COMPUTATIONAL OPTIMIZATION FOR INTEGRAL TRANSFORM ALGORITHMS APPLIED TO THE LID-DRIVEN CAVITY FLOW PROBLEM

J. M. B. S. Guigon[a],

J. S. Pérez Guerrero[b]

and R. M. Cotta[c]

[a] Laboratório de Métodos Computacionais em Engenharia - PEC/COPPE - UFRJ

Caixa Postal 68552 – Cidade Universitátia - CEP 21945-970

Rio de Janeiro, RJ - Universidade Federal do Rio de Janeiro, Brasil jaci.guigon@lab2m.coppe.ufrjbr

[b] CNEN – Comissão Nacional de Energia Nuclear

Rua General Severiano 90 - CEP. 22290-901–RJ

c Laboratório de Transmissão e Tecnologia do Calor – DEM/EE & PEM/COPPE - UFRJ

Caixa Postal 68503 – Cidade Universitária - CEP 21945-970

Rio de Janeiro, RJ - Universidade Federal do Rio de Janeiro, Brasil.

ABSTRACT

An analysis is presented of the computational optimization for integral transform algorithms using the streamfunction only formulation of the Navier-Stokes equations, as applied to the steady incompressible laminar flow of a Newtonian fluid in two-dimensional formulation. The classical lid-driven rectangular cavity flow problem is considered in order to revise the conventional development of the Generalized Integral Transform Technique (GITT). The GITT is applied transforming the partial differential equation into a system of coupled ordinary differential equations, which is numerically solved by a general algorithm for boundary value problems, using a subroutine from the IMSL library with automatic error control. The conventional algorithm written in FORTRAN language is modified and tested seeking its optimization. A few different strategies of applying the technique are considered to achieve improved computational performance and allowing the inspection of convergence rates in the eigenfunction expansion of the original potentials for high Reynolds numbers. These different algorithm alternatives are analyzed and the relative merits are discussed. Results for different values of the Reynolds number and cavity aspect ratios are presented in tabular and graphical forms and fully converged results are critically compared against previously published findings.

Keywords: GITT, hybrid methods, integral transforms, lid-driven cavity, computational optimization, incompressible flow, Navier-Stokes equations.

NOMENCLATURE

Re	Reynolds number
$A_{ijk,}$	transformation coefficients
$B_{ijk,}$	transformation coefficients
$C_{ijk,}$	transformation coefficients
$D_{ijk,}$	transformation coefficients
W	cavity width
U	wall velocity
p	parametrization coefficient
Y_i	vector elements

Greek symbols

μ	fluid absolute viscosity
$\psi(x, y)$	streamfunction
$\overline{\psi}_i$	transformed potential

Mathematical symbols

∇^2	laplacian
∇^4	bi-harmonic operator

Subscripts

b, t	bottom and top of the cavity

INTRODUCTION

Along the last few years, the classical discrete approaches for the approximate solution of partial differential equations have gained a significant impulse towards cost effectiveness and robustness, which allowed for the construction and dissemination of powerful multipurpose automatic solvers in CFD analysis, widely and commercially available nowadays.

The continuous progress on computer technology, resulting in growing CPU processing speed associated to new architectures that maximize the use of the computational resources, has on the other hand led to a great deal of progress on numerical methods development, as for instance in relation with heat transfer and fluids mechanics applications, of particular interest within the present context. Modern computational techniques such as parallelization and vectorization were then advanced and are already traditionally applied to purely numerical methods, contributing to increase their computational performance.

Also within mainly the last two decades, the Generalized Integral Transform Technique (GITT), a hybrid numerical-analytical methodology for partial differential equations, has been gradually advancing into the solution of progressively more involved convection-diffusion problems, of both linear and nonlinear nature (Cotta, 1993; Cotta and Mikhailov, 1997; Cotta, 1998; Cotta and Mikhailov, 2001, Cotta and Mikhailov, 2005). Its analytical heritage allows the generation of fairly straightforward algorithms with automatic error control scheme for the desired potentials and always maintaining a moderate increase of the computational effort with the increase on the number of independent variables (Cotta, 1993; Cotta, 1998). Such characteristics make it an attractive alternative to the more conventional purely numerical methods, provided the resulting additional analytical involvement is presumably accommodated by the user. Despite the strong analytical component inherent to this class of methods, a number of algorithm and numerical analysis questions have been posed along the years of its development, and further research is of interest in the computational optimization of integral transform solutions of different classes of partial differential problems. So, it is intended through this work, to investigate some of the computational aspects that influence the performance of the Integral Transform Method, as applied to the solution of steady fluid flow problems governed by the Navier-Stokes equations (Perez Guerrero, 1991, Perez Guerrero & Cotta, 1992, Perez Guerrero, 1992).

The chosen test-problem was the laminar driven flow in a rectangular cavity (Pan and Acrivos, 1967), due to the challenging nature of its formulation and the vast literature available on it in the realm of numerical analysis and comparative exercises. Seeking to minimize the CPU time spent to solve the model and for improving convergence, modifications on the parameterization scheme, on the vectorial arrangement of the equations and several other modifications in the structure of the algorithm are here proposed and critically analyzed. As a by-product of the present effort, new sets of benchmark results for the lid-driven cavity flow problem are presented, as obtained through the GITT approach for different values of the governing parameters, Reynolds number and aspect ratio.

PROBLEM FORMULATION

The Classical lid-driven rectangular cavity flow problem

The fluid flow inside lid-driven cavities belongs to a special class of problems represented by patterns of closed streamlines. The classical square cavity problem, already discussed and solved by the GITT (Perez Guerrero, 1991, Perez Guerrero & Cotta, 1992, Perez Guerrero, 1992), is represented by an enclosure of square section completely filled by a Newtonian fluid and possessing a moving end wall, in general with uniform speed. The formulation for the original problem is then presented in terms of the streamfunction as:

$$\frac{\partial \psi}{\partial y}\cdot\frac{\partial^3 \psi}{\partial x^3}+\frac{\partial \psi}{\partial y}\cdot\frac{\partial^3 \psi}{\partial x \partial y^2}- \frac{\partial \psi}{\partial x}\cdot\frac{\partial^3 \psi}{\partial x^2 \partial y}-\frac{\partial \psi}{\partial x}\cdot\frac{\partial^3 \psi}{\partial y^3}=\frac{1}{\mathrm{Re}}\nabla^4 \psi \tag{1}$$

with boundary conditions:

$$\psi = 0; \quad \frac{\partial \psi}{\partial x}=0; \quad x=0 \tag{2}$$

$$\psi = 0; \quad \frac{\partial \psi}{\partial y}=0; \quad y=0 \tag{3}$$

$$\psi = 0; \quad \frac{\partial \psi}{\partial x}=0; \quad x=1 \tag{4}$$

$$\psi = 0; \quad \frac{\partial \psi}{\partial y}=-1; \quad y=1 \tag{5}$$

The homogeneous linear case $\nabla^4\psi=0$, known as Stokes problem, has several analytical solutions reported in the literature, as described in (Burgraff, 1966; Joseph and Sturges, 1978; Hellebrand, 1996).

VARIATIONS OF THE CLASSICAL PROBLEM

Besides the classical problem, other formulations have been previously considered for benchmarking purposes, such as the Stokes problem in rectangular cavities with different dispositions of moving end walls. These models constitute useful approximations for many flows of interest in engineering practice. In (Guigon, 2000), examples of low Reynolds numbers flows were analyzed for different boundary condition configurations.

The first model to be presented involves a solid-wall rectangular cavity where the flow is driven by the motion of its horizontal bounding walls (Fig. 1). The second model involves a rectangular cavity with a combination of upper free surface and the flow driven by a horizontal-moving lower wall (Fig. 2), possessing industrial application such as in the production of fine ceramic layers used in electronics (Gaskell, 1998).

Such cases have thus been studied using the GITT for a wide range of Reynolds number and aspect ratio, considering variations of the flow patterns, as represented by the outlines in Fig.3 below.

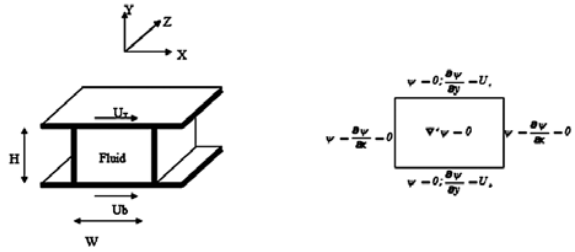

Figure 1. Cavity with two horizontal-moving end walls and the associated boundary conditions.

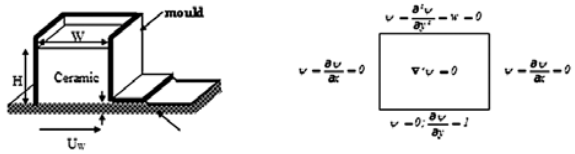

Figure 2. "*Ceramic Tape Caster*" and associated boundary conditions.

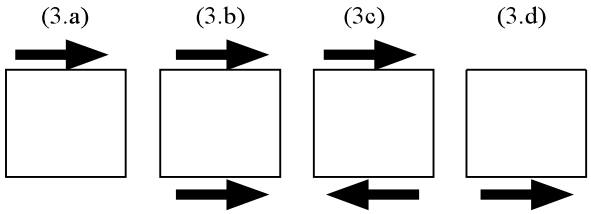

Figure 3. Variations of the elliptical problem formulation - Stokes Problem - $\nabla^4\Psi=0$.

Applying the GITT to the formulation given by equations (Eqs. (1)-(5)) the partial differential problem is transformed into an infinite coupled ordinary differential equations system, (Pérez-Guerrero et al., 1992, Guigon, 2000). The numerical solution for the transformed potentials is then achieved after truncation of the infinite system to a sufficiently large order N that reaches the prescribed accuracy target:

$$\frac{d^4\overline{\psi}_i}{dy^4} = -\mu_i^4\overline{\psi}_i + \sum_{j=1}^{N}\left\{ -2D_{ij}\frac{d^2\overline{\psi}_j}{dy^2} + \right.$$

$$\left. \mathrm{Re}\left[\sum_{k=1}^{N}\left(A_{ijk}\frac{d\overline{\psi}_j}{dy}\overline{\psi}_k + + B_{ijk}\frac{d\overline{\psi}_j}{dy}\frac{d^2\overline{\psi}_k}{dy^2} - \right.\right.\right. \qquad (6)$$

$$\left.\left.\left. C_{ijk}\overline{\psi}_j\frac{d\overline{\psi}_k}{dy} - B_{ijk}\overline{\psi}_k\frac{d^3\overline{\psi}_j}{dy^3}\right)\right]\right\}$$

COMPUTATIONAL IMPLEMENTATION

The major numerical task in the application of the GITT is therefore the numerical solution of the associated ODE system for the transformed potentials, here represented by Eq. (6) and related boundary conditions, for instance obtained from the integral transformation of Eq. (3) and Eq. (5) for the classical problem. The BVPFD routine is one of the best-known and commonly used to solve ordinary differential systems associated with stiff boundary value problems, such as those that appear from application of the Integral Transform Method to elliptic partial differential equations. Applying this routine, following (Pérez-Guerrero et al., 1992; Shankar, 1993), Eq. (6) must be redefined as a first order system (Pereyra, 1978). For implementation of the different proposed models (Fig.3), it was only necessary to modify the specification of one of the subroutines used by BVPFD, responsible for the definition of the boundaries limits. For all other parts of the algorithm, it was therefore employed the same methodology as for the classical case. We now discuss some of the modifications performed on the basic algorithm and improvements achieved from the present study.

THE FORTRAN90 IMPLEMENTATION

The original code written in Fortran77 (Perez Guerrero et al., 1992) was reorganized, making use of the resources and structures automatically available in the Fortran90 compiler. The resulting CPU time gain obtained through the modified code, rewritten in Fortran90 (Stephen, 1998), was significant. As an example, we take the case of Re=1000, N=25 terms, and with a relative error control of 10^{-5}, then the Fortran77 code was 77% slower than the correspondent Fortran90 implementation. This comparison has been produced using the automatic parameterization provided by the BVPFD routine, from the IMSL library.

Strategies for the Parametrization Scheme

As typical of various numerical schemes for nonlinear ODE´s, when increasing the nonlinear characteristic of the problem, numerical results may be obtained through the solution for intermediate problems with the gradual increase of the governing parameter in the nonlinear terms, here the Reynolds number. Therefore, each obtained intermediate result serves as an initial estimate for the following problem along the iterative procedure and the Reynolds number works as a natural parametrization coefficient in the present equations system. This approach demands that the programmer specifies the control of iterations inside the algorithm, but one is required to know which variation interval of the Reynolds number will produce a more efficient processing.

Another strategy is to use of the automatic parameterization process already provided by the BVPFD routine that consists in the solution of the following parametrized system:

$$y^{iv} = p.f(y, y', y'', y'''), 0 < p < 1 \quad (7)$$

Using the built in parametrization scheme, the solution process starts using an easier problem (with the p value < 1 associated with the nonlinear term), reaching the original system formulation when p=1. Some results are here compared obtained from the implemented algorithm using the natural parameterization (user chosen variations of the Reynolds number) and the automatic parameterization (BVPFD), as shown in Table 1. All the cases shown were executed for the model 3.a. The parametrization was directly applied to the first 29 terms of the finite series in Eq. (6) and for a desired final Reynolds number Re=1000.

Table 1. Comparison of CPU time units for different parametrization tests.

Natural Parame-triza-tion	**Reynolds Number**	**Tol.**	**Final Mesh**	**CPU Time Units**
0.10	0, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000	1.d-5	235	1.86
0.20	0, 200, 400, 600, 800, 1000	1.d-5	257	1.32
0.25	0, 250, 500, 750, 1000	1.d-5	265	1.32
0.30	0, 300, 600, 900, 1000	1.d-5	245	1.84
0.40	0, 400, 800, 1000	1.d-5	248	1.05
0.45	0, 450, 900, 1000	1.d-5	252	2.35
0.50	0, 500, 1000	1.d-5	271	1.18
BVPFD Parame-triza-tion	**Reynolds Number**	**Tol.**	**Final Mesh**	**CPU Time Units**
0.1	1000	1.d-5	268	1.00

For the tests using the natural parameterization, the best relative result in terms of CPU time was obtained for the parameter value 0.4. The variation in the size of the converged final mesh for several values of the parameter is not observed to be significant. Using the parameterization from BVPFD, in spite of reaching a larger final mesh, the resulting lower computational processing time demonstrates that this alternative was finally the most efficient among our various attempts. It is important to note that the automatic parametrization scheme of BVPFD is not, of course, always guaranteed to achieve convergence. For higher Reynolds numbers it might be necessary to appropriately experiment with different values of the parameter "p" in order to reach convergence.

DATA REARRANGEMENT IN MEMORY

Applications that use vector and matrix structures should be implemented exploring the best hardware characteristics of the computer, to enhance performance when processing their elements. Using Fortran codes, for example, the CPU time can be reduced eliminating the stride (spacing at the start of each block in the memory), measured by the Fortran compiler as the number of contiguous elements for each column. In one-dimensional arrays, it is desirable that the elements processed inside a loop be located the closest possible in the memory. For multi-dimensional arrays (in Fortran), the best obtainable performance will occur when the variable of the most internal loop is related with the left array index.

The transformation of Eq.6 to a first order system generates the following specification:

$$Y_i(y) = \overline{\psi}_i; \quad Y_{N+i}(y) = \frac{d\overline{\psi}_i}{dy}; \quad Y_{2N+i}(y) = \frac{d^2\overline{\psi}_i}{dy^2};$$

$$Y_{3N+i}(y) = \frac{d^3\overline{\psi}_i}{dy^3}; \quad \frac{dY_{3N+i}}{dy} = \frac{d^4\overline{\psi}_i}{dy^4} \quad (8a\text{-}e)$$

Fig.4 below shows the allocation for the vector elements employed to store data obtained during the processing for finite series of three terms (N=3), as specified in Eqs. 8.

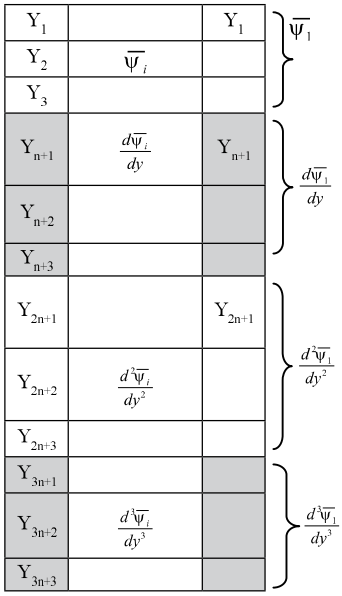

Figure 4. Original memory allocation for values of the transformed potentials and derivatives with N=3.

This memory allocation model presents two inconveniences, inherent to the solution methodology application. First, increasing the nonlinear characteristic of the problem, it becomes necessary to increase the number

of terms in the eigenfunction expansion (Eq. 6), to enhance convergence and thus achieve more accurate results. For this purpose, calculations are organized with gradual increase of the truncation orders; in other words, the solution scheme is constructed so that it is possible to use the solution vector obtained for a certain number of terms as an approximation for the next calculation with a larger truncation order. It is therefore necessary to reallocate the spaces inside the vector, reusing the data located in the same position. Figure 5 below illustrates the space reallocated in the solution vector, when the number of terms in the series increases (from 3 to 4 in this example). This memory reallocation scheme has been commonly applied to the codes developed to date for elliptical problems using the Integral Transform Method.

Second, the access to the elements of the array inside a loop is not sequentially processed. In this conventional model, the larger the number of terms in the series, more distant will be the memory addresses accessed by the potentials and the various derivatives in the solution vector.

Y_1	→	Y_1	
Y_2		Y_2	
Y_3		Y_3	
Y_{n+1}		....	y_4
Y_{n+2}		Y_{n+1}	
Y_{n+3}		Y_{n+2}	
Y_{2n+1}		Y_{n+3}	
Y_{2n+2}		....	y_{N+4}
Y_{2n+3}		Y_{2n+1}	
Y_{3n+1}		Y_{2n+2}	
Y_{3n+2}		Y_{2n+3}	
Y_{3n+3}		....	
Memory available		Y_{3n+1}	
		Y_{3n+2}	
		Y_{3n+3}	
		....	y_{2N+4}

Figure 5. Redistribution of the vector elements for a truncation order increase to N=4.

Therefore, a new array arrangement is proposed, which stores the transformed potentials and derivatives values as described below in Fig. 6:

$$Y(i-1)n+1=\overline{\psi}_i; \qquad Y(i-1)n+2=\frac{d\overline{\psi}_i}{dy};$$

$$Y(i-1)n+3=\frac{d^2\overline{\psi}_i}{dy^2}; \qquad Y(i-1)n+4=\frac{d^3\overline{\psi}_i}{dy^3} \qquad \text{(9a-d)}$$

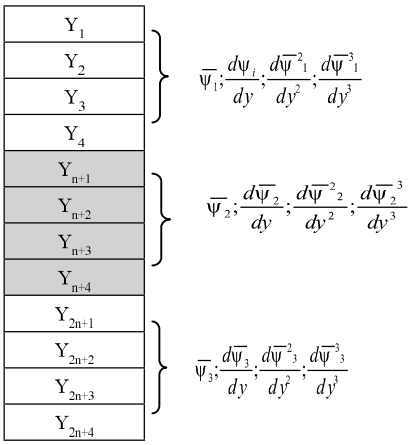

Figure 6. Memory array allocation for the new vectorial arrangement proposed - N=3.

Using this new model brings two advantages. First, the access to the several derivatives at each step of the loops is performed in closer or contiguous memory positions. Second, when the truncation order is increased the inclusion of new terms is only necessary at the end of the array, without reallocation of memory space in order to re-write the older elements.

Table 3 presents some comparative results for the original and optimized codes, under the memory reallocation scheme analysis. In agreement with our expectations, the new allocation of data in memory, in combination with the parameterization for the boundary conditions (Guigon, 2000), is the alternative that presents the best performance. With this information, the modified code was consolidated, making the program intrinsically more efficient with the increase of the truncation order and providing significant reduction on algorithm size as well.

CONVERGENCE FOR HIGH REYNOLDS NUMBER

As expected from eigenfunction expansion type approaches, as the importance of source terms increase, there should be a slowing down on convergence rates of the original potentials series representation. There is a visible reduction of the convergence rates with higher Reynolds numbers due to the stronger resulting nonlinear source terms. For such cases, a larger number of terms in the finite series representations are fairly expected, requiring a larger computational effort.

Table 4 shows a comparison of the algorithms used for a higher value of the Reynolds number, Re=5000, when the larger computational effort can be verified. For the original

code, the computational algorithm was developed according to the stride model shown in Figs. 4-5. In the modified version, it has been used the scheme depicted in Fig. 6 and Eq. 9, with reduction on the strides. As expected, the improvement in the CPU time becomes more evident while increasing the number of terms in the eigen-function expansions, here represented by a reduction of 50% on the total execution time obtained relative to the original algorithm version.

Table 3. Comparison of CPU time for optimized code as compared with the original algorithm–Tol= 10^{-5} - Re=1000.

Inter-mediary Trunca-tions	Max. Trunc.	Original		Modified	
		Final Mesh	CPU Time Unit	Final Mesh	CPU Time Unit
5	25	211	0.002	211	0.001
9	25	211	0.019	226	0.01
15	25	226	0.092	226	0.07
21	25	226	0.25	226	0.184
23	25	226	0.34	226	0.26
25	25	226	0.43	226	0.32
Total (in CPU time Units)		**1.14**		**0.84**	
25 terms (without intermediary truncation)		**Original: Time(s) = 0.35**			
		Modified: Time(s) = 0.28			

Table 4. Comparison of original and modified algorithms for large Reynolds number situation (Case 3.a - moving end wall - With and Without Stride - Initial Mesh=131pts.- Re=5000).

N	Original		Modified	
	Mesh	CPU Time Units	Mesh	CPU Time Units
5	240	0.004	240	0.004
9	285	0.048	285	0.043
13	297	0.29	297	0.21
17	297	0.65	297	0.52
21	399	1.73	399	1.26
25	399	3.54	399	2.87
29	399	5.15	399	4.29
33	399	11.82	399	5.77
37	399	13.56	399	8.11
41	561	38.15	531	17.52
Total CPU Units	**74.95**		**40.61**	

In both cases, the programs were executed considering a relative error control of 10^{-3} for up to 37 terms in the expansion. In the final execution, for which 41 terms were used, the tolerance was decreased to 10^{-4}. This justifies the significant increase of the size on the final mesh and the larger CPU time spent at the final execution of the algorithm.

APPLICATIONS AND REFERENCE RESULTS

After accomplishing an optimized algorithm for Integral Transforms applied to elliptical problems, numerical results for the models specified in Fig. 3 are here reported, as well as the comparisons of such converged values with those obtained through other methods, according to (Gaskell et al., 1998). The first analysis consists of the comparison among values of the stream-function obtained through analytical, numerical and hybrid (GITT) methods, as applied to the Stokes problem solution approximation. These results complement the comparisons presented in (Gaskell et al., 1998).

The values presented in Table 5, obtained for the model 3.b, correspond to points inside the rectangular cavity, with two moving end walls, U_t=2 (superior wall), U_b=1 (inferior wall), aspect ratio=5, where the various solutions presented are: **Sh** = Shankar method (Shankar, 1993); **JS** = Joseph and Sturges method (Joseph & Sturges, 1978); **Numerical** = Finite Elements (2592 elements, 5393 nodes). The last row in each case consolidates the reference results provided by the GITT approach after full convergence has been achieved for a five significant digits user prescribed accuracy control. The overall agreement is indeed very good, but closer between the GITT results and the more refined implemention of (Shankar, 1993), than with the purely numerical solution reported in (Gaskell et al., 1998). Table 6 confirms the converged values obtained through GITT and reported as reference results in Table 5.

The second analysis corresponds to the comparison of values and flow patterns for several models applied to rectangular cavities, not only for the Stokes flow case but also extending the calculation to the nonlinear case, allowing for the comparison of the corresponding streamline patterns. The variations in the flow behavior can be verified by changing the dimensionless parameters of the problem, while also changing the boundary conditions specification according to the previously discussed models. Figs. 7-14 show the streamlines with flow patterns for different cavity models and different aspect ratio. Figs.7-8 for instance, show the classical case of the square cavity flow problem, respectively for Re=0 (Stokes flow) and Re=1000. Figs. 9-10 then show the case of a rectangular cavity of aspect ratio 5, and again for the two values of Re=0 and 1000. The secondary recirculation zones are noticeably captured, where the potential gradients are

fairly high and highly dependent on the Reynolds number and aspect ratio. Figs.11-12 and 13-14 bring the different situation of a cavity with two moving end walls, with a top lid velocity U_t=-2 and a bottom lid velocity of U_b=-1, which induce the appearance of entirely different secondary flow patterns, especially for the larger value of Reynolds number.

Table 5. Comparison of Stokes flow solutions, including the present GITT hybrid approach and analytical and numerical solutions reported in Gaskell et al., 1998 (Cavity Ω: [-1,1] x [0,10] – Aspect Ratio 5).

X =	**0.0**	**0.25**	**0.50**	**0.75**
Y = 7.5				
Sh N=200,M=200	-3.7974 E-3	-3.0985 E-3	-1.5392 E-3	-2.8947 E-4
JS N=200	-3.7974 E-3	-3.0985 E-3	-1.5392 E-3	-2.8947 E-4
Sh N=20,M=20	-3.7970 E-3	-3.0981 E-3	-1.5390 E-3	-2.8940 E-4
JS N=20	-3.7961 E-3	-3.0973 E-3	-1.5385 E-3	-2.8925 E-4
Numerical	-3.8118 E-3	-3.1130 E-3	-1.5513 E-3	-2.9807 E-4
GITT Converged	**-3.7965 E-3**	**-3.0969 E-3**	**-1.5387 E-3**	**-2.9020 E-4**
Y = 5.0				
Sh N=200,M=200	1.8652 E-5	1.5741 E-5	8.8501 E-6	2.3507 E-6
JS N=200	1.8652 E-5	1.5741 E-5	8.8501 E-6	2.3507 E-6
Sh N=20,M=20	1.8651 E-5	1.5740 E-5	8.8496 E-6	2.3505 E-6
JS N=20	1.8645 E-5	1.5735 E-5	8.8465 E-6	2.3496 E-6
Numerical	1.8713 E-5	1.5796 E-5	8.8909 E-6	2.3744 E-6
GITT Converged	**1.8649 E-5**	**1.5735 E-5**	**8.8489 E-6**	**2.3540 E-6**
Y = 2.5				
Sh N=200,M=200	1.8985 E-3	1.5491 E-3	7.6952 E-4	1.4471 E-4
JS N=200	1.8985 E-3	1.5491 E-3	7.6952 E-4	1.4471 E-4
Sh N=20,M=20	1.8983 E-3	1.5489 E-3	7.6941 E-4	1.4471 E-4
JS N=20	1.8983 E-3	1.5489 E-3	7.6938 E-4	1.4465 E-4
Numerical	1.9057 E-3	1.5463 E-3	7.7576 E-4	1.4901 E-4
GITT Converged	**1.8980 E-3**	**1.5484 E-3**	**7.6930 E-4**	**1.4507 E-4**

Table 6a. Convergence behavior of GITT solution for the case 3.b (N<25, Aspect Ratio=5, Stokes flow) – top of cavity.

N	**Y=7.5 X = 0.0**	**Y = 7.5 X = 0.25**
5	-.37898E-02	-.31035E-02
10	-.37958E-02	-.30972E-02
15	-.37966E-02	-.30970E-02
20	-.37965E-02	-.30969E-02
21	-.37965E-02	-.30969E-02
24	-.37965E-02	-.30969E-02
25	-.37965E-02	-.30969E-02
N	**Y = 7.5 X = 0.5**	**Y = 7.5 X = 0.75**
5	-.15324E-02	-.29716E-03
10	-.15394E-02	-.28909E-03
15	-.15388E-02	-.29028E-03
20	-.15387E-02	-.29027E-03
21	-.15387E-02	-.29024E-03
24	-.15387E-02	-.29020E-03
25	-.15387E-02	-.29020E-03

Table 6b. Convergence behavior of GITT solution for the case 3.b (N<25, Aspect Ratio=5, Stokes flow) – center of cavity.

N	**Y=5.0 X = 0.0**	**Y = 5.0 X = 0.25**
5	.18631E-04	.15754E-04
10	.18647E-04	.15736E-04
15	.18645E-04	.15736E-04
20	.18649E-04	.15735E-04
21	.18649E-04	.15735E-04
24	.18649E-04	.15735E-04
25	.18649E-04	.15735E-04
N	**Y = 5.0 X = 0.5**	**Y = 5.0 X = 0.75**
5	.88313E-05	.23732E-05
10	.88508E-05	.23507E-05
15	.88492E-05	.23541E-05
20	.88488E-05	.23541E-05
21	.88489E-05	.23540E-05
24	.88489E-05	.23540E-05
25	.88489E-05	.23540E-05

Table 6c. Convergence behavior of GITT solution for the case 3.b (N<25, Aspect Ratio=5, Stokes flow) – bottom of cavity

N	Y=2.5 X = 0.0	Y = 2.5 X = 0.25
5	.18947E-02	.15516E-02
10	.18977E-02	.15485E-02
15	.18980E-02	.15485E-02
20	.18980E-02	.15483E-02
21	.18980E-02	.15483E-02
24	.18980E-02	.15483E-02
25	.18980E-02	.15483E-02
N	**Y = 2.5 X = 0.5**	**Y = 2.5 X = 0.75**
5	.76611E-03	.14855E-03
10	.76960E-03	.14452E-03
15	.76933E-03	.14511E-03
20	.76930E-03	.14510E-03
21	.76930E-03	.14508E-03
24	.76930E-03	.14507E-03
25	.76930E-03	.14507E-03

CONCLUDING REMARKS

This work proposes a computational optimization for integral transform algorithms applied to the Navier-Stokes equations in streamfunction-only formulation. Steady lid driven cavity flow was taken for the test-case problem, as a typical two-dimensional example of a nonlinear elliptical formulation. The optimization explored the features of the Fortran90 compiler, modifying several arrays structures previously used for the data representation, and thus significantly improving the algorithm and reducing CPU time. The combined use of the best alternative of the parameterization provided by the IMSL routine BVPFD and the new memory allocation scheme for the data structures have also produced a significant reduction in computational cost for large values of the main parameter, the Reynolds number, recommending the modified approach use in the Integral Transform algorithms developed for elliptical problems.

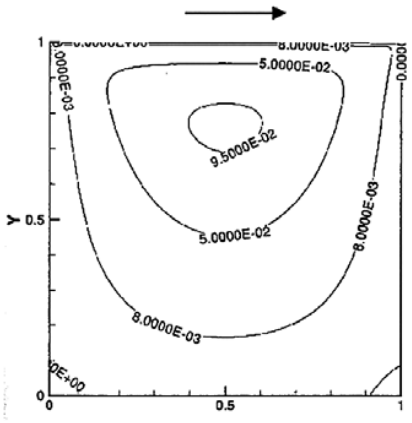

Figure 7. One moving end wall - Re=0
U_t =-1; Aspect Ratio=1

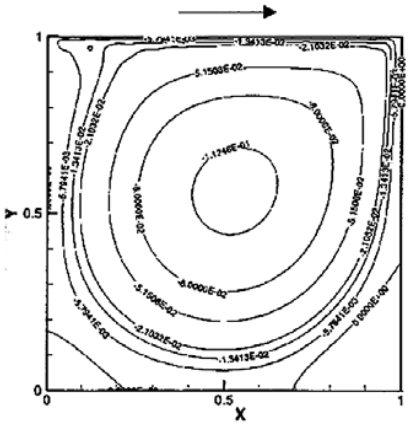

Figure 8. One moving end wall - Re=1000
U_t =-1; Aspect Ratio=1

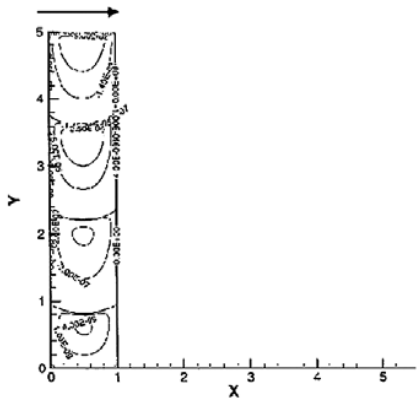

Figure 9. One moving end wall - Re=0 ; U_t=-1; Aspect Ratio=5

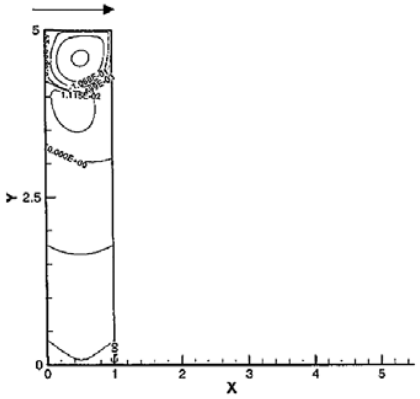

Figure 10. One moving end wall - Re=1000 ; U_t =-1; Aspect Ratio=5

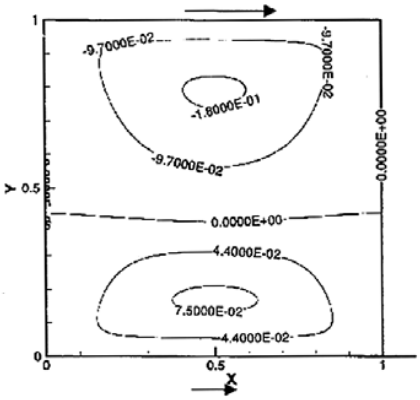

Figure 11. Two moving end walls - Re=0
U_t =-2, U_b =-1; Aspect Ratio=1

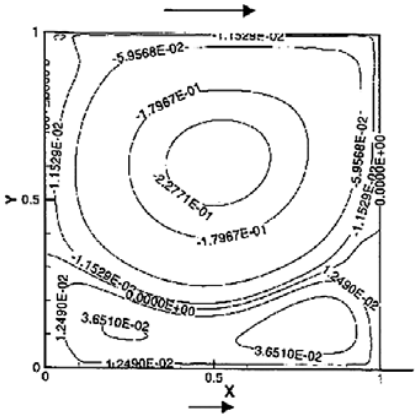

Figure 12. Two moving end walls - Re=1000
U_t =-2, U_b =-1; Aspect Ratio=1

Finally, a few flow configurations were analyzed under proposed boundary conditions combinations, for different values of the governing parameters, Reynolds number and aspect ratio, allowing for the establishment of new reference results represented by the fully converged Integral Transform computations.

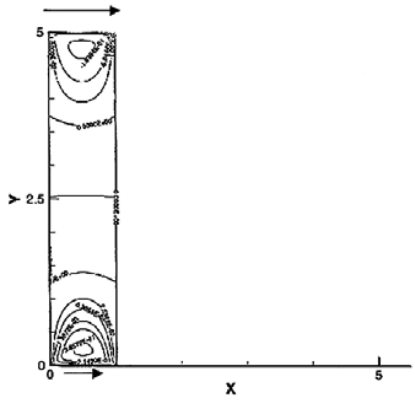

Figure 13. Streamlines for the case of two moving end walls - Re=0; U_t =-2; U_b =-1 ; Aspect Ratio=5

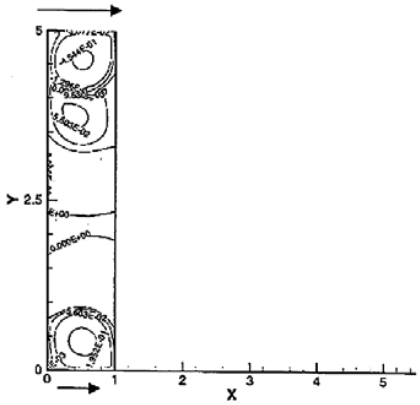

Figure 14. Streamlines for the case of two moving end walls - Re=1000; U_t =-2; U_b =-1 ; Aspect Ratio=5.

REFERENCES

Burggraf, O. R., 1966, Analytical and Numerical Studies of the Structure of Steady Separated Flows, J. Fluid Mech., Vol. 24, pp. 113-151.

Cotta, R. M., 1993, *Integral Transforms in Computational Heat and Fluid Flow,* CRC Press, FL.

Cotta, R. M., 1998, *The Integral Transform Method in Thermal and Fluids Science and Engineering*, Begell House, New York.

Cotta, R. M., Mikhailov, M. D., 1997, *Heat Conduction - Lumped Analysis, Integral Transforms, Symbolic Computation*, Wiley-Interscience, New York, 1997.

Cotta, R. M., and M. D. Mikhailov, 2001, Hybrid Approaches in Convective Heat Transfer, in: Benchmark Results for Convective Heat Transfer in Ducts: - The Integral Transform Approach, eds. C.A.C. Santos, J.N.N. Quaresma, and J.A. Lima, ABCM Mechanical Sciences Series, Editora E-Papers, Rio de Janeiro, Part 1, Chapter II, pp.17-38.

Cotta, R. M., and Mikhailov, M. D., 2005, Hybrid Methods and Symbolic Computations. In: Handbook of Numerical Heat Transfer, 2nd edition, Chapter 16, Eds. W. J. Minkowycz, E. M. Sparrow, and J. Y. Murthy, John Wiley, New York.

Gaskell, P. H., Savage, M. D., Summers, J. L., and Thompson, H. M., 1998, Stokes Flow in Closed, Rectangular Domains, Applied Mathematical Modeling, Vol. 22, pp.727-743.

Guigon, J. M. B. S., 2000, Otimização Computacional em Algoritmos de Transformação Integral para as Equações de Navier-Stokes, COPPE/UFRJ, M.Sc., Tese, Rio de Janeiro.

Hellebrand, H., 1996, Tape Casting. In: Brook, R. J. (ed.), Processing of Ceramics Part I, VCH Publishers.

IMSL Library, 1989, Math.Lib., Houston, Texas.

Joseph, D. D., Sturges, L., 1978, The Convergence of Bi-orthogonal Series for Bi-harmonic and Stokes Flow Edge Problems, SIAM J. Appl. Math., Vol. 34, pp.7-26.

Pan, F., Acrivos, A., 1967, Steady Flows in Rectangular Cavities, J. Fluid Mech., Vol. 28, pp. 643-655.

Pereyra, V., 1978, PASVA3: An Adaptative Finite-Difference FORTRAN Program for First Order Nonlinear Boundary Value Problems, Lec. Notes in Comp. Science, Vol. 76, Springer-Verlag, Berlin, pp.67-88.

Perez Guerrero, J. S., 1991, Solução das Equações de Navier-Stokes em Formulação de Função Corrente via Transformação Integral, COPPE/UFRJ, M.Sc., Tese, Rio de Janeiro.

Pérez-Guerrero, J. S., Cotta, R. M., 1992, Integral Transform Solution for the Lid-Driven Cavity Flow Problem in Stream-Function-Only Formulation, Int. J. for Num. Methods in Fluids, Vol. 15, pp. 399-409.

Pérez-Guerrero, J. S., Cotta, R. M., Scofano Neto, F., 1992, Hybrid Solution of the Incompressible Navier-Stokes Equations via Integral Transformation. In: Proc. of the 2nd Int. Conf. Advanced Computational Methods in Heat Transfer, Heat Transfer 92, Vol. 1, pp. 735-750, Milan, Italy.

Shankar, P.N., 1993, The Eddy Structure in Stokes flow in a Cavity, J. Fluid Mechanics, Vol. 250, pp.371-383.

Received: September 06, 2006
Revised: October 06, 2006
Accepted: November 06, 2006