Sparse-Grid Solution of the Steady Euler Equations of Gas Dynamics

B. Koren [1], **P.W. Hemker** [2] and **C.T.H. Everaars** [3]

Abstract. A significant difficulty of standard multigrid methods for 3D problems, when compared to application to 2D problems, is that the requirements to be imposed on the smoother are much more severe. As a remedy, we investigate three different possibilities of multiple semi-coarsening: full, sparse and semi-sparse. Numerical results are presented for a standard 3D transonic test case. The good parallel computing properties of the sparse-grid and the semi-sparse-grid approaches are also investigated. The first speed-up results are promising. The paper contributes to the state-of-the-art in efficiently solving 3D fluid-flow equations.

1 INTRODUCTION

With standard multigrid methods, the total amount of work on the coarse grids is relatively smaller in the 3D case than in the 2D case. However, the reverse side is that in 3D only a relatively small amount of error components can be annihilated by the coarse-grid corrections. When cells are used as grid elements, in 3D, standard coarsening implies restriction from each set of $2 \times 2 \times 2$ cells to a single cell only. Because the set of eight cells can support more high-frequency errors than the two-dimensional 2×2-set, 3D standard multigrid imposes stronger requirements on the smoother than 2D standard multigrid. Standard multigrid may not perform satisfactory for 3D generalizations of 2D problems, for which it does perform well. A fix might be found in deriving a more powerful smoother, keeping the other components of the numerical method the same. A more natural remedy is not to apply standard, i.e., full coarsening, but to use multiple semi-coarsening instead (Figure 1). When multiple semi-coarsening is applied to solve a system of equations defined on the single, finest grid $\Omega_{l_{\max},m_{\max},n_{\max}}$, and when *all* coarser grids $\Omega_{l,m,n}$, level$\equiv l+m+n < l_{\max}+m_{\max}+n_{\max}$ contribute to the solution process, we speak of *full-grid-of-grids* semi-coarsening [14]. A disadvantage of full-grid-of-grids semi-coarsening is that many grid cells are needed in total. With N^3 the total number of cells on the finest grid $\Omega_{l_{\max},m_{\max},n_{\max}}$, in 3D, asymptotically standard multigrid uses $\frac{9}{8}N^3$ grid cells versus $8N^3$ cells for the full-grid-of-grids approach. An efficiency improvement can be achieved by thinning out the grid-of-grids, e.g., by deleting fine grids. This may lead to the sparse-grid-of-grids and the semi-sparse-grid-of-grids approaches, to be discussed in this paper.

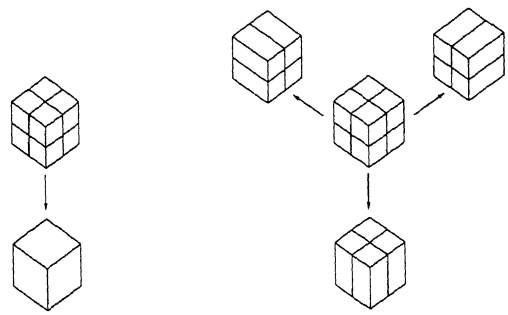

Figure 1. Two types of 3D coarsenings, left: full coarsening, right: multiple semi-coarsening.

The contents of the paper is as follows. In Section 2, we briefly describe the 3D discrete equations and the 3D test case to be considered throughout the paper. In Section 3, we describe the four different multigrid strategies to be compared: (i) standard multigrid, (ii) full-grid-of-grids multigrid, (iii) sparse-grid-of-grids multigrid and (iv) semi-sparse-grid-of-grids multigrid. Per type of multigrid strategy, we illustrate the performance for the

[1] CWI, P.O. Box 94079, 1090 GB Amsterdam, The Netherlands
[2] CWI, P.O. Box 94079, 1090 GB Amsterdam, The Netherlands
[3] CWI, P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

ECCOMAS 98.
Published in 1998 by John Wiley & Sons, Ltd.

test case chosen. Finally, in Section 4, we exploit the good parallelization properties of the sparse-grid and the semi-sparse-grid method.

2 TEST SET

2.1 Equations

The steady, non-isenthalpic, 3D Euler equations of gas dynamics are considered. The equations are discretized in their integral form. The computational domain Ω is divided, in a regular manner, into cell-centered finite volumes. These finite volumes are arbitrarily shaped hexahedra. Following the Godunov approach, along each cell face the flux vector is assumed to be constant and to be determined by a uniformly constant left and right state. To solve the resulting 1D Riemann problem, we apply the 3D extension of the 2D P-variant [9] of Osher's approximate Riemann solver [17]. For the left and right cell-face states, we take the first-order accurate approximations. At a later stage, these approximations can be replaced by higher-order accurate ones, in which case also limiters can be introduced. We emphasize that the major challenge is to know how to solve *first-order accurate* discretized, steady 3D Euler equations at efficient, grid-independent convergence rates. Once this is known, solving higher-order accurate discrete, steady 3D Euler equations can be done by a standard procedure, e.g., by a defect correction method as outer and the efficient multigrid method as inner iteration [11, 12].

2.2 Flow problem

As test case we consider the ONERA-M6 half-wing at $M_\infty = 0.84, \alpha = 3.06°$. The grids used are of C-O-type. (Graphs are given in [14].) The wing as well as the grids are symmetric with respect to the plane through the wing's leading and trailing edges. In the results to be presented hereafter, the finest grid considered is a $64 \times 16 \times 16$-grid.

3 MULTIGRID METHODS AND RESULTS

3.1 Standard multigrid

3.1.1 Method

First we briefly describe the standard 3D multigrid algorithm. The multigrid methods to be described hereafter are based on it. We use the 3D generalization of the optimal 2D multigrid approach, that was originally described in [8, 9]. As the smoothing technique for the first-order discrete Euler equations, collective symmetric point Gauss-Seidel relaxation is applied. The four different symmetric relaxation sweeps that are possible on a regular 3D grid, are performed alternatingly. At each volume visited during a relaxation sweep, the system of five nonlinear equations is solved by Newton iteration.

As standard multigrid method we apply the nonlinear version (FAS, [3, 6]), preceded by nested iteration (FMG, [3, 6]). For this we construct a nested set of grids such that each finite volume on a coarse grid is the union of $2 \times 2 \times 2$ volumes on the next finer grid. Let $\Omega_0, \Omega_1, \ldots, \Omega_{\lambda_{max}}$ be the sequence of such nested grids, with Ω_0 the coarsest and $\Omega_{\lambda_{max}}$ the finest grid. Then, nested iteration is applied to obtain a good initial solution on $\Omega_{\lambda_{max}}$, whereas nonlinear multigrid is applied to converge to the solution on the finest grid, $q_{\lambda_{max}}$. The first iterand for the nonlinear multigrid cycling is the solution obtained by nested iteration. We proceed by discussing both stages in more detail.

Nested iteration The nested iteration starts with a user-defined initial estimate for q_0, the solution on the coarsest grid. To obtain an initial solution on a finer grid $\Omega_{\lambda+1}$, first the solution on the coarser grid Ω_λ is improved by a single nonlinear multigrid cycle. Hereafter, this solution is prolongated to the finer grid $\Omega_{\lambda+1}$. These steps are repeated until the highest level (finest grid $\Omega_{\lambda_{max}}$) has been reached.

Nonlinear multigrid iteration Let $N_\lambda(q_\lambda) = 0$ denote the nonlinear system of first-order accurate discretized equations on Ω_λ. Then, a single nonlinear multigrid cycle is recursively defined by the following steps:

1. Improve on Ω_λ the latest obtained solution q_λ by application of n_{pre} relaxation sweeps.
2. Compute on the next coarser grid $\Omega_{\lambda-1}$ the right-hand side $r_{\lambda-1} = N_{\lambda-1}(q_{\lambda-1}) - I_\lambda^{\lambda-1} N_\lambda(q_\lambda)$, where $I_\lambda^{\lambda-1}$ is a restriction operator for right-hand sides.
3. Approximate the solution of $N_{\lambda-1}(q_{\lambda-1}) = r_{\lambda-1}$ by the application of n_{FAS} nonlinear multigrid cycles. Denote the approximation obtained as $\tilde{q}_{\lambda-1}$.
4. Correct the current solution by: $q_\lambda = q_\lambda + \tilde{I}_{\lambda-1}^\lambda (\tilde{q}_{\lambda-1} - q_{\lambda-}$ where $\tilde{I}_{\lambda-1}^\lambda$ is a prolongation operator for solutions.
5. Improve again q_λ by application of n_{post} relaxations.

Steps (2),(3) and (4) form the coarse-grid correction. The restriction operator $I_\lambda^{\lambda-1}$ and the prolongation operator $\tilde{I}_{\lambda-1}^\lambda$ are the usual operators that are consistent with the piecewise constant approximation (for more details, see [14]).

3.1.2 *Results*

Convergence results are given in Figure 2. In both graphs, the residual ratio is defined as $\|R^i\|_{L_1}/\|R^1\|_{L_1}$, where R^i is the mass defect of the discrete Euler equations and where i refers to the status after the i-th iteration. For the standard multigrid convergence results presented in Figure 2b, we took $n_{pre} = 0$, $n_{post} = 1$, i.e., sawtooth-cycles. Though – of course – to a lesser extent than the single-grid convergence results (Figure 2a), the standard multigrid method's convergence results appear to be rather grid-dependent (Figure 2b). As mentioned in Section 1, the expected cure is to apply multiple semi-coarsening instead of standard, i.e., full coarsening. In the next sections we proceed by discussing this alternative coarsening.

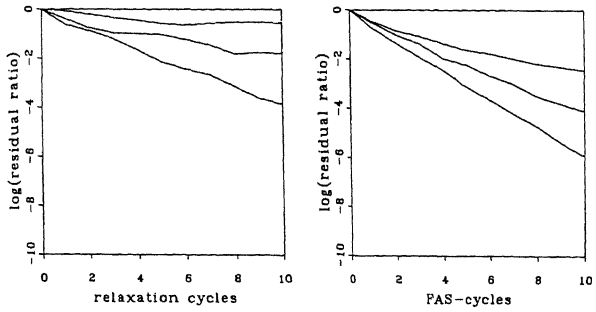

Figure 2. Convergence behaviors of two solution methods, ONERA-M6 half-wing at $M_\infty = 0.84$, $\alpha = 3.06°$, $\Omega_{\lambda_{max}} = (8 \times 2 \times 2) \times 2^{\lambda_{max}}$-grid, $\lambda_{max} = 1,2,3.$, left: single-grid, right: standard multigrid.

3.2 Full-grid-of-grids multigrid

3.2.1 *Method*

Pioneering work has been done by Mulder [15], who has introduced multiple semi-coarsening as a fix to the poor convergence results observed in computing nearly grid-aligned flows governed by the steady, 2D Euler equations. In [18], Radespiel and Swanson embroider on Mulder's approach for the steady, 2D Euler equations. Here we consider multiple semi-coarsened multigrid for the steady, 3D Euler equations, and pay particular attention to the prolongation operators.

Also in the case of the semi-coarsened multigrid method we use FAS as the basic multigrid algorithm, and on each grid collective symmetric point Gauss-Seidel relaxation is applied as the smoothing technique. As mentioned in Section 1, in the semi-coarsened multigrid method, the sequentially ordered set of grids $\Omega_\lambda, \lambda = 0,\ldots,\lambda_{max}$, is replaced by a partially ordered set of grids $\Omega_{l,m,n}$, $l = 0,1,\ldots,l_{max}$, $m = 0,1,\ldots,m_{max}$, $n = 0,1,\ldots,n_{max}$, with $\Omega_{0,0,0}$ the coarsest and $\Omega_{l_{max},m_{max},n_{max}}$ the finest grid. In the full-grid-of-grids variant of multiple semi-coarsening, all grids $\Omega_{l,m,n}$ play a role in the solution process. The nesting and the semi-coarsening relation between these grids and more data structure aspects are described in [10].

Nested iteration Also with semi-coarsening, nested iteration (FMG) is applied to obtain a good initial solution on the finest grid. We proceed to discuss the present nested iteration and nonlinear multigrid iteration in more detail. The nested iteration starts with a user-defined initial estimate on the coarsest grid, $\Omega_{0,0,0}$, which is improved by relaxation. To continue, the following two options exist:

- The approximate solution $q_{0,0,0}$ is prolongated to all grids up to and including level 3, with the 3D prolongation according to formula (37) in [7] (see [14] for the implementation in the present 3D Euler context). Next, the solution $q_{1,1,1}$ is improved by a single nonlinear multigrid cycle and prolongated to all grids up to and including level 6. For simplicity, we assume that $l_{max} = m_{max} = n_{max}$. Then, the above process can be repeated in a straightforward manner up to and including level $3l_{max}$. Notice that solution improvements are only made at $\Omega_{0,0,0}$,$\Omega_{1,1,1}$,$\Omega_{2,2,2}$, ...
- The approximate solution $q_{0,0,0}$ is prolongated to the three grids $\Omega_{1,0,0}$, $\Omega_{0,1,0}$ and $\Omega_{0,0,1}$ on the next level, with the same 3D prolongation as mentioned above. Next, the three solutions $q_{1,0,0}$, $q_{0,1,0}$ and $q_{0,0,1}$ are first improved by a single nonlinear multigrid cycle and then prolongated to all six grids on level 2. The above process is repeated up to and including level $l_{max} + m_{max} + n_{max}$. Notice that here, as opposed to the previous strategy, solution improvements are made on *all* grids, *level-by-level.*

Nonlinear multigrid iteration A single nonlinear multigrid cycle on level $l + m + n$ is recursively defined by the following steps:

1. Compute the same right-hand sides as in standard multigrid, on all grids at the next coarser level $(l + m + n) - 1$, but use as restriction operator the natural one described in [14] (natural because it just sums defects over the sub-cells).
2. Approximate the solutions on the coarser level $(l + m + n) - 1$ by the application of a single nonlinear multigrid cycle.

3. Correct the current solutions on level $l + m + n$ by one of two alternative correction prolongations. The first prolongation can be seen as an extension to 3D and to systems of equations, of the prolongation due to Naik and Van Rosendale [16]. It uses prolongation weights that are proportional to the absolute values of the restricted defect components. The second correction prolongation is the hierarchical one proposed in [7], equation (36). It has a-priori known prolongation weights $+1$ and -1.
4. Improve the solutions on level $l + m + n$ by the application of n_{post} relaxation sweeps.

3.2.2 *Results*

We first compare the two correction prolongations just mentioned: the one with defect-dependent weights and the one with fixed hierarchical weights. The nested iteration applied is the first one described in Section 3.2. Convergence results are shown in Figure 3. In the two graphs, the residual ratio is defined as $\|R^{i_{\text{FAS}}}\|_{L_1}/\|R^1\|_{L_1}$, where $R^{i_{\text{FAS}}}$ is the first component (i.e., the mass component) of $N_{l_{\max},m_{\max},n_{\max}}(q^{i_{\text{FAS}}}_{l_{\max},m_{\max},n_{\max}})$ and where i_{FAS} refers to the status after the i_{FAS}-th FAS-cycle. Similar as for the standard multigrid convergence results (Figure 2b), here we also used sawtooth cycles ($n_{\text{pre}} = 0$, $n_{\text{post}} = 1$). The improvement of both semi-coarsened multigrid methods with respect to the standard multigrid method is significant. Of both methods, the one with the fixed hierarchical prolongation weights (Figure 3b) performs better than the one with defect-dependent prolongation weights (Figure 3a).

The convergence results may still be further improved.

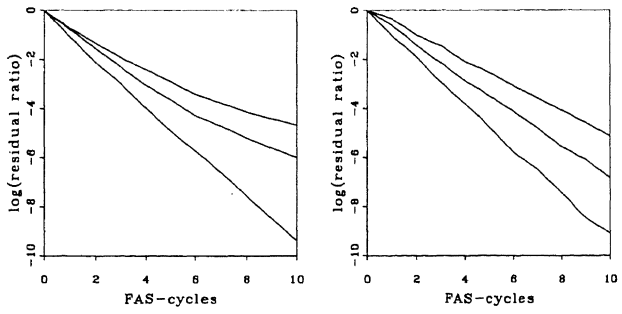

Figure 3. Convergence behaviors of two semi-coarsened multigrid methods, ONERA-M6 half-wing at $M_\infty = 0.84$, $\alpha = 3.06°$, $\Omega_{l_{\max},m_{\max},n_{\max}} = (8 \times 2^{l_{\max}}) \times (2 \times 2^{m_{\max}}) \times (2 \times 2^{n_{\max}})$-grid, $l_{\max} = m_{\max} = n_{\max} = 1,2,3$, left: with defect-dependent prolongation weight, right: with fixed prolongation weights.

In Figure 4 we present results for the same solution strategy as that of Figure 3b, but now with V-cycles ($n_{\text{pre}} = n_{\text{post}} = 1$) and with the more elaborate, level-by-level nested iteration described in Section 3.2.

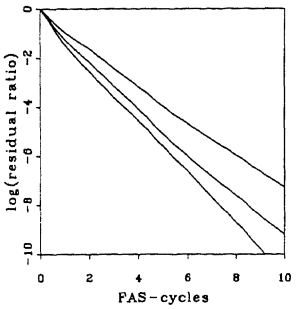

Figure 4. Convergence behavior of semi-coarsened multigrid method with fixed prolongation weights, V-cycles and level-by-level nested iteration, for ONERA-M6 half-wing at $M_\infty = 0.84$, $\alpha = 3.06°$, $\Omega_{l_{\max},m_{\max},n_{\max}} = (8 \times 2^{l_{\max}}) \times (2 \times 2^{m_{\max}}) \times (2 \times 2^{n_{\max}})$-grid, $l_{\max} = m_{\max} = n_{\max} = 1,2,3$.

3.3 Sparse-grid and semi-sparse-grid multigrid

3.3.1 *Methods*

As mentioned in Section 1, a disadvantage of full-grid-of-grids semi-coarsening is that $8N^3$ grid cells are needed in total (N^3 being the number of grid cells on $\Omega_{l_{\max},m_{\max},n_{\max}}$). An efficiency improvement can be achieved by deleting fine grids. Then, if no finest grid is available anymore, accurate approximations can still be constructed either by extrapolation or by the use of hierarchical bases. Most ambitious is the sparse-grid-of-grids approach. With the full grid-of-grids depicted as a cube in Figure 5, the corresponding sparse grid-of-grids is the subset under the dashed line, only grids $\Omega_{l,m,n}$, level$\leq l_{\max}$ contribute. The reduction in the numbers of grid cells is enormous. The computational complexity of the sparse-grid-of-grids approach is $\mathcal{O}(N\log^2 N)$, i.e., almost the complexity of a 1D problem only. Theoretically, the sparse-grid-of-grids approach has the best ratio of discrete accuracy over number of grid points used [5], the loss of accuracy is only a logarithmic factor when compared with the full-grid-of-grids approach. In practice, although very fast, the accuracy of the sparse-grid approximations is slightly disappointing. It appears that more accurate approximations are obtained *not* by only increasing the number of levels, but also by dropping the cells with extreme aspect ratios.

This leads to the compromise of the semi-sparse grid-of-grids. This uses the family of grids $\Omega_{l,m,n}$, level$\leq 2l_{\max}$, $\max(l,m,n) \leq l_{\max}$ (Figure 5, under the dotted line), which (asymptotically) still has a computational complexity which is smaller than that of the single-grid approach, viz. $\mathcal{O}(N^2 \log^2 N)$, i.e., almost the complexity of a 2D problem. So, though to a lesser extent than the genuine sparse-grid approach, it still is a cure to cubic complexity, the 'curse of 3D'.

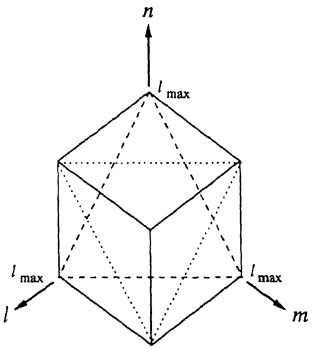

Figure 5. Cubic, full grid-of-grids and the corresponding sparse (dashed) and semi-sparse grid-of-grids (dotted).

3.3.2 Results

The numerical ingredients of both approaches are identical to those in the full-grid-of-grids approach applied in obtaining Figure 4. Exactly the same method is applied, with as the only difference that in the sparse-grid case the multi-level semi-coarsening solver stops its work at level $l_{\max}$. From there the solution is prolongated to the very finest grid at level $3l_{\max}$. The prolongation is done by the 3D extension of the combination extrapolation given on p. 290 in [19]. In the semi-sparse-grid approach the semi-coarsened multi-level algorithm is only stopped at level $2l_{\max}$ and from there, by the same combination technique, the finest-grid solution at $3l_{\max}$ is computed. A particular advantage of the semi-sparse-grid approach as compared to the sparse-grid approach, is that the 3D extrapolation rule can be applied for all remaining grids to be filled, including the grids along the boundaries of the grid-of-grids. In the sparse-grid approach this is not possible. There, for all boundary grids in between $l_{\max}$ and $2l_{\max}$ one has to make a compromise, for instance, by applying a 2D combination extrapolation, which will inevitably result in some additional loss of accuracy.

In [13], we give an impression of the accuracy of the numerical solutions, obtained by the three different grid-of-grids approaches depicted in Figure 5. In Table 1 we give the relative computing times used.

Table 1. Cost (for the ONERA-M6 case) of the three types of grid-of-grids methods.

grid-of-grids method	full	semi-sparse	sparse
(scaled) CPU time	150	35	1

4 PARALLELIZATION OF SPARSE-GRID AND SEMI-SPARSE-GRID MULTIGRID

The pre- and post-relaxations, steps 1 and 5 in the nonlinear multigrid iteration (Section 3.1), are done by a procedure for performing a user-defined operation on all grids $\Omega_{l,m,n}$ at grid level $l+m+n$. In this case, the user-defined procedure is the point Gauss-Seidel relaxation on all cells of grid $\Omega_{l,m,n}$. Because the relaxation subroutine only reads and writes data concerning its own grid, the relaxations can be done directly in parallel for all grids visited at a certain grid level. Given the fact that almost all computing time consumed by the total program, is used in the relaxations, parallel implementation is expected to pay off.

Parallelization is done through the MANIFOLD coordination language. MANIFOLD is a language for managing complex, dynamically changing interconnections among sets of independent, concurrent, cooperating processes [1]. MANIFOLD is based on the IWIM model of communication [2].

Details of the parallel implementation in MANIFOLD can be found in [4]. Here, we further restrict ourselves to giving some results of performance measurements. The speed-up results for the sparse- and the semi-sparse grid approach are given in Table 2. They show the elapsed time versus the grid level. To even out such unpredictable effects as network traffic and file server delays, etc., we have run the two versions of the application on each of the three levels close to each other in real time, and for each version of the application, a few times on each level. From Table 2, where the average times are given, it appears that the parallel version takes good advantage of the parallelism offered by the four processors of the machine. For the sparse-grid and the semi-sparse-grid application, the parallel computing times are about 3.25 and 3.75 times smaller, respectively, than the sequential-code times. So, in both cases we have obtained a nearly linear speed-up.

Table 2. The average elapsed times (in hours:minutes:seconds) for the sparse- and semi-sparse-grid-of-grids approaches.

	level	*sparse*	*semi-sparse*
sequential	1	11.24	50.43
	2	1:37.42	18:02.10
	3	9:15.56	4:36:08.54
parallel	1	5.84	27.33
	2	34.06	5:58.72
	3	2:47.40	1:14:44.04

5 CONCLUSIONS

The intrinsically low computational complexity of sparse-grid and semi-sparse-grid techniques, plus the additional gains in computing time by parallelization, make both methods challenging for very computing-intensive work.

An interesting possibility for future research is the application of local 3D semi-refinement. For this, the relative truncation errors that are available between all grids on two consecutive grid levels, can serve in the grid-adaptation criterion. Work in this direction is in progress.

REFERENCES

[1] F. Arbab, 'Coordination of massively concurrent activities', Technical Report CS-R9565, CWI, Amsterdam, (1995). Available on-line at http://www.cwi.nl/ftp/CWIreports/IS/CS-R9565.ps.Z.

[2] F. Arbab, *The IWIM model for coordination of concurrent activities*, volume 1061 of *Lecture Notes in Computer Science*, 34–56, Springer, Berlin, 1996.

[3] A. Brandt, *Guide to multigrid development*, volume 960 of *Lecture Notes in Mathematics*, 220–312, Springer, Berlin, 1982.

[4] C.T.H. Everaars and B. Koren, 'Using coordination to parallelize sparse-grid methods for 3D CFD problems', *Parallel Computing*, (to appear).

[5] M. Griebel, C. Zenger, and S. Zimmer, 'Multilevel Gauss-Seidel-algorithms for full and sparse grid problems', *Computing*, **50**, 127–148, (1993).

[6] W. Hackbusch, *Multi-Grid Methods and Applications*, Springer, Berlin, 1985.

[7] P.W. Hemker, 'Sparse-grid finite-volume multigrid for 3D-problems', *Advances in Computational Mathematics*, **4**, 83–110, (1995).

[8] P.W. Hemker and B. Koren, *A non-linear multigrid method for the steady Euler equations*, volume 26 of *Notes on Numerical Fluid Mechanics*, 175–196, Vieweg, Braunschweig, 1989.

[9] P.W. Hemker and S.P. Spekreijse, 'Multiple grid and Osher's scheme for the efficient solution of the steady Euler equations', *Applied Numerical Mathematics*, **2**, 475–493, (1986).

[10] P.W. Hemker and P.M. de Zeeuw, *BASIS3, a data structure for 3-dimensional sparse grids*, volume 57 of *Notes on Numerical Fluid Mechanics*, 443–484, Vieweg, Braunschweig, 1997.

[11] B. Koren, 'Defect correction and multigrid for an efficient and accurate computation of airfoil flows', *Journal of Computational Physics*, **77**, 183–206, (1988).

[12] B. Koren, 'Multigrid and defect correction for the steady Navier-Stokes equations', *Journal of Computational Physics*, **87**, 25–46, (1990).

[13] B. Koren, P.W. Hemker, and C.T.H. Everaars. Multiple semi-coarsened multigrid for 3D CFD. AIAA paper 97-2029.

[14] B. Koren, P.W. Hemker, and P.M. de Zeeuw, *Semi-coarsening in three directions for Euler-flow computations in three dimensions*, volume 57 of *Notes on Numerical Fluid Mechanics*, 547–567, Vieweg, Braunschweig, 1997.

[15] W.A. Mulder, 'A new multigrid approach to convection problems', *Journal of Computational Physics*, **83**, 303–323, (1989).

[16] N.H. Naik and J. Van Rosendale, 'The improved robustness of multigrid elliptic solvers based on multiple semi-coarsened grids', *SIAM Journal on Numerical Analysis*, **30**, 215–229, (1993).

[17] S. Osher and F. Solomon, 'Upwind difference schemes for hyperbolic systems of conservation laws', *Mathematics of Computation*, **38**, 339–374, (1982).

[18] R. Radespiel and R.C. Swanson, 'Progress with multigrid schemes for hypersonic flow problems', *Journal of Computational Physics*, **116**, 103–122, (1995).

[19] U. Rüde, *Multilevel, extrapolation, and sparse grid methods*, volume 116 of *International Series of Numerical Mathematics*, 281–294, Birkhäuser, Basel, 1994.