Edwin van der Weide*and Magnus Svärd†

I. Background information for the SBP-SAT scheme

As is well-known, stability of a numerical scheme is a key property for a robust and accurate numerical solution. Proving stability for high-order finite-difference schemes on bounded domains is a highly non-trivial task. One successful way to obtain stability proofs is to employ so-called Summation-by-Parts (SBP) schemes with Simultaneous Approximation Terms (SAT) for imposing boundary conditions. With a simple example, we will briefly describe how stability proofs can be obtained.

Consider the scalar advection equation,

$$\begin{aligned} &u_t + a u_x = 0, \quad 0 < x < 1, \quad 0 < t \leq T \\ &a^+ u(0,t) = a^+ g_l(t) \\ &a^- u(1,t) = a^- g_r(t) \end{aligned} \tag{1}$$

where $a^+ = \max(a,0)$ and $a^- = \min(a,0)$. Furthermore, we augment the equation with initial data $u(x,0) = f(x)$, bounded in L^2. To demonstrate well-posedness, we employ the energy method.

$$\begin{aligned} &\|u\|_t^2 + a \int_0^1 u u_x \, dx = 0 \\ &\|u\|_t^2 \leq a u^2(0,t) - a u^2(1,t) \leq a^+ g_l(t)^2 - a^- g_r(t)^2 \end{aligned} \tag{2}$$

Integrating in time gives the bound

$$\|u(\cdot,T)\| \leq \|f\| + a^+ \int_0^T g_l(t)^2 \, dt - a^- \int_0^T g_r(t)^2 \, dt. \tag{3}$$

For linear PDEs, such a bound is sufficient to prove well-posedness.

Next, we turn to the SBP-SAT semi-discretization of (1). To this end, we introduce the computational grid, $x_i = ih$, $i \in \{0,1,2,...,N\}$ and $h > 0$ is the grid spacing. For the moment, we keep time continuous. With each grid point x_i, we associate a value $v_i(t)$, and define a grid function $v(t) = (v_0, v_1, v_2, ...)^T$. The SBP difference operator, D is a matrix with the following properties: $D = P^{-1}Q$ where P and Q are two matrices; $P = P^T > 0$ and $Q + Q^T = B = \text{diag}(-1,0,...,0,1)$. The matrix P can be used to define a weighted l^2 equivalent norm as $\|v\|^2 = v^T P v$. We will also need the vectors $e_0 = (1,0,0...,0)^T$ and $e_N = (0,...,0,1)^T$.

Let w denote a smooth function and define a grid function $\bar{w} = (w(x_0),...,w(x_N))^T$ and $\bar{w}_x = (w_x(x_0),...,w(x_N))^T$. It turns out that the SBP property precludes the accuracy of D to be uniform in space. We have

$$D\bar{w} = \bar{w}_x + \bar{T}$$

where $\bar{T}$ is the truncation error. In general, it takes the form,

$$\bar{T}^T = (\mathcal{O}(h^s),...,\mathcal{O}(h^s),\mathcal{O}(h^p),...,\mathcal{O}(h^p),\mathcal{O}(h^s),...,\mathcal{O}(h^s)). \tag{4}$$

where $s < p$ and the lower accuracy is confined to a few (finite) number of points close to the boundary. SBP operators exist with various orders of accuracy, [1]. In particular, if P is a diagonal matrix, there are SBP operators with p even and $p \leq 8$, and $s = p/2$. If P is allowed to have off-diagonal elements for a few points near the boundary $s = p - 1$ can be achieved.

*Department of Engineering Technology, University of Twente, the Netherlands, e-mail: e.t.a.vanderweide@ctw.utwente.nl
†School of Mathematics, University of Edinburgh, e-mail: Magnus.Svard@ed.ac.uk

Using the SBP operators, we now define a semi-discrete scheme for (1).

$$v_t + aDv = \sigma_l a^+ P^{-1} e_0 (v_0 - g_l(t)) + a^- \sigma_r P^{-1} e_N (v_N - g_r(t))$$

The right-hand side are the SAT:s, which impose the boundary conditions weakly. (Originally proposed in [2].) $\sigma_{l,r}$ are two scalar parameters, to be determined by the stability analysis. Multiplying by $2v^T P$, we obtain

$$\|v\|_t^2 - a(v_0^2 - v_N^2) = 2\sigma_l a^+ v_0 (v_0 - g_l(t)) + 2a^- \sigma_r v_N (v_N - g_r(t)) \tag{5}$$

For stability, it is sufficient to obtain a bound with $g_{l,r} = 0$. In that case, it is easy to see that we must require $\sigma_l \leq -1/2$ and $\sigma_r \geq 1/2$ to obtain a bounded growth of $\|v\|$. More generally, allowing boundary data to be inhomogeneous when deriving a bound leads to *strong stability*. (See [3]. The benefit of proving strong stability as opposed to stability is that less regularity in the boundary data is required.) For strong stability, it can be shown that $\sigma_{l,r}$ must satisfy $\sigma_l < -1/2$ and $\sigma_r < 1/2$, i.e., strict inequalities. As an example, the choice $\sigma_l = -1, \sigma_r = 1$ leads to

$$\|v\|_t^2 - a(v_0^2 - v_N^2) = -2a^+ v_0 (v_0 - g_l(t)) + 2a^- v_N (v_N - g_r(t))$$

or

$$\|v\|_t^2 \leq -a^+ (v_0 - g)^2 + a^+ g_l(t)^2 + a^- (v_N - g_r(t)^2) - a^- g^2 \tag{6}$$

If $v_0 = g_l, v_N = g_r$, (6) is the same as (2), but this is not the case and the additional terms add a small damping to the boundary. Upon integration of (6) in time, an estimate corresponding to (3) is obtained. We also remark that the SAT terms are accurate as they do not contribute to a truncation error in the scheme. Furthermore, semi-discrete stability guarantees stability of the fully discrete problem obtained by employing Runge-Kutta schemes in time, [4].

The above example, demonstrates the general procedure for obtaining energy estimates for an SBP-SAT scheme. Naturally, for systems of PDEs, in 3-D with stretched and curvi-linear multi-block grids, and with additional parabolic terms, the algebra for proving stability becomes more involved. However, the resulting schemes are still fairly straightforward to use. For the linearized Euler and Navier-Stokes equations, semi-discrete energy estimates have been derived. (See [5–7] and references therein.) Different boundary types, including far-field, walls and grid block interfaces are included in the theory. For flows with smooth solutions, linear stability implies convergence as the grid size vanish. (See [8].)

II. Code description

As a general code is still under development, specialized codes for the test cases computed have been written. These codes assume a single block grid and do not have parallel capabilities, hence they are relatively easy to modify for testing purposes. Due to the fact that these codes can only be used for one specific test case their efficiency is expected to be higher than for a general purpose code.

The discretization schemes used are finite difference SBP-SAT schemes, see section I, of order 2 to 5. Thanks to the energy stability property of these schemes no or a significantly reduced amount of artificial dissipation is needed compared to schemes which do not posses this (or a similar) property. This leads to a higher accuracy of the numerical solutions. The set of nonlinear algebraic equations is solved using the nonlinear solver library of PETSc [9]. This library requires the Jacobian matrix of the spatial residual, which is computed via dual numbers [10] and appropriate coloring of the vertices of the grid, for which the PETSc routines are used. Initial guesses are obtained via grid sequencing, where appropriate. The solution of the linear systems needed by PETSc's nonlinear solution algorithm is obtained either by a direct solver or by Block ILU preconditioned GMRES, depending on the case. It was found that this approach resulted in a speedup of at least a factor of 100 compared to standard explicit time integration techniques.

For the postprocessing standard commercially available software, such as Tecplot, is used. Grid adaption has not been carried out.

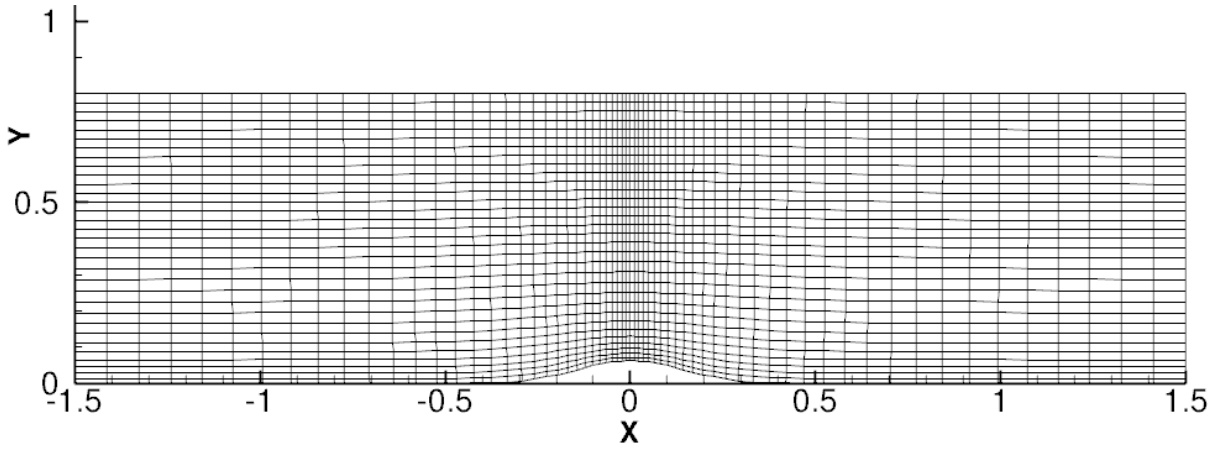

Figure 1. 64×32 **grid for test case C1.1. This grid is obtained from the fine grid by applying a regular coarsening twice**

III. Case summary

In this section the test cases computed will be presented in each of the subsections. All the results have been obtained on a Linux work station running Ubuntu 10.04 with an Intel i7-2600 CPU running at 3.4 GHz, with 8 Mb of cache. The machine contains 16 Gb of RAM memory with an equivalent amount of swap. Running the Taubench on this machine led to a CPU time of 5.59 seconds (average over 4 runs).

III.A. Case C1.1

Test case C1.1, the internal inviscid flow over a smooth bump, has been computed on three structured grids containing 64×32, 128×64 and 256×128 cells respectively. The finest grid is obtained by applying elliptic smoothing using the Pointwise software. The point distribution on the boundary is created using the hyperbolic tangent stretching from Vinokur [11]. The coarser grids are obtained by deleting every other grid line from the fine grid. The coarsest grid used is shown in figure 1. Numerical solutions are obtained using the nonlinear solver from PETSc, where a direct solver is used to solve the linear system of equations. For

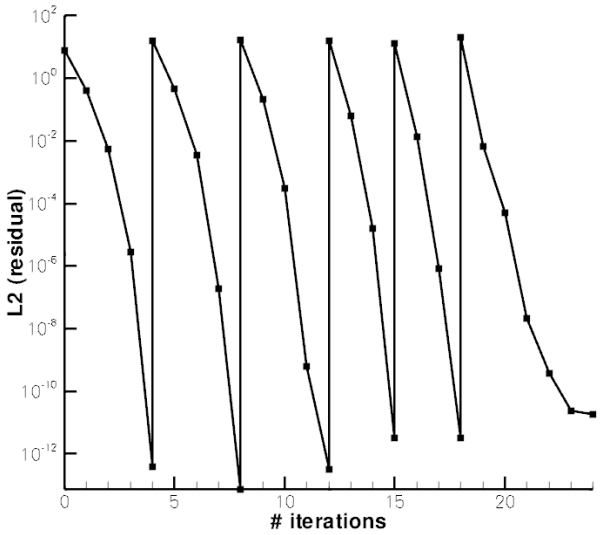

Figure 2. Convergence history (using grid sequencing) for the 5^{th} **order scheme on the fine grid (**256×128 **cells) for test case C1.1. On the coarser grids a** 2^{nd} **order scheme is used. The total amount of CPU time needed is 91 times the cost of the Taubench.**

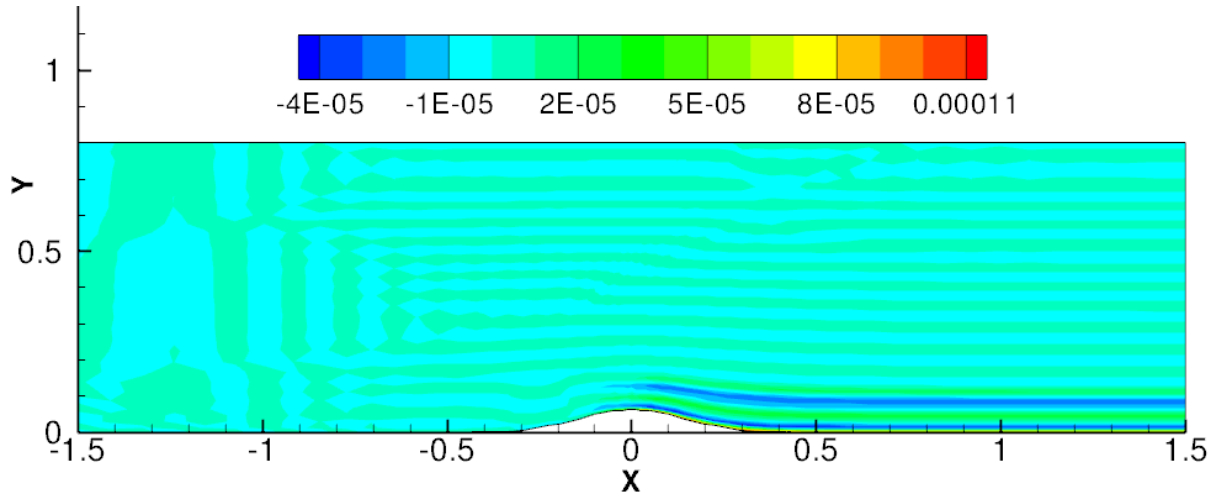

Figure 3. Contour plot of the entropy error for the 5^{th} order scheme on the coarse grid (64×32 cells) for test case C1.1.

this case no artificial dissipation is needed for the discretization itself. However, a small amount of artificial dissipation must be added to the Jacobian matrix to keep the iteration process stable. This of course does not have any influence on the converged solution. A typical convergence history for the 5^{th} order scheme on the finest (256×128 cells) grid is shown in figure 2, which clearly shows the grid sequencing as well as the convergence to machine zero. On the coarser grids a second order discretization is used to reduce the computational costs. The total amount of CPU time required for this case is 510 seconds, which corresponds to 91 times the cost of the Taubench. As the exact solution for this case is isentropic, the deviation from the exact entropy can be used as a measure of the error, i.e.

$$\text{error} = \frac{\exp(\mathrm{s}) - \exp(\mathrm{s}^{\text{exact}})}{\exp(\mathrm{s}^{\text{exact}})}, \quad \exp(\mathrm{s}) = \frac{\mathrm{p}}{\rho^{\gamma}}. \tag{7}$$

The contour plot of this error on the coarsest grid for the 5^{th} order scheme is shown in figure 3, while the L^2 norm of this quantity, defined as

$$L^2_{\text{error}} = \left[\frac{\sum_{i=1}^{N} \text{error}_{\mathrm{i}} \, |\mathrm{J_i}|}{\sum_{i=1}^{N} |J_i|} \right]^{1/2},$$

$|J_i|$ being the determinant of the metric Jacobian, is found in table 1 for the three different grids used. In this table also the order of the scheme relative to the previous grid is given. The 5^{th}-order scheme, has 4^{th}-order truncation errors close to the boundaries. Since the interior scheme is more accurate and since the problem is hyperbolic, the global convergence rate is expected to be 5. (See [3]). However, this convergence rate only applies to the solution variables. The entropy, being deduced from the solution variables will not enjoy the same convergence rate. Instead, with 4^{th}-order truncation errors confined to vanishing strips close

Grid	L^2 norm entropy error	Order of accuracy
64×32	1.16e-5	-
128×64	2.60e-7	5.47
256×128	1.22e-8	4.41

Table 1. L^2 norms of the entropy error for test case C1.1 and the actual order of accuracy on the three grids used. Discretization scheme is the 5^{th} order scheme.

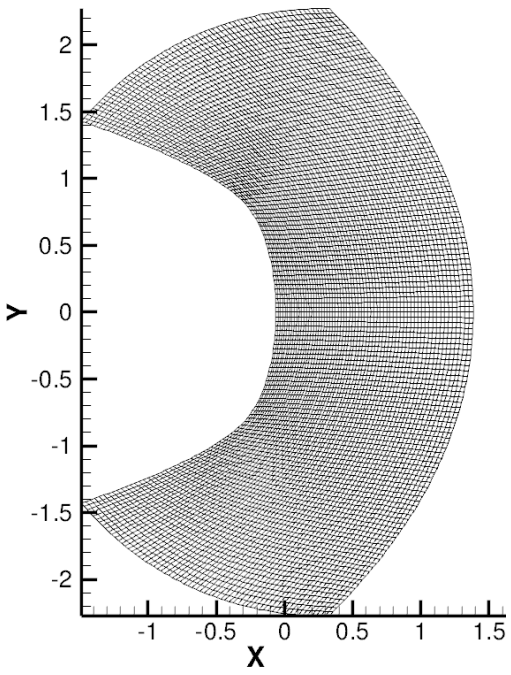

Figure 4. 128×64 **grid for test case C1.2. This grid is obtained from the fine grid by applying a regular coarsening.**

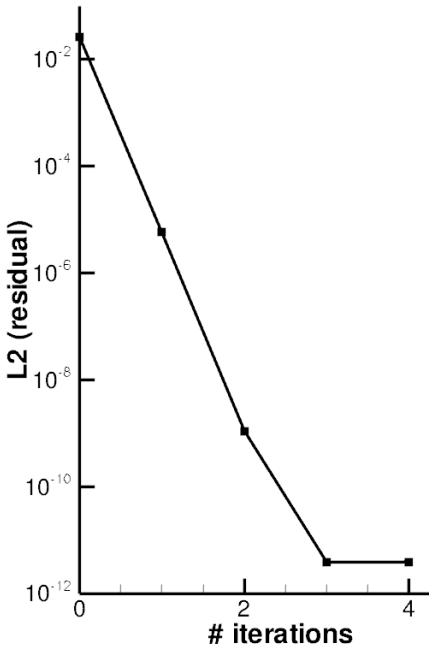

Figure 5. Convergence history for the 5^{th} **order scheme on the fine grid** (256×128 **cells) for test case C1.2. The total amount of CPU time needed is 61.7 times the cost of the Taubench.**

to the boundary, it is easy to see that in L^2 the convergence rate should be 4.5, which indeed is close to what we record in our grid refinement study.

III.B. Case C1.2

Test case C1.2, the transonic Ringleb flow, has been computed on two structured grids containing 128×64 and 256×128 cells respectively. The finest grid is obtained by applying elliptic smoothing using the Pointwise software. The point distribution on the boundary is created using the hyperbolic tangent stretching from Vinokur [11]. The coarse grid is obtained by deleting every other grid line from the fine grid. The coarse grid used is shown in figure 4.

It was found that at least a 3^{rd} order discretization must be used to obtain a stable solution. For the 2^{nd} order scheme the computation of the metric terms is not accurate enough to represent the boundary correctly. Numerical solutions are obtained using the nonlinear solver from PETSc, where a direct solver is used to solve the linear system of equations. In contrast to test case C1.1, a small amount of artificial dissipation is needed for stability. A typical convergence history for the 5^{th} order scheme on the finest (256×128 cells) grid is shown in figure 5, which clearly shows the convergence to machine zero. As a restart is carried out from the exact solution, no grid sequencing needs to be used here. The total amount of CPU time required is 345 seconds, which corresponds to 61.7 times the cost of the Taubench. Contour plots of the Mach number and entropy error, defined in equation (7), for the fine grid are shown in figures 6 and 7 respectively. The L^2 norm of the entropy error is shown in table 2 for the three different grids used. In this table also the order of the scheme relative to the previous grid is given.

Grid	L^2 norm entropy error	Order of accuracy
128×64	8.31e-7	-
256×128	3.12e-8	4.73

Table 2. L^2 **norms of the entropy error for test case C1.2 and the actual order of accuracy on the two grids used. Discretization scheme is the** 5^{th} **order scheme.**

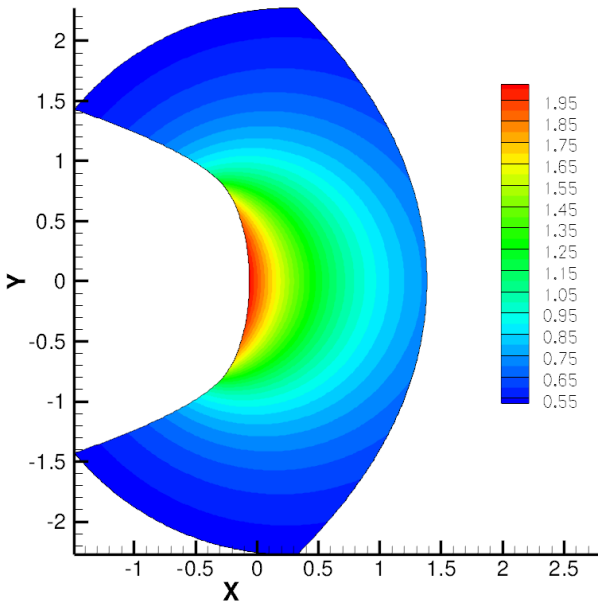

Figure 6. Contour plot of the Mach number for the 5^{th} order scheme on the fine grid (256×128 cells) for test case C1.2.

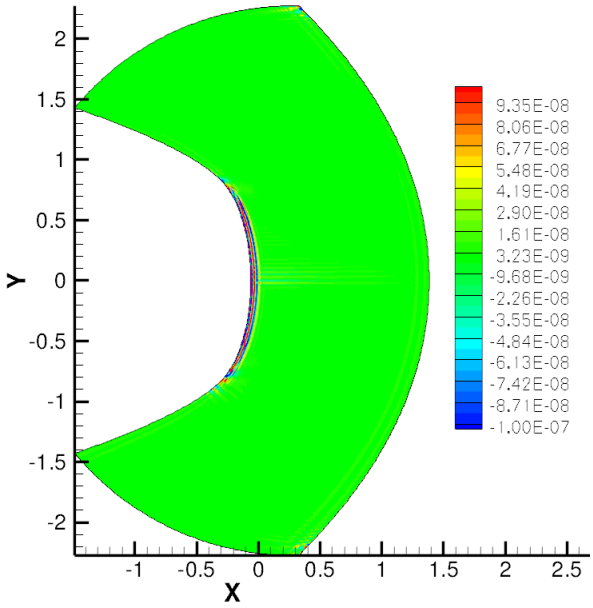

Figure 7. Contour plot of the entropy error for the 5^{th} order scheme on the fine grid (256×128 cells) for test case C1.2.

III.C. Case C1.3

This test case considers three flow conditions for the NACA 0012 airfoil of which only condition a), subsonic inviscid flow with $M_\infty = 0.5$ and 2 degrees angle of attack, has been considered so far. The first requirement is to determine the farfield location whose effect on the lift and drag coefficients is less than 0.01 counts. This task has been carried out on an O-grid topology, see figure 8, which is generated using the hyperbolic grid generation capabilities of Pointwise [12]. This grid contains 256 cells along the airfoil while the number of cells normal to the airfoil depends on the location of the farfield. Three farfield locations have been considered, namely 100 chord lengths away (96 cells), 20,000 chord lengths away (120 cells) and 130,000 chord lengths away (128 cells). A typical convergence history for the 5^{th} order scheme on the 256×128 grid is shown in figure 9. This case required a CPU time of 522.5 seconds, which is 93.5 times the cost of the Taubench. Contour plots of the Mach number and entropy error, defined in equation (7), are shown in figures 10 to 12. Due to the presence of large gradients in especially the leading and trailing edge region of the flow, some artificial dissipation is needed for stability.

It is clear from especially figure 12 that the main source of entropy error is the trailing edge of the airfoil. The reason is that the geometry is not smooth in this region, which results in the large entropy error originating from the trailing edge. Additional tests with a C-grid topology showed exactly the same behavior, hence the O-grid topology is not the cause of this error. As a consequence, the schemes will not obtain the design accuracy, at least not for the inviscid case, and especially the drag convergence will only be first order. This is not shown in this abstract, but will be shown in the final results for the workshop.

Two options for the farfield boundary condition have been used, where the difference between the two formulations is whether or not a vortex correction is applied. From linear theory it is clear that the induced velocities from the lift generating vortex are inversely proportional to the distance from the airfoil. When an accuracy of 1.e-6 is required for the lift and drag coefficients, it can therefore be expected that the farfield boundary must be located extremely far away to accomplish this when no vortex correction is applied. This is confirmed by results shown in table 3. In this table the lift and drag coefficients for the NACA0012 are shown for $M_\infty = 0.5$ and 2 degrees angle of attack as a function of the distance of the farfield location. It is clear that only when the farfield is located more than 100,000 chords away the lift coefficient is close the desired accuracy of 1.e-6 when no vortex correction is applied. When a vortex correction is applied a farfield location of 100 chords suffices.

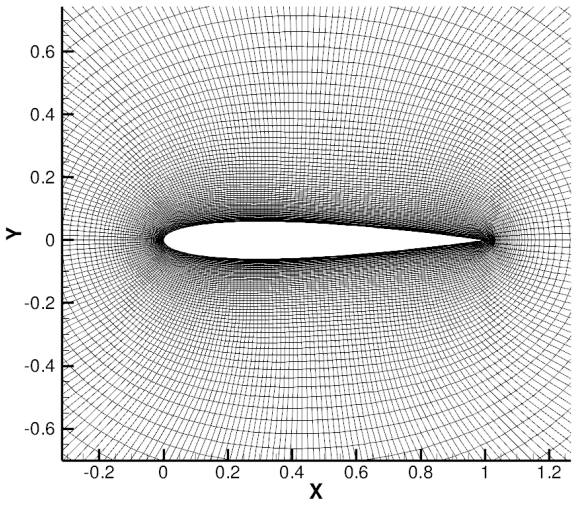

Figure 8. O-grid for the NACA0012. The grid contains 256 cells in the direction along the airfoil and 96 to 128 cells n the direction normal to the airfoil, depending on the farfield location.

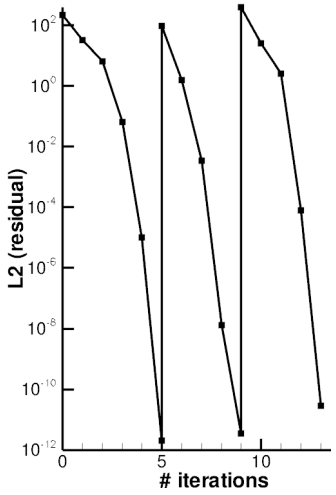

Figure 9. Convergence history for the 5^{th} order scheme on the 256×128 grid for test case C1.3. The total amount of CPU time needed is 93.5 times the cost of the Taubench.

Farfield location	C_l no vortex correction	C_l with vortex correction	C_d no vortex correction	C_d with vortex correction
100	0.285430	0.286700	8.84819e-05	4.43426e-05
20,000	0.286693	0.286699	4.45542e-05	4.43381e-05
130,000	0.286698	0.286699	4.43730e-05	4.43379e-05

Table 3. C_l and C_d values for the NACA0012, $M_\infty = 0.5$ and 2 degrees angle of attack, for different locations of the farfield. Discretization scheme is the 5^{th} order scheme.

References

[1] Strand, B., "Summation by Parts for Finite Difference Approximations for d/d x," *J. Comput. Phys.*, Vol. 110, 1994.

[2] Carpenter, M. H., Gottlieb, D., and Abarbanel, S., "Time-stable boundary conditions for finite-difference

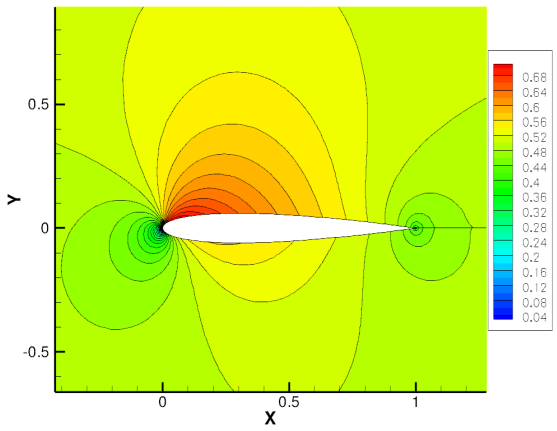

Figure 10. Mach number isolines for the 5^{th} order scheme for the NACA0012 on the 256×128 grid.

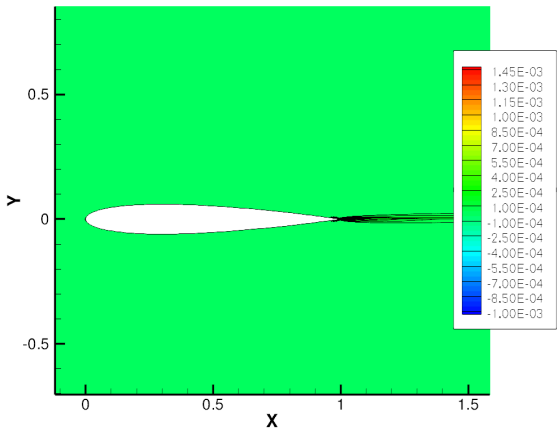

Figure 11. Isolines of the entropy error for the 5^{th} order scheme for the NACA0012 on the 256×128 grid.

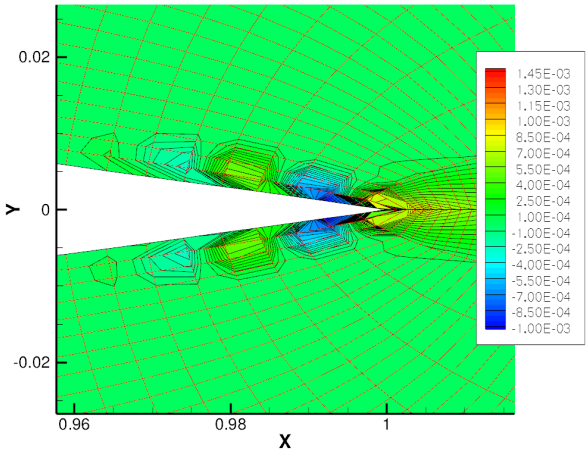

Figure 12. Isolines of the entropy error for the 5^{th} order scheme for the NACA0012 on the 256×128 grid. Zoom near the trailing edge. The grid lines are shown in red and the isolines in black.

schemes solving hyperbolic systems: Methodology and application to high-order compact schemes," *J. Comput. Phys.*, Vol. 111(2), 1994.

[3] Gustafsson, B., Kreiss, H.-O., and Oliger, J., *Time dependent problems and difference methods*, John Wiley & Sons, Inc., 1995.

[4] Kreiss, H.-O. and Wu, L., "On the stability definition of difference approximations for the initial boundary value problem," *Applied Numerical Mathematics*, Vol. 12, 1993, pp. 213–227.

[5] Svärd, M., Carpenter, M., and Nordström, J., "A stable high-order finite difference scheme for the compressible Navier-Stokes equations, far-field boundary conditions," *Journal of Computational Physics*, Vol. 225, 2007, pp. 1020–1038.

[6] Svärd, M. and Nordström, J., "A stable high-order finite difference scheme for the compressible Navier-Stokes equations, no-slip wall boundary conditions," *J. Comput. Phys.*, Vol. 227, 2008, pp. 4805–4824.

[7] Nordström, J., Gong, J., van der Weide, E., and Svärd, M., "A stable and conservative high order multi-block method for the compressible Navier-Stokes equations," *J. Comput. Phys.*, Vol. 228, 2009, pp. 9020–9035.

[8] Strang, G., "Accurate partial difference methods II. Non-linear problems," *Num. Math.*, Vol. 6, 1964, pp. 37–46.

[9] Balay, S., Brown, J., Buschelman, K., Eijkhout, V., Gropp, W., Kaushik, D., Knepley, M., McInnes, L. C., Smith, B., and H.Zhang, "PETSc Users Manual, Revision 3.2," Tech. rep., Argonne National Laboratory, 2011.

[10] Fike, J., Jongsma, S., Alonso, J., and van der Weide, E., "Optimization with Gradient and Hessian Information Calculated Using Hyper-Dual Numbers," *AIAA Paper 2011-3807*, 2011.

[11] Vinokur, M., "On One-Dimensional Stretching Functions for Finite-Difference Calculations," *Journal of Computational Physics*, Vol. 50(2), May 1983, pp. 215 – 234.

[12] "Pointwise User Manual," Tech. rep., www.pointwise.com, 2011.