^k$	$a_{j+4,j}^k$	$\dots$
0	0	0	$\dots$	$\dots$	$\dots$	$\dots$	0	1	0	$\dots$	$\dots$	$\dots$	$\dots$
1	2	-1	$\dots$	$\dots$	0	-$\frac{1}{8}$	$\frac{1}{2}$	$\frac{5}{8}$	0	$\dots$	$\dots$	$\dots$	$\dots$
1	2	0	$\dots$	$\dots$	$\dots$	0	$\frac{1}{8}$	1	-$\frac{1}{8}$	0	$\dots$	$\dots$	$\dots$
1	2	1	$\dots$	$\dots$	$\dots$	$\dots$	0	$\frac{11}{8}$	-$\frac{1}{2}$	$\frac{1}{8}$	0	$\dots$	$\dots$
2	4	-2	0	$-\frac{7}{128}$	$\frac{19}{64}$	-$\frac{11}{16}$	$\frac{61}{64}$	$\frac{63}{128}$	0	$\dots$	$\dots$	$\dots$	$\dots$
2	4	-1	$\dots$	0	$\frac{3}{128}$	-$\frac{9}{64}$	$\frac{13}{32}$	$\frac{49}{64}$	-$\frac{7}{128}$	0	$\dots$	$\dots$	$\dots$
2	4	0	$\dots$	$\dots$	0	-$\frac{3}{128}$	$\frac{11}{64}$	1	-$\frac{11}{64}$	$\frac{3}{128}$	0	$\dots$	$\dots$
2	4	1	$\dots$	$\dots$	$\dots$	0	$\frac{7}{128}$	$\frac{79}{64}$	-$\frac{13}{32}$	$\frac{9}{64}$	-$\frac{3}{128}$	0	$\dots$
2	4	2	$\dots$	$\dots$	$\dots$	$\dots$	0	$\frac{193}{128}$	-$\frac{61}{64}$	$\frac{11}{16}$	-$\frac{19}{64}$	$\frac{7}{128}$	0

Table 2: Coefficients $a_{r,j}^k$ to stencil $S_{k,j}^{\mu} = \{j - s + \mu + l \mid l = 0, \dots, N\}$

We prefer onesided stencils, because this is more rigorous from a mathematical point of view. The definition of artificial quantities is based on physical arguments. For the stencils

$$S_{k,j}^{\mu} := \{j - s + \mu + l \mid l = 0, \dots, N\}, \quad \mu = -s, \dots, s$$

the coefficients $a_{r,j}^k$ are summarized in Tab. 2.

9 Application

The calculation of steady hypersonic blunt body flow is of great importance in the design of re-entry vehicles because there is a high heat transfer at the body surface. The flow field is characterized by a detached bow shock and a boundary layer at the wall (cf. Fig. 2).

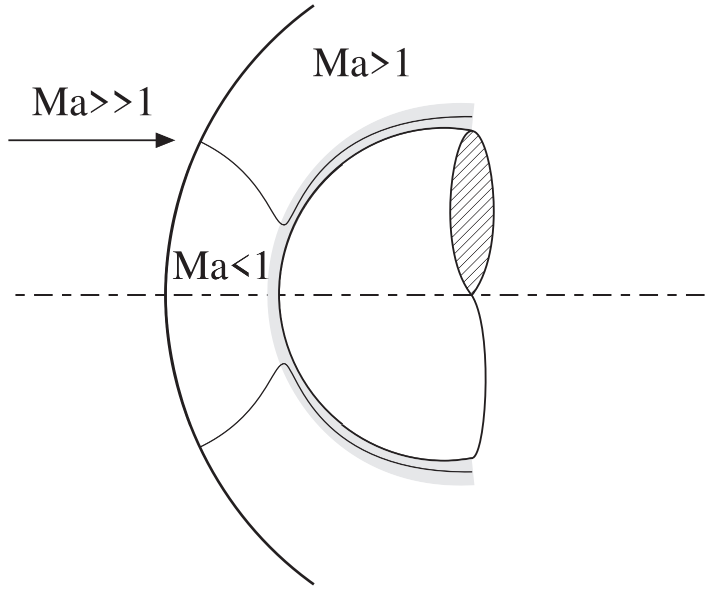

Figure 2: Hypersonic blunt body flow

For a rotational symmetric flow around a sphere with radius R_{sph} Klomfass and Müller (cf. [10]) derived a dimensionally reduced model. This can be written in the form

$$\frac{\partial\,\mathbf{u}(t,r)}{\partial\,t} + \frac{\partial\,\mathbf{f}(\mathbf{u}(t,r))}{\partial\,r} = \mathbf{s}(r,\mathbf{u}(t,r)), \qquad 0 < t,\ a < r < b \tag{89}$$

where the flux $\mathbf{f}$ and the source $\mathbf{s}$ consist of two parts

$$\mathbf{f}(\mathbf{u}) = \mathbf{f}^I(\mathbf{u}) + \mathbf{f}^V\left(\mathbf{u}, \frac{\partial\,\mathbf{u}}{\partial\,r}\right), \qquad \mathbf{s}(\mathbf{u}) = \mathbf{s}^I(\mathbf{u}) + \mathbf{s}^V\left(\mathbf{u}, \frac{\partial\,\mathbf{u}}{\partial\,r}\right),$$

namely the inviscid part describing the convection in the flow field and the viscous part modeling the dissipative effects due to viscosity and heat conduction:

$$\mathbf{u} = \begin{pmatrix} \rho \\ \rho\,\overline{v}_r \\ \rho\,\overline{v}_\theta \\ \rho\,\overline{E} \end{pmatrix}, \quad \mathbf{f}^I(\mathbf{u}) = \begin{pmatrix} \rho\,\overline{v}_r \\ \rho\,\overline{v}_r^2 + p \\ \rho\,\overline{v}_r\overline{v}_\theta \\ \rho\,\overline{v}_r\,\overline{H} \end{pmatrix}, \quad \mathbf{f}^V\left(\mathbf{u}, \frac{\partial\,\mathbf{u}}{\partial\,r}\right) = -\begin{pmatrix} 0 \\ \overline{\tau}_{rr} \\ \overline{\tau}_{\theta\theta} \\ \overline{\tau}_{rr}\,\overline{v}_r - \overline{q}_r \end{pmatrix},$$

$$\mathbf{s}^V\left(r, \mathbf{u}, \frac{\partial\,\mathbf{u}}{\partial\,r}\right) = \frac{1}{r}\begin{pmatrix} 0 \\ \overline{\tau}_{rr} + \overline{\tau}_{r\theta} - \overline{\tau}_{\theta\theta} \\ 3\,\overline{\tau}_{r\theta} - \overline{\tau}_{\theta\theta} \\ 2\,\left(\overline{\tau}_{rr}\,\overline{v}_r + \left(\overline{\tau}_{r\theta} + \overline{\tau}_{\theta\theta}\right)\,\overline{v}_\theta - \overline{q}_r\right) \end{pmatrix},$$

$$\mathbf{s}^I(r,\mathbf{u}) = -\frac{1}{r}\begin{pmatrix} 2\,\rho\,(\overline{v}_r+\overline{v}_\theta) \\ 2\,\rho\,\overline{v}_r(\overline{v}_r+\overline{v}_\theta) \\ 3\,\rho\,\overline{v}_\theta(\overline{v}_r+\overline{v}_\theta) - 2\,\overline{p} \\ 2\,\rho\,\overline{H}(\overline{v}_r+\overline{v}_\theta) \end{pmatrix},$$

where $\overline{E}$ and $\overline{H}$ denote total energy and total enthalpy

$$\overline{E} = \overline{e} + \frac{1}{2}\overline{v}_r^2, \quad \overline{H} = (\overline{E} + p/\rho),$$

$\overline{\tau}_{rr},\ \overline{\tau}_{r\theta},\ \overline{\tau}_{\theta\theta}$ the elements of the stress tensor

$$\begin{aligned} \overline{\tau}_{rr} &= \frac{4}{3}\mu\left\{\frac{\partial\,\overline{v}_r}{\partial\,r} - \frac{\overline{v}_r+\overline{v}_\theta}{r}\right\} = -2\,\overline{\tau}_{\theta\theta}\,, \\ \overline{\tau}_{r\theta} &= \mu\left\{\frac{\partial\,\overline{v}_\theta}{\partial\,r} - \frac{\overline{v}_r+\overline{v}_\theta}{r}\right\} \end{aligned}$$

and $\overline{q}_r$ the heat flux

$$\overline{q}_r = -\lambda\,\frac{\partial\,\overline{T}}{\partial\,r}\,.$$

For the viscosity coefficient μ we apply Sutherland's law

$$\mu = \mu_\infty\,\frac{\overline{T}^{3/2}}{\overline{T}+110.4}$$

and determine the heat conduction coefficient λ from the Prandtl number Pr which we assume to be constant

$$\lambda = \frac{\mu\,c_p}{Pr}, \quad Pr = 0.7\,.$$

In our calculations we only consider a perfect gas, i.e. the specific heats c_p and c_v — and hence the ratio γ —are constant:

$$c_p = 969.9\overline{3}\ \ [\text{J/kg K}], \quad c_p = 692.8\overline{3}\ \ [\text{J/kg K}], \quad \gamma = \frac{c_p}{c_v} = 1.4\,.$$

Finally, the system is closed by the thermal and caloric equation of state

$$\begin{aligned} p &= \rho\,R\,\overline{T} = p_\infty + \overline{p}, \quad R = 277.1\overline{3}\ \ [\text{J/kg K}], \\ \overline{e} &= c_v\,\overline{T} = \frac{p}{\rho(\gamma-1)}, \end{aligned}$$

where p_∞ is the free-stream pressure.

In the following, system (89) is referred to as the Navier-Stokes equations or the Euler equations, if the second order terms are neglected.

We want to emphasize that in the previous sections only hyperbolic conservation laws are considered. However, system (89) is no longer hyperbolic but parabolic. But for high Reynolds-number flows, i.e. the Reynolds number Re

$$Re = \frac{\overline{v}_{r,\infty}\,\rho_\infty\,R_{sph}}{\mu_\infty}$$

is sufficiently large, it is dominated by convective effects which are described by the first order terms. The subscript ∞ denotes the free-stream condition.

In our computations we examine only one flow field prescribed by the free-stream conditions for the Mach number, the velocity and the pressure

$$Ma_\infty = 10,\ \overline{v}_r = 2928\ [\mathrm{m/s}]\ ,\ p_\infty = 19.16\ [\mathrm{Pa}]\ . \tag{90}$$

These render the initial conditions for the conservative quantities

$$\mathbf{u}(0,r) = \begin{pmatrix} \rho \\ \rho\,\overline{v}_r \\ \rho\,\overline{v}_\theta \\ \rho\,\overline{E} \end{pmatrix}_\infty = \begin{pmatrix} 3.1288\times 10^{-4} & [\mathrm{kg/m^3}] \\ -9.1612\times 10^{-1} & [\mathrm{kg/m^2\ s}] \\ 9.1612\times 10^{-1} & [\mathrm{kg/m^2\ s}] \\ 1.3891\times 10^{3} & [\mathrm{J/m^3}] \end{pmatrix}.$$

From this we calculate

$$T_\infty = 220.97\quad [\mathrm{K}],\qquad \mu_\infty = 1.458\times 10^{-6}\quad [\mathrm{kg/m\,s}],\qquad Re = 95759\,.$$

The computational domain is given by

$$a = -0.1784,\qquad b = -R_{sph} = -0.1524.$$

At the inflow boundary we put the free-stream conditions

$$\mathbf{u}(t,a) = \mathbf{u}_\infty\,.$$

The wall is assumed to be non-adiabatic, i.e. we precribe a constant wall temperature

$$\overline{T}(t,b) = 1500\quad [\mathrm{K}]\,,$$

and the pressure is approximately constant in the boundary layer

$$\frac{\partial\, p(t,b)}{\partial\, r} = 0\,.$$

For the velocity we apply the no-slip condition

$$\overline{v}_r(t,b) = \overline{v}_\theta(t,b) = 0\,.$$

In the case of an inviscid flow, i.e. the second order terms in (89) are neglected, we choose the slip condition at the wall

$$\overline{v}_r(t,b) = 0$$

and vanishing gradients for the remaining quantities

$$\frac{\partial\, \rho(t,b)}{\partial\, r} = \frac{\partial\, \overline{v}_\theta(t,b)}{\partial\, r} = \frac{\partial\, \overline{T}(t,b)}{\partial\, r} = 0\,.$$

The inviscid part of (89) is approximated by an ENO scheme of second order. This is based on the reconstruction of the characteristic variables via primitive function, the so-called characteristic reconstruction (cf. [7]). The Riemann problems at the cell interfaces are solved by the approximate Riemann solver of Roe (cf. [14]). The viscous fluxes at the cell interfaces are approximated by central differences inside the computational domain and onesided differences at the boundaries. The number of floating point operations of the numerical flux and the source per component are

$$F_{f^I} = 266, \qquad F_{f^V} = 19, \qquad F_{s^I} = 6, \qquad F_{s^V} = 16\,. \tag{91}$$

In our calculations we apply the multiscale concept based on the coefficients due to Harten for $s = 1$ to each component of the vector of the conservative quantities (cf. Sec. 8). The number of the refinement levels is $L = 6$ and the number of points on the coarsest grid $N_0 = 10$. For the truncation process we replace $|d_{k,j}|$ by $\|\mathbf{d}_{k,j}\|_\infty$ (cf. Alg. 11.3) and choose $\overline{K} = 1$. The thresh-hold value for the truncation on the finest level is $\epsilon_L = 0.05$. In order to achieve the steady state we execute 60000 time steps with a fixed time discretization $\tau = 8 \times 10^{-9}$ for the Euler calculations and 240000 time steps with $\tau = 2 \times 10^{-9}$ for the Navier-Stokes calculations. All calculations are performed on an SGI workstation with R8000 processor.

In Fig. 3 and 4 the temperature is plotted for different time levels. We notice that the shock wave detaches from the wall — here at the right side — and runs to the left. The shock speed decreases towards zero with increasing integration time. At time 4.8×10^{-4} the steady state is approached. If dissipative effects are taken into account a boundary layer developes additionally. In the shock layer the temperature is approximately constant, because chemical and thermal relaxation processes are neglected. For a more realistic calculation they have to be taken into account.

In inviscid stagnation-point calculations the temperature at the wall is of special interest. For a perfect gas the analytic solution can be calculated explicitly (cf. [1]) from

$$\overline{H}_s = \overline{H}_\infty, \ \overline{\mathbf{v}}_s = 0$$

as

$$T_s = T_\infty \left(1 + \frac{\gamma - 1}{2}\, Ma_\infty^2\right) .$$

From (90) we get for the present test case $T_s = 4640$ K, whereas our calculations with and without multiresolution predict $T_s = 4649$ K.

Opposite to the inviscid flow field, the wall heat-flux q_w is a very important quantity if viscosity and heat conduction are taken into account. But there exists no explicit solution even in the case of an ideal gas. A reference value for the stagnation-point heat-flux can be determined with the formula of Fay and Riddell [5]

$$q_w = 0.76\,(Pr)^{-0.6}\,(\rho_e\,\mu_e)^{0.4}\,(\rho_w\,\mu_w)^{0.1}\,(H_e - H_w)\sqrt{\left(\frac{\partial\, v}{\partial\, y}\right)_s}$$

derived from numerical solutions of the boundary layer equations at the stagnation streamline. Here $(\partial\, v/\partial\, y)_s$ is the stagnation-point velocity gradient that can be estimated by the Newtonian theory (cf. [2]), i.e. $Ma \to \infty$, as

$$\left(\frac{\partial\, v}{\partial\, y}\right)_s = \frac{1}{R_{sph}}\sqrt{\frac{2(p_e - p_\infty)}{\rho_e}}\,.$$

Subscript e and w denote the value of the quantity at the outer edge of the boundary layer and at the wall respectively. The first are identified with the stagnation-point conditions predicted by the Euler equations:

$$H_e = 4500959 \text{ [J/kg]} \ , \ T_e = 4648 \text{ [K]} \ , \ p_e = 2475 \text{ [Pa]} \ , \ \rho_e = 1.921 \times 10^{-3} \text{ [kg/m}^3] \ .$$

From these we calculate $q_w = 127660$ [W/m^2], whereas our calculations with and without multiresolution technique predict $q_w = 106970$ [W/m^2] and $q_w = 107800$ [W/m^2], respectively.

From the previous discussion we can conclude that the numerical results are not essentially influenced by the truncation process. In the following we want to examine the gain of efficiency that is related to the calculations. For different time levels the number of significant points, i.e. $|\mathcal{D}^{n+1}|$, as well as the temperature are presented for the Euler (cf. Figs. 5 - 12) and the Navier-Stokes (cf. Figs. 13 - 20) calculation. From the distribution of the significant points we can conclude that the multiresolution technique is capable to detect local changes in the gradients. The coefficients of the finest grid level are only involved around the bow shock and close to the wall. Although no temperature gradient exists at the wall in the inviscid flow field (cf. Fig. 11), we know from earlier investigations that the error due to numerical dissipation results in a gradient of the total enthalpy (cf. [13]). This seems to explain the significant points near the wall.

In Fig. 21 and 22 the number of significant points is plotted for certain time levels. We notice a strong deviation in the Euler and the Navier-Stokes calculation. In the latter case, the number is approximately constant (cf. Fig. 22). This is different in the inviscid flow field. When the bow shock detaches, this number increases until the shock speed is approximately zero, i.e. at $t \approx 1 \times 10^{-4}$. Approaching the steady state, the gradients within the shock layer are diminishing (cf. Fig. 3). The number of significant points is reduced correspondingly. This behaviour might be explained by the different degree of smoothness in the shock layer. The Euler equations form a system of first order derivatives only, whereas the Navier-Stokes equations are based on second order derivatives. Some more investigations are necessary to examine the relation between the truncation process and the conservation laws in order to determine an optimal thresh-holding procedure. Recently, Bihari developed a method for conservation laws with viscosity (cf. [3]).

The computational costs are directly related to $|\mathcal{D}^{n+1}|$ (cf. Figs. 21, 23 and 22,24). From (68) and (91) we know that at least a reduction factor of $R_{sp,min} = 0.12$ is needed in order to achieve a speed up of the computation. In Figs. 25, 26 the reduction factor R_{sp} at certain time levels is presented. Throughout the Euler computation the number of significant points is reduced by an average value of 60%. Thus, we attend from (69, 70) with $\alpha \approx 0.95, \beta \approx 0.12$ a gain of computational time of about 45%. However, a pure reduction of the numerical flux evaluations does not result in a speed up of the computation. The computation based on the multiresolution strategy takes us 4214 seconds for the inviscid and 19478 seconds for the viscous flow field. The same calculation without the multiresolution transformation takes us 5614 and 27542 seconds. Thus, the gain of computational time by the multiresolution technique in connection with the thresh-holding strategy is about 25% and 29%, respectively. Although, this is a good increase in efficiency, it is not as good as the value predicted from (69). It can be explained by the neglected costs for the case decisions in (66, 67).

10 Conclusion

In the present work we have derived a more general multiscale concept for conservation laws. In contrast to Harten's ansatz our prediction is based on general reconstruction functions not necessarily restricted to polynomials (cf. [16]) instead of the interpolation of the primitive function. Hence, our strategy can be extended in a straight forward way on to multidimensional and unstructured grids. First investigations have been performed by Sonar and Schröder-Pander(cf. [17]).

Application to dimensionally reduced hypersonic blunt body flow rendered a gain of computational time of 25% for inviscid and 29% for viscous flow fields. Further investigations are necessary in order to verify the speed up by the multiscale strategy for multidimensional problems. It will be interesting to see whether the strategy of Sjögreen (cf. [15]) in two space dimensions based on Harten's one-dimensional multiscale transformation can be improved by our technique.

Acknowledgement

The authors want to thank Wolfgang Dahmen for many helpful remarks.

11 Appendix

In the following the algorithm derived in the present paper are summarized. We always refer to the governing equations.

11.1 Algorithm

Algorithm 11.1 (Encode)

Input: $\hat{\boldsymbol{v}}_L^n$

Output: $\hat{\boldsymbol{v}}_{MR,L}^n$

<u>for</u> $k := L-1$ <u>to</u> 0 <u>do</u>

    <u>for</u> $j := 1$ <u>to</u> N_k <u>do</u>

$$\hat{v}_{k,j}^n \stackrel{(24)}{=} \sum_{r=1}^{N_{k+1}} g_{j,r}^k \; \hat{v}_{k+1,r}^n \stackrel{(30)}{=} \frac{1}{2}\left\{\hat{v}_{k+1,2j-1}^n + \hat{v}_{k+1,2j}^n\right\};$$

    <u>od</u> ;

<u>od</u> ;

<u>for</u> $k := L-1$ <u>to</u> 0 <u>do</u>

    <u>for</u> $j := 1$ <u>to</u> N_k <u>do</u>

$$d_{k,j}^n \stackrel{(25)}{=} \sum_{r=1}^{N_{k+1}} \tilde{g}_{j,r}^k \; \hat{v}_{k+1,r}^n \, ; \stackrel{(30,37)}{=} \hat{v}_{k+1,2j-1}^n - \sum_{r=1}^{N_k} a_{r,j}^k \; \hat{v}_{k,r}^n \, ;$$

    <u>od</u> ;

<u>od</u> ;

Algorithm 11.2 (Decode)

Input: $\hat{\boldsymbol{v}}_{MR,L}^n$

Output: $\hat{\boldsymbol{v}}_L^n$

<u>for</u> $k := 0$ <u>to</u> $L-1$ <u>do</u>

    <u>for</u> $j := 1$ <u>to</u> N_{k+1} <u>do</u>

$$\begin{aligned} \hat{v}_{k+1,j}^n &\stackrel{(23)}{=} \sum_{r=1}^{N_k} m_{j,r}^k \; \hat{v}_{k,r}^n + \sum_{r=1}^{N_k} \tilde{m}_{j,r}^k \; d_{k,r}^n \\ &\stackrel{(33,34)}{=} \begin{cases} \sum_{r=1}^{N_k} a_{r,(j+1)/2}^k \; \hat{v}_{k,r}^n + d_{k,(j+1)/2}^n & , \; j \text{ odd} \\ 2\, \hat{v}_{k,j/2}^n - \sum_{r=1}^{N_k} a_{r,j/2}^k \; \hat{v}_{k,r}^n - d_{k,j/2}^n & , \; j \text{ even} \end{cases} \qquad ; \end{aligned}$$

    <u>od</u> ;

<u>od</u> ;

Algorithm 11.3 (Thresh-Holding (cf. Harten [9], p. 27))

Input: $\hat{\boldsymbol{v}}^n_{MR,L}$

Output: $\mathcal{D}^{n+1}$

<u>for</u> $k := 0$ <u>to</u> $L-1$ <u>do</u>

    <u>for</u> $j := 1$ <u>to</u> N_k <u>do</u>

        <u>if</u> ($|d^n_{k,j}| \leq \epsilon_k$) <u>then</u>

            $d^n_{k,j} \stackrel{(60)}{=} 0$;

        <u>else</u>

            <u>for</u> $l = -\overline{K}$ <u>to</u> $\overline{K}$ <u>do</u>

                $(k, j-l) \stackrel{(64)}{\in} \mathcal{D}^{n+1}$;

            <u>od</u> ;

            <u>if</u> ($|d^n_{k,j}| \geq 2^{p+1}\epsilon_k$ <u>and</u> $k > 1$) <u>then</u> ;

                $(k-1, 2j-1) \stackrel{(65)}{\in} \mathcal{D}^{n+1}$;

                $(k-1, 2j) \stackrel{(65)}{\in} \mathcal{D}^{n+1}$;

            <u>fi</u> ;

        <u>fi</u> ;

    <u>od</u> ;

<u>od</u> ;

Algorithm 11.4 (Update with Thresh-Holding)

Input: $\hat{\boldsymbol{v}}_L^n,\ \hat{\boldsymbol{s}}_L^n,\ \hat{\boldsymbol{f}}_0^n,\ \hat{\boldsymbol{v}}_{MR,L}^n,\ \mathcal{D}^{n+1}$

Output: $\hat{\boldsymbol{v}}_{MR,L}^{n+1}$

$\underline{\text{for}}\ k := L-1\ \underline{\text{downto}}\ 0\ \underline{\text{do}}$

$\quad \underline{\text{for}}\ j := 1\ \underline{\text{to}}\ N_k\ \underline{\text{do}}$

$$\hat{s}_{k,j}^n \overset{(55)}{=} \tfrac{1}{2}\left(\hat{s}_{k+1,2j-1}^n + \hat{s}_{k+1,2j}^n\right)\ ;$$

$\quad \underline{\text{od}}$;

$\underline{\text{od}}$;

$\underline{\text{for}}\ j := 1\ \underline{\text{to}}\ N_0\ \underline{\text{do}}$

$$v_{0,j}^{n+1} \overset{(45)}{=} \hat{v}_{0,j}^n - \lambda_0\left[\hat{f}_{0,j}^n - \hat{f}_{0,j-1}^n\right] + \tau\,\hat{s}_{0,j}^n\,;$$

$\underline{\text{od}}$;

$\underline{\text{for}}\ k := 0\ \underline{\text{to}}\ L-1\ \underline{\text{do}}$

$\quad \underline{\text{for}}\ j := 1\ \underline{\text{to}}\ N_k\ \underline{\text{do}}$

$$\tilde{f}_{k+1,2j-1}^n \overset{(51)}{=} \hat{f}_{k,j-1}^n + \tfrac{1}{2}\sum_{r=1}^{N_k} a_{r,j}^k\ \left(\hat{f}_{k,r}^n - \hat{f}_{k,r-1}^n\right)\ ;$$

$$\tilde{s}_{k+1,2j-1}^n \overset{(52)}{=} \sum_{r=1}^{N_k} a_{r,j}^k\ \hat{s}_{k,r}^n\ ;$$

$$d_{k,j}^n(\hat{\boldsymbol{s}}_k^n) \overset{(47)}{=} \hat{s}_{k+1,2j-1}^{n+1} - \tilde{s}_{k+1,2j-1}^n\,;$$

$$\hat{f}_{k+1,2j}^n \overset{(54)}{=} \hat{f}_{k,j}^n\,;$$

$$\hat{s}_{k+1,2j}^n \overset{(55)}{=} 2\ \hat{s}_{k,j}^n - \hat{s}_{k+1,2j-1}^n\,;$$

$\quad\quad \underline{\text{if}}\ (\ (k,j) \in \mathcal{D}^{n+1}\ )\ \underline{\text{then}}$

$$\hat{f}_{k+1,2j-1}^n \overset{(18)}{=} \hat{f}_{k,j}^n(\hat{v}_{L,2^{L-k-1}(2j-1)-p+1}^n, \ldots, \hat{v}_{L,2^{L-k-1}(2j-1)+p}^n)\,;$$

$$d_{k,j}^n\left(\hat{\boldsymbol{f}}_k^n\right) \overset{(49)}{=} \hat{f}_{k+1,2j-1}^n - \tilde{f}_{k+1,2j-1}^n\,;$$

$$d_{k,j}^{n+1} \overset{(46)}{=} d_{k,j}^n - \lambda_{k+1}\ d_{k,j}^n\left(\hat{\boldsymbol{f}}_k^n\right) + \tau\ d_{k,j}^n(\hat{\boldsymbol{s}}_k^n)\,;$$

$\quad\quad \underline{\text{else}}$

$$\hat{f}_{k+1,2j-1}^n \overset{(62)}{=} \tilde{f}_{k+1,2j-1}^n\,;$$

$$d_{k,j}^{n+1} \overset{(63)}{=} d_{k,j}^n + \tau\, d_{k,j}^n(\hat{\boldsymbol{s}}_k^n)\,;$$

$\quad\quad \underline{\text{fi}}$;

$\quad \underline{\text{od}}$;

$\underline{\text{od}}$;

11.2 Figures

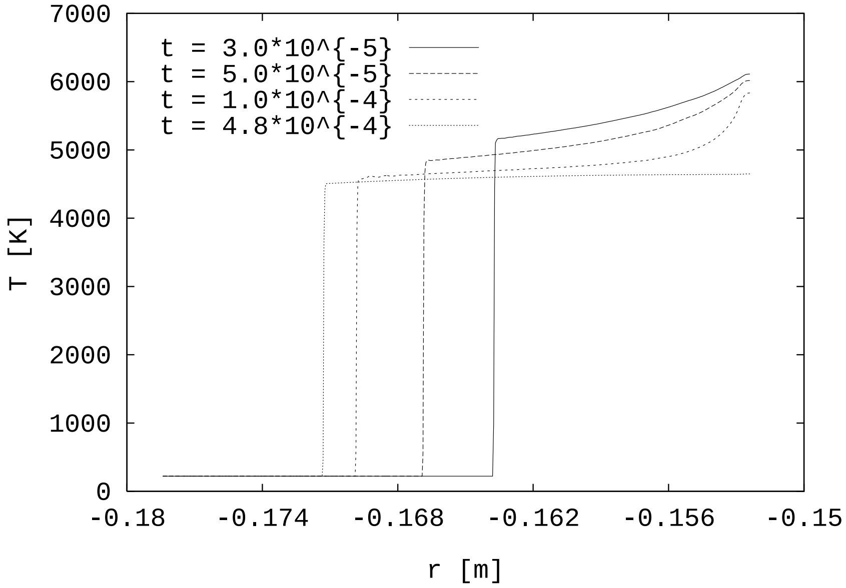

Figure 3: Temperature, Euler calculation

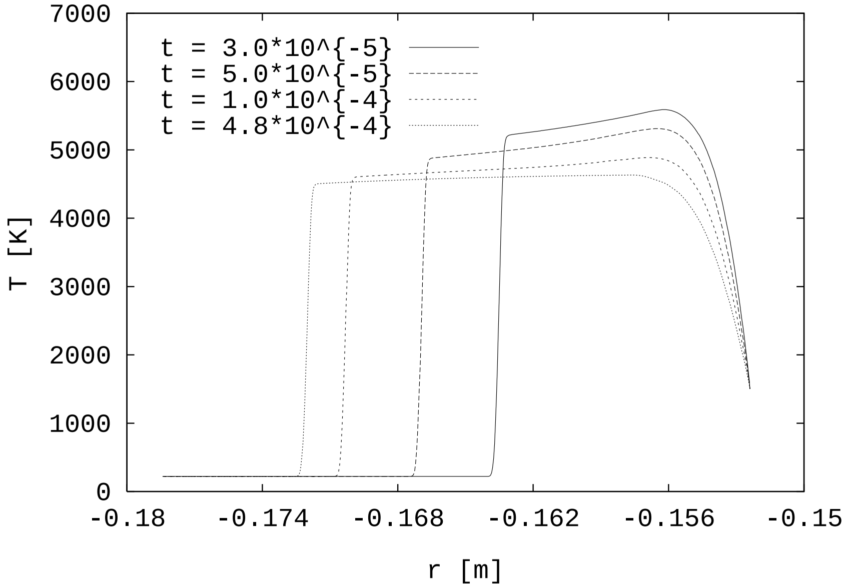

Figure 4: Temperature, Navier-Stokes calculation

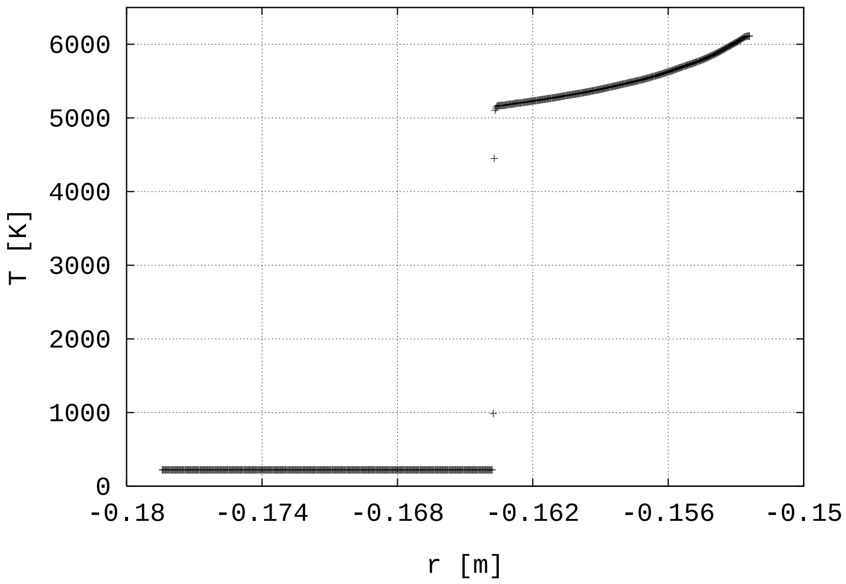

Figure 5: Temperature at time 3×10^{-5}, Euler calculation

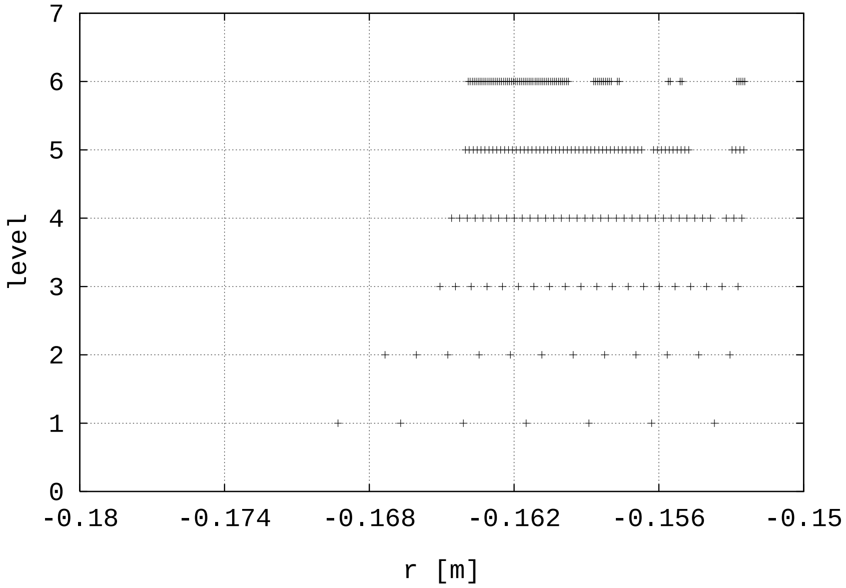

Figure 6: Significant points at time 3×10^{-5}, Euler calculation

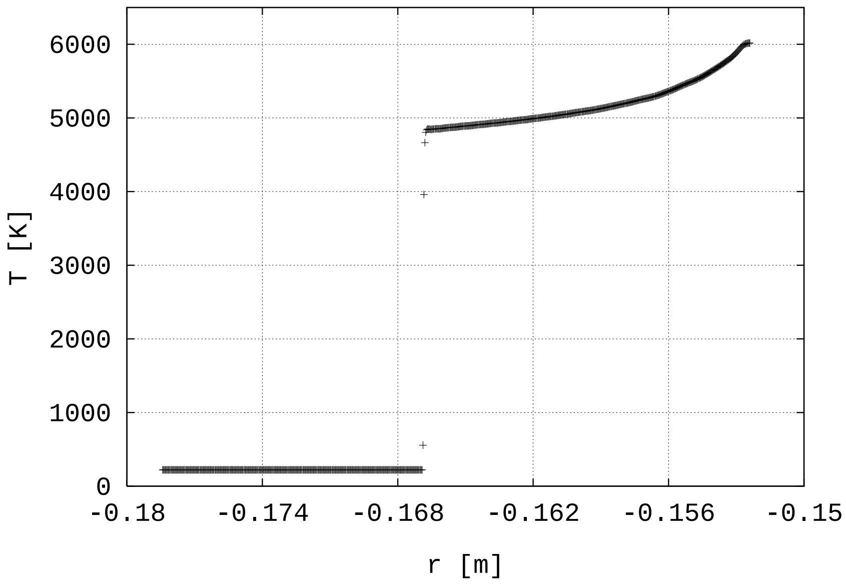

Figure 7: Temperature at time 5×10^{-5}, Euler calculation

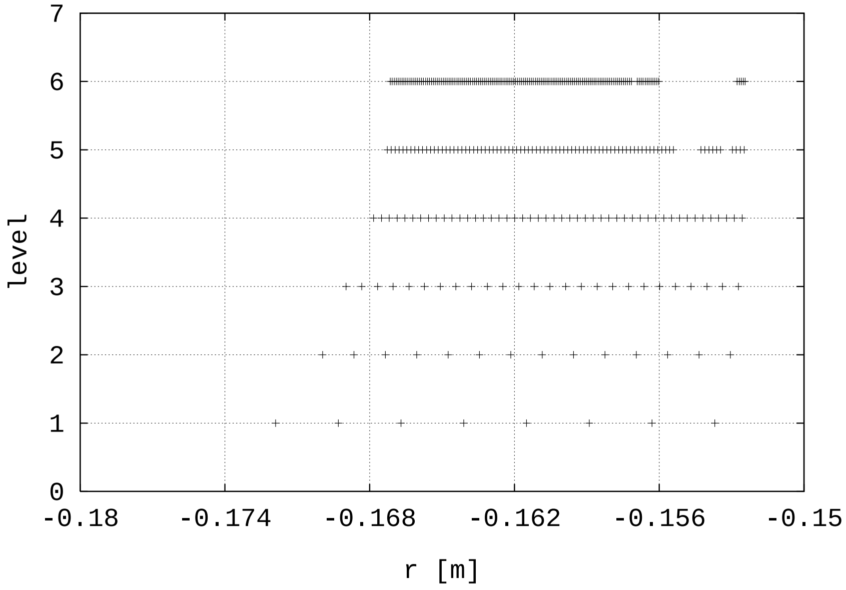

Figure 8: Significant points at time 5×10^{-5}, Euler calculation

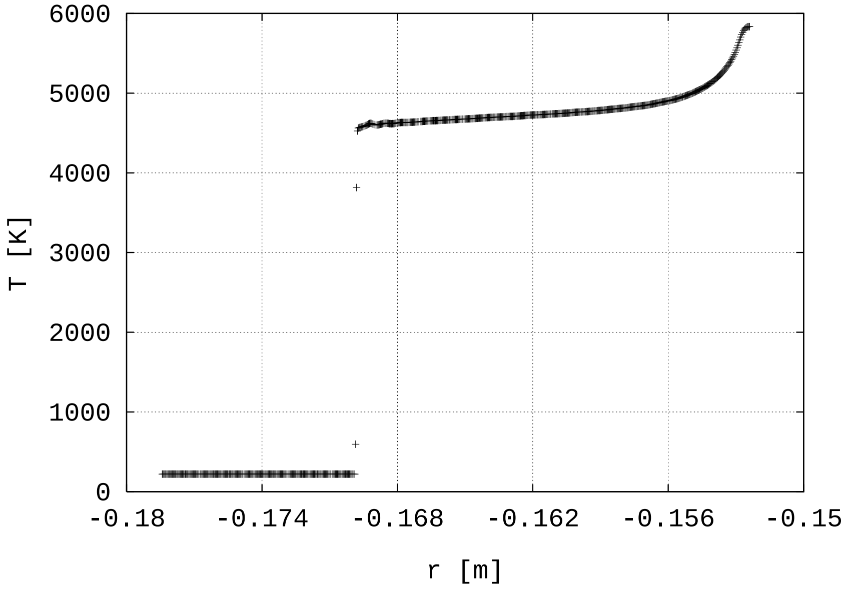

Figure 9: Temperature at time 1×10^{-4}, Euler calculation

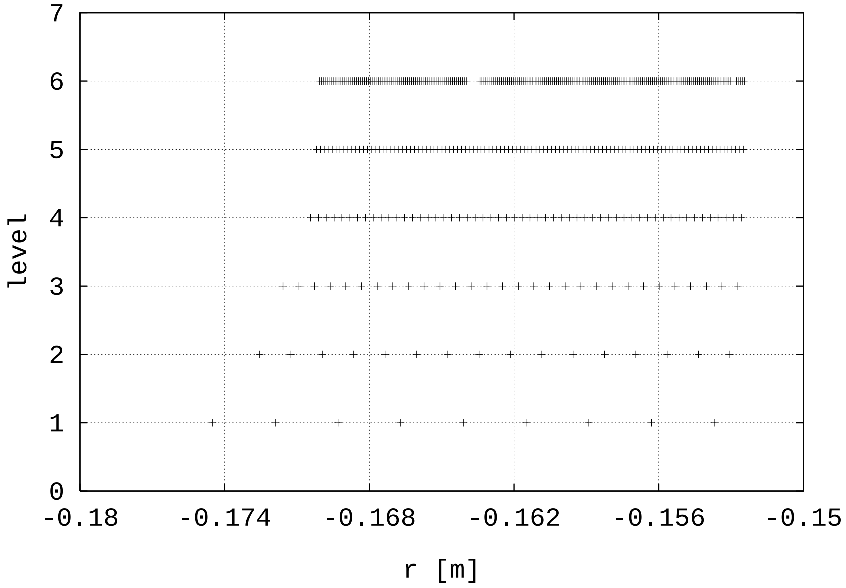

Figure 10: Significant points at time 1×10^{-4}, Euler calculation

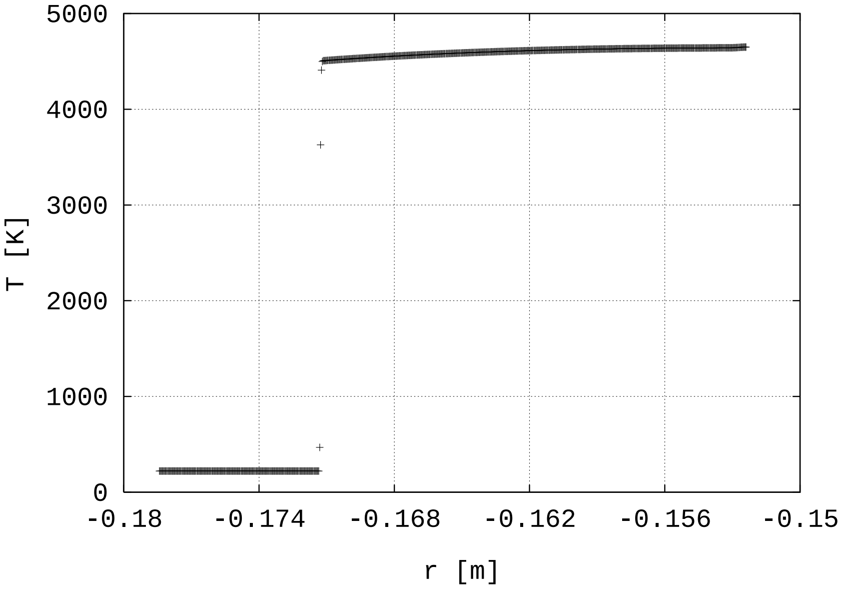

Figure 11: Temperature at time 4.5×10^{-4}, Euler calculation

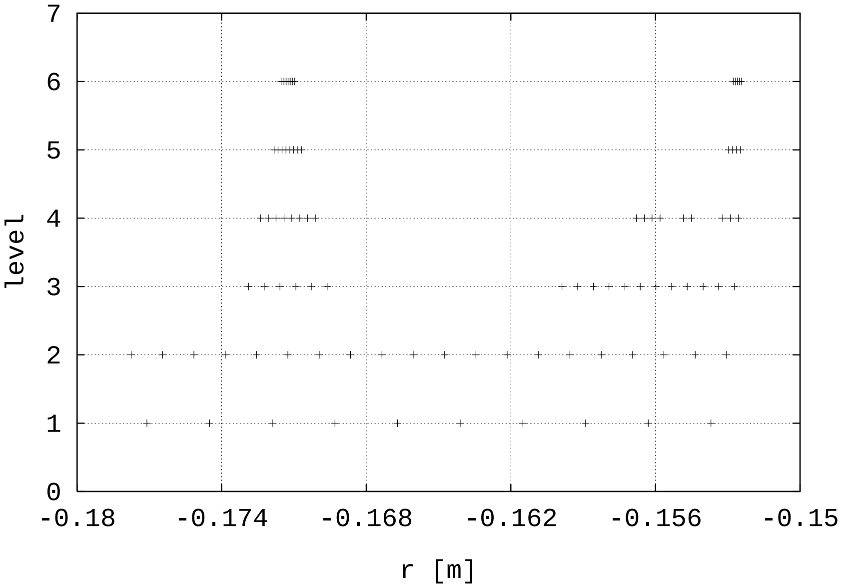

Figure 12: Significant points at time 4.5×10^{-4}, Euler calculation

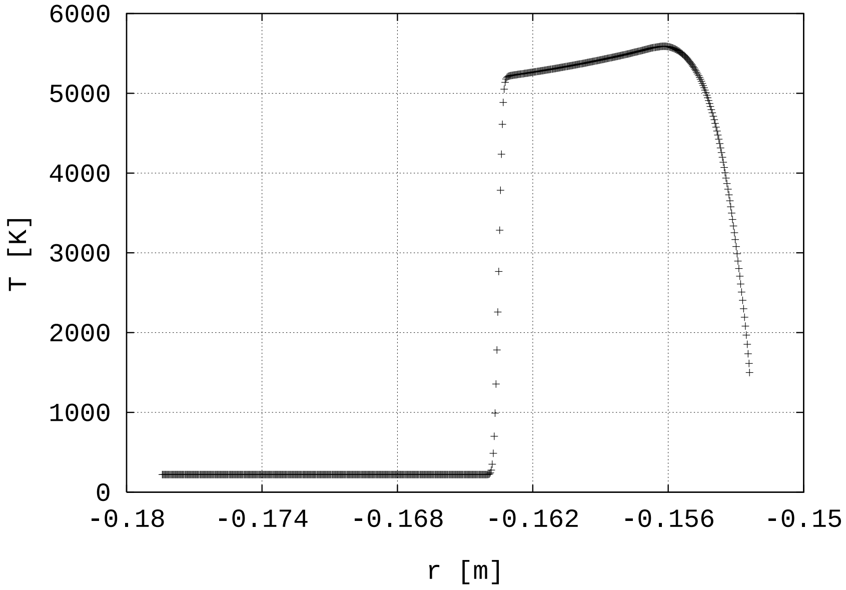

Figure 13: Temperature at time 3×10^{-5}, Navier-Stokes calculation

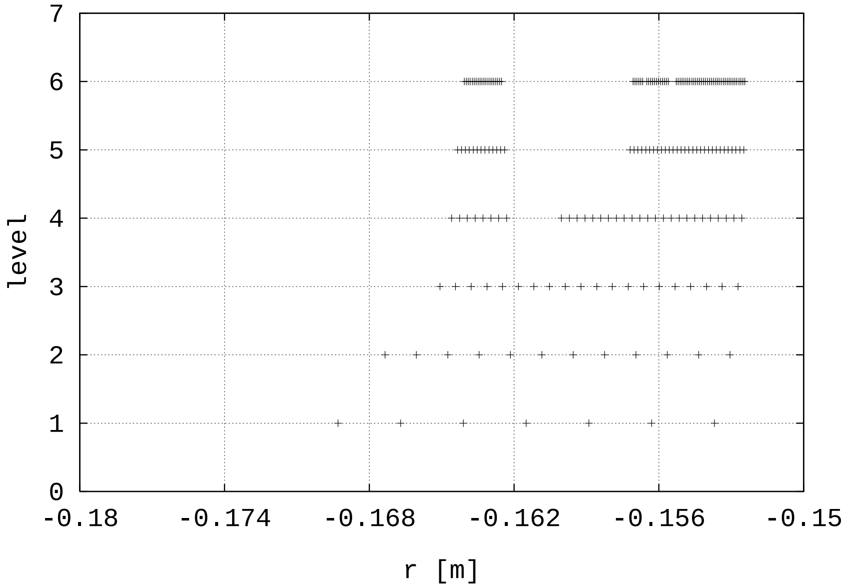

Figure 14: Significant points at time 3×10^{-5}, Navier-Stokes calculation

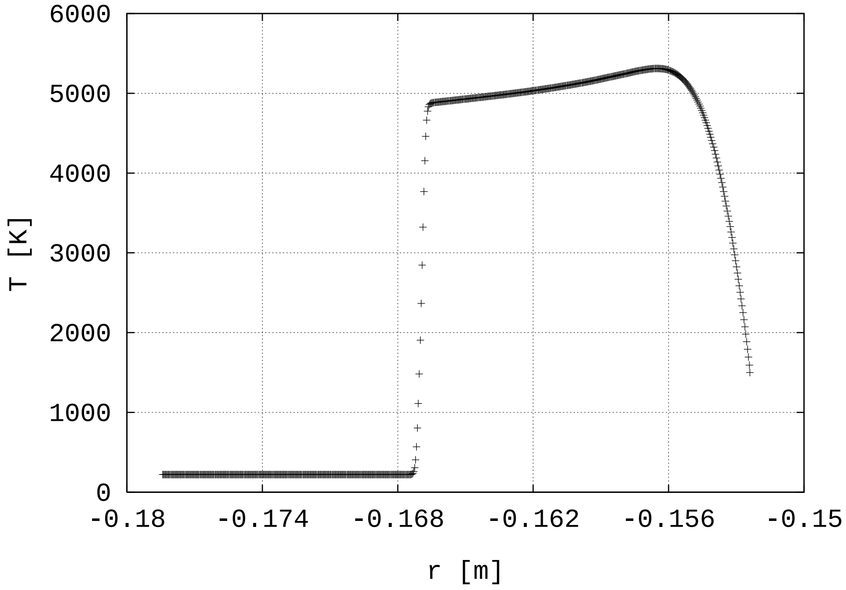

Figure 15: Temperature at time 5×10^{-5}, Navier-Stokes calculation

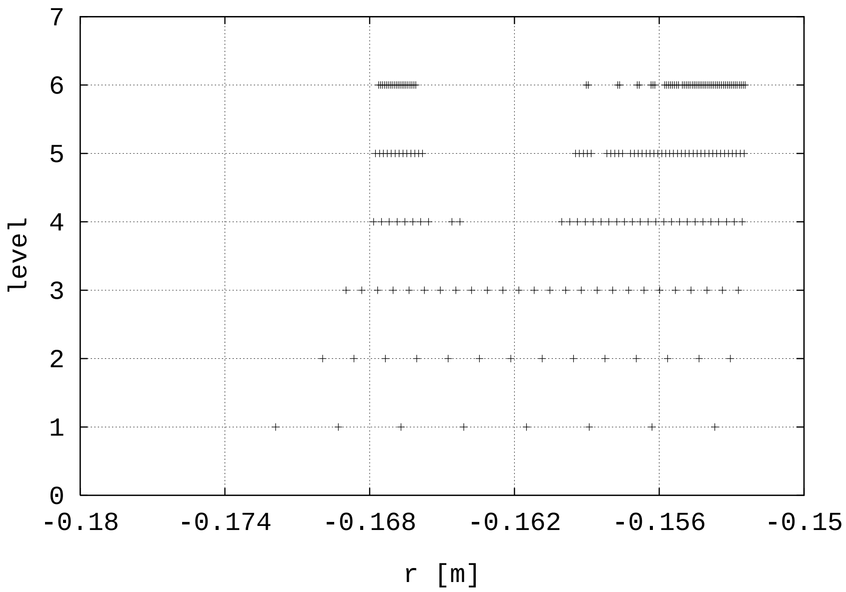

Figure 16: Significant points at time 5×10^{-5}, Navier-Stokes calculation

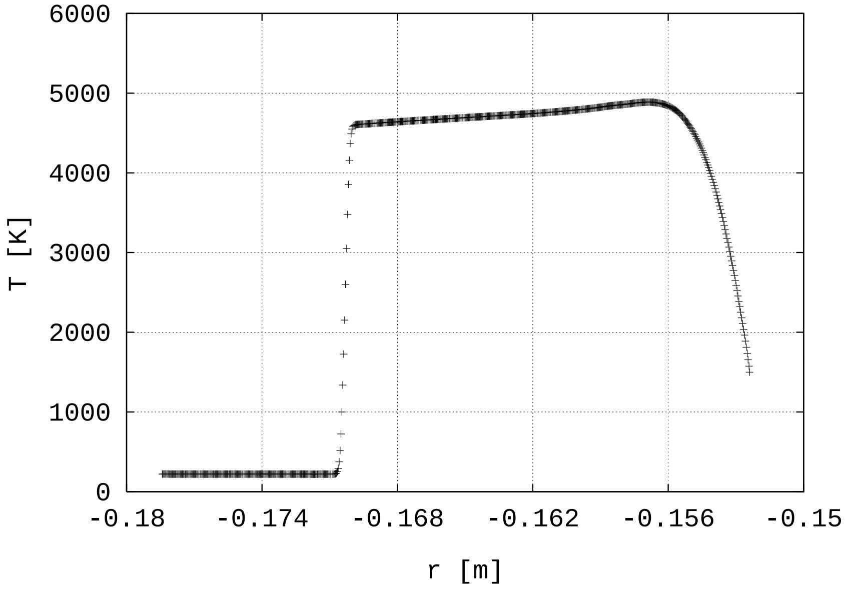

Figure 17: Temperature at time 1×10^{-4}, Navier-Stokes calculation

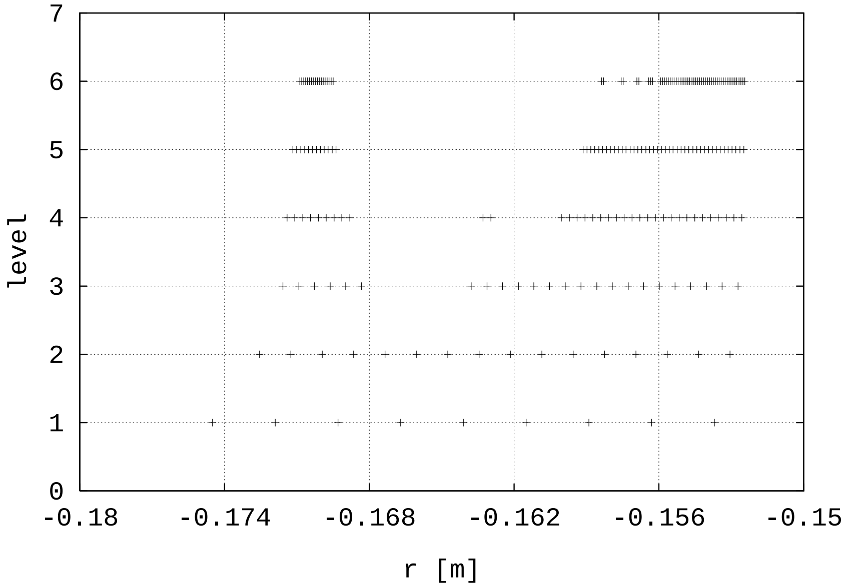

Figure 18: Significant points at time 1×10^{-4}, Navier-Stokes calculation

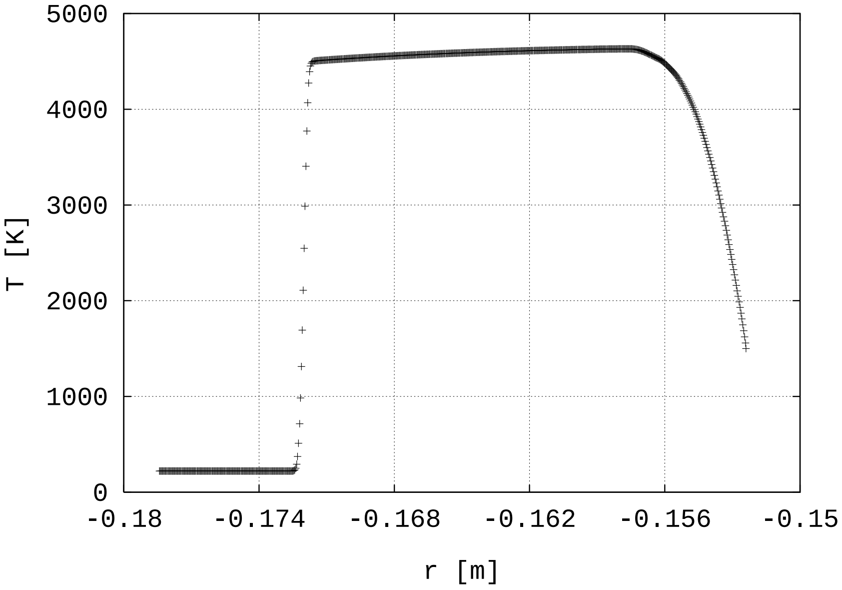

Figure 19: Temperature at time 4.5×10^{-4}, Navier-Stokes calculation

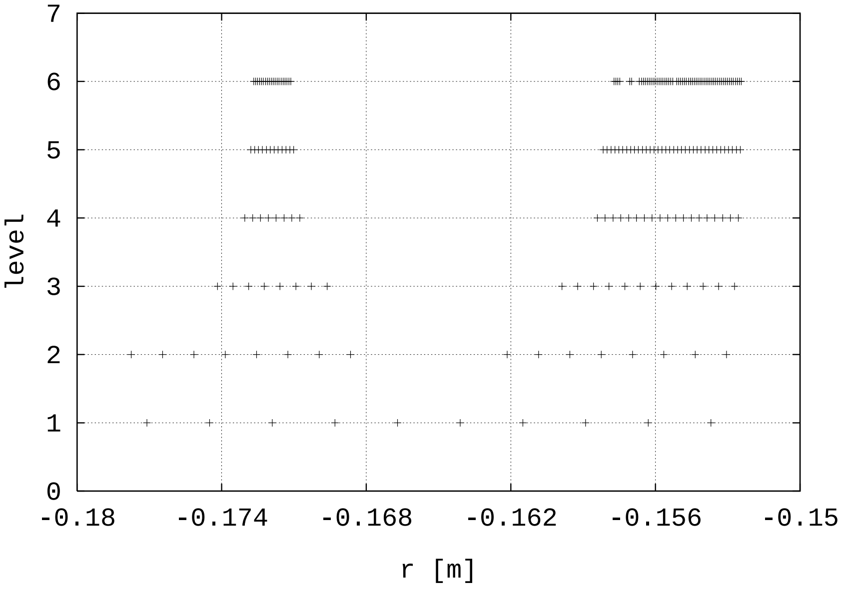

Figure 20: Significant points at time 4.5×10^{-4}, Navier-Stokes calculation

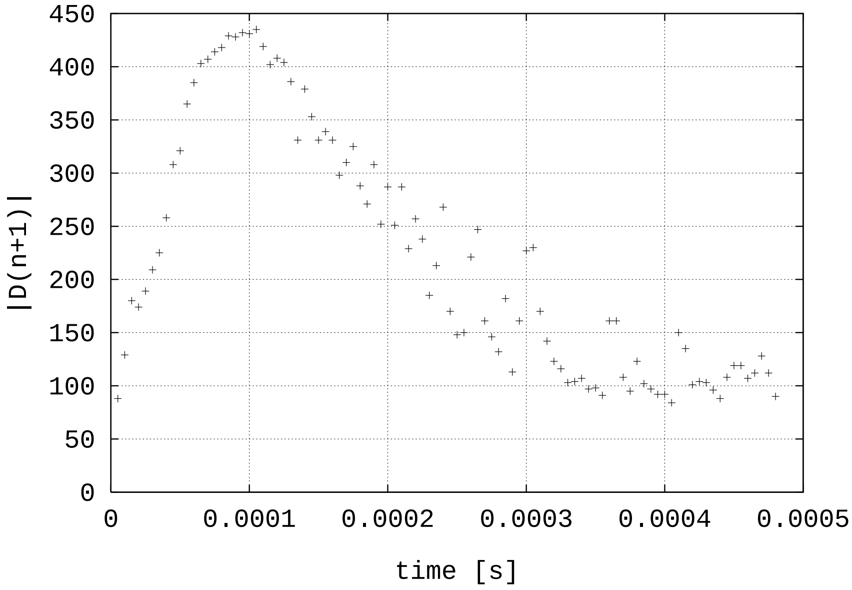

Figure 21: Number of significant points, Euler calculation

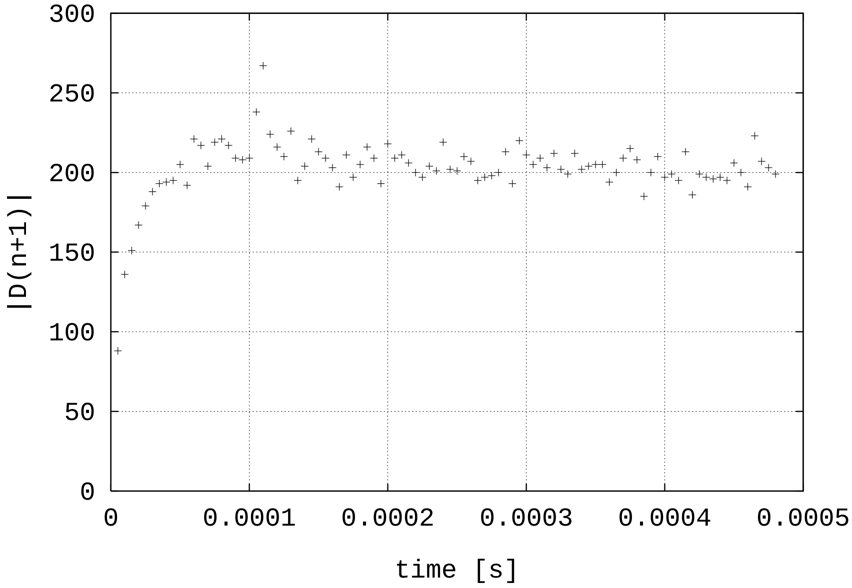

Figure 22: Number of significant points, Navier-Stokes calculation

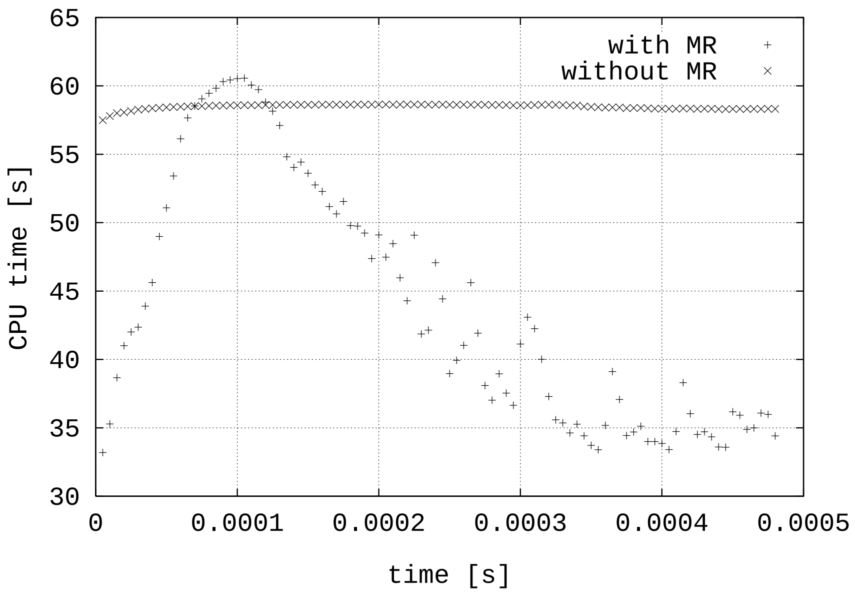

Figure 23: CPU time for 625 time steps, Euler calculation

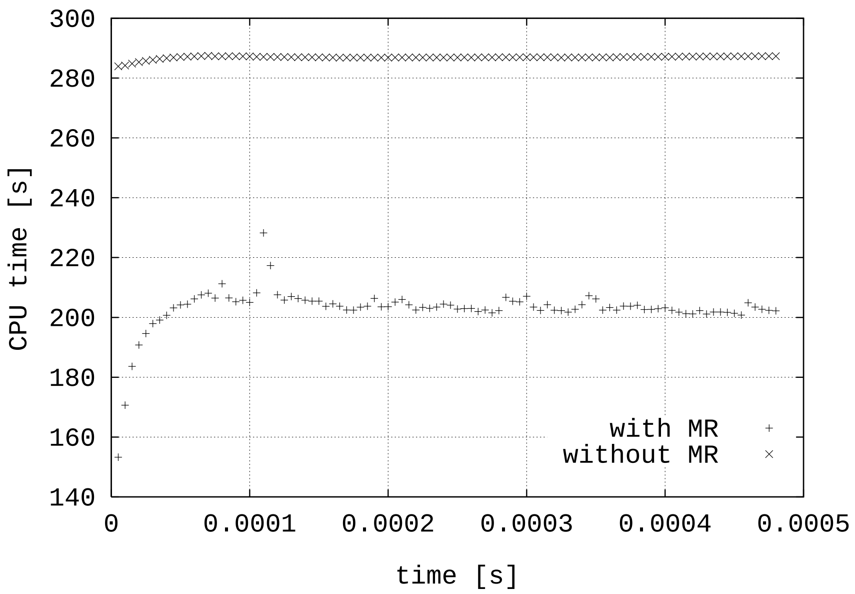

Figure 24: CPU time for 2500 time steps, Navier-Stokes calculation

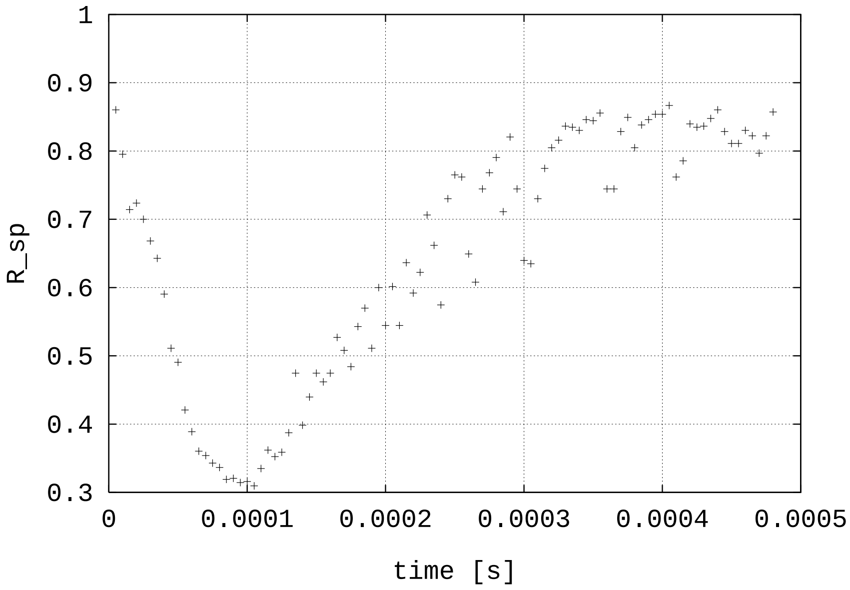

Figure 25: Reduction factor, Euler calculation

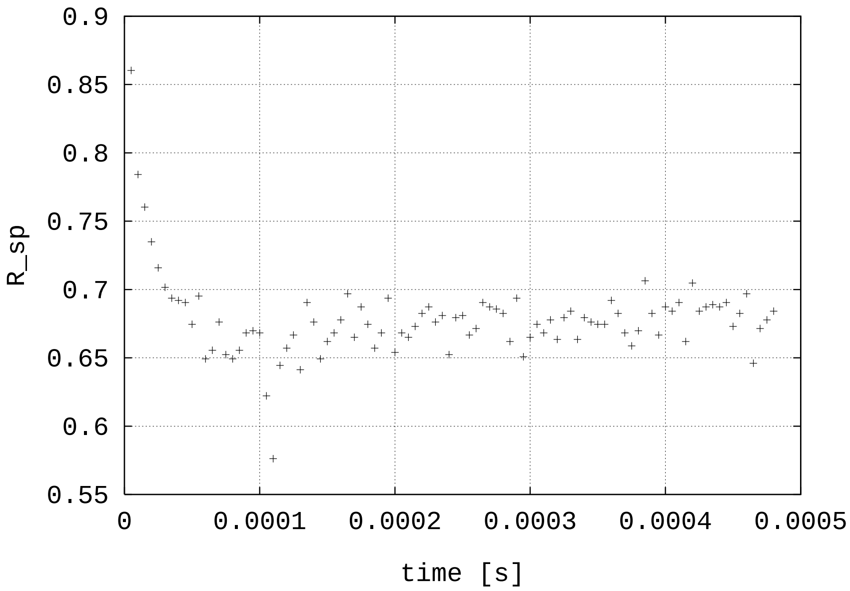

Figure 26: Reduction factor, Navier-Stokes calculation

References

[1] J. D. Anderson: Modern Compressible Flow, MacGraw-Hill Series in Mechanical Engineering, 1982

[2] J. D. Anderson: Hypersonic and High Temperature Gas Dynamics, MacGraw-Hill Series in Aeronautical and Aerospace Engineering, 1989

[3] B. L. Bihari: Multiresolution Schemes for Conservation Laws with Viscosity, J. Comp. Phys., 123, 1996, p. 207 - 225

[4] W. Dahmen: Multiresolution Methods, Lecture hold at the RWTH Aachen, 1995

[5] J. A. Fay, F. R. Riddell: Theory of Stagnation Point Heat Transfer in Dissociated Air, J. Aero. Sci. 25(2), 1958, p. 73 - 85

[6] B. Gottschlich-Müller: A Multiresolution Scheme based on Biorthogonal Wavelets for Scalar Conservation Laws in one Space Dimension, in preparation

[7] A. Harten, B. Engquist, S. Osher, S. R. Chakravarthy: Uniformly high order accurate Essentially Non-Oscillatory Schemes III, J. Comp. Phys., 71, 1987, p. 231 - 303

[8] A. Harten, S. R. Chakravarthy: Multidimensional ENO Schemes for General Geometries, ICASE Report, NASA Contractor Report NAS1-18605

[9] A. Harten: Multiresolution Algorithms for the Numerical Solution of Hyperbolic Conservation Laws, UCLA CAM Report 93-03, 1993, to appear in Comm. Pure Appl. Math.

[10] A. Klomfass, S. Müller: A quasi-onedimensional approach for hypersonic stagnation point flows, Internal Report No. 121, Institut für Geometrie und Praktische Mathematik, RWTH Aachen, 1996

[11] S. N. Kruzkov: First order quasilinear equations with several space dimensions, Math. USSR Sb., 10, 1970, p. 217 - 243

[12] R. LeVeque: Numerical Methods for Conservation Laws, ETH Zürich, Birkhäuser, 1990

[13] S. Müller: Erweiterung von ENO-Verfahren auf zwei Raumdimensionen und Anwendung auf hypersonische Staupunktprobleme, Dissertation, RWTH Aachen, Shaker-Verlag, 1993

[14] P. L. Roe: Approximate Riemann Solvers, Parameter Vectors and Difference Schemes, J. Comp. Phys., 43, 1981, p. 357 - 372

[15] B. Sjögreen: Numerical experiments with the Multiresolution Scheme for the Compressible Euler Equations, J. Comp. Phys., 117, 1995, p. 251 - 261

[16] Th. Sonar: Multivariate Rekonstruktionsverfahren zur numerischen Berechnung hyperbolischer Erhaltungsgleichungen, Habilitationsschrift, TH Darmstadt, 1995

[17] F. Schröder-Pander, Th. Sonar: Preliminary Investigations on Multiresolution Analysis on Unstructured Grids, DLR Forschungsbericht IB 223-95 A 36, 1995

[18] H. C. Yee: A class of High-Resolution Explicit and Implicit Shock-Capturing Methods, VKI Lecture Series 04-1989