REVISTA INTERDISCIPLINAR DE PESQUISA EM ENGENHARIA

CILAMCE 2016
XXXVII IBERIAN LATIN AMERICAN CONGRESS ON COMPUTATIONAL METHODS IN ENGINEERING
BRASÍLIA - DF - BRAZIL

EFFECT OF PARAMETERS OF MULTIGRID METHOD ASSOCIATED WITH EXTRAPOLATORS IN CFD PROBLEMS

Anunciação, M. A. M.

marcioalexandro@ufpr.br

Federal University of Paraná, Post-Graduate Course of Numerical Methods in Engineering, Centro Politécnico, 81531-980 Curitiba-PR, Brazil.

Pinto, M. A. V., Araki, L. K.

marcio_villela@ufpr.br; lucaraki@ufpr.br;

Federal University of Paraná, Department of Mechanical Engineering, Curitiba - PR, Brazil.

Martins, M. A.

mandre@unicentro.br

State University of Centro Oeste, Department of Mathematics, Guarapuava - PR, Brazil.

Gonçalves, S.F.T.

simone.tg@ufpr.br

Federal University of Paraná, Department of Mechanical Engineering, Curitiba - PR, Brazil.

Abstract. *The focus of this work is analyzing the behavior of the following parameters: the iteration error, the processing time (CPU time) and the convergence factors for two problems of Computational Fluid Dynamics (CFD): the two-dimensional linear heat diffusion problem, governed by a Poisson-like equation, with Dirichlet's boundary conditions, and it is solved by using the Geometric Multigrid Method associated to the following extrapolation methods: Aitken, Empiric, Mitin, Epsilon (scalar and topological), Rho (scalar and topological), Multiple Aitken Extrapolations and Multiple Mitin Extrapolations; and the square lid-driven cavity, governed by Burgers' equations, with Dirichlet's boundary conditions, solved by using the Geometric Multigrid associated to the Topological Epsilon Extrapolation Method during the Multigrid cycles. According to numerical results, it was observed: the reduction of the magnitude of iteration error, the reduction of non-dimensional residual based on the initial estimate and the reduction of the convergence factor, with a CPU time compatible to the pure Multigrid Method for both problems.*

Keywords: *extrapolation methods, Multigrid, acceleration of convergence, iteration error.*

1 INTRODUCTION

Systems of algebraic equations with the form $Au = f$, where A is the matrix of coefficients, f is the vector of independent terms and u is the vector of unknowns, are commonly obtained in the discretization process of a mathematical model in the Computational Fluid Dynamics (CFD). The solution of such systems is usually obtained by the use of iterative methods, which although are faster than direct methods, frequently present slow convergence for higher-order matrix problems. The acceleration of the convergence for iterative processes can be achieved by two distinct ways: by modifying the iterative process or by changing the sequence which converges slowly by another one with better convergence properties (Brezinski and Zaglia, 2008).

In last decades, an alternative of unambiguous efficiency in the acceleration of iterative methods is the Multigrid Method (Briggs et al., 2000 and Trottenberg et al., 2001). Its philosophy is based on the employment of grids of different sizes, which are covered during the iterative process. Another methodology which aims the acceleration of the convergence for iterative methods is to associate them to Extrapolation Methods, whose focus is to change a sequence of vectors in another one with faster convergence features.

There are other techniques to speed-up of the Multigrid Method in literature. Among several works, Shen et al. (2000) can be cited. In such work, ideal pre-conditioners to the Multigrid Method are used, based on the interchange of lines in the Gauss-Seidel smoothing process using Chebyshev-collocation for second order elliptical equations. Other authors work with parallelization using GPUs in the solution of problems associated to the Multigrid Method, as can be seen in Liu et al. (2015). In Zhang et al. (2010) the Pseudospectral Chebyshev Method is used with the Multigrid Method to solve the primitive variables of the Navier-Stokes Equations.

In the current work, Extrapolation Methods were associated to the Multigrid Method, in order to accelerate the convergence process and reduce the iteration error. Such formulation is attractive since there is no works focusing on it.

Next sections of the current work are organized as follows. Theoretical and computational features of the Multigrid and Extrapolation Methods are presented in Section 2. Mathematical and numerical models of adopted test problems are shown in Section 3. Numerical results and associated discussions are provided in Section 4. And the closure of this work is done in Section 5, with the most important remarks.

2 NUMERICAL METHODS

2.1 Multigrid method

The use of the Multigrid Method is related to the smoothing properties of error by classical iteration methods: it is faster in initial iterations for oscillatory components, while for smooth components a large number of iterations is required and these classical methods lose efficiency. In this way, the Multigrid Method deals with a sequence of coarse auxiliary grids (with a lower quantity of nodal points) in which the error components are quickly smoothed, to later return to the original grid. Information is transferred between two grids by operators, called restriction operators (in this case, information is transferred from a fine grid to the next coarser one), generically represented by $\left[I\right]_h^H$ and defined as

$$v^H = \left[I\right]_h^H v^h \tag{1}$$

or prolongation operators (in this case, information is transferred from a coarse grid to the next finer one), generically represented by $\left[I\right]_H^h$ and defined as

$$v^h = \left[I\right]_H^h v^H \tag{2}$$

The restriction operator employed in the current work was the full weighting interpolation (Briggs et al., 2000 and Trottenberg et al., 2001), while the prolongation operator was the bilinear interpolation (Briggs et al., 2000, Trottenberg et al., 2001 and Hackbush 1985), both commonly found in Multigrid literature.

V-cycle was also used in this work by its efficiency (Briggs et al., 2000, Trottenberg et al., 2001 and Hackbush 1985). To solve the Poisson-type equation, which is a linear problem, the Correction Scheme (CS) was used; such scheme transfers only the residual value to the coarser grids (Briggs et al., 2000, Trottenberg et al., 2001 and Hackbush 1985). For the Burgers' equations, which are a system of non-linear equations, the Full Approximation Scheme (FAS) was used; in such scheme both the residual value and the approximation for solution are transferred to coarser grids (Trottenberg et al., 2001).

The coarsening ratio, for the two-dimensional problem, by using uniform grids, is defined as $r = H/h$, where *h* is related to the spacing of the fine grid, Ω^h, and *H* is the spacing size of the immediately coarser grid, Ω^H. In the current work, the standard coarsening ratio, $r = 2$, was employed (Briggs et al., 2000).

The algorithms for CS and FAS schemes with V-cycle for several grid levels, until the achievement of a stop criterion or a chosen maximum number of cycles, can be found in Briggs et al. (2000), Trottenberg et al. (2001) and Hackbush (1985). Such authors also present the recursive versions for these schemes, which are generally the most used ones.

2.2 Extrapolation Methods

The purpose of Extrapolation Methods is to change a sequel with slow convergence in another one with better convergence properties. Such extrapolators can be classified as scalar and vector ones, according to the way information is dealt.

Mitin (1985) considers a iterative process in a Hilbert space, where the vectors $C_1, C_2, ..., C_k, ..., C_\infty$ are obtained in steps during the process. The vector $\Delta_k = C_k - C_\infty$ is defined. In general: $\Delta_{k+1} = F\left(\Delta_k\right)$, where F is the operator which defines the iterative process. In such work, an expression, which will support the formulae to the three first extrapolators of the current work, is defined:

$$C_{\infty,j} = \frac{C_{k+2,j}C_{k,j} - {C_{k+1,j}}^2}{C_{k+2,j} - 2C_{k+1,j} + C_{k,j}}, \tag{3}$$

For the subsequent iterations, with $k=1$, $C_1=\phi_1$, $C_2=\phi_2$, $C_3=\phi_3$ and for all j-th components satisfying the relation described by Eq. (3), the following expression can be written:

$$\phi_{\infty}^{Aitken}=\frac{\phi_1\phi_3-\phi_2{}^2}{\phi_3-2\phi_2+\phi_1}, \tag{4}$$

which is the formula used to the Aitken extrapolation (Burden and Faires, 2005)

Based on the Empiric Estimator Martins and Marchi (2008) proposed:

$$\phi_{\infty}^{Empírico}=\phi_3+\frac{\left(\phi_3-\phi_2\right)^2}{2\phi_2-\phi_3-\phi_1}, \tag{5}$$

which is a relation with the same properties of the Aitken extrapolator.

Considering now the subsequent iterations with $k=1$, $C_1=\phi_1$, $C_2=\phi_2$, $C_3=\phi_3$, $C_4=\phi_4$, $C_5=\phi_5$ and all the *j*-th components satisfying the relation described by Eq. (3), the following expression is written:

$$\phi_{\infty}^{Mitin}=\frac{\phi_1\phi_5-\phi_3{}^2}{\phi_5-2\phi_3+\phi_1}, \tag{6}$$

which is a formula used to the Mitin extrapolation (Mitin, 1985).

Other two extrapolation methods presented in literature are the Epsilon and Rho algorithms, which present a formal similarity, but significantly differ in their capability to accelerate the convergence. The properties of Epsilon and Rho algorithms are, in a certain manner, complementary.

The importance of studying the Epsilon algorithm is based on its potential to be applied on the acceleration of convergence of iterative solution for discretized differential equations. However, all algorithms present their domain of validity. The Epsilon algorithm, for example, fails to logarithmically convergent sequences (which converge very slowly) and it cannot achieve the fixed point of generators of sequences which diverge very fast. The Epsilon algorithm generally fails to such sequences and Delahaye (1988) and Delahaye and Germain-Bonne (1982) demonstrated that there is no universal accelerator to logarithmically convergent sequences (Graves-Morris et al., 2000). The Rho algorithm does not accelerate linear convergent sequences, but it is powerful to logarithmically convergent sequences (Graves-Morris et al., 2000 and Gao et al., 2010).

The Rho and Epsilon Extrapolation Methods can be generalized to the vector case, i. e., involving vector operations. In addition, such approach can be recursively applied and then it

is known as topological formulation (Brezinski and Zaglia, 2008); it is also used in the current work.

With the aim of using a scalar extrapolator to a vector sequence it must be applied simultaneously to each vector component. Nonetheless a disadvantage of using such technique is the fact that the connection among the components are neglected, in other words, the components are treated as independent scalars. Attention must also be paid to the vector characteristics, since vectors arise from the iterative process and components can assume different orders of convergence, when independently analyzed. In this case, numerical instabilities are not discharged (for some components).

In these cases the vector approach is recommended; however, it does not imply in a higher level of complexity. As example, to generalize the Scalar Epsilon Extrapolator to vector case, the inverse of a vector is needed to be defined. A possibility considered by Wynn (1962) is the use of inverse defined by

$$z^{-1} = \frac{z}{\|z\|^2}, \; z \in \mathrm{R}^N . \tag{7}$$

Therefore, for vector sequences, the vector Epsilon extrapolator is defined by

$$\varepsilon_{-1}^{(n)} = 0; \;\; \varepsilon_0^{(n)} = S_n; \;\; \varepsilon_{k+1}^{(n)} = \varepsilon_{k-1}^{(n+1)} + \left[\varepsilon_k^{(n+1)} - \varepsilon_k^{(n)} \right]^{-1}, \;\; k, n = 0,1,... \tag{8}$$

Another generalization of Scalar Epsilon Extrapolation for vector sequences is proposed in Brezinski and Zaglia (2008). It is a recursive form of vector Epsilon algorithm, in which the evaluation of vector inverses is not needed: the called Topological Epsilon Algorithm (TEA), whose formula is given recursively by:

$$\begin{cases} \varepsilon_{-1}^{(n)} = 0; \qquad \varepsilon_0^{(n)} = S_n; \qquad \varepsilon_{2k+1}^{(n)} = \varepsilon_{2k-1}^{(n+1)} + \dfrac{y}{\left\langle y, \Delta\varepsilon_{2k}^{(n)} \right\rangle}; \\ \varepsilon_{2k+2}^{(n)} = \varepsilon_{2k}^{(n+1)} + \dfrac{\Delta\varepsilon_{2k}^{(n)}}{\left\langle \Delta\varepsilon_{2k+1}^{(n)}, \Delta\varepsilon_{2k}^{(n)} \right\rangle}, \quad k, n = 0,1,... \end{cases} \tag{9}$$

being y an arbitrary vector, orthogonal to the j-th component of residual value generalized between $e_{k,j}(S_n)$ and $e_{k,j-1}(S_n)$ and

$$\langle \alpha, \beta \rangle = \sum_{i=1}^{p} \alpha_i \beta_i \tag{10}$$

the inner product of two vectors α and β, where α_i and β_i are the components of α and β vectors, respectively, and *p* is the number of components of vectors.

The difference operator Δ acts on the superscript *n* and

$$\varepsilon_{2k}^{(n)} = e_k\left(S_n\right) = s\,, \tag{11}$$

and

$$\varepsilon_{2k+1}^{(n)} = \frac{y}{\left\langle y, e_k\left(\Delta S_n\right)\right\rangle},\ k, n = 0,1,.... \tag{12}$$

Beyond the cited extrapolators, repeated extrapolations (Burden and Faires, 2005) (extrapolation of already extrapolated data) were also done for the Aitken and Mitin methods.

For the Repeated Aitken Extrapolation (RAE), for example, the five last obtained solutions by the Multigrid Method $\left(\phi_{1,0}, \phi_{2,0}, \phi_{3,0}, \phi_{4,0}, \phi_{5,0}\right)$ are used, where $\phi_{g,m}$ represents the g^{th}-solution and m^{th}-extrapolation level. For the first solution level, in which none extrapolation was done $\left(m = 0\right)$, extrapolations are evaluated by using: $\left(\phi_{1,0}, \phi_{2,0}, \phi_{3,0}\right)$, $\left(\phi_{2,0}, \phi_{3,0}, \phi_{4,0}\right)$ and $\left(\phi_{3,0}, \phi_{4,0}, \phi_{5,0}\right)$, which generates, respectively, $\phi_{3,1}$, $\phi_{4,1}$ and $\phi_{5,1}$, where *m* = 1 represents the first extrapolation level; as an example, $\phi_{3,1}^{Aitken} = \phi_{3,1}$ in equation

$$\phi_{3,1}^{Aitken} = \frac{\phi_{1,0}\phi_{3,0} - \phi_{2,0}^2}{\phi_{3,0} - 2\phi_{2,0} + \phi_{1,0}}\,. \tag{13}$$

After that, these three extrapolated solutions $\left(\phi_{3,1}, \phi_{4,1}, \phi_{5,1}\right)$ are used to evaluate a new extrapolation, generating the second extrapolation level solution $\phi_{5,2}$. Figure 1 shows a scheme of the Repeated Aitken Extrapolation for two levels.

The procedure for the Repeated Mitin Extrapolation (RME) is analogous to the previously presented one; taking into account for RME at least nine different solutions are necessary to evaluate two extrapolation levels.

3 MATHEMATICAL AND NUMERICAL MODELS

3.1 Mathematical method

The aim of this work is to show how the use of extrapolators associated to the Multigrid Method can act on the iteration error and the iterative processing time. To do such a thing, two problems were chosen, a simpler one (Poisson-type equation), and a more complex one

(Burgers' equations). The first problem was chosen to particularly analyze the effects of the cited methodology, allowing the detection of which extrapolator presents the best results. In the sequence, such extrapolator was used to the second problem, a non-linear system of equations, to analyze its behavior.

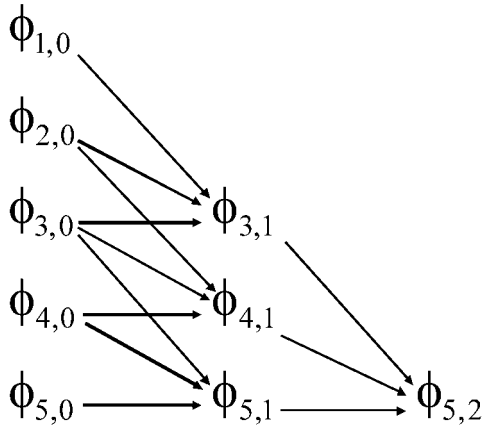

Figure 1. Schematic representation of Multiple Aitken Extrapolation with two levels.

Thereby, in this work the steady linear two-dimensional heat transfer problem, described by a Poisson-type equation for a unity-square domain, was solved

$$\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial y^2}=S(x,y),\quad 0\le x,y\le 1, \tag{14}$$

with Dirichlet's boundary conditions, $T(0,y)=T(x,0)=T(1,y)=T(x,1)=0$, source term

$$S(x,y)=-2\left[\left(1-6x^2\right)y^2\left(1-y^2\right)+\left(1-6y^2\right)x^2\left(1-x^2\right)\right], \tag{15}$$

and analytical solution obtained by the method of manufactured solutions, presented by Tannehill et al. (1997) as

$$T(x,y)=\left(x^2-y^4\right)\left(y^4-x^2\right). \tag{16}$$

Afterwards, the advection-diffusion problem, governed by the Burgers equations, which is a non-linear partial differential equations, was solved. Such equations are a simplified form of the Navier-Stokes equations, reduced to the Linear Momentum Equations, since the pressure field is prescribed and the mass conservation equation is not needed. The model considered in this work presents the analytical solution and the pressure field given by Shih et al. (1989) and are presented in the sequence. Considering constant properties, Cartesian

coordinates and steady state, the Burgers' equations in conservative form are given by Tannehill (1997)

$$\frac{\partial u^2}{\partial x}+\frac{\partial (uv)}{\partial y}=-\frac{\partial p}{\partial x}+\frac{1}{\mathrm{Re}}\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right),$$

$$\frac{\partial (uv)}{\partial x}+\frac{\partial v^2}{\partial y}=-\frac{\partial p}{\partial y}+\frac{1}{\mathrm{Re}}\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}\right)-S(x,y,\mathrm{Re}), \tag{17}$$

where p is the static pressure, u and v are the velocity components in x and y directions, respectively, Re is the number of Reynolds, evaluated by $\mathrm{Re}=U_\infty l/\nu_c$ where U_∞ and l are the velocity and the length of reference, $\nu_c=\mu_d/\rho$ is the cinematic viscosity, where μ_d is the dynamic viscosity and ρ is density. In this work, the simulations were done for Re = 1000. The source term $S(x,y,\mathrm{Re})$ and the pressure field p, given by Shih et al., (1989), are respectively

$$\begin{aligned} S(x,y,\mathrm{Re})=&-\frac{8}{\mathrm{Re}}\left[24F(x)+2f'(x)g''(x)+f'''(x)g(y)\right]-\\ &64\left[F_2(x)G_1(x)-g(x)g'(y)F_1(x)\right] \end{aligned} \tag{18}$$

and

$$p(x,y,\mathrm{Re})=\frac{8}{\mathrm{Re}}\left[F(x)g'''(y)+f'(x)g'(y)\right]+64F_2\left\{g(y)g''(y)-\left[g'(y)\right]^2\right\}, \tag{19}$$

$$\begin{cases} f(x)=x^4-2x^3+x^2 \\ g(y)=y^4-y^2 \\ F(x)=0,2x^5-0,5x^4+\dfrac{1}{3}x^3 \\ F_1(x)=4x^6+12x^5-14x^4+8x^3-2x^2 \\ F_2(x)=0,5(x^4-2x^3+x^2)^2 \\ G_1(y)=-24y^5+8y^3-4y \end{cases}. \tag{20}$$

The Dirichlet boundary conditions are:

$$\begin{aligned} v(x,0) = v(0,y) = v(1,y) = v(x,1) = 0, \\ u(x,0) = u(0,y) = u(1,y) = 0, \\ u(x,1) = 16(x^4 - 2x^3 + x^2) \end{aligned} \quad . \tag{21}$$

The analytical solution, obtained by the manufactured solutions technic (Roy, 2005), is given by

$$u(x,y) = 8(x^4 - 2x^3 + x^2)(4y^3 - 2y) \tag{22}$$

and

$$v(x,y) = -8(-4x^3 - 6x^2 + 2x)(y^4 - y^2) \tag{23}$$

In this work, the Burgers' equations are solved in a closed region and the problem is known as the square lid-driven cavity problem. In this case, the fluid moves from left to right in the upper lid with the velocity distribution in *x*-direction given by Eq. (22), while the other boundaries present null velocities (non-slip condition).

3.2 Numerical model

The first numerical model was obtained by the discretization of the Poisson-type equation by the use of the Finite Difference Method (FDM) (Tannehill, 1997), with Central Difference Scheme (CDS). The discretization of the Burgers' equations was done by the use of the Finite Volume Method (FVM) (Maliska, 2004), with structured grids, uniform for each direction. In this proceeding, diffusive terms were approximated by CDS; on the other hand, the advective terms were approximated by CDS with deferred correction. The Dirichlet boundary conditions were applied by the use of ghost cells. Such technique, beyond its easy application and the fact of respecting the conservation principia to all domain (Maliska, 2004), favors the implementation of restriction and prolongation procedures of the Multigrid Method. For both models, it was used uniform square grids with $N = N_x N_y$ nodes, generating a grid with nodal spacing $h_x = 1/(N_x - 1)$ and $h_y = 1/(N_y - 1)$, where N_x and N_y are the quantity of nodes of the fine grid in *x*- and *y*- coordinate directions, respectively. In current work, uniform grids are used, i.e., $N_x = N_y$ and $h_x = h_y = h$.

Gauss-Seidel Red-Black (GS-RB) (Wynn, 1965) was used as smoother, what benefits the parallel computation but also presents good results for serial programming (Zhang, 1996). The convergence criterion used to interrupt the iteration steps is the l_2-norm of residual by the l_2-norm of residual at initial guess, given by:

$$\|R\|_2 = \frac{\|R^{it}\|_2}{\|R^0\|_2}, \tag{24}$$

where R^{it} is the residual of current iteration and R^0 is the residual at initial guess. Null vector was used as initial guess for all variables of interest and the number of nodes for the coarsest grid was equal to 3 in each direction; for all simulations, the Multigrid Method was applied from the finest grid to the coarsest possible one. For both problems, the number of inner iterations was $\nu_1 = \nu_2 = 3$, where ν_1 and ν_2 are, respectively, the number of pre- and post-smoothing iterations. For the Burgers' equations, the Full Multigrid (FMG) was used; such method is based on the idea that a good initial guess can considerably reduce the processing time, since the solution of a problem with good initial guess requires few iterations in the fine grids. In such way, in FMG, to obtain a good initial guess the solution is interpolated from the coarse grid to the finer one. Because of this, errors are efficiently smoothed, which guarantees an optimum convergence rate. According to Trottenberg et al. (2001) such properties make the FMG the most efficient version of Multigrid and Zhang (1996) and Thekale et al. (2010) consider the FMG the preferred method to accelerate the Multigrid.

4 NUMERICAL RESULTS

For the Poisson-type equation, nine cases, resulting from the combination of values related to the sizing of finest grid $\left(N = 129\times129,\ N = 1025\times1025,\ N = 4097\times4097\right)$ and the stopping criterion $\left(10^{-6}, 10^{-10}, 10^{-15}\right)$, were studied. Extrapolations were inserted in two different phases, in order to evaluate their performance: in the end of iterative process and during the iterative process.

4.1 Use of extrapolations in the end of the Multigrid

For this methodology, each case was solved in three distinct ways:

(a) using only the Multigrid Method, until the achievement of the stopping criterion (MG);

(b) using the Multigrid Method with an additional V-cycle, in relation to the number of iteration steps necessary to achieve the stopping criterion (MG + 1 ITE);

(c) using the Multigrid Method until the achievement of the stopping criterion, associated to the extrapolation of the last numerical results (MG + Extrapolator).

The analyzed parameters in the current work were: the processing time or CPU time (t_{CPU}), in seconds [s]; the highest value of memory usage (M), in Megabytes [MB]; the norm of residual by the norm at initial guess $\left(\|R\|_2\right)$; empirical average convergence factor $\left(\hat{q}^{(k)}\right)$ (Tannehill, 1997); infinity norm of iteration error $\left(\|E_n\|_\infty\right)$; and Euclidian-norm of iteration error $\left(\|E_n\|_2\right)$. Iteration error is evaluated by the difference between the numerical solution at n-iteration and the solution obtained when the iterative procedure is done until the achievement of the round-off error. Table 1 present numerical results for $N = 1025\times1025$ and tolerance of 10^{-15}. Other numerical results are omitted since they presented analogous behavior.

Table 1. Results for studied parameters. Extrapolations in the end of the Multigrid.

Methodology	t_{CPU} **[s]**	M **[MB]**	$\|R\|_2$	$q^{(k)}$	$\hat{q}^{(k)}$	$\|E_n\|_\infty$	$\|E_n\|_2$
MG	21.871	119.640	$6.272\text{x}10^{-16}$	$4.238\text{x}10^{-2}$	$4.149\text{x}10^{-2}$	$3.672\text{x}10^{-17}$	$1.816\text{x}10^{-17}$
MG + 1 ITE	23.775	119.636	$2.663\text{x}10^{-17}$	$4.246\text{x}10^{-2}$	$4.157\text{x}10^{-2}$	$1.553\text{x}10^{-18}$	$7.749\text{x}10^{-19}$
MG + Aitken	119.076	218.468	$3.145\text{x}10^{-14}$	$5.014\text{x}10^{+1}$	$7.495\text{x}10^{-2}$	$4.118\text{x}10^{-18}$	$7.301\text{x}10^{-20}$
MG + Empirical	119.076	218.468	$3.145\text{x}10^{-14}$	$5.014\text{x}10^{+1}$	$7.495\text{x}10^{-2}$	$4.118\text{x}10^{-18}$	$7.301\text{x}10^{-20}$
MG + Mitin	119.029	251.360	$8.701\text{x}10^{-14}$	$1.387\text{x}10^{+2}$	$8.158\text{x}10^{-2}$	$9.630\text{x}10^{-18}$	$3.349\text{x}10^{-19}$
MG + Scalar Epsilon	29.874	251.556	$3.044\text{x}10^{-11}$	$4.854\text{x}10^{+4}$	$1.329\text{x}10^{-1}$	$5.152\text{x}10^{-15}$	$7.111\text{x}10^{-18}$
MG + Scalar Rho	29.921	251.564	$4.511\text{x}10^{-12}$	$7.193\text{x}10^{+3}$	$1.133\text{x}10^{-1}$	$2.066\text{x}10^{-14}$	$1.000\text{x}10^{-14}$
MG + Topological Epsilon	121.525	399.444	$1.648\text{x}10^{-19}$	$2.628\text{x}10^{-4}$	$2.721\text{x}10^{-2}$	$1.213\text{x}10^{-21}$	$5.165\text{x}10^{-22}$
MG + Topological Rho	31.169	399.656	$3.498\text{x}10^{-13}$	$5.577\text{x}10^{+2}$	$9.161\text{x}10^{-2}$	$2.066\text{x}10^{-14}$	$9.998\text{x}10^{-15}$
MG + Repeated Aitken Extrapolation	29.952	251.576	$4.716\text{x}10^{-12}$	$7.519\text{x}10^{+3}$	$1.137\text{x}10^{-1}$	$7.045\text{x}10^{-16}$	$1.856\text{x}10^{-18}$
MG + Repeated Mitin Extrapolation	30.405	317.368	$4.702\text{x}10^{-10}$	$7.497\text{x}10^{+5}$	$1.669\text{x}10^{-1}$	$6.402\text{x}10^{-14}$	$1.098\text{x}10^{-16}$

For this case the use of the Topological Epsilon Extrapolator was the methodology with the best results for the residual norm, the convergence factors and the iteration error norms. So it can be concluded that the Topological Epsilon was the most efficient for most parameters. Otherwise, related to the CPU time and memory requirements, the values observed for the Topological Epsilon Extrapolator were higher, compared to all other extrapolation schemes. The storage of numerical solutions and the extra numerical computations for the extrapolations increase both the memory and the CPU time requirements, when compared to the single use of a Multigrid cycle. The results for all other cases presented analogous behavior.

4.2 Use of extrapolations during the Multigrid cycles

In this approach, the Topological Epsilon Extrapolator was employed during the Multigrid cycles, i.e., for each group of five iterations, numerical solutions were combined with the extrapolator to generate a new solution vector. Such vector serves as an initial estimate for the next Multigrid cycle. Completing a new group of five iterations, a new extrapolation is done and so on, until the achievement of a determined stopping criterion. This study was also made for the Burgers' equations.

Table 2 presents results for N = 4097x4097 with tolerance of 10^{-15} for the Poisson-type equation and results for N = 1025x1025 with tolerance of 10^{-12} for Burgers' equations for u and v (velocity components). Results concerned to other cases are omitted since they present similar behavior. In cited table, only Multigrid (MG) and Topological Epsilon Extrapolator during the Multigrid (MG + Topological Epsilon) are compared.

Table 2. Results for studied parameters. Extrapolations during the Multigrid cycles.

Poisson-type Equation						
Methodology	t_{CPU} **[s]**	***M*** **[MB]**	$\|R\|_2$	$\hat{q}^{(k)}$	$\|E_n\|_\infty$	$\|E_n\|_2$
Pure MG	351.579	1.844	$6.270\text{x}10^{-16}$	$4.148\text{x}10^{-2}$	$3.673\text{x}10^{-17}$	$1.818\text{x}10^{-17}$
MG + Topological Epsilon	364.699	3.946	$2.320\text{x}10^{-16}$	$1.831\text{x}10^{-2}$	$1.912\text{x}10^{-18}$	$8.251\text{x}10^{-19}$

Burgers' Equations										
Methodology	t_{CPU} **[s]**	***M*** **[MB]**	***u***				***v***			
			$\|R\|_2$	$\hat{q}^{(k)}$	$\|E_n\|_\infty$	$\|E_n\|_2$	$\|R\|_2$	$\hat{q}^{(k)}$	$\|E_n\|_\infty$	$\|E_n\|_2$
Pure MG	3.25	290.1	$1.38\text{x}10^{-13}$	$2.68\text{x}10^{-3}$	$3.39\text{x}10^{-12}$	$6.69\text{x}10^{-13}$	$2.24\text{x}10^{-10}$	$1.17\text{x}10^{-3}$	$1.54\text{x}10^{-12}$	$6.32\text{x}10^{-13}$
MG + Topological Epsilon	3.349	596.1	$5.94\text{x}10^{-14}$	$6.24\text{x}10^{-3}$	$2.94\text{x}10^{-13}$	$1.33\text{x}10^{-13}$	$7.20\text{x}10^{-11}$	$2.04\text{x}10^{-2}$	$9.65\text{x}10^{-14}$	$1.27\text{x}10^{-14}$

Taking into account this case, it can be noticed a slight increase of CPU time when results are compared to pure Multigrid (i.e., Multigrid without the use of any extrapolation method) for both Poisson-type and Burgers' equations problems. For the non-dimensional norm of residual, the use of the extrapolator showed to be more efficient than the pure Multigrid for both problems; the same behavior is observed to the average convergence rate. When the iteration error is analyzed, its magnitude is reduced of about 95% for the Poisson-like equation and 98% for Burgers' equations.

Figure 2 presents the behavior of l_2-norm of residual (in logarithmic scale) as function of the number of iterations for the Poisson-type equation.

According to Fig. 2, until the 5th iteration, the residual reduces similarly for both methodologies. However, since in the 6th iteration occurs the first extrapolation, the norm of residual for the methodology of Multigrid with extrapolations presents a smaller magnitude than the use of the pure Multigrid (about 200 times smaller). It motivates the achievement of the stopping criterion in the 9th iteration for the MG + Topological Epsilon while for the pure Multigrid 11 iterations are needed. The results for all other cases presented analogous behavior.

In order to achieve more accurate results of CPU time, additional tests were made with N varying from 33×33 to 8193×8193 and tolerance of 10^{-20} for the Poisson-type equation and with N varying from 5×5 to 1025×1025 and tolerance of 10^{-12} for Burgers' equations. In such tests, it was realized that the use of extrapolations is more advantageous for higher values of N, since it reduces the CPU time compared to the pure Multigrid. Such affirmation is based on the evaluation of the speed-up of the pure Multigrid in relation to the MG + Topological Epsilon. The speed-up (S_p) is defined as the ratio of CPU time of two algorithms (Galante, 2006):

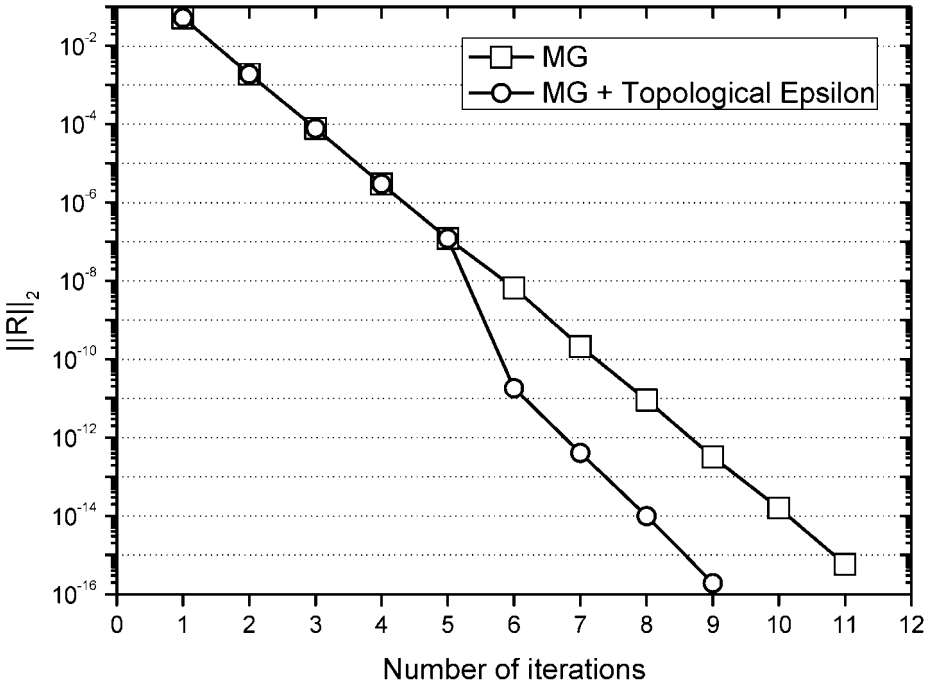

Figure 2. Behavior of $||R||_2$ as function of the number of iterations for N = 4097x4097

$$S_p = \frac{t_{CPU}(\text{algorithm A})}{t_{CPU}(\text{algorithm B})}, \tag{25}$$

Table 3 presents the speed-up of pure Multigrid (algorithm A) in relation to MG + Topological Epsilon (algorithm B) for a tolerance of 10^{-20} to the Poisson-type equation and a tolerance of 10^{-12} to the Burgers' equations.

Table 3. Speed-up of the pure Multigrid in relation to the MG + Topological Epsilon; $\varepsilon = 10^{-20}$ for the Poisson-type Equation and $\varepsilon = 10^{-12}$ for the Burgers' equations.

N	$S_p(Poisson)$	$S_p(Burgers)$
5x5	-	1.000
9x9	-	1.000
17x17	-	0.500
33x33	0.966	0.500
65x65	0.929	0.714
129x129	0.908	0.769
257x257	0.930	0.900
513x513	0.938	0.958
1025x1025	1.013	0.970
2049x2049	1.011	-
4097x4097	1.011	-
8193x8193	1.011	-

Based on Tab. 4, it is observed that for $N = 1025 \times 1025$, or larger, the values of speed-up are higher than the unity, indicating that MG + Topological Epsilon is faster than the pure Multigrid. Besides, a curve fitting by total least squares was made:

$$t_{CPU}(N) = c\, N^{p}, \tag{26}$$

where p describes the complexity of the algorithm, c is a coefficient dependent on both the chosen method and the solver and N is the number of unknowns of the system. Results for both the Poisson and Burgers' equations are presented in Tab. 4.

Table 4. Values of *p* and *c* of Eq. (26), for both algorithms of studied methodologies.

Metodology	Poisson		Burgers	
	c	***p***	***c***	***p***
Pure MG	0.0755	1.03258	1.3547×10^{-6}	1.05962
MG + Topological Epsilon	0.0850	1.02957	3.599×10^{-6}	0.99137

According to Tab. 5, both methodologies, pure Multigrid and MG + Topological Epsilon, are nearly equivalent. The advantage of the latter methodology is the fact that it accelerates the Multigrid method with the increase of the number of unknowns of the problem.

From Fig. 3, the RAM memory requirement increases with the use of extrapolators; the relative increment, however, presents an asymptotic behavior, which indicates that such methodology can be applied to problems with refined grids.

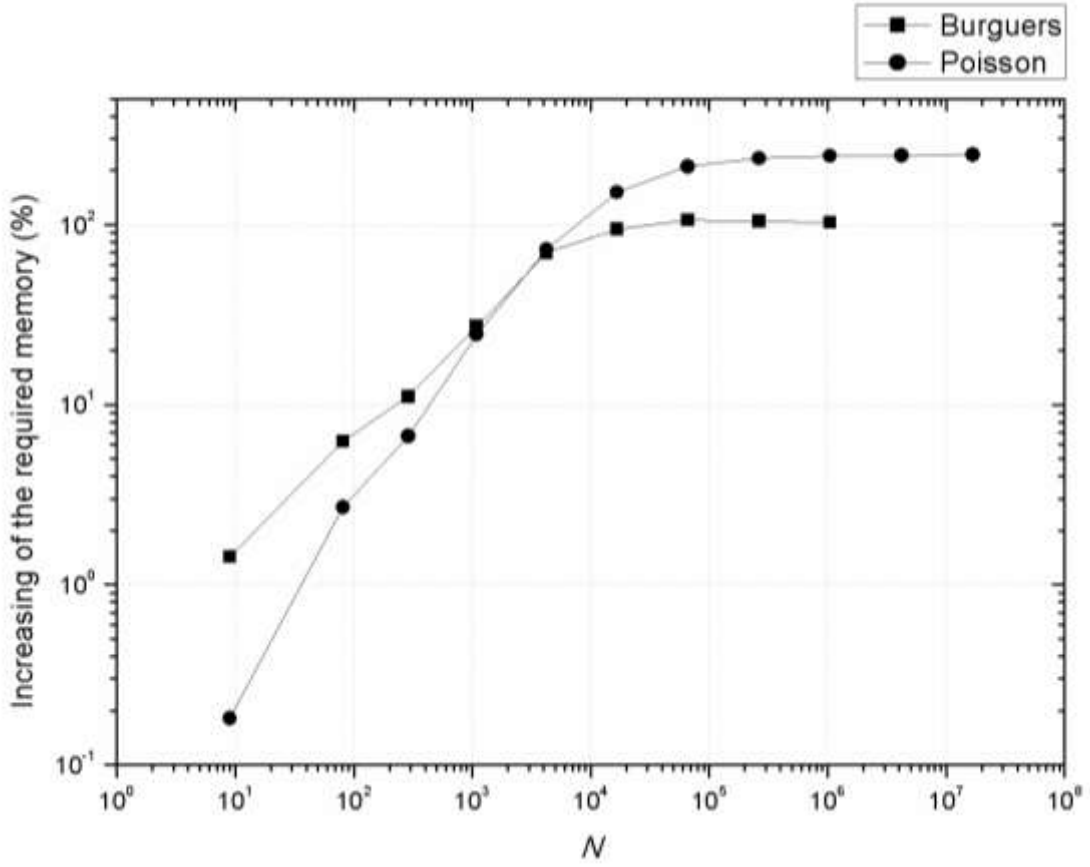

Figure 3. Relative increment of RAM memory requirement as function of the number of unknowns of the problems.

5 CONCLUSION

Based on the results presented in this work:

1) The use of extrapolators in the end of the Multigrid, for the Poisson-type equation, increases the CPU time in relation to an additional Multigrid cycle.

2) The RAM memory requirement is higher when an extrapolator is employed; however, the increasing is asymptotic with the growth of problem size.

3) The use of the Topological Epsilon Extrapolator, both in the end of and during the Multigrid cycles, was the methodology which reduced more considerably the non-dimensional residual norm, the convergence factors and the magnitude of iteration error for the Poisson-type equation.

4) The use of the Topological Epsilon Extrapolator during the Multigrid cycles reduced considerably the non-dimensional residual norm, the convergence factors and the magnitude of iteration error for the Burgers' equations.

5) The CPU time for extrapolations during the Multigrid cycles is nearly equal to the obtained for the pure Multigrid; otherwise, the methodology of using an extrapolator is slightly faster for more refined grids (when the number of unknowns increases).

ACKNOWLEDGEMENTS

The authors would acknowledge the infrastructure provided by the Department of Mechanical Engineering of UFPR (Federal University of Paraná). Authors also acknowledge the financial support of CNPq (National Council for Scientific and Technological Development), AEB (Brazilian Space Agency) by Uniespaço Program, CAPES (Coordination for the Improvement of Higher Education Personnel) and Araucaria Foundation.

REFERENCES

Brezinski C., Zaglia R. M., 2008. A Review of Vector Convergence Acceleration Methods, With Applications to Linear Algebra Problems. *Int. J. Quantum Chem.*, 109, pp. 1631–1639.

Briggs W. L., Henson V. E., McCormick S. F., 2000. *A Multigrid Tutorial*, 2 ed. Philadelphia: SIAM.

Burden R. L., Faires J. D., 2005. *Numerical analysis*, eighth edition, Cengage Learning – Brooks Cole, Florence.

Delahaye J. P., 1988. *Sequence Transformations*. Berlin: Springer.

Delahaye J. P., Germain-Bonne B., 1982. *The set of logarithmically convergent sequences cannot be accelerated*. Philadelphia: SIAM.

Galante G., 2006. *Multigrid Parallel Methods on Non-structured grids Applied to Simulation of Computational Fluid Dynamic and Heat Transfer Problems* (in Portuguese). PhD Thesis. Federal University of Rio Grande do Sul, Porto Alegre, RS.

Gao Q., Jiang Z., Liao T., Song K., 2010. Application of the vector ε and ρ extrapolation methods in the acceleration of the Richardson–Lucy algorithm, *Opt. Commun*., 283, pp. 4224–4229.

Graves-Morris P. R., Roberts D. E., A. Salamc, 2000. The epsilon algorithm and related topics, J. Comput. *Appl. Math*., 122, pp. 51-80.

Hackbush W., 1985. *Multigrid Methods and Applications*. Berlin: Springer-Verlag.

Liu H., Yang B., Chen Z., 2015. Accelerating algebraic Multigrid solvers on NVIDIA GPUs. *Comput. Math. Appl.*, 70, pp. 1162-1181.

Maliska C. R., 2004. *Computational Heat Transfer and Fluid Mechanics* [in Portuguese], 2. ed. Rio de Janeiro: Livros Técnicos e Científicos.

Martins M. A., Marchi C. H.. 2008. Estimate of iteration errors in Computational Fluid Dynamics, Num. *Heat Tr. B-Fund*., 53, pp. 234-245.

Mitin A. V., 1985. Linear extrapolation in an iterative method for solving systems of equations, U.S.S.R. *Comput. Math.* Math.+, 25-2, pp. 1-6.

Roy C. J., 2005. Review of Code and Solution Verification Procedures for Computational Simulation. *J. Comput. Phys*., 205, pp. 131-156.

Shen J., Wang F., Xu J., 2000. Finite element Multigrid preconditioner for Chebyshev–collocation methods. *Appl. Numer. Math.*, 33, pp. 471-477.

Shih T. M., Tan C. H., Hwang B. C., 1989. Effects of grid staggering on numerical scheme. *Int. J. Num. Met. Fluids*, 9, pp. 193-212.

Tannehill J. C., Anderson D. A., Pletcher R. H., 1997. *Computational Fluid Mechanics and Heat Transfer*, Washington: Taylor & Francis.

Thekale A., Grandl T., Klamroth K., Rüde U., 2010. Optimizing the number of Multigrid cycles in full Multigrid algorithm. *Numer. Linear Algebr*., 17, pp. 199-210.

Trottenberg U., Oosterlee C., Schüller A., 2001. Multigrid, St Augustin, Germany: Academic Press.

Wynn P., 1962. Acceleration techniques for iterated vector and matrix problems. *Math. Comput*., 16, pp. 301-322.

Zhang J., 1996. Acceleration of Five-Point Red-Black Gauss-Seidel in Multigrid for Poisson equation. *Appl. Math. Comput*., 80, pp. 73-93.

Zhang W., Zhang C.H., Xi G., 2010. An explicit Chebyshev pseudospectral Multigrid method for incompressible Navier–Stokes equations. *Comput. Fluids*, 39, pp. 178-188.