

# The Effect of the Schedule on the CPU Time for 2D Poisson Equation

**Fabiane de Oliveira,**

State University of Ponta Grossa, Department of Mathematics and Statistics  
Campus Uvaranas, Block L, 4748 Carlos Cavalcanti Street, 84030-900 Ponta Grossa, PR, Brazil  
E-mail: [faboliveira@uepg.br](mailto:faboliveira@uepg.br)

**Marcio Augusto Villela Pinto, Carlos Henrique Marchi**

Federal University of Paraná, Department of Mechanical Engineering  
Curitiba, PR, Brazil  
E-mail: [marcio\\_villela@yahoo.com.br](mailto:marcio_villela@yahoo.com.br)  
[marchi@ufpr.br](mailto:marchi@ufpr.br)

**Abstract.** *The aim of this paper is to minimize the CPU time necessary for solving the 2D Poisson equation with Dirichlet boundary conditions. We study the influence of: several schedules for the inner iterations ( $v$ ) in the red-black Gauss-Seidel solver; number of levels of grids ( $L$ ); and unknowns number ( $N$ ) for grids of  $33 \times 33$  to  $8193 \times 8193$  nodes. It was verified that: (1) the small CPU time was obtained with the Hortmann schedule; (2) the order of  $p$  in the best schedule (type I Hortmann) is 0.99996.*

**Key-words:** *geometric multigrid, schedule, finite difference, inner iterations, CFD.*

## 1. Introduction

The multigrid method [2] belongs to the group of iterative solvers and it is one of the most efficient and widespread methods to solve large systems of linear equations [20]. It consists of 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. The transference is done through of restriction and 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) [1,21].

Throughout 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 [12].

There are many parameters that can be selected more or less arbitrarily when using the multigrid method. Among these parameters one can include: the type of cycle, CS or FAS scheme, number of iterations in the smoother, number of grids, the smoother, and restriction and prolongation schemes. Nevertheless, according to Ferziger and Peric [3], the ratio of CPU time from the work to the best set of parameters is probably smaller than a factor of two. However, Trottenberg et al. [21] state that the choice of parameter can significantly affect the efficiency of the multigrid. There exist works available in the literature concerning the influence of parameters on the multigrid method [15,16,13,18]. In Pinto et al. [15], the authors presented the optimum parameters of multigrid for 1D diffusion equations, advection-diffusion and Burgers; Rabi and De Lemos [16] studied the optimum parameters for a 2D advection-diffusion problem; Oliveira et al. [13] investigated types of rate coarsening for heat diffusion problems and anisotropic grids used the Poisson and Laplace equation; and Suero et al. [18] analyzed of algebraic multigrid parameters for two-dimensional steady-state heat diffusion equations.

The reduction CPU time for solving a determined problem results in the reduction of project costs. An increased in the efficiency method also allows, in the same time computation, to solving a problem in the grid more refined, namely, with more number of nodes; these

signify to obtaining a numeric solution with less discretization error [17], improving the quality and reliability of projects.

The purpose of this work is to determine which is the schedule resulting in smaller CPU time among the more used in the literature, considering the number of inner iterations. Denominate schedule the different ways to go through a cycle. The schedule considered for the number of inner iterations ( $v$ ) are: totally constant, dynamic, constant in the restriction in the prolongation, type I Hortmann and type II Hortmann and type I sawtooth and type II sawtooth. These schedules are present on the section 4. Not is founding in literature a study wide and detailed about the number of inner iterations and schedule how the presenting in this work. In this work we utilized CPU time instead work units due the reasons pointed by Trottenberg et al. [21].

The present work involves the heat diffusion linear two-dimensional, governed for Poisson equation with Dirichlet boundary conditions. The Finite Difference Method [18,19] utilized with central approximate scheme (CDS) and uniform grids.

## 2. Mathematical and Numeric Models

A two-dimensional heat transfer problem in the unitary square domain, with Dirichlet boundary conditions, is investigated in this work, being governed by the Poisson equation in the Cartesian coordinate system [11]:

$$\frac{\partial^2 T}{dx^2} + \frac{\partial^2 T}{dy^2} = S, \quad (1)$$

where  $x$  and  $y$  are the coordinate directions,  $T$  is the temperature, and  $S$  is a source term, defined as:

$$S = -2 \left[ (1 - 6x^2)y^2(1 - y^2) + (1 - 6y^2)x^2(1 - x^2) \right], \quad (2)$$

The boundary condition are  $T(0, y) = T(x, 0) = T(1, y) = T(x, 1) = 0$ . The analytical solution is given by  $T(x, y) = (x^2 - x^4)(y^4 - y^2)$ .

The Finite Difference Method [19] is employed in the discretization of Eq. (1), 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 ( $N = N_x \times N_y$ ). 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  $hx$ , in this work a two-dimensional element is delimited by four neighbor nodes, which are located in the vertices of a square, with sides of length  $hx$ . 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 must be solved. By the discretization procedures applied, the coefficients matrix  $A$  is pentadiagonal, symmetric and positive definite,  $T$  is the solution temperature vector and  $b$  is the independent vector.

## 3. Multigrid and details

The system of linear equations, Eq. (1), is solved using the geometric multigrid method, as described by Wesseling [22], with the correction scheme (CS). According to Fletcher [4], when compared to CS, the full approximation scheme (FAS) demands 5–10% more computational effort in each multigrid cycle, since 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 [9].

Bilinear interpolation and injection restriction operators [21] were chosen for the prolongation in all the studied cases.

The error smoothers or solver algorithms are employed in this work, based on the red-black Gauss–Seidel [14].

In all the numerical simulations, the number of grid levels ( $L$ ) was taken in a such way that  $1 \leq L \leq L_{\text{maximum}}$ , where  $L_{\text{maximum}}$  is the maximum number of different grids which can be employed in the multigrid cycle. The rate coarsening  $r = 2$  is used.

Each V-cycle is repeated until the achievement of a given stop criterion, which (in this work) is based on the non-dimensional l2-norm of the residual – the reference is the l2-norm of the initial guess – as found in [21,23]. The null-value was taken as the initial guess for the whole domain, except by the boundaries. The admitted tolerance was equal to  $10^{-10}$  for all the analyzed cases in this work. The numerical codes were generated using the Fortran Visual Studio 2008, using quadruple precision.

#### 4. Schedules

Some authors [2,10,21,22] have shown several strategies for determining the moment of changing of grid in the multigrid method, using strategy of dynamic type or cycle. In this work these strategies are called by schedules.

The dynamic schedule consists in monitoring the convergence rate of numeric solution which can be determined for the rate between the residual norm of two successive iterations. More details on dynamic strategy can be found in [1,7]. Wesseling [22] refers the dynamic strategy like “adaptative schedule”.

The cycle criterion consists in specifying the number of inner iterations in each level of grid. The literature, in general, uses the cycle strategy. Some applications of cycle strategy can be found in Hortmann et al. [10], where is defined the Hortmann schedule; Gerolymos and Vallet [6] and Wesseling [22] use the sawtooth schedule; Pinto et al. [15] and Oliveira et al. [13] used the schedule that consider the number inner iterations totally constant.

The schedules used in this work are presented in the following.

- a) Number iterations totally constant: The schedule with  $v$  totally constant is the most used in the literature [2]. It consists in using the same number iterations in all levels of grid, both in the restriction ( $v_1$ ) as the prolongation ( $v_2$ ), this is  $v_1 = v_2 = v$ ;
- b) Number iterations constant in the restriction and prolongation: The schedule that differentiates the inner iterations for restriction ( $v_1$ ) of the number iterations for prolongations ( $v_2$ ) has been used by Trottenberg et al. [21]. Is considered  $v_1 \neq v_2$  always constant in each one these process, being in general different from each other;
- c) Type I Hortmann: This schedule was proposed by Hortmann et al. [10] for full multigrid, seeking at solving a problem of laminar flow, without the objective of optimized the schedule. In this work, Hortmann schedule was adapted for V-cycle. It consist in specific the number iterations for restriction and prolongation that varies with the level in each grid ( $l$ ). Begins with  $v = 2$  and in subsequent grid have one increase of one unit the each level of grid. In the coarser grid  $v$  is equal to  $l+1$ ;
- d) Type II Hortmann: This schedule, proposed in this work, used one  $v$  bigger on the finest grids, beginning with  $v = L+1$ . In the subsequent grids this  $v$  have one decrease of one unit. In the coarser grid  $v$  is equal to 2;
- e) Type I sawtooth: According to [22], the type I sawtooth is the special case of V-cycle, where the smoothing is performed only in the prolongation;
- f) Type II sawtooth: This schedule smoothing only in restriction, in the prolongation is done only the transference of information. The smoothing is realized only in the finer grid. More details of this schedule can be found in [6];

- g) Dynamic: This schedule follows the scheme presented in [1]. It consist in carrying out a necessary number of inner iterations in each cycle  $V$ , until reach an inner tolerance established by  $Tol\_d$ . Where  $Tol\_d$  is defined as dynamic tolerance, in the other words, the rate between actual solution and previous solution.

## 5. Numeric Results

About 800 numeral simulations were performed with the purpose of analyzing the influence of different multigrid components on the CPU time in isotropic heat diffusion problems. The methodology employed consists on, for a given component of interest, keeping the other ones with a fixed value and, by comparison, choosing the set of components which has shown the best performance. The numerical simulations belong to two categories: number of smoothing steps (inner iterations) and number of grids. After performing the comparison among schedules. This optimization process is detailed in the next pages.

The optimum number of smoothing steps (or inner iterations), which provides the minimum CPU time for a given set of components was analyzed. In order to reduce the number of numerical simulations and all the dependent variables for the CPU time minimization, the number of levels is fixed equal to  $L_{\text{maximum}}$ , while the chosen restriction operator is the full weighting one and the error smoother (solver) is red-black Gauss–Seidel.

The primary analyses were obtained for (a) schedule with grids of 1,089; 4,225; 16,641; 66,049; 263,169; 1,050,625; 4,198,401; 16,785,409; and 67,125,249 nodes. It was observed that minimum CPU time was obtained for  $v=1$ . This result corroborates [5,18] with algebraic multigrid studies.

For tests with the (b) schedule,  $v_1$  could vary from 1 to 20 and  $v_2$  could vary from 1 to 4. It was verified that the smaller CPU time was obtained with one inner iteration. Therefore, the optimum number of (b) schedule is the totally constant schedule.

For the (g) schedule, were realized simulations for  $Tol\_d = 0.5, 0.2, E-01, E-02, E-03, E-04, E-05$  and  $E-06$ . Defined like dynamic tolerance the rate between actual solution and previous solution. The smaller CPU time was obtained with tolerance of  $E-04$ .

The comparison between the CPU time obtained with the (c) and (d) schedules (c) was done. It was verified that the best schedule is the (c) schedule. The (c) schedule is in average 3.82 times faster than (d) schedule.

The comparison between the CPU time obtained with the (e) and (f) schedules was done. It was verified that the best schedule is (e) schedule, in average 1.33 times faster than (f) schedule. Through of simulations it was verify that the (e) schedule present small CPU time when uses  $v_2 = 3$  and for (f) schedule for  $v_1 = 2$ .

The analysis of the schedule which provides the minimum CPU time for a given set of parameters was done. In order to reduce the number of numerical simulations, all dependent variables for the CPU time minimization and other parameters were fixed. The schedules described in Section 4 are compared in Figure 1. It is easily observed that the type I Hortmann is slightly better schedule, following by number of inner iterations totally constant, type I sawtooth and dynamic schedules.

We have computed the exponent.  $p$ , obtained using the least square method, for the function  $\text{CPUtime}(N) = cN^p$ , where  $p$  is the order solver associate to method used,  $c$  is the coefficient that depends of each method.  $N$  is the number of unknowns of the system.

For the ideal multigrid,  $p=1$ , meaning the computational effort increases linearly with the size of grid [1,9,21]. So, for a given hardware and compiler, as the value of  $p$  decreases, the efficiency of the algorithm increases. The Table 1 shown the coefficient for the geometric curve fitting. It was verify that the best schedule is the type I Hortmann. In the literature this schedule is not very used, but presents the best results than schemes used like sawtooth and number of inner iterations constant.

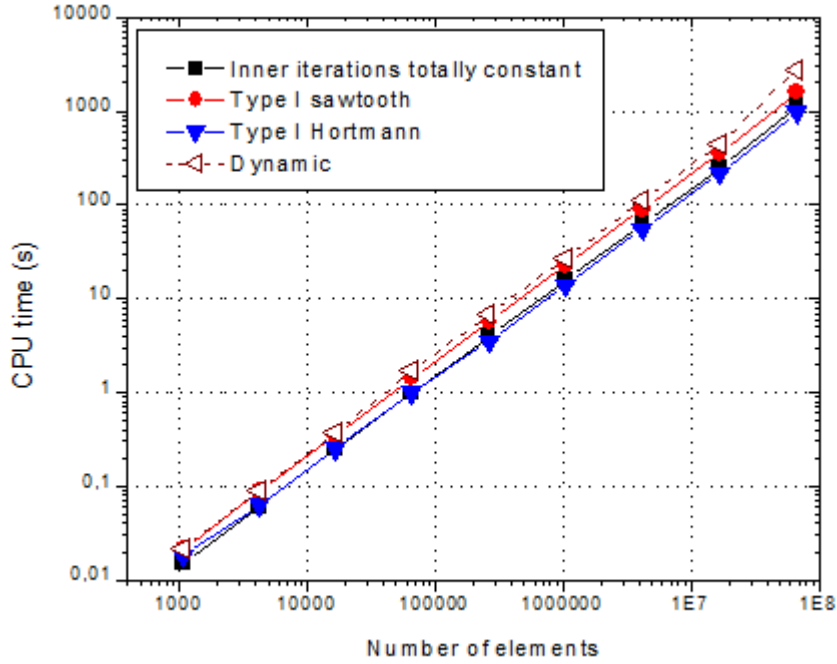


Figure 1: CPU time versus number of elements for the schedules in study

	$c$	$P$
Constant	1.293302151E-05	0.99996
Type I sawtooth	1.825029967 E-05	1.01217
Type II sawtooth	1.460057636E-05	1.04657
Type I Hortmann	1.733285085E-05	0.98433
Type II Hortmann	1.688184856E-05	1.07603
Dynamic	1.408023814E-05	1.04810

Table 1: Coefficient for the geometric curve fitting

The analysis of the number of levels which provides the minimum CPU time for a given set of parameters was realized. Some authors like [13,16,18] also analyzed the number of levels for problems involving the multigrid method. In the last subsections, the number of grids used for the multigrid was kept invariable and equal to  $L_{\text{maximum}}$ . Since other components that influence in the CPU time performance were previously studied, in this subsection the effect of  $L$  on the CPU time performance is evaluated. It is observed that the minimum CPU time is achieved when the maximum number of grids is used, not depending on the size of grid.

The results of the current work are in agreement with the ones presented [19]. They investigated the existence of the optimal components for the multigrid method in a two-dimensional Laplace problem, using an isotropic  $128 \times 128$  elements grid, 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. 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 to numerical results of the current work, the recommended value of  $L$  is always the  $L_{\text{maximum}}$ , which agrees to both works. It must be noticed, however, that both previous works of Tannehill et al. [19] and Rabi and De Lemos [16] do not provide a general recommendation about the value of  $L$  for the Hortmann schedule.

## 6. Conclusion

The geometric multigrid components for CS and isotropic grids were studied in the current work. For the numerical analysis, a two-dimensional heat diffusion problem governed by

Poisson equation, with Dirichlet boundary conditions. It was employed uniform grids, discretized with the Finite Difference Method and second order CDS approximations.

Based on the results of this work, it was found that:

- 1) The type I Hortmann schedule present the smaller CPU time than type II Hortmann schedule, about 3.82 times faster.
- 2) The type I sawtooth schedule present the smaller CPU time than type II sawtooth schedule, about 1.33 times faster.
- 3) The type I Hortmann present the smaller CPU time than type I sawtooth schedule and number of iterations totally constant. It is about 1.12 times faster than number of inner iterations totally constant.
- 4) The type I Hortmann is slightly better schedule, following by number of inner iterations totally constant, type I sawtooth, type II sawtooth, dynamic and type II Hortmann.

## Acknowledgements

This work was partially supported by the Brazilian agencies CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico - Brazil), AEB (Agência Espacial Brasileira, by the Uniespaço Program), and CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brazil). The third author is supported by a CNPq scholarship. The first author thanks the State University of Ponta Grossa for the financial support. We thank the support of the laboratory of Numerical Experimentation (LENA) of the Department of Mechanical Engineering of the UFPR allowing us to use its infrastructure. We are grateful also to reviewers of this work by important remarks.

## References

- [1] Brandt A., *Multi-level adaptive solutions to boundary-value problems*, Math. Comput., 31, (1977) 333–390.
- [2] Briggs W.L., Henson V.E., McCormick S.F., “A Multigrid Tutorial”, second ed., SIAM, Philadelphia, 2000.
- [3] Ferziger, J. H.; Peric, M., *Computational Methods for Fluid Dynamics*, 3 ed., Berlin: Springer, 2002.
- [4] Fletcher C.A.J., “Computational Techniques for Fluid Dynamics”, v1, Second ed., Springer, Berlin, 1997.
- [5] Gaspar F.J., Gracia J.L., Lisbona F.J., C. Rodrigo, *On geometric Multigrid methods for triangular grids three-coarsening strategy*, Appl. Numer. Math., 59, (2009) 1693-1708.
- [6] Gerolymos, G.A.; Vallet I. *Mean-Flow-Multigrid for Implicit Reynolds-Stress-Model Computations*. AIAA Journal. 43, n. 9, September, (2005).
- [7] Ghia, U.; Ghia, K. N.; Shin, C. T. *High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method*. J. Comp. Physics, 48, (1982) 387-411.
- [8] Golub, G. H.; Ortega, J. M. “Scientific Computing and Differential Equations: an Introduction to Numerical Methods”, Academic Press, Inc., 1992.
- [9] Hirsch C., “Numerical Computational of Internal and External Flows”, vol. 1, John Wiley & Sons, Chichester, 1988.

- [10] Hortmann, M.; Peric. M.; Scheuerer, G. *Finite Volume Multigrid Prediction of Laminar Natural Convection: Bench-Mark Solutions*. Int. J. Numer. Methods Fluid, 11, (1990) 189-207.
- [11] Incropera F.P., DeWitt D.P., Bergman T.L., Lavine A.S., “Fundamentals of Heat and Mass Transfer”, sixth ed., John Wiley & Sons, 2007.
- [12] Mohamed S.A., *Optimally efficient multigrid algorithm for incompressible Euler equations*, Int. J. Numer. Methods Heat Fluid Flow, 18, (2008) 783–804.
- [13] Oliveira F., Pinto M.A.V., Marchi C.H., Araki L.K., *Optimized Partial Semicoarsening Multigrid Algorithm, for Heat Diffusion Problems and Anisotropic Grids*. Appl. Math. Modell., 36, (2012) 4665-4676.
- [14] Parter, S.V. *Estimates for Multigrid Methods Based on Red-Black Gauss-Seidel Smooth*, Numerical Math., 52, (1988) 701-723.
- [15] Pinto, M. A V; Santiago, C. D.; Marchi, C.H. Effect of Parameters of a Multigrid Method on CPU Time for One-dimensional Problems. Proceedings of COBEM, 2005.
- [16] Rabi J.A., De Lemos M.J.S., *Optimization of convergence acceleration in multigrid numerical solutions of conductive–convective problems*, Appl. Math. Comput., 124 (2001) 215-226.
- [17] Roache, P. J. “Fundamentals of Computational Fluid Dynamics”, Albuquerque, USA: Hermosa Publishers, 1998.
- [18] Suero R., Pinto M.A.V., Marchi C.H., Araki L.K., Alves A.C., *Analysis of the algebraic Multigrid parameters for two-dimensional steady-state diffusion equations*. Appl. Math. Modell., 36, (2012) 2996-3006.
- [19] Tannehill, J.C., Anderson D.A., Pletcher R.H., “Computational Fluid Mechanics and Heat Transfer”, second ed., Taylor & Francis, Washington, 1997.
- [20] Thekale A., Gradl T., Klamroth K., Råde U., *Optimizing the number of multigrid cycles in the full multigrid algorithm*, Numer. Linear Algebra Appl., 17, (2010) 199-210.
- [21] Trottenberg U., Oosterlee C., A. Schüller, “Multigrid”, Academic Press, San Diego, 2001.
- [22] Wesseling P., “An Introduction to Multigrid Methods”, John Wiley & Sons, Philadelphia, 1992.
- [23] Zhang J., *Multigrid method and fourth-order compact scheme for 2D Poisson equation with unequal mesh-size discretization*, J. Comput. Phys., 179, (2002) 170-179.