Applied Mathematical Modelling 36 (2012) 4665–4676

Contents lists available at SciVerse ScienceDirect

Applied Mathematical Modelling

journal homepage: www.elsevier.com/locate/apm

Optimized partial semicoarsening multigrid algorithm for heat diffusion problems and anisotropic grids

F. Oliveira [a,*], M.A.V. Pinto [b], C.H. Marchi [b], L.K. Araki [b]

[a] *State University of Ponta Grossa, Department of Mathematics and Statistics, Campus Uvaranas, Block L, 4748 Carlos Cavalcanti Street, 84030-900 Ponta Grossa, PR, Brazil*

[b] *Federal University of Paraná, Department of Mechanical Engineering, Curitiba, PR, Brazil*

ARTICLE INFO

Article history:
Received 18 April 2011
Received in revised form 16 November 2011
Accepted 27 November 2011
Available online 3 December 2011

Keywords:
Coarsening algorithm
Geometric multigrid
Restriction operator
V-cycle
Correction scheme

ABSTRACT

The purpose of this work is to reduce the CPU time necessary to solve three two-dimensional linear diffusive problems governed by Laplace and Poisson equations, discretized with anisotropic grids. The Finite Difference Method is used to discretizate the differential equations with central differencing scheme. The systems of equations are solved with the lexicographic and red–black Gauss–Seidel methods associated to the geometric multigrid with correction scheme and V-cycle. The anisotropic grids considered have aspect ratios varying from 1/1024 up to 16,384. Four algorithms are compared: full coarsening, semicoarsening, full coarsening followed by semicoarsening and partial semicoarsening. Three new restriction schemes for anisotropic grids are proposed: geometric half weighting, geometric full weighting and partial weighting. Comparisons are made among these three new schemes and some restriction schemes presented in literature: injection, half weighting and full weighting. The prolongation process used is the bilinear interpolation. It is also investigated the effects on the CPU time caused by: the number of inner iterations of the smoother, the number of grids and the number of grid elements. It was verified that the partial semicoarsening algorithm is the fastest. This work also provides the optimum values of the multigrid components for this algorithm.

1. Introduction

The multigrid method [1] belongs to the group of iterative solvers and is one of the most efficient and widespread methods to solve large systems of linear equations [2]. It consists on the transference of information among a refined grid, in which the numerical solution is desired, and coarse grids, in which numerical smoothers (or solvers) are more efficient, by means of restriction and prolongation operators. Several components can be modified in the multigrid method, such as the smoother (solver), the sequence in which each grid is employed (defined as multigrid cycle: V-cycle, W-cycle, F-cycle, among other ones) and the restriction and the prolongation operators. The kind of information that is transferred among the grids defines the multigrid scheme, which can be the CS one (correction scheme, in which only the residual is transferred to the coarser grids) or the FAS (full approximation scheme, in which both the residual and the solution are transferred to the coarser grids) [3].

Over the years, multigrid has become closely intertwined with Computational Fluid Dynamics (CFD), and has become an ingredient in major CFD codes. However, the full theoretical multigrid efficiency has not yet been achieved for realistic engineering applications in CFD [4]. It is observed, for example, in grids with cells presenting high-aspect ratios, which is known

[*] Corresponding author. Tel.: +55 42 33203050/44203288.
E-mail addresses: faboliveira@uepg.br (F. Oliveira), marcio_villela@ufpr.br (M.A.V. Pinto), marchi@ufpr.br (C.H. Marchi), lucaraki@ufpr.br (L.K. Araki).

0307-904X/$ - see front matter © 2011 Elsevier Inc. All rights reserved.
doi:10.1016/j.apm.2011.11.084

as anisotropy. Another reason is the fact that the governing equations may show elliptic or parabolic behavior in one part of the domain and hyperbolic behavior in another part. According to the methodology commonly used for the study of theoretical multigrid efficiency, the observed difficulties should be isolated, analyzed and solved systematically using a carefully constructed series of model problems [4].

The focus of this article is the optimization of the geometric multigrid method in anisotropic grids. This anisotropy can be related to: (1) different quantities of nodes in each direction (N_x and N_y); (2) different distance between two neighbor nodes in each direction, being $h_x = 1/(N_x - 1)$ and $h_y = 1/(N_y - 1)$; and (3) different lengths of the domain in each direction (C_x and C_y). In this work, the studied anisotropy corresponds to $h_x \neq h_y$, $N_x \neq N_y$ and $C_x = C_y = 1$. The anisotropy is evaluated by the aspect ratio (Q), given by

$$Q = h_x/h_y. \tag{1}$$

For $Q = 1$, there is no anisotropy and the problem is called isotropic.

Several strategies have been used recently trying to improve the multigrid performance for anisotropic problems. Liu [5] proposed new transfer operators based on the finite difference approximations, which are employed in the prolongation of the residual error from the coarser to the finer grids. The prolongation operators for anisotropies were also studied by Gee et al. [6], in order to obtain an efficient methodology that can be applied to the algebraic multigrid method (AMG), conforming the operator a desired sparcity pattern, which provides an effective method. Bin Zubair et al. [7] presented techniques to create general adaptations for high-dimensional problems as well as proposed grid-coarsening strategies to handle anisotropies induced by geometric factors; such strategies involves doubling and quadrupling coarsening and second and forth-order finite difference operators.

Some other works focus the use of semicoarsening (SC) algorithms for anisotropic grids. Larsson et al. [8] used a SC algorithm – in the case, a conditional SC algorithm – for heat transfer and flow problems. Mulder [9] employed multiple SC for convection problems, while Naik and Van Rosendale [10] created a variant of the Mulder's algorithm, which could be applied to both elliptic and hyperbolic problems. Hyperbolic problems with SC algorithms were also studied by Radespiel and Swanson [11] for the hypersonic viscous flow over an airfoil. Zhang [12] proposed a partial semicoarsening (PSC) algorithm, comparing the results with the full coarsening (FC) in grids whose aspect ratios were 1/2, 1/4, 1/8 and 1/16 for a two-dimensional Poisson-type equation. The FC and the PSC algorithms were also employed by Ge [13], who also used a forth-order compact difference discretization scheme for anisotropic grids to solve a three-dimensional Poisson-type equation in a cubic domain, with point and plane Gauss–Seidel smoothers.

Wang and Zhang [14] combined the multigrid method with Richardson extrapolations to solve isotropic and anisotropic 2D Poisson-type equations in rectangular grids, in order to increase a forth-order discretization method to a sixth-order one. The line-Gauss–Seidel smoother was used in the anisotropic case and it was concluded that this technique provides accurate solutions with good computational performance. Another approach was taken by Fischer and Huckle [15,16], who proposed new algorithms for the two-level Toeplitz matrices, which can be used even when the anisotropy is not only along a coordinate axis, but also along other directions.

According to Larsson et al. [8], for simple isotropic problems, pointwise smoothers, like the point Gauss–Seidel, reduce efficiently the error in all the coordinated directions, which cannot happen for anisotropic problems. In this case, a pointwise smoother is efficient in just some directions, or even in only one direction. Thus, there are essentially two possibilities to ensure the convergence of the multigrid method for anisotropic problems: (1) to adapt the smoother, so that it operates efficiently in all the directions; or (2) to adapt the restriction and prolongation operators. In studies involving the PSC algorithm [12,13], the four-coloring and red–black Gauss Seidel were more robust for moderate anisotropies, while the plane Gauss–Seidel relaxation showed to be the most efficient for strong anisotropies. These results encourage the performance analysis of the point lexicographic Gauss–Seidel (GS–LEX) and the red–black one (GS–RB), combined to the proposal of new restriction operators.

Motivated by all the previous works, this study is made in order to provide an optimized geometric multigrid method for anisotropic grids. Such optimization involves the study of the multigrid components, like: the number of inner iterations of the solver (ν); the number of grid levels (L); the number of elements (E = 4096; 65,536; 262,144 and 1,048,576); the aspect ratio – Q for the range from 1/1024 up to 16,384; the solver – lexicographic and red–black Gauss–Seidel; the restriction operators – injection (INJ), half-weighting (HW), full-weighting (FW), geometric half-weighting (GHW), geometric full-weighting (GFW) and partial weighting (PW), the last three ones proposed in this work; and the coarsening algorithms – the full coarsening (FC), the semicoarsening (SC), the full coarsening followed by semicoarsening (FC–SC) and the partial semicoarsening (PSC).

The main intentions of this work are: (1) testing strong anisotropies ($0 < Q << 1$ and $Q >> 1$), for values not previously presented in literature and using four different coarsening algorithms; (2) proposing a suitable restriction operator for anisotropic grids; and (3) optimizing the multigrid components for the best algorithm found in the first objective.

2. Mathematical and numerical models

2.1. Mathematical models and discretization

Three two-dimensional heat transfer problems, with Dirichlet boundary conditions, are studied in this work, being governed by the following differential equation in the Cartesian coordinate system [17]:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = S, \tag{2}$$

where x and y are the coordinate directions, T is the temperature, and S is a source term, which can be null (providing the Laplace equation) or either a constant or a function of x and/or y (providing a Poisson type equation). While the Laplace equation is employed twice, for two different sets of boundary conditions, listed in Table 1, the Poisson type equation has as source term $S = -2[(1-6x^2)y^2(1-y^2) + (1-6y^2)x^2(1-x^2)]$. In order to distinguish the two sets of boundary conditions employed for the Laplace equation, the labels Laplace 1 and Laplace 2 are used in this work, according to the boundary conditions listed in Table 1.

The Finite Difference Method [18] is employed in the discretization of Eq. (2), with the use of the second order central differencing scheme (CDS). The unitary square domain is divided into N_x nodes in the x-direction and N_y nodes in the y-direction, using uniform grids in each direction, totalizing N nodes ($N = N_xN_y$). Analogously to the concept of nodal element in one-dimensional grids, in which the element is delimited by two neighbor nodes, separated by a distance h_x, in this work a two-dimensional element is delimited by four neighbor nodes, which are located in the vertices of a rectangle, whose sides measures h_x and h_y in the x and y directions, respectively. The number of elements, in this work, is denoted by E. For each single node, a linear equation arises from the discretization process and, for the whole domain, a system of linear equations of the type

$$\mathbf{AT} = \mathbf{b}, \tag{3}$$

must be solved. By the discretization procedures applied, the coefficients matrix $\mathbf{A}$ in Eq. (3) is pentadiagonal, symmetric and positive definite; $\mathbf{T}$ is the solution temperature vector and $\mathbf{b}$ is the independent vector.

2.2. Multigrid method: interpolation operators and other details

The system of linear equations, Eq. (3), is solved using the geometric multigrid method, as described by Wesseling [19], with the correction scheme (CS). According to Fletcher [20], when compared to CS, the full approximation scheme (FAS) demands 5–10% more computational effort in each multigrid cycle, once FAS needs to restrict both the residual and the numerical solution in the coarser grids. Among the several multigrid cycling schemes available in literature, the V-cycle was the chosen one by its simplicity for programming implementation and by its smaller computational work involved: W-cycles, for example, are roughly 50% more expensive than V-cycles [21].

While bilinear interpolation operator [22] was chosen for the prolongation in all the studied cases, six different operators are compared for the restriction: injection (INJ) [1,19,21]; half-weighting (HW) and full-weighting (FW) [23]; geometric half-weighting (GHW), geometric full-weighting (GFW) and partial weighting (PW). These last three operators (GHW, GFW and PW, in this order) are proposed in this work and present as stencils, for the coarsening in the x-direction:

$$I_h^H = \frac{1}{4Q+4}\begin{bmatrix} 0 & Q & 0 \\ 1 & 2Q+2 & 1 \\ 0 & Q & 0 \end{bmatrix}_h^H, \quad I_h^H = \frac{1}{2(6Q+2)}\begin{bmatrix} Q & 2Q & Q \\ 2 & 4Q & 2 \\ Q & 2Q & Q \end{bmatrix}_h^H, \quad I_h^H = \frac{1}{4}\begin{bmatrix} 0 & 0 & 0 \\ 1 & 2 & 1 \\ 0 & 0 & 0 \end{bmatrix}_h^H, \tag{4}$$

in which Q is evaluated according to Eq. (1). In Eq. (4), I_h^H is the restriction operator with H and h representing the coarse and the fine grids, respectively. It must be noted that, for the PW operator (last stencil), the weighting factors exist only in the direction in which the coarsening is done. The ratio between grid spacings is chosen equal to two ($r = 2$), once this is nearly a universal practice [1].

Two error smoothers or solver algorithms are employed in this work, based on the Gauss–Seidel type iteration [22,23]: the point lexicographic Gauss–Seidel (GS–LEX) and the red–black Gauss–Seidel (GS–RB). The number of smoothing steps (or inner iterations) in each grid level is symbolized by v, and present, in this work, the same value for the pre- and the post-smoothing steps (v_1 and v_2, respectively), that is, $v = v_1 = v_2$.

In all the numerical simulations, the number of grid levels (L) was taken in a such way that $1 \leqslant L \leqslant L_{\max}$, where $L_{\max}$ is the maximum number of different grids which can be employed in the multigrid cycle, for a given grid spacings ratio. For example, if the $E = 512 \times 512$ elements grid is employed, with a ratio $r = 2$, all the auxiliary grids are 256×256, 128×128,

Table 1
Problem labels, boundary conditions and analytical solutions for Eq. (2).

Problem label	Boundary conditions	Analytical solution
Laplace 1	$T(0,y) = T(x,0) = 0$ $T(x,1) = x, T(1,y) = y$	$T(x,y) = xy$
Laplace 2	$T(0,y) = T(x,0) = T(1,y) = 0$ $T(x,1) = \sin(\pi x)$	$T(x,y) = \sin(\pi x)\frac{\sinh(\pi y)}{\sinh(\pi)}$
Poisson	$T(0,y) = T(x,0) = 0$ $T(1,y) = T(x,1) = 0$	$T(x,y) = (x^2 - x^4)(y^4 - y^2)$

64×64, 32×32, 16×16, 8×8, 4×4 and 2×2, which corresponds to the coarsest grid, with only one inner node, totalizing 9 grids (the basis and 8 auxiliary grids, that is, $L_{\max} = 9$).

Each V-cycle is repeated until the achievement of a given stop criterion, which, in this work, is based on the non-dimensional l_2-norm of the residual – the reference is the l_2-norm of the initial guess – as found in Trottenberg et al. [22] and Zhang [12]. The null-value was taken as the initial guess for the whole domain, except by the boundaries. The admitted tolerance (ε) was equal to 10^{-10} for all the analyzed cases in this work.

The numerical codes were generated using the Fortran 2003 Language, with the Intel 9.1 Visual Fortran Compiler and double precision floating-point arithmetic. All the numerical results were obtained in a computer with an Intel Core2Duo 2.66 GHz processor, 8 GB RAM and Windows XP 64 bits OS.

The main intention of this work is the minimization of the CPU time (t_{cpu}) needed to solve the two-dimensional heat transfer problems listed in Table 1. CPU time, in this work, is concerned about the time intervals spent with the grids generation (the basis and the auxiliary grids), the appliance of the initial guess, the coefficients evaluation and the solution of the system of linear equations represented by Eq. (3), until the achievement of the admitted tolerance. In order to evaluate the t_{cpu}, the subroutine CPU_TIME of the Intel 9.1 Visual Fortran is employed.

2.3. Coarsening algorithms

Four different coarsening algorithms are employed in this work for the geometric anisotropic problems:

(a) Full coarsening (FC): presented by Brandt [3]. The coarsening process is taken in both directions simultaneously. An example of such procedure is shown for two grids at Fig. 1(a).

(b) Semicoarsening (SC): presented by Mulder [9]. The coarsening process is taken in only one direction – the one which presents more elements. This algorithm is employed until the achievement of an isotropic grid. Such a procedure is employed in anisotropic problems in which the strong coupling direction is previously known. At Fig. 1(b) this algorithm is schematically shown for two grids in the x direction.

(c) Full coarsening followed by semicoarsening (FC–SC): presented by Pinto and Marchi [24]. In this case, firstly the FC algorithm is employed, as many times as possible; after that, the SC algorithm is used, until the achievement of the coarsest grid. Fig. 1(c) shows this procedure schematically for three grids in the x direction.

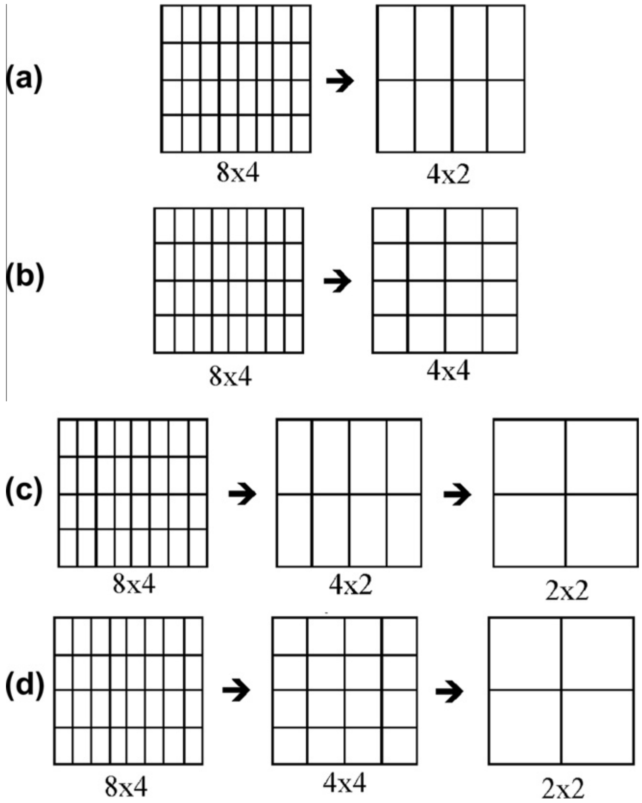

Fig. 1. Examples of the application of the coarsening algorithms: (a) FC, (b) SC, (c) FC–SC and (d) PSC. The numbers represent the number of elements in each grid in x and y directions.

(d) Partial semicoarseninig (PSC): presented by Zhang [12]. In this case, the SC algorithm is employed until the achievement of an isotropic grid. Subsequently, the FC algorithm is used. Such procedure is illustrated in Fig. 1(d) for three grids in the x direction. This algorithm will be called in this work as PSC 1.

3. Numerical results

With the purpose of analyzing the influence of different multigrid components on the t_{cpu} in anisotropic heat diffusion problems, about 2800 numerical simulations were done. The methodology employed consists on, for a given component of interest, keeping the other ones with a fixed value and, by comparison, choose the set of components which has shown the best performance. The numerical simulations belong to four categories: number of smoothing steps (inner iterations), restriction operators, number of grids and comparison among coarsening algorithms. An analysis of each one of these categories is done in the following subsections.

In this work, two variations of the PSC coarsening algorithm are considered, namely: PSC 1 and PSC 2. It must be noticed that the PSC 2 is obtained by the optimization of the multigrid components of the PSC 1. This optimization process is detailed in the next pages. Table 2 presents the overall results for the multigrid components.

3.1. Number of inner iterations (ν) and coarsening algorithms

The main purpose of this subsection is the establishment of the optimum number of smoothing steps (or inner iterations), $\nu_{optimum}$, which provides the minimum t_{cpu} for a given set of components for anisotropic grids. In order to reduce the number of numerical simulations and all the dependent variables for the t_{cpu} minimization, the number of levels is fixed equal to L_{max}, while the chosen restriction operator is the injection one and the error smoother (solver) is the lexicographic Gauss–Seidel (GS–LEX). Each one of the four coarsening algorithms (FC, SC, FC–SC and PSC 1) is employed for different aspect ratios (Q), being Q given by Eq. (1). The primary analyses are obtained for the Laplace 2 problem, for $Q = 1/64$, 1/16, 1/4, 1, 4, 16 and 64 and grids with $E = 65{,}536$ and 262,144 elements. The number of elements (E) is taken as the primary component in comparisons involving different aspect ratios, once it can be fixed as constant, differently from the number of nodes (N). For all the cases, when $Q = 1$, only the FC algorithm is employed, once the grid is already isotropic. In order to extend the results for the other two problems (Laplace 1 and Poisson), supplementary simulations are done for grids with $E = 262{,}144$ elements and aspect ratios of $Q = 1/16$, 1 and 16.

Except by the PSC 1 algorithm, the $\nu_{optimum}$ for all the algorithms is strongly influenced by the values of E and Q, assuming values in the range of 10–250. According to the numerical results, there is not a standard behavior which allows the determination of the $\nu_{optimum}$ for the FC, SC and FC–SC algorithms. Otherwise, the PSC 1 presents relatively small values for the $\nu_{optimum}$ in all the situations, associated to the smallest t_{cpu}, when compared to the other algorithms. The value of the $\nu_{optimum}$ for the PSC 1 algorithm, nevertheless, is influenced by the preferential direction of the oscillatory modes of the error, as can be seen in Fig. 2. Both the Laplace 1 and the Poisson problems present oscillatory profiles which are symmetric and, because of this, the value of $\nu_{optimum}$ is the same for $Q = 1/64$ (Fig. 2(a)) and $Q = 64$ (Fig. 2(b)), assuming the value of $\nu_{optimum} = 4$ for both aspect ratios and both problems. For the Laplace 2 problem, as the oscillatory behavior is not the same for both directions, the values of $\nu_{optimum}$ are different for $Q = 1/64$ and $Q = 64$, being equal to 2 and 4, respectively. The fact that, for the Poisson problem and $Q = 1/64$, the value of $\nu_{optimum}$ is also equal to 2 might be related to random fluctuations in the numerical simulations, the order of the nodes visited by the GS–LEX algorithm or even to the imprecision of the function employed in the t_{cpu} determination.

The similar behavior, which can be observed for all the curves presented in Fig. 2, is expected. It can be explained by the fact that, for both Laplace problems, the only difference is observed for the boundary conditions and, their evaluation does not influence on the CPU time. For the Poisson problem, the difference is observed for the source term and the boundary conditions; because of this, its CPU time is proportional to the Laplace problems ones. Although it is not presented in this work, such behavior is observed for all the values of Q.

Briggs et al. [1] and Trottenberg et al. [22] made detailed analyses of ν only for isotropic grids. Reminding that ν_1 and ν_2 are the number of pre- and post-smoothing, respectively, Briggs et al. [1] analyzed the Poisson 2D equation, verifying numerically that the recommended values are $\nu_1 = 2$ and $\nu_2 = 1$. On the other hand, Trottenberg et al. [22] also used the Poisson 2D equation, but with the use of rigorous Fourier analysis, and verified that $\bar{\nu} = \nu_1 + \nu_2 \leqslant 3$. It must be noticed, however, that

Table 2
Multigrid components for the PSC 1 and PSC 2 algorithms.

Component	PSC 1	PSC 2
Smoother (solver)	GS–LEX	GS–RB
Restriction operator	INJ	PW (anisotropic grids) FW (isotropic grids)
Number of inner iterations (ν)	Laplace 1: $\nu = 2$ for $Q < 1$ and $\nu = 4$ for $Q > 1$ Laplace 2: $\nu = 4$, $\forall Q$ Poisson: $\nu = 3$, $\forall Q$	$\nu = 1$ for all the problems and $Q \neq 1$ (anisotropic grids) $\nu = 2$ for all the problems and $Q = 1$ (isotropic grids)

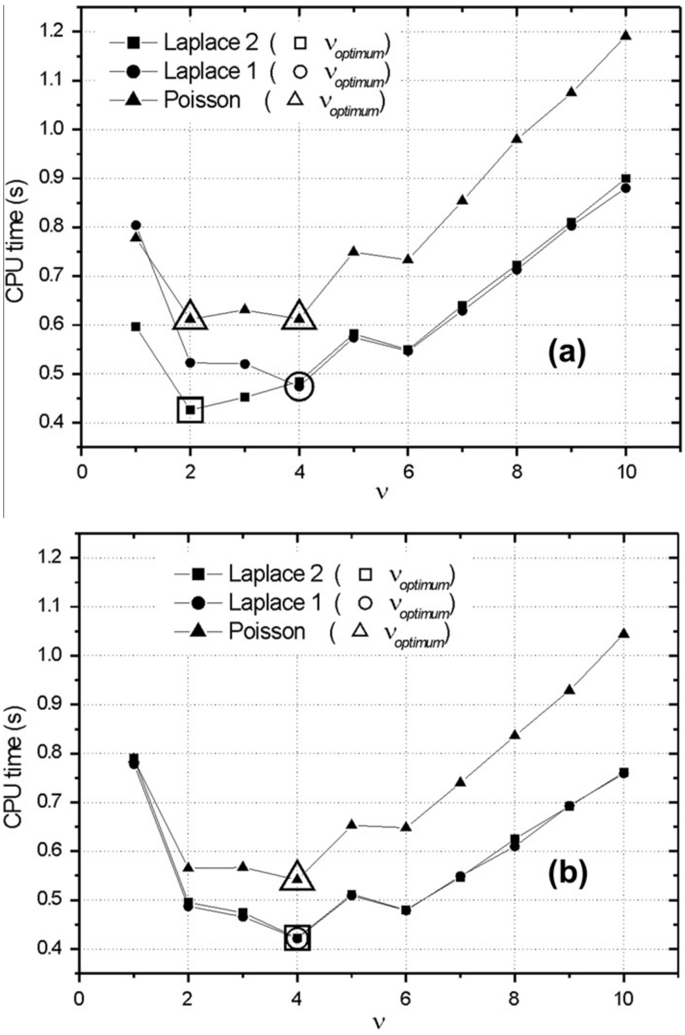

Fig. 2. CPU time versus ν for $E = 262{,}144$ and PSC 1: (a) $Q = 1/64$ and (b) $Q = 64$.

both Briggs et al. [1] and Trottenberg et al. [22] used the same $\bar{\nu}$ obtained in the isotropic grids studies for anisotropic ones, without any special Fourier analysis for this grid condition. This kind of analysis was made by Zhang [12], who studied the Poisson 2D equation discretized with anisotropic grids and employed $\nu_1 = \nu_2 = 1$; in this case, $\nu = \nu_1 = \nu_2 = 1$. These values are corroborated numerically in current work by the analysis of different quantities for ν, according to the results presented in the next subsection.

3.2. Restriction operators

Once the t_{cpu} obtained for the PSC 1 algorithm is the smallest one, for all the aspect ratios and for both grid sizes ($E = 65{,}536$ and 262,144 elements), some other simulations are conducted based on such algorithm. These analyses involve modifications in the choice of the error smoother (solver) and in the restriction operator, providing a modified PSC 1 algorithm, which is denoted in this work as PSC 2. Two smoothers (GS–LEX and GS–RB) and six restriction operators (INJ, HW, FW, GHW, GFW and PW) are tested, so as to find the value of $\nu_{optimum}$ which provides the smallest CPU time (t_{cpu}).

Since the PSC 1 algorithm is composed basically by two steps, the first based on the semicoarsening (SC) algorithm until the achievement of an isotropic grid, followed by the full coarsening (FC) algorithm, until the coarsest grid, the adopted procedure for the numerical simulations is made in two moves. In the first one, the components related to the FC algorithm are kept fixed (INJ as restriction operator and GS–LEX as solver) and both the restriction operator (INJ, HW, FW, GHW, GFW and PW) and the solver are analyzed for the SC algorithm. In the second move, the components related to the SC algorithm are now fixed, chosen accordingly to the results obtained in the first move, while the restriction operator (INJ, HW and FW) and the solver for the FC algorithm are analyzed. For these analyses, the number of inner iterations (ν) is kept constant and equal to the values obtained in the Section 3.1 as optimum components. Once the components for both the FC and the SC algorithms are chosen, then a study to determine the $\nu_{optimum}$ for the set of selected components is executed.

From the first set of numerical simulations, it is observed that the PW operator is the best one, for both the GS–LEX and the GS–RB solvers, although the GS–RB presents always a better performance than the GS–LEX. From the second set of simulations, the best results are obtained with the FW operator, associated to the GS–RB solver. These results are valid for anisotropic grids ($Q \neq 1$), once for the limit case of isotropic grids ($Q = 1$), the best results are observed for the FC algorithm, associated to the GS–RB solver; the influence of the restriction operator, in this case, is non-significant. In contrast, the use of the full weighting restriction (FW) and GS–RB solver for isotropic grids is a common practice, as can be seen in the works of Briggs et al. [1], Lai et al. [25], Lent and Vandewalle [26] and Weib et al. [27]. These authors, however, do not provide the motives of these choices and/or any reference to anisotropic grids.

As previously presented, the PSC 2 algorithm is obtained by the optimization of the multigrid components of the PSC 1 one, originating the Table 2. This optimization is made with a separate study of the components of the multigrid method in the two moves (SC and FC), which provide the following multigrid components: GS–RB solver, PW operator of the SC process and FW for the FC one. Therefore, the next step to the achievement of the minimum t_{cpu} is the determination of the new values for the $\nu_{optimum}$. It is needed once the other multigrid components were changed in comparison to the previous subsection. As can be seen in Fig. 3, $\nu = \nu_{FC} = \nu_{SC} = 1$ (where FC and SC, in this order, correspond to FC and SC blocks of the partial semicoarsening) provides the smallest t_{cpu}, for both types of anisotropy: $0 < Q < 1$ (Fig. 3(a)) and $Q > 1$ (Fig. 3(b)), which also demonstrate the better stability of the PSC 2 algorithm in comparison to the PSC 1. Although these results are related to the Laplace 2 problem, similar behavior is observed for the other problems. In the limit case of $Q = 1$ (isotropic grid), $\nu_{optimum} = 2$, for all the three analyzed problems. It must be noticed that it is not found in literature analyses of the number of inner iterations for each different block of the partial semicoarsening (PSC) algorithm, i.e., studies with $\nu_{SC} \neq \nu_{FC}$.

3.3. Number of grids (L)

In the last subsections, the number of grids used for the multigrid was kept invariable and equal to L_{max}. Once other components that influence in the t_{cpu} performance were previously studied, in this subsection the effect of L on the t_{cpu} performance is evaluated. For such study, all the coarsening algorithms are considered, with the $\nu_{optimum}$ values obtained in Sections 3.1 and 3.2. The basis problem for all the analyses is the Laplace 2, since it presents a different oscillatory behavior in each coordinate direction.

Fig. 4 summarizes the numerical results obtained for the effect of L on t_{cpu}, assuming the PSC 1 algorithm and $E = 262{,}144$ elements. Similar behavior is seen for the other coarsening algorithms, including the PSC 2 one. It is easily observed that the minimum t_{cpu} is achieved when the maximum number of grids is used, not depending on the aspect ratio (Q), whose definition was presented by Eq. (1). In Fig. 4, it also must be noted that the value of L_{max} varies according to the Q: for the isotropic grid, in which $E = 262{,}144 = 512 \times 512$ elements, the value of L_{max} is 9, as presented in Section 2.2. For anisotropic grids, using the PSC algorithm, L_{max} is the number of grids employed in the semicoarsening direction, until the achievement of an isotropic grid, summed with the maximum number of grids which can be employed in the multigrid cycle. For example, for $Q = 16$ and $E = 262{,}144 = 128 \times 2048$ elements with ratio $r = 2$, all the auxiliary grids are: 128×1024, 128×512, 128×256, 128×128, 64×64, 32×32, 16×16, 8×8, 4×4 and 2×2, which correspond to the coarsest grid, with only one inner node, totalizing 11 grids (the basis and 10 auxiliary grids, that is, $L_{max} = 11$).

The results of the current work are in agreement with the ones presented by Tannehill et al. [18]. They investigated the existence of the optimal components for the multigrid method in a two-dimensional Laplace problem, using an isotropic 128×128 elements grid, the FC algorithm and from 2 to 7 grids. According to their numerical results, for the work units, it was observed that using four or five grids gave nearly the same performance as using seven grids (which corresponds to L_{max}). Another study involving the existence of the optimal components for the multigrid method was made by Rabi and De Lemos [28]. They studied a two-dimensional steady-state conductive–convective problem, using a modest anisotropy ($Q = 3/2$) for a rectangular domain and the FC algorithm. Numerical results suggested the existence of an optimum value for the number of grids L and their recommendation is the use of at least 4 grid levels for the V-cycle. According the numerical results of the current work, the recommended value of L is always the L_{max}, which agrees to both works. It must be noticed, however, that both previous works [18,28] do not provide a general recommendation about the value of L for the best multigrid performance; in the current work, also, strong anisotropies were studied, differently from the previous cited ones.

3.4. Comparison of coarsening algorithms

The four basic coarsening algorithms (FC, SC, FC–SC and PSC 1) listed in Section 2.3 are compared in Fig. 5, along with the PSC 2, for aspect ratios varying from $Q = 1/64$ up to 64 and the Laplace 2 problem. This study is based on the results presented by Wesseling and Oosterlee [29], in which it was observed that the performance of the multigrid method is deteriorated by anisotropic grids, with poor performance or even divergence of the numerical solution [8]. As can be seen, for all the anisotropic grids, the performance of the PSC 2 is the best one, followed by the PSC 1. For both algorithms, the aspect ratio Q seems to have only little influence on the t_{cpu}, at least for modest anisotropies (moderate values of Q), as can be seen at Fig. 5: nearly constant values of the t_{cpu} for such algorithms. This behavior, however, is not observed for the other algorithms: both the FC and the FC–SC show increasing t_{cpu} values when the grid becomes more anisotropic – for values of $Q \to 0$ or $Q \to \infty$. On the

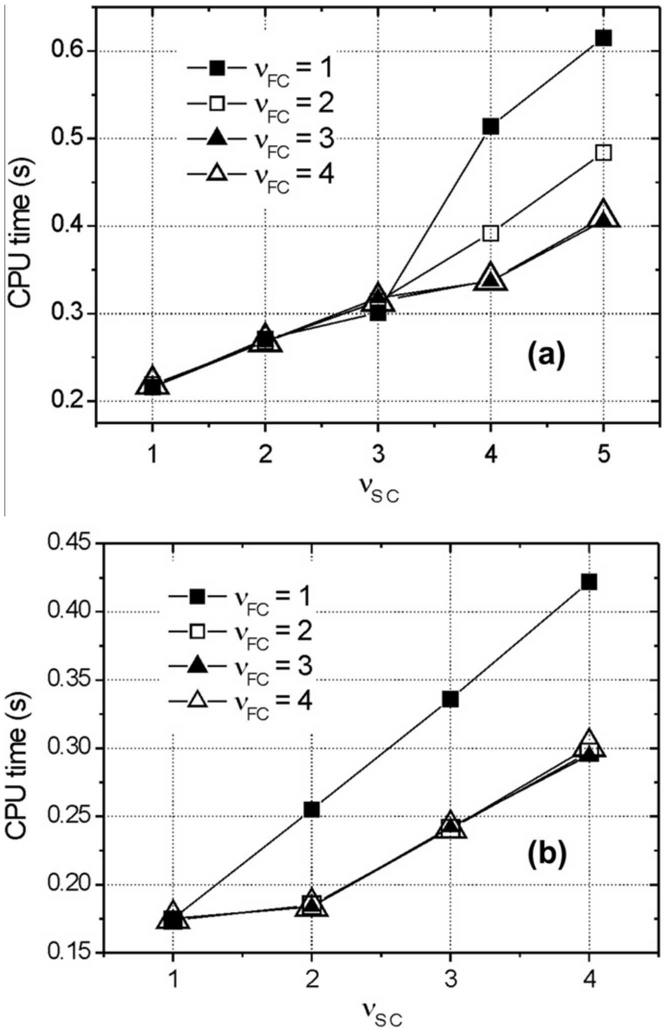

Fig. 3. CPU time versus ν_{SC} for $E = 262{,}144$ and PSC 2: (a) $Q = 1/64$ and (b) $Q = 64$.

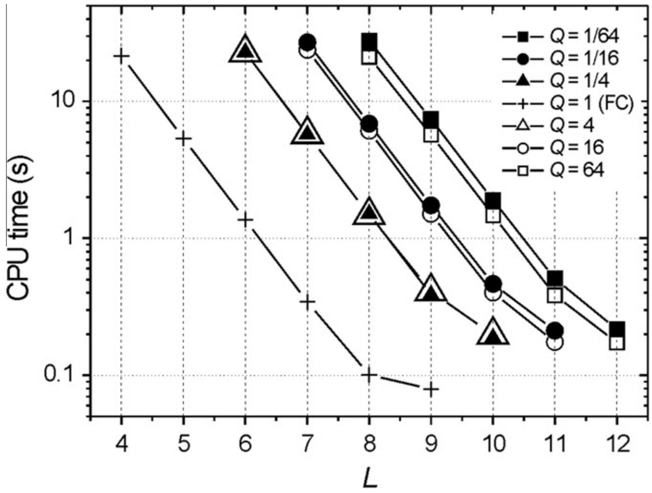

Fig. 4. CPU time versus number of grids (L) for $E = 262{,}144$, PSC 2 algorithm and Laplace 2 problem.

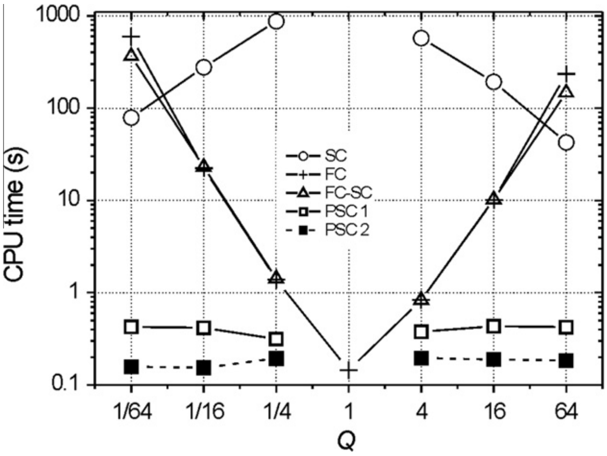

Fig. 5. CPU time versus aspect ratio (Q) for five coarsening algorithms, $E = 262{,}144$ and Laplace 2 problem.

other hand, the SC presents decreasing t_{cpu} values for other anisotropic grids; the performance of such algorithm, however, is always lower than the PSC 1 and the PSC 2, at least for the studied grids. This general behavior is also seen for the Laplace 1 and the Poisson problems.

These results confirm and extend a previous study presented by Zhang [12], in which he observed that the PSC 1 algorithm was faster than the FC one for a two-dimensional Poisson-type equation, with aspect ratios lower than the unity

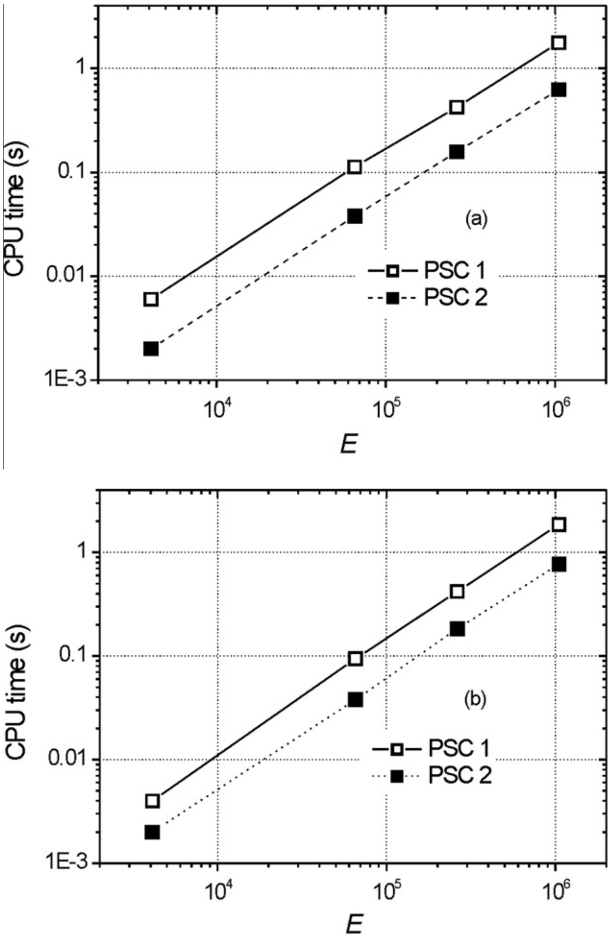

Fig. 6. CPU time versus number of elements (E) for Laplace 2 problem: (a) $Q = 1/64$ and (b) $Q = 64$.

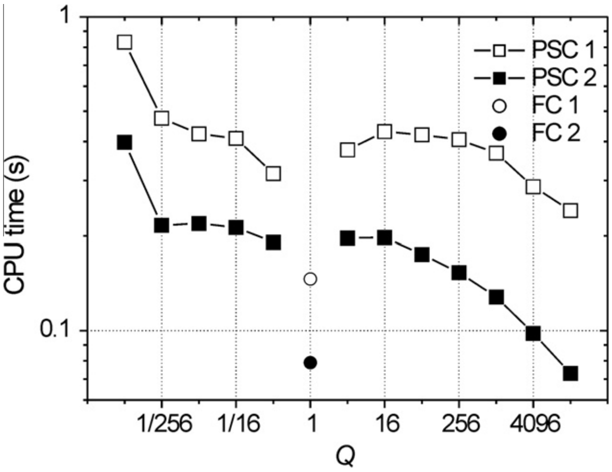

Fig. 7. CPU time versus aspect ratio (Q) for Laplace 2 problem.

($Q = 1/2$, 1/4, 1/8 and 1/16) and problems with at the most 1,048,576 elements. The current work also extends the Zhang's results for problems involving up to 67,108,864 elements and other anisotropic grids ($0 < Q << 1$ and $Q >> 1$). The difference of the performances among the algorithms PSC 1, PSC 2 and FC is more impressive in more anisotropic grids: the t_{cpu} ratio between the FC and the PSC 1 can be of only 2 (for $Q = 4$) up to 2786 (for $Q = 1/64$). Such behavior shows and confirms the superiority of the PSC algorithms (i.e., PSC 1 [12] and PSC 2) for anisotropic grids.

3.5. Analysis the multigrid components of PSC 1 and PSC 2

The PSC 1 and PSC 2 algorithms are compared for different numbers of elements, as seen at Fig. 6. The performance of the PSC 2 is always better than the one of the PSC 1 algorithm, taking only 1/2 t_{cpu} (or even less) of the equivalent simulation using the PSC 1. In the case of the Poisson problem, for example, with $Q = 16$, the PSC 2 need about 1/3 t_{cpu} of the PSC 1. This behavior is observed for all the problems and all the aspect ratios considered.

Fig. 7 also presents a comparison between the performances of the PSC 1 and the PSC 2 algorithms for the Laplace 2 problem, with 262,144 elements and different Q values (from $Q = 1/1024$ up to $Q = 16,384$). In this figure, for $Q = 1$ (isotropic grid), the PSC algorithms are not employed; in this case, the FC algorithm is used for comparison. The notations FC 1 and FC 2 represent the FC algorithm with the PSC 1 and PSC 2 multigrid components, respectively. Differently as seen in Fig. 5, for which only small or even moderate values of Q (modest anisotropies) were employed, in Fig. 7, strongly anisotropic grids are used. In this case, the t_{cpu} cannot be considered independent of the values of Q: for $Q > 1$, the t_{cpu} tends to decrease for more anisotropic grids, while for $Q < 1$, the t_{cpu} presents an opposite tendency. Such behavior can be attributed to the preferential directions of the oscillatory modes of error: in Laplace 2 problem, the preferential direction is related to the y-axis. When anisotropic grids are used, with $Q > 1$, the refinement of the grid coincides with the y-axis: the use of greater values of Q emphasizes this tendency. Otherwise, for $Q < 1$, the refinement of grid is made in the x-axis, opposite to the direction of the oscillatory error modes; in this case, the refinement of the grid (with lower values of Q) tends to worsen this effect. The use of more refined grids in the preferential direction of the oscillatory error modes tends to speed up the algorithms: in both cases, the better performance in Fig. 7 is observed for increasing values of Q.

Another interesting remark is related to the performance of the PSC 2 against the PSC 1 algorithm for $Q > 1$: the implemented modifications present more significant results for more anisotropic grids. For the Laplace 2 problem, while for $Q = 4$ the t_{cpu} evaluated to the PSC 2 corresponds to 52% of the t_{cpu} of the PSC 1 algorithm, for $Q = 16,384$ the t_{cpu} of the PSC 2 is only 30% of the t_{cpu} of the PSC 1 algorithm.

The results obtained in the current work could also be used, for example, for other algorithms, such as the one proposed by Wang and Zhang [14]. In this case, it is expected to obtain a better CPU time performance for that algorithm.

4. Conclusion

The geometric multigrid components for CS and anisotropic grids were studied in the current work. For the numerical analyses, three two-dimensional heat diffusion problems governed by Laplace and Poisson equations, with Dirichlet boundary conditions, were employed, discretized with the Finite Difference Method and second order CDS approximations.

In order to optimize the performance of the geometric multigrid method in heat diffusion problems, some components were studied in four algorithms: full coarsening (FC), semicoarsening (SC), full coarsening followed by semicoarsening (FC–SC) and partial semicoarsening (PSC 1). The optimization of the multigrid components of the PSC 1 generated an algorithm called PSC 2 in the current work. The studied multigrid components are: the number of inner iterations (ν); the number of grids (L); six restriction operators – injection (INJ), half-weighting (HW), full-weighting (FW), geometric half-weighting (GHW), geometric full-weighting (GFW) and partial weighting (PW); the number of elements of grids (E) and the grid aspect ratios (Q).

Based on numerical results, these are the most interesting remarks:

(1) The variation of the grid aspect ratio (Q) results in large variation of $\nu_{optimum}$ to the FC, SC and FC–SC algorithms. In PSC 1 algorithm, $\nu_{optimum}$ remains constant for all the aspect ratios. In the PSC 2 algorithm, the components which presents the smaller CPU time are: $\nu = \nu_{FC} = \nu_{SC} = 1$ for $Q \neq 1$ (anisotropic grids), which corroborates numerically the results presented by Zhang [12]. For $Q = 1$ (isotropic grids), the FC algorithm with $\nu = \nu_{FC} = 2$ presents the smaller CPU time.

(2) The maximum L results in the smallest CPU time for all the analyzed coarsening algorithms, all aspect ratio (Q), all number of elements of grids, and all problems.

(3) For anisotropic grids, red–black Gauss–Seidel solver results in smaller CPU time compared to the lexicographic Gauss–Seidel, corroborating the results of Zhang [12] and Ge [13].

(4) Three new restriction operators, the geometric half-weighting (GHW), the geometric full-weighting (GFW) and the partial weighting (PW) were proposed for anisotropic grids. Among all the studied operators for anisotropic grids, the PW presented the best performance. For isotropic grids, the choice of the restriction operator does not affect significantly the CPU time.

(5) For the same components (ν, Q, L and E), the comparison among four coarsening algorithms confirms the results of Zhang [12], in which the performance of the PSC 1 algorithm was the best one for anisotropic grids. The current work, however, also extends his results, once some even more anisotropic grids ($0 < Q << 1$ and $Q >> 1$) were considered, with aspect ratios varying from $Q = 1/1024$ up to $Q = 16,384$.

(6) Based on the PSC 1 algorithm, some studies were made in order to reduce the t_{cpu} associated to such algorithm. The new multigrid components of this algorithm, called in this work PSC 2, is more stable and presents better performance than the other algorithms, something that is more appreciable for stronger anisotropic grids (with the range from $Q = 1/1024$ up to $Q = 16,384$). It is in average, 2.3 times faster when compared to PSC 1.

(7) The qualitative behavior of the studied multigrid components remains the same to Laplace 1, Laplace 2 and Poisson problems.

There are also evidences which suggest that the results obtained in the current work can be extended to other partial differential equations, such the 3D Poisson equation, and 2D and 3D convection–diffusion equations. These results can also be used for example in Wang and Zhang [14] algorithm to improve the accuracy order and CPU time.

Acknowledgements

The first author would like to acknowledge the infrastructural support provided by the Laboratory of Numerical Experimentation (LENA) of the Department of Mechanical Engineering (DEMEC), Federal University of Paraná (UFPR) and the State University of Ponta Grossa (UEPG), by the financial support. The third author is scholarship of CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico, Brazil). The authors would like also to acknowledge the financial support provided by CNPq, CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, Brazil), Fundação Araucária (Paraná, Brazil) and the Brazilian Space Agency (AEB), by the UNIESPAÇO Program. The authors would like also to acknowledge the anonymous reviewers of this article.

References

[1] W.L. Briggs, V.E. Henson, S.F. McCormick, A Multigrid Tutorial, second ed., SIAM, Philadelphia, 2000.

[2] A. Thekale, T. Gradl, K. Klamroth, U. Rüde, Optimizing the number of multigrid cycles in the full multigrid algorithm, Numer. Linear Algebra Appl. 17 (2010) 199–210.

[3] A. Brandt, Multi-level adaptive solutions to boundary-value problems, Math. Comput. 31 (1977) 333–390.

[4] S.A. Mohamed, Optimally efficient multigrid algorithm for incompressible Euler equations, Int. J. Numer. Methods Heat Fluid Flow 18 (2008) 783–804.

[5] Z. Liu, Optimal multigrid methods with new transfer operators based on finite difference approximations, Acta Appl. Math. 111 (2010) 83–91.

[6] M.W. Gee, J.J. Hu, R.S. Tuminaro, A new smoothed aggregation multigrid method for anisotropic problems, Numer. Linear Algebra Appl. 16 (2009) 19–37.

[7] H. Bin Zubair, C.W. Oosterlee, R. Wienands, Multigrid for high-dimensional elliptic partial differential equations on non-equidistant grids, SIAM J. Sci. Comput. 29 (2007) 1613–1636.

[8] J. Larsson, F.S. Lien, E. Yee, Conditional semicoarsening multigrid algorithm for the Poisson equation on anisotropic grids, J. Comput. Phys. 208 (2005) 368–383.

[9] W.A. Mulder, A new multigrid approach to convection problems, J. Comput. Phys. 83 (1989) 303–323.

[10] N.H. Naik, J. Van Rosendale, The improved robustness of multigrid elliptic solvers based on multiple semicoarsened grids, SIAM J. Numer. Anal. 30 (1993) 215–229.

[11] R. Radespiel, R.C. Swanson, Progress with multigrid schemes for hyperbolic flow problems, J. Comput. Phys. 116 (1995) 103–122.

[12] J. Zhang, Multigrid method and fourth-order compact scheme for 2D Poisson equation with unequal mesh-size discretization, J. Comput. Phys. 179 (2002) 170–179.

[13] Y. Ge, Multigrid method and forth-order compact difference discretization scheme with unequal meshsizes for 3D Poisson equation, J. Comput. Phys. 229 (2010) 6381–6391.

[14] Y. Wang, J. Zhang, Sixth order compact scheme combined with multigrid method and extrapolation technique for 2D Poisson equation, J. Comput. Phys. 228 (2009) 137–146.

[15] R. Fischer, T. Huckle, Multigrid methods for anisotropic BTTB systems, Linear Algebra Appl. 417 (2006) 314–334.

[16] R. Fischer, T. Huckle, Multigrid solution techniques for anisotropic structured linear systems, Appl. Numer. Math. 58 (2008) 407–421.

[17] F.P. Incropera, D.P. DeWitt, T.L. Bergman, A.S. Lavine, Fundamentals of Heat and Mass Transfer, sixth ed., John Wiley & Sons, 2007.

[18] J.C. Tannehill, D.A. Anderson, R.H. Pletcher, Computational Fluid Mechanics and Heat Transfer, second ed., Taylor & Francis, Washington, 1997.

[19] P. Wesseling, An Introduction to Multigrid Methods, John Wiley & Sons, Philadelphia, 1992.

[20] C.A.J. Fletcher, Computational Techniques for Fluid Dynamics, v 1, Second ed., Springer, Berlin, 1997.

[21] C. Hirsch, Numerical Computational of Internal and External Flows, v 1, John Wiley & Sons, Chichester, 1988.

[22] U. Trottenberg, C. Oosterlee, A. Schüller, Multigrid, Academic Press, San Diego, 2001.

[23] R.L. Burden, J.D. Faires, Numerical Analysis, eighth ed., Brooks-Cole Publishing, 2004.

[24] M.A.V. Pinto, C.H. Marchi, Effect of Parameters of Multigrid Method CS and FAS on the CPU time for Laplace bidimensional Equation (in Portuguese), in: Proceedings of the XI Brazilian Congress of Thermal Sciences and Engineering (ENCIT), Curitiba, Brazil, 2006.

[25] M.C. Lai, C.T. Wu, Y.H. Tseng, An efficient semi-coarsening multigrid method for variable diffusion problems in cylindrical coordinates, Appl. Numer Math. 57 (2007) 801–810.

[26] J.V. Lent, S. Vandewalle, Multigrid waveform relaxation for anisotropic partial differential equation, Numer. Algorithm 31 (2001) 1–4.

[27] C. Weib, W. Karl, M. Kowarshick, U. Rude, Memory characteristics of iterative methods, in: Proceedings of Supercomputing Conference: High performance networking an computing, Portland, OR, 1999.

[28] J.A. Rabi, M.J.S. De Lemos, Optimization of convergence acceleration in multigrid numerical solutions of conductive–convective problems, Appl. Math. Comput. 124 (2001) 215–226.

[29] P. Wesseling, C. Oosterlee, Geometric multigrid with applications to computational fluid dynamics, J. Comput. Appl. Math. 128 (2001) 311–334.