BABEŞ-BOLYAI UNIVERSITY OF CLUJ-NAPOCA
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE

Gabriela Florentina NUŢ

The Multigrid Method for Differential Equations

PHD THESIS SUMMARY

Scientific Supervisor:
PhD Professor Petru BLAGA

2014

Contents

Introduction

The evolution of the multigrid method

The differential equations with partial derivatives are often used in order to describe real phenomena and processes from various fields (fluids mechanics, thermodynamics, economy, meteorology, image processing,...). Solving such equations is thus very important from both the theoretical and practical point of view. As the analytical solution isn't always accesible, it is important to build numerical methods in order to approximate the solution.

Recent simulations of these problems are based on more and more complicated models. The discretization process of differential equations leads to big systems that become ill-conditioned (small perturbations of constant term lead to bigg variations of the solution) as the working grid step becomes smaller. The necessity of a numerical method to solve these systems is thus obvious. Also there is a growing need of computer memory and of reducing the processing time, as the problems are solved on grids that have a growing number of cells. Thus there is not only a mathematical challenge, but also one of computer programming.

The system obtained after the discretization has a very large number of unknowns (for example for a grid of the seventh order there are 2^8 equations on each dimension, thus 2^{16} in the bidimensional case, or 2^{24} in the tridimensional one) and has to be solved as accurately as possible, in a reasonabily short computing time. Solving such a system using for example Gauss's elimination method becomes unpractical as it requires a huge number of operations and uses too much of the computer's memory. This is the reason why numerical iterative methods are used. But it is well known that these methods have a very poor convergence. They reduce very well the oscillating components of the error, but the smooth ones remain almost unchanged, being thus very slowly reduced toward zero. So, the classical iterative methods are inefficient as they generate smooth errors that are very slowly approaching to zero.

One of the quickest methods used for solving a big system is the multigrid method, which has been proven to be a very good tool for increasing the convergence speed through introducing a new component: the coarse grid correction. This reduces only the smooth components of the error, being a complementary method for the classical numerical iterative methods. The main idea of this method is that, when passing to a coarser grid, only smooth components can be well approximated on this grid, and then, on the coarse grid, the smooth components become oscillant, so that they can be efficiently reduced by a numerical iterative method. Moreover, the number of unknowns from the system that has to be solved becomes smaller so that the solving time decreases. This is one of the main proper-

ties of the multigrid method: it is rapidly convergent. Multigrid method can be applied for solving many types of differential equations, with a cost of $O(n)$-arithmetic operations for n unknowns, thus being among the most efficient numerical methods used for solving a systems of equations. Also, the the convergence of the method does not depend on the grid step.

The idea of the multigrid method was first introduced by R. P. Fedorenko, in 1964. He presented a multigrid algorithm for a five points standard discretization for the Poisson equation on the [0,1]×[0,1] domain. Since then, a large number of authors (B.E. Bank, T.F. Dupont, A. Brandt, W. Hackush, P. Wesseling, S.Mc. Cormick, ...) have shown that this is a really usefull technique, practical and theoretical.

The implementation of the multigrid method on computers started in 1977 with the work of A. Brandt that contains all the main processes used in the method. He obtained the first practical results and showed the efficiency of the method. A. Brandt and W. Hackbush introduced the multigrid method for nonliniar problems and proved the convergence of the method for some elliptical problems. Since then the interest in the multigrid method has become even bigger, although initially there was a lot of scepticism, due to the fact that the theory was not well developed. Another important step was the combination of the multigrid method with the nested iteration techniques. The multigrid method was initially used only for elliptic problems, it's implementation requiring substantial effort. Further researches allowed using the method to a larger class of problems. Numerous forms of the multigrid method have been developed, from the geometrical ones that use structured grids, to the pure algebraic ones in which the accent is set on the algebraic aspects of the equation, not requiring any information about the grids on which the problem is discretised.

The growing interest for this method determined even the organizing of at least two multigrid conferences: "European Multigrid Conference" and "Copper Mountain Conference on Multigrid Methods".

The purpose and motivation of the work

In this work the purpose was to obtain efficient solutions for the systems of equations generated by the discretization of partial derivatives differential equations that model physical phenomena such as the ones mentioned above. We used the local Fourier analysis in order to build efficient multigrid methods by choosing it's components according to the conclusions of this analysis. We also analized the error and the convergence of the multigrid method. The problems have been chosen from different classes of partial derivatives equations, both stationary and nonstationary.

The discretization methods used have been the finite differences of second and fourth order and the finite element method, on rectangular systems of grids. For the nonstationary equations we used backward Euler scheme for the discretization in the time direction.

Any multigrid method strongly depends on the problem to be solved, thus there isn't a unique multigrid algorithm. This is why in this work we used the local Fourier analysis for designing an efficient method for the problem that has to be

solved.

The convergence of the multigrid method for convection-diffusion problems has been studied in the stationary case by many researchers ([29, 53, 62, 72, 73, 86, 89, 90] and so on). The novelty in this work is the fact that *the local Fourier analysis is extended for the nonstationary convection-diffusion equations, for both the onedimensional case* (paper [58]), *and the bidimensional one* (paper [59]). We also have studied with this method the influence of the discretization and the numerical iterative methods on the results obtained with the multigrid method, on the convergence rate and on the error reduction.

While we made the Fourier analysis we introduced a *new definition for the smoothing factor* (paper [61]), as compared with the one used for example by U. Trottenberg, in [86], R. Wienands and W. Joppich in [90] or P. Wesseling in [89], With this definition we made a different splitting of the frequencies spectrum from the components of the error.

While the method is rapidly convergent for model problems like Poisson equation on a square, the rate of convergence can be drasticaly modified by the presence of large variations in the coefficients or of complicated domains. Also the convection-diffusion equations lead to poor convergence of the multigrid method, and this type of equations are very often the mathematical model for which a solution is needed. This is the reason why in order to solve convection-diffusion problems we introduced a *new method that uses a system of grids,* named *"stellar prolongation method"* (works [56], [57]). The results obtained with this method are compared with the ones obtained with the multigrid method for several problems. We used the stellar prolongation method for convection-diffusion equations in the case of dominant convection, for which it is well known (for example [63], [86]) that the multigrid method leads to unsatisfying results. The practical results showed that this method can be succesfully applied for solving this kind of problems.

The structure of the work

In the first chapter, **"Multigrid method"**, we presented a classification of the partial derivative equations and the border condition needed by each of these for the solution to be stable and unique. The motivation of this presentation is also related to the fact that the multigrid method will be applied and studied in this work for different practical problems from all these classes: elliptic, parabolic and hyperbolic.

In the second part of the chapter some discretization methods are described: finite differences of second and fourth order and finite element and the numerical iterative methods that will be used for solving the systems of equations obtained after the discretization process.

The third part of the first chapter deals with the multigrid method, starting from the local Fourier analysis of the iterative numerical methods. The informations obtained with this analysis can be used in designing efficient multigrid method .

The second chapter, **"The study of the convergence of the multigrid method"**, presents the local Fourier analysis of the multigrid method.

At the begining are presented a few theoretical notions about the Fourier transform of a function, then the Fourier analysis for classical iterative methods is presented. The model problem used for the convergence and the error analysis of the

multigrid method is a time dependent convection-diffusion equation, for which the theoretical results obtained are then confirmed by the practical ones. The conclusions of the analysis allow an efficient design of the multigrid method.

Chapter 3, **"Multigrid method for nonstationary convection-diffusion problems "**, presents the multigrid method and it's analysis for a nonstationary convection-diffusion problem in two dimensions, and it is a generalization of the results obtained in **Chapter 2**. As convection-diffusion problems often lead to instable solutions, we also presented the streamline diffusion method, which has the role of reducing these oscillations for the case when the convection is dominant. The 3-dimensions multigrid method is presented next, and the errors obtained for a nonstationary convection-diffusion problem.

The convection-diffusion equation is also studied in the particular case when there is no convection. The diffusion equation is solved with the multigrid method, for the case when the physical process is nonstationary and takes place in a three layers environement with different physical properties, where the active substance is transported by nanoparticles. For the error to be comprised between certain values, we also determined the number of iterations needed by the multigrid method.

Chapter 4, **"The stellar prolongation method"**, introduces a new numerical method that uses a system of more grids for solving diferential equations with partial derivatives.

At the end we presented the main ideas of this work, the original ideas, respectively what novelties this work brings and some directions for future work.

The original contributions are in *Theorems* 2.2.1, 2.2.2, 2.2.3, 2.2.4, 2.3.1, 2.3.2, 3.2.1, 3.2.2, *Properties* 2.3.1, 2.3.2, 2.3.3, 2.3.4, 2.3.5, 2.3.6, 2.3.7, 2.3.8, 3.1.1, 3.1.2, 3.2.1, 3.2.2, 3.2.3, 3.2.4, 3.5.1, 4.2.1, 4.2.2, 4.2.3, 4.2.4, 4.3.1, 4.3.2, 4.3.3, *Corolary* 2.2.1, *Remarks* 2.2.1, 2.2.2, 2.2.3, 2.2.4, 2.3.1, 2.3.2, 3.2.1, 3.2.2, 3.3.1, 3.5.1, *Examples and problems* 3.4.1, 3.4.2, 4.4.1-4.4.5.

Keywords:
Multigrid method, finite differences method, finite elements method, partial derivatives differential equations, convection-diffusion equation, error reduction factor, smoothing factor, amplification factor, asymptotic convergence factor, stellar prolongation

I would like to thank all those who have been next to me in my work and encouraged me to keep on going. Without their help I couldn't have done it.

First of all I am gratefull to my scientific adviser, Prof. Univ. Dr. Petru Blaga for accepting me to work under his supervision and for the scientific guidance he gave me all these years.

Very special thaks to Conf. Univ. Dr. Ioana Chiorean for allways being there for me. I will never forget her patience and the help I received from her each time I nedeed it.

I would also like to express my gratitude to my parents for supporting me in everything I do.

Chapter 1

The Multigrid Method

As the purpose of this work is to numerically solve differential equations with the multigrid method, in Section 1.1 we presented a classification of the differential equations with partial derivatives ([84]). Section 1.3 describes some classical methods for solving the systems of equations ([22]) obtained after the discretization processes ([22], [85]) described in Section 1.2. Then, in Section 1.4 we presented the multigrid method as introduced in [34], [86], [89].

1.1 Differential equations with partial derivatives - clasiffication and canonical form

In this section are presented the three classes of differential equations that will be used in this work and the boundary conditions needed for the equation to have a unique and stable solution.

1.2 Discretization methods of a differential equation

We present here the two discretization methods used further: the finite differences method of second and fourth order and the finite elements method.

1.3 Numerical iterative methods for solving systems of equations

For the numerical solution of a system of equations the iterative methods used in this work, such as Jacobi, pondered Jacobi or Gauss-Seidel methods are presented in this section.

1.4 Multigrid method

The multigrid method presentation from this section follows the main ideas from [34]. In order to illustrate the multigrid method ([34], [89]), the model problem was Poisson equation on a square.

Chapter 2

The convergence of the multigrid method

The convergence of the multigrid method has been studied for the Poisson equation [35], [86], [89]. In more recent years, it has also been established for the convection-diffusion equation [62], [73].

The novelty in this chapter is that we study the convergence and error properties for a time dependent convection-diffusion equation in two dimensions, on a domain containing three layers with different physical properties (paper [58]). Also, we defined a new frequencies spectrum and proposed a new definition for the smoothing factor of the multigrid method, (paper [61]). We used for our work the Fourier analysis method, which is often applied in order to design efficient multigrid approaches. For the begining, in Section 2.1 are presented a few theoretical aspects about the Fourier expansion ([49]) and the local Fourier analysis ([86, 90]) in Section 2.2 for the classical numerical methods .

2.1 The Fourier transform of a function

In order to use the local Fourier analysis, we briefly present in this section the Fourier transform and the discrete Fourier transform.

2.2 Local Fourier analysis of the convergence for the classical numerical methods

For the study of iterative numerical methods, one of the most efficient and frequently used methods is the local Fourier analysis (LFA). This method is used in order to compute the amplification factor, the smoothing factor, the error reduction factor and the convergence factor for a numerical method. The computing of the smoothing factor is very important because it's analysis allows designing efficient components of the multigrid method, being well known that this method has to be adapted to each type of problem that has to be solved. The LFA, first introduced by A. Brandt in [8], [9], [10], has become a widely used method [29], [86], [89] on a large variety of problems.

We propose here a new definition (paper [61]) for the smoothing factor- used for both the design and analysis of the multigrid method, and the new way to split the frequency spectrum into high and low frequencies for the one dimensional case (paper [61]).

The model problem used here is the mathematical representation of the stationary convection-diffusion process:

$$\begin{cases} -\varepsilon u''(x) + au'(x) = f(x), & x \in \Omega = (0,1), \\ u(x) = 0, & x \in \Gamma = \{0,1\}, \end{cases} \tag{2.1}$$

where u is the concentration of the substance, f is a possible perturbation term for the concentration, due for example to chemical reactions, ε is the diffusion coefficient and a is the convection coefficient.

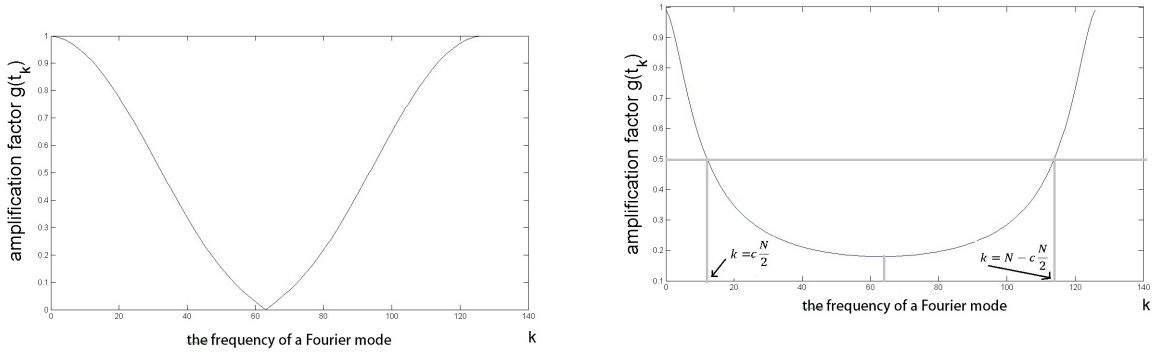
2.2.1 The amplification factor and the smoothing factor of a numerical method

Definition 2.2.1. *[89] For a numerical iterative method,* ***the amplification factor*** $g(t_k)$ *is the ratio between the coefficient* $c_k^{(m)}$ *after* m *iterates and the coefficient from the previous iterative step,* $c_k^{(m-1)}$.

Property 2.2.1. *[89] The convergence speed of a numerical method is better when the module of the amplification factor smaller than 1.*

Theorem 2.2.1. *[61] The amplification factor of the Gauss-Seidel method for the problem (2.1) has the module*

$$|g(t_k)| = \frac{|2\varepsilon - ah_l|}{\sqrt{(2\varepsilon + ah_l)^2 + (4\varepsilon)^2 - 8\varepsilon(2\varepsilon + ah_l)\cos(t_k)}}, k = 0, \ldots, n_l. \tag{2.2}$$



(a) Pondered Jacobi method for $\omega = 0.5$ **(b)** Gauss-Seidel method

Figure 2.1: The module of the amplification factor for $l = 6, \varepsilon = 0.1$ şi $a = 10$ in the one-dimensional case

Theorem 2.2.2. *[61] For the pondered Jacobi method the module of the amplification factor for the problem (2.1) is*

$$|g(t_k)| = \sqrt{(1-\omega+\omega\cos t_k)^2 + \left(\frac{ah\omega}{2\varepsilon}\sin t_k\right)^2}. \tag{2.3}$$

Remark 2.2.1. *[61] In **Figure 2.1b** and **Figure 2.1a** it can be seen that for the frequencies $k \in \left[c\frac{N}{2}, N - c\frac{N}{2}\right], c \in (0,1)$ the amplification factor has the module $|g(t_k)| \leq \frac{1}{2}$. This means that for the error components that have these frequencies, the numerical iterative method is efficient because in the worst case they are reduced by a factor of at least $\frac{1}{2}$ on each iteration step. On the other hand, for the components having frequencies between $\left(0, c\frac{N}{2}\right) \cup \left(N - c\frac{N}{2}, N\right)$, the amplification factor is almost 1, thus for these frequencies the method is not efficient as these components remain almost unchanged after one iterative step.*

Definition 2.2.2. *[86], [89] The **smoothing factor** of a numerical iterative method having the iteration matrix M, is the worst amplification factor module, taken for all the high frequencies and is denoted by $\rho(M) = \max\{|g(t_k)|, t_k \in T_{high}\}$.*

level	$l=1$	$l=2$	$l=3$	$l=4$	$l=5$	$l=6$
$\rho(M_J)$	1	1	1	1	1	1
$\rho(M_{\omega J})$	0	0.5	0.8536	0.9619	0.9904	0.9976
$\rho(M_{GS})$	0.3333	0.4472	0.6786	0.8756	0.9637	0.9905

Table 2.1: The smoothing factor of Jacobi (J), Jacobi pondered (ωJ) and Gauss-Seidel (GS) methods for problem (2.1), $a = 0, \varepsilon = 1$

The data from **Table 2.1** show that the convergence speed of the studied methods decreases as the grid step becomes smaller due to the poor reduction of the low frequencies. Thus, these numerical methods are slowly convergent.

2.2.2 The frequencies spectrum

One of the most efficient methods to overcome this disadvantage of a classical iterative method is the multigrid method, which combines the property of such a method to reduce the high frequencies with the coarse grid correction method that has complementary properties: it reduces well the low frequencies.

Theorem 2.2.3. *[61] The smoothing factor of an iterative numerical method that has the iteration matrix M is*

$$\rho(M) = \max_k \left\{ |g(t_k)|, t_k \in T_h \right\}, \tag{2.4}$$

with: $T_{high} = \left\{k | k = c\frac{N}{2}, ..., \frac{N}{2} - 1\right\}$, *the high frequencies domain from the expansion and where:* $g(t_k) = \frac{c_k^{(m)}}{c_k^{(m-1)}}, k = 1, ..., \frac{N}{2} - 1$ *is the ratio of the coefficients of order* k *in the Fourier-transform expansion of the error,* $c \in (0, 1)$ *being a fixed constant.*

Corolary 2.2.1. *[61] The Fourier expansion in a point* $x_s, s = 0, ..., N-1$ *for a function that has* N *real values* $e_0, e_1, ..., e_{N-1}$, *is*

$$E_s = B_0 + \sum_{k=1}^{N/2-1} B_k \sin\left(\frac{2\pi ks}{N} - \varphi\right), \qquad B_0, B_k \in \mathbb{R}, k = 1, 2, ..., \frac{N}{2} - 1. \tag{2.5}$$

Theorem 2.2.4. *[61] The smoothing factor of the Gauss-Seidel method for problem (2.1), is*

$$\rho(M_{GS}) = \frac{|2\varepsilon - ah_l|}{\sqrt{(2\varepsilon + ah_l)^2 + (4\varepsilon)^2 - 8\varepsilon(2\varepsilon + ah_l)\cos\pi c}},$$

and for the pondered Jacobi method

$$\rho(M_{\omega J}) = \sqrt{(1 - \omega + \omega\cos\pi c)^2 + \left(\frac{ah\omega}{2\varepsilon}\sin\pi c\right)^2}.$$

2.2.3 The results obtained using the Fourier analysis for the numerical methods studied here

Remark 2.2.2. *The Jacobi method should not be used as a smoother in the multigrid method due to the fact that it does not have the usual property of a numerical iterative method to efficiently reduce the high frequency error components, but it reduces only the middle part of the frequency spectrum (*$k \in (0, N/2 - 1)$*).*

Remark 2.2.3. *For the Gauss-Seidel or pondered Jacobi method, the property of reducing the high frequencies determined using* ***Theorem 2.2.3*** *is even better than was computed untill now using* ***Definition 2.2.2*** *for each iterative step and applying these methods more times makes them even more efficient.*

Remark 2.2.4. *For the case of dominant convection (Table 2.3) as the number of layers used is growing, the amplification factor becomes smaller, thus it is better to use the numerical iterative method on a grid having more levels (at least six for the problem studied here) in order to have a reduction of the low frequencies components of the error, and even so the reduction is not efficient.*

2.3 The convergence of the Multigrid Method using the-Local Fourier Analysis

2.3.1 Mathematical model

In order to apply the local Fourier analysis, we used the convection-diffusion equation

$$\begin{cases} c\dfrac{\partial u(\boldsymbol{x}, t)}{\partial t} + \boldsymbol{v} \cdot \bigtriangledown u(\boldsymbol{x}, t) = \boldsymbol{d} \bigtriangleup u(\boldsymbol{x}, t) + \alpha u(\boldsymbol{x}, t) + f(\boldsymbol{x}), & t \geq 0, \boldsymbol{x} \in \Omega \\ u(\boldsymbol{x}_0, t) = u_0, & t \geq 0. \end{cases} \tag{2.6}$$

$a=0, e=1$		$l=3$	$l=4$	$l=5$	$l=6$
c=0.5	GS	0.4472	0.4472	0.4472	0.4472
		0.8756	0.9637	0.9905	0.9976
	J	0.9239	0.9808	0.9952	0.9988
		1.0000	1.0000	1.0000	1.0000
	ωJ	0.5000	0.5000	0.5000	0.5000
	($\omega = 0.5$)	0.9619	0.9904	0.9976	0.9994

Table 2.2: The smoothing factor of Gauss-Seidel (GS) and pondered Jacobi (ωJ) methods for model problem (2.1), $a = 0, \varepsilon = 1$ **-pure diffusion**

$a=10, e=0.1$		$l=3$	$l=4$	$l=5$	$l=6$
c=0.5	GS	0.4635	0.1730	0.0817	0.2502
		0.8845	0.7851	0.7632	0.9911
	J	3.1250	1.5625	0.9981	0.9990
		3.1250	1.5625	1.0000	1.0000
	ωJ	1.6406	0.9276	0.6345	0.5368
	($\omega = 0.5$)	1.6406	1.0167	0.9983	0.9994

Table 2.3: The smoothing factor of Gauss-Seidel (GS) and pondered Jacobi (ωJ) methods for model problem (2.1) and $a = 10, \varepsilon = 0.1$ - **dominant convection**

Here, $u(\boldsymbol{x}, t)$ represents the concentration of the substance transported through the blood stream, $\boldsymbol{v}$ is the convection coefficient and $\boldsymbol{d}$ is the diffusion coefficient.

The substance applied on the skin has a constant concentration at any moment of time $u(0,t) = u_0,\ t \geq 0$. On the frontiers between the skin layers the law of flux conservation gives

$$\left[\!\left[-d\frac{\partial u(x,t)}{\partial x}\right]\!\right] = 0, x = x_{0i}\ , i = 1, 2, ..., n_d, t \geq 0, \tag{2.7}$$

n_d is the number of layers where the diffusion takes place and $[\![a(x,t)]\!] = a(x^+, t) - a(x^-, t)$.

Property 2.3.1. *[58] For finite differences method of discretization, the coefficients from system obtained are:*

$$\begin{aligned} q_0 &= -\frac{2\alpha h}{3} + \frac{2ch}{3dt} + \frac{2d}{h}, \\ q_1 &= -\frac{\alpha h}{6} + \frac{ch}{6dt} - \frac{v}{2} - \frac{d}{h}, \\ q_2 &= -\frac{\alpha h}{6} + \frac{ch}{6dt} + \frac{v}{2} - \frac{d}{h}, \\ f_i &= f(x_i) + \frac{ch}{6dt}\left(u_{i-1}^{ant} + 4u_i^{ant} + u_{i+1}^{ant}\right). \end{aligned} \tag{2.8}$$

Property 2.3.2. *[58] For finite differences method with backward Euler discretization scheme for the time direction, the coefficients in the system obtained after the*

discretization process will be:

$$q_0 = -\alpha + \frac{c}{dt} + \frac{2d}{h^2}, q_1 = -\frac{v}{2h} - \frac{d}{h^2}, q_2 = \frac{v}{2h} - \frac{d}{h^2}, f_i = f(x_i) + c\frac{u_i^{ant}h}{dt}. \tag{2.9}$$

2.3.2 The components of the multigrid method

Property 2.3.3. *[58] The matrix of the operator L_h of the system obtained after the discretization process is*

$$\widehat{L}_h = \begin{pmatrix} \widetilde{L}_h(\theta) & 0 \\ 0 & \widetilde{L}_h(\overline{\theta}) \end{pmatrix}, \tag{2.10}$$

where $\widetilde{L}(\theta) = q_0 + q_1 e^{-i\theta} + q_2 e^{i\theta}$.

Property 2.3.4. *[58] The matrix of the smoothing operator corresponding to the Gauss-Seidel method is*

$$\widehat{S}_h = \widehat{S}_h^n \widehat{S}_h^r = \frac{1}{4}\begin{pmatrix} (a+1)^2 - (a-1)(b-1) & (b-1)(a-b) \\ -(a-1)(a-b) & (b+1)^2 - (a-1)(b-1) \end{pmatrix} \tag{2.11}$$

where $a = 1 - \frac{\omega}{q_0}\widetilde{L}(\theta), b = 1 - \frac{\omega}{q_0}\widetilde{L}(\overline{\theta})$.

Property 2.3.5. *[58] The restriction operator has the matrix*

$$\widehat{I}_h^{2h} = \frac{1}{2}\begin{pmatrix} 1 + cos\theta & 1 + cos\overline{\theta} \end{pmatrix}. \tag{2.12}$$

Property 2.3.6. *[58] The matrix of the coarse grid operator is*

$$\widehat{L}_{2h}^{-1}(2\theta) = \frac{1}{\widetilde{L}_{2h}(2\theta)}. \tag{2.13}$$

Property 2.3.7. *[58] The matrix of the prolongation operator is*

$$\widehat{I}_{2h}^{h} = \frac{1}{2}\begin{pmatrix} 1 + cos\theta \\ 1 + cos\overline{\theta} \end{pmatrix}. \tag{2.14}$$

Two-grid operator

The matrices (2.10)- (2.14) are used for the two-grid operator for the multigrid method

$$\widehat{M}_h^{2h} = \widehat{S}_h^{\nu_2} \widehat{K}_h^{2h} \widehat{S}_h^{\nu_1}, \tag{2.15}$$

where

$$\widehat{K}_h^{2h} = \widehat{I}_h - \widehat{I}_{2h}^{h}(\widehat{L}_{2h})^{-1}\widehat{I}_h^{2h}\widehat{L}_h \tag{2.16}$$

is the matrix of the coarse grid correction operator.

Definition 2.3.1. *[86]* ***The asymptotic convergence factor of the 2-grid method is***

$$\rho_{loc}(M_h^{2h}) = sup\left\{\rho_{loc}(\widehat{M}_h^{2h}(\theta)), \theta \in T^l = \left[-\frac{\pi}{2}, \frac{\pi}{2}\right], \theta \notin \Lambda\right\}. \tag{2.17}$$

Definition 2.3.2. *[86]* ***The error reduction factor of the 2-grid method is***

$$\sigma_{loc}(M_h^{2h}) = sup\left\{||(\widehat{M}_h^{2h}(\theta))||, \theta \in T^l = \left[-\frac{\pi}{2}, \frac{\pi}{2}\right], \theta \notin \Lambda\right\}. \tag{2.18}$$

Here $||\cdot||$ *is the Euclidian norm in* $\mathbb{C}^2$ *and*

$$\Lambda = \left\{\theta \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right], \widetilde{L}_h(\theta) = 0 \quad or \quad \widetilde{L}_{2h}(\theta) = 0\right\}. \tag{2.19}$$

Definition 2.3.3. *[86]* ***The smoothing factor of the 2-grid method is***

$$\mu_{loc}(S_h, \nu) = sup\left\{\sqrt[\nu]{\rho_{loc}(\widehat{S}_h^{\nu_2}(\theta)\widehat{Q}_h^{2h}\widehat{S}_h^{\nu_1}(\theta))}, \theta \in T^l = \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]\right\}. \tag{2.20}$$

where $\rho_{loc}(\widehat{S}(\theta))$ *is the spectral radius of* $\widehat{S}(\theta)$, $\nu = \nu_1 + \nu_2$.

$$Q_h^{2h}\varphi(\theta, \cdot) = \begin{cases} 0, & \theta \in T^l = \left[-\frac{\pi}{2}, \frac{\pi}{2}\right] \\ \varphi(\theta, \cdot), & \theta = \overline{\theta} \in T^h \end{cases},$$

$$\widehat{Q}_h^{2h}(\theta) = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}, \theta \in T^l.$$

2.3.3 Numerical results obtained with the local Fourier analisys for the studied problem

The smoothing factor

Theorem 2.3.1. *[58] If* ω*=1 then the matrix (2.11) of the smoothing operator of the Gauss-Seidel method after* $\nu = \nu_1 + \nu_2$ *steps is*

$$\widehat{S}_h^{\nu}(\theta) = \frac{1}{4^{\nu-1}}(a-b)^{2(\nu-1)}\widehat{S}_h(\theta)$$

and has the eigenvalues $\lambda_1 = 0$ *and* $\lambda_2 = \frac{(a-b)^{2(\nu-1)}}{4^{\nu}}[(b+1)^2 - (a-1)(b-1)]$, a *and* b *being given in Property 2.3.2.*

Theorem 2.3.2. *[58] The smoothing factor for the Gauss-Seidel method for model problem (2.6) is*

$$\mu_{loc}(S_h, \nu) = \left(\frac{q_1 + q_2}{q_0}\right)^2 \sqrt[\nu]{\frac{q_0 + q_1 + q_2}{2(q_1 + q_2)}}. \tag{2.21}$$

For the finite differences method, q_0, q_1, q_2 *are the coefficients*

$q_0 = -\alpha + \frac{c}{dt} + \frac{2d}{h^2}, q_1 = -\frac{v}{2h} - \frac{d}{h^2}, q_2 = \frac{v}{2h} - \frac{d}{h^2}, f_i = f(x_i) + c\frac{u_i^{ant}h}{h_t},$

and for the finite elements method

$q_0 = -\frac{2\alpha h}{3} + \frac{2ch}{3h_t} + \frac{2d}{h}, q_1 = -\frac{\alpha h}{6} + \frac{ch}{6h_t} - \frac{v}{2} - \frac{d}{h}, q_2 = -\frac{\alpha h}{6} + \frac{ch}{6h_t} + \frac{v}{2} - \frac{d}{h},$

$f_i = f(x_i) + \frac{ch}{6h_t}\left(u_{i-1}^{ant} + 4u_i^{ant} + u_{i+1}^{ant}\right),$

On the boundaries between the layers

$q_0 = \frac{1}{h}(d(x_i^-) + d(x_i^+)), q_1 = -\frac{1}{h}d(x_i^-), q_2 = -\frac{1}{h}d(x_i^+)), f_i = 0.$

For $a = 10^{-4}, ct = 1, d_1 = 1 \cdot 10^{-12}, d_2 = 1 \cdot 10^{-10}, d_3 = 3 \cdot 10^{-10}$, $v_1 = 1 \cdot 10^{-9}, v_2 = 1 \cdot 10^{-6}, v_3 = 1 \cdot 10^{-6}$ the smoothing factors for the Gauss-Seidel relaxation method are presented in **Table 2.4**.

	Finite Differences Method				
	ν=1	ν=2	ν=3	ν=4	ν=5
$l = 3$	0.0058	$8.8272 \cdot 10^{-4}$	$4.7260 \cdot 10^{-4}$	$3.4581 \cdot 10^{-3}$	$2.8670 \cdot 10^{-3}$
$l = 4$	0.0215	0.0066	0.0044	0.0037	0.0032
$l = 5$	0.0667	0.0409	0.0348	0.0321	0.0306
	Finite Elements Method				
	ν=1	ν=2	ν=3	ν=4	ν=5
$l = 3$	0.3493	0.2801	0.2603	0.2509	0.2454
$l = 4$	0.2809	0.2125	0.1937	0.1849	0.1798
$l = 5$	0.0991	0.0533	0.0434	0.0391	0.0368

Table 2.4: The smoothing factor as a function of $\nu = \nu_1 + \nu_2$ and l

Remark 2.3.1. *The data from **Table 2.4** show that the Gauss-Seidel red-black relaxation method is a very good smoother for this problem as the smoothing factors in the cases presented here are $\leq$ 0.5; both the discretization methods lead to good smoothing factors.*

Asymptotic convergence factor and error reduction factor

Property 2.3.8. *[58] The matrix of the two-grid operator for the problem (2.6) is*

$$\widehat{M}_h^{2h} = \widehat{S}_h^{\nu_2}(\theta)\left[\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix} - \frac{1}{4\widetilde{L}_{2h}(2\theta)}\begin{pmatrix}(1+cos\theta)^2\widetilde{L}_h(\theta) & (1-cos^2\theta)\widetilde{L}_h(\overline{\theta})\\ (1-cos^2\theta)\widetilde{L}_h(\theta) & (1+cos\theta)^2\widetilde{L}_h(\overline{\theta})\end{pmatrix}\right]\widehat{S}_h^{\nu_1}(\theta). \quad (2.22)$$

Number of	**Finite differencesmethod**		**Finite element method**	
smoothig steps	$\rho_{loc}(M_h^{2h})$	$\sigma_{loc}(M_h^{2h})$	$\rho_{loc}(M_h^{2h})$	$\sigma_{loc}(M_h^{2h})$
$\nu_1 = 0, \nu_2 = 1$	0.1224	0.1731	0.0153	0.1989
$\nu_1 = 1, \nu_2 = 0$	0.1224	0.3297	0.0153	0.2420
$\nu_1 = 2, \nu_2 = 2$	$7.5569 \cdot 10^{-4}$	0.0019	$5.6099 \cdot 10^{-6}$	$2.0141 \cdot 10^{-5}$
$\nu_1 = 3, \nu_2 = 2$	$1.3862 \cdot 10^{-4}$	$3.5209 \cdot 10^{-4}$	$4.8333 \cdot 10^{-7}$	$1.7152 \cdot 10^{-6}$

Table 2.5: Asymptotic convergence factor and error reduction factor (l=6)

Remark 2.3.2. *The data from **Table 2.5** show that the multigrid method is very rapidly convergent: if at least one smoothing step is performed before and after the coarse grid correction, then the error is reduced by at least a 10^{-2} factor per multigrid cycle.*

Numerical results

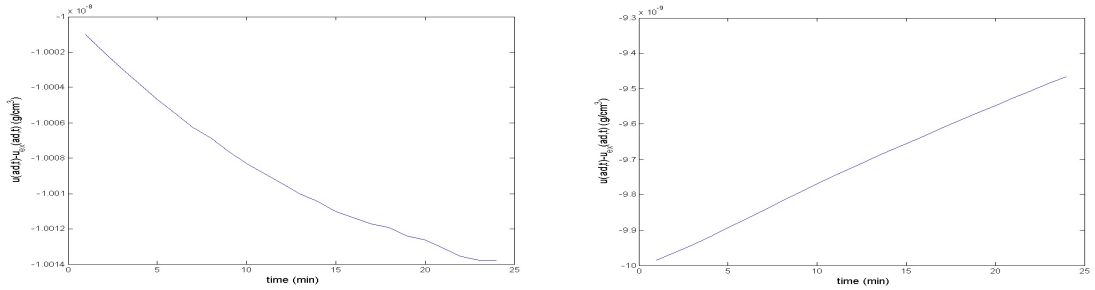
The problem (2.6) has been solved on a domain containing tree layers with different diffusion and convection coefficients ([55], [71]).

The error was computed for the following exact solution

$$u_{ex}(x,t) = x^2 + t, max(|ue(x,t)|) = 1.44 \cdot 10^3, x \in \Omega = [0, 1620nm], t \in [0, 24min].$$

The time step in the discretization process has been $dt = 60s$.

Figures 2.2 and Table 2.6 represent the error after eight multigrid cycles, with two smoothing steps before and two after the coarse grid correction.



(a) Finite Differences Method **(b)** Finite Elements Method

Figure 2.2: The multigrid error at ad=100nm in the skin for $v_1 = 1.0 \cdot 10^{-10}; v_2 = 1.0 \cdot 10^{-7}; v_3 = 1.0 \cdot 10^{-7}; d_1 = 1 \cdot 10^{-12}; d_2 = 1 \cdot 10^{-10}; d_3 = 3 \cdot 10^{-10}; c = 10^4; a = 0.$

	FD		FEM		
	$max_i\|u(x_i) - u_{ex}(x_i)\|$	$\|\|u - u_{ex}\|\|$		$max_i\|u(x_i) - u_{ex}(x_i)\|$	$\|\|u - u_{ex}\|\|$
l=3	$1.0000 \cdot 10^{-8}$	$4.8970 \cdot 10^{-8}$	l=3	$9.9847 \cdot 10^{-9}$	$4.7554 \cdot 10^{-8}$
l=4	$1.0057 \cdot 10^{-8}$	$3.1812 \cdot 10^{-8}$	l=4	$1.6648 \cdot 10^{-8}$	$5.9871 \cdot 10^{-8}$
l=5	$1.0617 \cdot 10^{-7}$	$1.6775 \cdot 10^{-7}$	l=5	0.1079	0.2457
l=6	$8.3356 \cdot 10^{-7}$	$1.9383 \cdot 10^{-6}$	l=6	13.3535	30.8845
l=7	$5.5076 \cdot 10^{-6}$	$1.4308 \cdot 10^{-5}$	l=7	104.0864	245.1033

Table 2.6: Multigrid error for Finite Differences and Finite Elements Methods

Chapter 3

The multigrid method for nonstationary convection-diffusion problems

The general methodology and the local Fourier analysis described in the previous chapters lead to various multigrid algorithms according to the way that multigrid components are chosen and how they are used. All the components have to be adapted to the problem to be solved, thus there is no unique multigrid algorithm for all types of differential equations.

In Section 3.2, starting from the local Fourier analysis of the multigrid method, we study (paper [59]) the efficient design of the method, it's convergence and error reduction properties for a nonstationary convection-diffusion equation in two dimensions. This study is an extension of the results presented in Chapter 2.3.

In Section 3.3 we use the streamline diffusion method in order to reduce the oscillations that appear when the physical process the covection is dominant.

In Section 3.4 we present an example of using the multigrid method in the tridimensional space and in Section 3.5 in a medium that is made of three layers with different diffusion properties (paper [60]).

3.1 Mathematical model

The nonstationary convection-diffusion equation is

$$\begin{cases} c_t \dfrac{\partial u(\boldsymbol{x},t)}{\partial t} + V \cdot \bigtriangledown u(\boldsymbol{x},t) = D \bigtriangleup u(\boldsymbol{x},t) + \alpha u(\boldsymbol{x},t) + f(\boldsymbol{x},t), & t \geq 0, \boldsymbol{x} \in \Omega \\ u(\boldsymbol{x}_0,t) = u_0 & t \geq 0, \boldsymbol{x}_0 \in \partial\Omega. \end{cases} \tag{3.1}$$

Here $u(\boldsymbol{x},t)$ represents the concentration of the substance transported through the blood stream, $V = (V_1, V_2)$ is the vector of convection coefficients and D is the diffusion coefficient, $\Omega = [a,b] \times [c,d]$ is the domain where the physical process takes place.

The substance applied on the skin has a constant concentration at any moment of time

$$u((x,c),t) = u_0, \ t \geq 0, a \leq x \leq b. \tag{3.2}$$

On the frontiers between the skin layers the law of flux conservation gives

$$\left[\!\left[-D\frac{\partial u(\boldsymbol{x},t)}{\partial \boldsymbol{n}}\right]\!\right] = -D^{+}\frac{\partial u(x^{+},t)}{\partial \boldsymbol{n}} + D^{-}\frac{\partial u(x^{-},t)}{\partial \boldsymbol{n}} = 0,$$
$$\boldsymbol{x} = \boldsymbol{x}_{0i}, i = 1,2,...,n_d, t \geq 0, \tag{3.3}$$

n_d is the number of layers where the convection-diffusion process takes place.

The system obtained from equation (3.1) using a discretization method will be

$$\begin{cases} q_1 u_{i-1,j+1} + q_2 u_{i,j+1} + q_3 u_{i+1,j+1} + q_4 u_{i-1,j} + q_5 u_{i,j} + q_6 u_{i+1,j} + \\ \qquad + q_7 u_{i-1,j-1} + q_8 u_{i,j-1} + q_9 u_{i+1,j-1} = f_{i,j}, \quad i = 1,2,..,n_x, j = 1,2,..,n_y, \\ u|_{\partial\Omega} = u_0, \end{cases} \tag{3.4}$$

Property 3.1.1. *[59] If the finite differences discretization method with the explicit backward Euler scheme is used on the time direction for the equation (3.1), the coefficients from system (3.4) are*

$$q_5 = -\alpha + \frac{c_t}{h_t} + 2D\left(\frac{1}{h_x^2} + \frac{1}{h_y^2}\right) - V\left(\frac{1}{h_x} + \frac{1}{h_y}\right), q_4 = -\frac{d}{h_x^2}, q_6 = \frac{V}{h_x} - \frac{D}{h_x^2},$$
$$q_8 = -\frac{d}{h_y^2}, q_2 = \frac{V}{h_y} - \frac{D}{h_y^2}, f_{i,j} = f(x_i, y_j) + \frac{c_t}{h_t}u_{i,j}^{ant}, \qquad i = 1,...,n_x, j = 1,...,n_y.$$

Property 3.1.2. *[59] If the equation (3.1) is discretized using the finite elements method, the system obtained is*

$$\left[\!\left[\begin{array}{ccc} & K_{24}^{\Omega_4} & \\ K_{34}^{\Omega_1} + K_{21}^{\Omega_4} - k_{i,i-1} & K_{23}^{\Omega_4} + K_{14}^{\Omega_3} - k_{i,i+n_x} & \\ & K_{31}^{\Omega_1} & \end{array} \right.\right. \left.\left. \begin{array}{c} K_{13}^{\Omega_3} \\ K_{33}^{\Omega_1} + K_{44}^{\Omega_2} + K_{11}^{\Omega_3} + K_{22}^{\Omega_4} - k_{i,i} \\ K_{32}^{\Omega_1} + K_{41}^{\Omega_2} - k_{i,i-n_x} \end{array} \right.\right. \left.\left. \begin{array}{c} \\ K_{43}^{\Omega_2} + K_{12}^{\Omega_3} - k_{i,i+1} \\ K_{42}^{\Omega_2} \end{array} \right.\right] u_i =$$
$$= f_3^{\Omega_1} + f_4^{\Omega_2} + f_1^{\Omega_3} + f_2^{\Omega_4} + \sum_{j=1}^{N} c_{ij} c_t \frac{u_j^{ant}}{h_t}, \qquad i = 1,...,N, \tag{3.5}$$

where

$$\left(K_{ij}^{\Omega_e}\right)_{i,j=\overline{1:4}} = \frac{D}{6}\begin{bmatrix} 2r_0 + \frac{2}{r_0} & -2r_0 + \frac{1}{r_0} & -r_0 - \frac{1}{r_0} & r_0 - \frac{2}{r_0} \\ -2r_0 + \frac{1}{r_0} & 2x + \frac{2}{r_0} & r_0 - \frac{2}{r_0} & -r_0 - \frac{1}{r_0} \\ -r_0 - \frac{1}{r_0} & r_0 - \frac{2}{r_0} & 2r_0 + \frac{2}{r_0} & -2r_0 + \frac{1}{r_0} \\ r_0 - \frac{2}{r_0} & -r_0 - \frac{1}{r_0} & -2r_0 + \frac{1}{r_0} & 2r_0 + \frac{2}{r_0} \end{bmatrix} +$$
$$+ \frac{V(b_1 - a_1)}{12}\begin{bmatrix} -2 & -1 & 1 & 2 \\ -1 & -2 & 2 & 1 \\ -1 & -2 & 2 & 1 \\ -2 & -1 & 1 & 2 \end{bmatrix} + \frac{V(d_1 - c_1)}{12}\begin{bmatrix} -2 & 2 & 1 & -1 \\ -2 & 2 & 1 & -1 \\ -1 & 1 & 2 & -2 \\ -1 & 1 & 2 & -2 \end{bmatrix} +$$
$$+ \left(\frac{ct}{h_t} - \alpha\right)\frac{(b_1 - a_1)(d_1 - c_1)}{36}\begin{bmatrix} 4 & 2 & 1 & 2 \\ 2 & 4 & 2 & 1 \\ 1 & 2 & 4 & 2 \\ 2 & 1 & 2 & 4 \end{bmatrix}, \quad r_0 = \frac{d_1 - c_1}{b_1 - a_1}, \tag{3.6}$$

$$k_{i,i-1} = -\frac{D^+ - D^-}{2}, k_{i,i} = 0, k_{i,i+1} = \frac{D^+ - D^-}{2}, k_{i,i-n_x} = 0, k_{i,i+n_x} = 0.$$

3.2 Local Fourier Analysis of the nonstationary convection-diffusion problem

3.2.1 The components of the multigrid method

The operators for each step from above and their corresponding matrices are given in the following.

For $\theta = (\theta_1, \theta_2) \in T^l$,

$$T^l = \left[-\frac{\pi}{2}, \frac{\pi}{2}\right) \times \left[-\frac{\pi}{2}, \frac{\pi}{2}\right) \tag{3.7}$$

being the set of low frequencies, as in [86] we consider the following frequencies:

$$\theta^{(1)} = (\theta_1, \theta_2) \in T^l, \theta^{(2)} = (\overline{\theta}_1, \theta_2) \in T^h, \theta^{(3)} = (\theta_1, \overline{\theta}_2) \in T^h, \theta^{(4)} = (\overline{\theta}_1, \overline{\theta}_2) \in T^h,$$

with

$$\overline{\theta}_i = \begin{cases} \theta_i + \pi, & \text{if } \theta_i < 0 \\ \theta_i - \pi, & \text{if } \theta_i > 0 \end{cases}, \qquad i \in \{1, 2\}.$$

Property 3.2.1. *[59] The matrix of the operator L_h of the system is*

$$\widehat{L}_h = \begin{pmatrix} \widetilde{L}_h(\theta^{(1)}) & 0 & 0 & 0 \\ 0 & \widetilde{L}_h(\theta^{(2)}) & 0 & 0 \\ 0 & 0 & \widetilde{L}_h(\theta^{(3)}) & 0 \\ 0 & 0 & 0 & \widetilde{L}_h(\theta^{(4)}) \end{pmatrix}, \tag{3.8}$$

where

$$\widetilde{L_h}(\theta_1, \theta_2) = q_1 e^{-i\theta_1} e^{i\theta_2} + q_2 e^{i\theta_2} + q_3 e^{i\theta_1} e^{i\theta_2} + q_4 e^{-i\theta_1} + q_5 + q_6 e^{i\theta_1} + + q_7 e^{-i\theta_1} e^{-i\theta_2} + q_8 e^{-i\theta_2} + q_9 e^{i\theta_1} e^{-i\theta_2}$$

is an eigenvalue of the operator L_h.

Property 3.2.2. *[59], The matrix of the smoothing operator obtained with the Gauss-Seidel method is $\widehat{S}_h = \widehat{S}_h^n \widehat{S}_h^r$, where*

$$\widehat{S}_h^r = \frac{1}{2} \begin{pmatrix} \widetilde{S}(\theta^{(1)}) + 1 & 0 & 0 & \widetilde{S}(\theta^{(4)}) - 1 \\ 0 & \widetilde{S}(\theta^{(2)}) + 1 & \widetilde{S}(\theta^{(3)}) - 1 & 0 \\ 0 & \widetilde{S}(\theta^{(2)}) - 1 & \widetilde{S}(\theta^{(3)}) + 1 & 0 \\ \widetilde{S}(\theta^{(1)}) - 1 & 0 & 0 & \widetilde{S}(\theta^{(4)}) + 1 \end{pmatrix}, \tag{3.9}$$

$$\widehat{S}_h^n = \frac{1}{2} \begin{pmatrix} 1 + \widetilde{S}(\theta^{(1)}) & 0 & 0 & \widetilde{1} - S(\theta^{(4)}) \\ 0 & 1 + \widetilde{S}(\theta^{(2)}) & 1 - \widetilde{S}(\theta^{(3)}) & 0 \\ 0 & 1 - \widetilde{S}(\theta^{(2)}) & 1 + \widetilde{S}(\theta^{(3)}) & 0 \\ 1 - \widetilde{S}(\theta^{(1)}) & 0 & 0 & 1 + \widetilde{S}(\theta^{(4)}) \end{pmatrix} \tag{3.10}$$

and $\widetilde{S}(\theta) = 1 - \frac{\omega}{q_5} \widetilde{L}(\theta)$.

Property 3.2.3. *[59] The operator that makes the fine to coarse grid transfer has the matrix*

$$\widehat{I}_h^{2h} = \left(\widetilde{I}_h^{2h}(\theta^{(1)}) \quad \widetilde{I}_h^{2h}(\theta^{(2)}) \quad \widetilde{I}_h^{2h}(\theta^{(3)}) \quad \widetilde{I}_h^{2h}(\theta^{(4)})\right), \tag{3.11}$$

with

$$\begin{aligned} &\widetilde{I}_h^{2h}(\theta^{(1)}) = \frac{1}{4}(cos\theta_1+1)(cos\theta_2+1), \widetilde{I}_h^{2h}(\theta^{(2)}) = \frac{1}{4}(1-cos\theta_1)(cos\theta_2+1),\\ &\widetilde{I}_h^{2h}(\theta^{(3)}) = \frac{1}{4}(cos\theta_1+1)(1-cos\theta_2), \widetilde{I}_h^{2h}(\theta^{(4)}) = \frac{1}{4}(1-cos\theta_1)(1-cos\theta_2). \end{aligned} \tag{3.12}$$

The operator of the system that generates the solution on the coarse grid has the matrix

$$\widehat{L}_{2h}^{-1}(2\theta) = \frac{1}{\widetilde{L}_{2h}(2\theta)}. \tag{3.13}$$

Property 3.2.4. *[59] The matrix of the prolongation operator is*

$$\widehat{I}_{2h}^{h} = \frac{1}{4}\begin{pmatrix} (1+cos\theta_1)(1+cos\theta_2)\\ (1-cos\theta_1)(1+cos\theta_2)\\ (1+cos\theta_1)(1-cos\theta_2)\\ (1-cos\theta_1)(1-cos\theta_2) \end{pmatrix}. \tag{3.14}$$

Two-grid operator

The smoothig steps (3.9), (3.10) and the coarse grid correction (3.11), (3.13) and (3.14) are the multigrid method components [35] and the two-grid operator is

$$\widehat{M}_h^{2h} = \widehat{S}_h^{\nu_2}\widehat{K}_h^{2h}\widehat{S}_h^{\nu_1} \tag{3.15}$$

where the matrix of the coarse grid correction operator is

$$\widehat{K}_h^{2h} = \widehat{I}_h - \widehat{I}_{2h}^{h}(\widehat{L}_{2h})^{-1}\widehat{I}_h^{2h}\widehat{L}_h. \tag{3.16}$$

3.2.2 The smoothing factor of the multigrid method for the convection-diffusion problem

Definition 3.2.1. *[86] The smoothing factor of a numerical method that has the matrix of the smoothing operator $\widehat{S}_h$, is*

$$\mu_{loc}(S_h,\nu) = sup\left\{\sqrt[\nu]{\rho_{loc}(\widehat{Q}_h^{2h}\widehat{S}_h^{\nu}(\theta))}, \theta\in T^l\right\}, \tag{3.17}$$

where ν represents the number of smoothing steps, T^{low} is defined by (3.7) and the rest of the parameters involved are defined in the following.

From Property 3.2.2, the matrix of the smoothing operator is

$$\widehat{S}_h = \frac{1}{4}\begin{pmatrix} 1+a & 0 & 0 & 1-b\\ 0 & 1+c & 1-d & 0\\ 0 & 1-c & 1+d & 0\\ 1-a & 0 & 0 & 1+b \end{pmatrix}\begin{pmatrix} a+1 & 0 & 0 & b-1\\ 0 & c+1 & d-1 & 0\\ 0 & c-1 & d+1 & 0\\ a-1 & 0 & 0 & b+1 \end{pmatrix}, \tag{3.18}$$

where

$$a = 1-\frac{\omega}{q_5}\left(q_5+q_4e^{-i\theta_1}+q_6e^{i\theta_1}+q_8e^{-i\theta_2}+q_2e^{i\theta_2}\right), b = 1-\frac{\omega}{q_5}\left(q_5-q_4e^{-i\theta_1}-q_6e^{i\theta_1}-q_8e^{-i\theta_2}-q_2e^{i\theta_2}\right),$$
$$c = 1-\frac{\omega}{q_5}\left(q_5-q_4e^{-i\theta_1}-q_6e^{i\theta_1}+q_8e^{-i\theta_2}+q_2e^{i\theta_2}\right), d = 1-\frac{\omega}{q_5}\left(q_5+q_4e^{-i\theta_1}+q_6e^{i\theta_1}-q_8e^{-i\theta_2}-q_2e^{i\theta_2}\right).$$

Theorem 3.2.1. *[59] After ν smoothing steps, the smoothing operator will have the matrix*

$$\widehat{S}_h^{\nu} = \frac{1}{2}\begin{pmatrix} a^{2\nu-1}(a+1) & 0 & 0 & -a^{2\nu-1}(a+1) \\ 0 & c^{2\nu-1}(c+1) & -c^{2\nu-1}(c+1) & 0 \\ 0 & -c^{2\nu-1}(c-1) & c^{2\nu-1}(c-1) & 0 \\ -a^{2\nu-1}(a-1) & 0 & 0 & a^{2\nu-1}(a-1) \end{pmatrix}. \tag{3.19}$$

The matrix $\widehat{Q}_h^{2h}\widehat{S}_h^{\nu}$ has the eigenvalues

$$\lambda_1 = \lambda_2 = 0, \lambda_3 = \frac{a^{2\nu-1}(a-1)}{2}, \lambda_4 = c^{2\nu}.$$

For $\theta_1, \theta_2 \in T^l$ these values are represented in **Figures 3.1** and **3.2**.

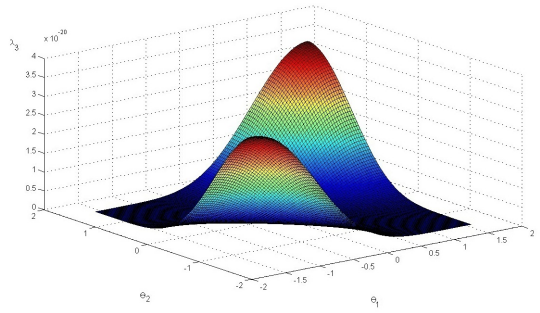


Figure 3.1: The eigenvalues λ_3 for $(\theta_1, \theta_2) \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right)\times\left[-\frac{\pi}{2}, \frac{\pi}{2}\right)$

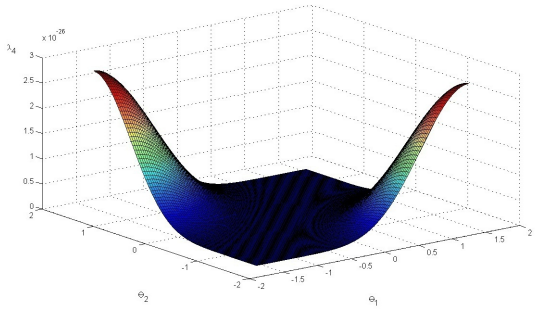


Figure 3.2: The eigenvalues λ_4 for $(\theta_1, \theta_2) \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right)\times\left[-\frac{\pi}{2}, \frac{\pi}{2}\right)$

Theorem 3.2.2. *[59] The eigenvalues λ_3 and λ_4 attain their maximum absolute values*

$$|\lambda_3|_{max} = \frac{\sqrt{q_5^2 + (q_4 - q_6 + q_8 - q_2)^2}|q_4 - q_6 + q_8 - q_2|^{2\nu-1}}{2q_5^{2\nu}}, \text{ for } \theta_1 = \theta_2 = \pm\frac{\pi}{2},$$

$$|\lambda_4|_{max} = \frac{|q_6 - q_4 + q_2 - q_8|^{2\nu}}{q_5^{2\nu}} \text{ for } \theta_1 = \mp\frac{\pi}{2}, \theta_2 = \pm\frac{\pi}{2}.$$

Remark 3.2.1. *The data in **Table 3.1** show that:*

-for the finite differences formula the smoothing factor increases as the number of grid involved in the multigrid method increases;

-if the finite element method is used, the smoothing factor remains constant when the number of grid increases, thus being independent of the grid size. So it is more efficient to use the finite element method when the grid number is higher.

	Finite Differences Method		
	ν=1	ν=2	ν=3
$l=3$	$2.5507 \cdot 10^{-8}$	$8.1475 \cdot 10^{-12}$	$5.5695 \cdot 10^{-13}$
$l=4$	$5.1015 \cdot 10^{-8}$	$2.3045 \cdot 10^{-11}$	$1.7682 \cdot 10^{-12}$
$l=5$	$1.0203 \cdot 10^{-7}$	$6.5180 \cdot 10^{-11}$	$5.6137 \cdot 10^{-12}$
$l=6$	$2.0406 \cdot 10^{-7}$	$1.8436 \cdot 10^{-10}$	$1.7822 \cdot 10^{-11}$
$l=7$	$4.0812 \cdot 10^{-7}$	$5.2144 \cdot 10^{-10}$	$5.6582 \cdot 10^{-11}$
	Finite Element Method		
	ν=1	ν=2	ν=3
$l=3$	0.2500	0.0625	0.0156
$l=4$	0.2500	0.0625	0.0156
$l=5$	0.2500	0.0625	0.0156

Table 3.1: The smoothing factor as a function of $\nu = \nu_1 + \nu_2$ and l

3.2.3 Asymptotic convergence factor and error reduction factor of the multigrid method for the convection-diffusion problem in two dimensions

Definition 3.2.2. *[86] The **asymptotic convergence factor** of the multigrid method is*

$$\rho_{loc}(M_h^{2h}) = sup\left\{\rho_{loc}(\widehat{M}_h^{2h}(\theta)), \theta \in T^l = \left[-\frac{\pi}{2}, \frac{\pi}{2}\right) \times \left[-\frac{\pi}{2}, \frac{\pi}{2}\right), \theta \notin \Lambda\right\} \tag{3.20}$$
$$with\ \Lambda = \left\{\theta \in T^l, \widetilde{L}_h(\theta) = 0 \quad or \quad \widetilde{L}_{2h}(\theta) = 0\right\}.$$

Definition 3.2.3. *[86] The **error reduction factor** of the multigrid method is*

$$\sigma_{loc}(M_h^{2h}) = sup\left\{||(\widehat{M}_h^{2h}(\theta))||, \theta \in T^l, \theta \notin \Lambda\right\} \tag{3.21}$$

Number of smoothing steps	**Finite differences method**		**Finite elements method**	
	$\rho_{loc}(M_h^{2h})$	$\sigma_{loc}(M_h^{2h})$	$\rho_{loc}(M_h^{2h})$	$\sigma_{loc}(M_h^{2h})$
$\nu_1=0, \nu_2=1$	$3.7215 \cdot 10^{-5}$	$4.7404 \cdot 10^{-7}$	0.6249	1.2866
$\nu_1=1, \nu_2=0$	$3.7215 \cdot 10^{-5}$	$4.7481 \cdot 10^{-7}$	0.6249	1.2866
$\nu_1=1, \nu_2=1$	$2.0619 \cdot 10^{-13}$	$5.6180 \cdot 10^{-14}$	0.2499	0.6152
$\nu_1=2, \nu_2=1$	$1.1424 \cdot 10^{-21}$	$1.2624 \cdot 10^{-26}$	0.0859	0.2276
$\nu_1=2, \nu_2=2$	$6.3298 \cdot 10^{-30}$	$2.8369 \cdot 10^{-39}$	0.0273	0.0753

Table 3.2: Asymptotic convergence factor and error reduction factor for l=6

Remark 3.2.2. *The data in **Table 3.2** show that it is sufficient to perform a small number of smoothing steps before and/or after the coarse grid correction for the error to be reduced efficiently when the finite differences method is used. And for the finite element method, the data show that at least one smoothing step should be performed before and after the coarse grid correction in order to reduce the error by at least a 0.6 factor per multigrid cycle.*

The results presented in **Table 3.1** and **Table 3.2** have been determined using the following parameters [71], [55]: $\Omega = [0,1] \times [0, 1620 \cdot 10^{-6}], c_t = 10^4, \alpha = 0, D = 10^{-12}, V = 10^{-9}, dt = 60sec.$

3.3 Streamline diffusion

This method was first described by T. Hughes and A. Brooks in [38], and by C. Johnson in [39], [40] for linear problems, then it was developed for nonstationary problems ([41]). The method mainly consists in adding a diffusion term (streamline upwind perturbation) in the flow direction to the weight functions

$$\int_a^b\int_c^d c_t \frac{\partial u}{\partial t} v dxdy + \int_a^b\int_c^d \boldsymbol{V} \bigtriangledown u\, v\, dxdy + \int_a^b\int_c^d D \bigtriangledown u \cdot \bigtriangledown v dxdy +$$

$$+ \int_a^b\int_c^d \mu \boldsymbol{V} \bigtriangledown u \cdot \boldsymbol{V} \bigtriangledown v dxdy - \oint_{\partial\Omega} D \frac{\partial u}{\partial n} v ds - \int_a^b\int_c^d \alpha u v dxdy =$$

$$= \int_a^b\int_c^d f v dxdy + \int_a^b\int_c^d f \mu \boldsymbol{V} \bigtriangledown v dxdy.$$

The parameter μ is defined as $\mu = \frac{\delta h}{|\boldsymbol{V}|}$, with

$$\delta = \begin{cases} 0, & Pe \leq 1 \\ \frac{1}{2}\left(1 - \frac{1}{Pe}\right), & Pe > 1 \end{cases}$$

This parameter determines the value of the perturbation added to the weight functions.

Definition 3.3.1. *Pe is the Péclet number $Pe = \frac{h|\boldsymbol{V}|}{2D}$ and it determines the ratio between the convection and the diffusion .*

In **Table 3.3** are presented the results obtained for the maximum value of the error in the grid points with the streamline diffusion method. The diffusion coefficients in the three layers of the domain were: $d_1 = 10^{-12}, d_2 = 10^{-10}, d_3 = 3 \cdot 10^{-10}, v_{x,1} = v_{y,1} = 10^{-9}, v_{x,2} = v_{y,2} = 10^{-6}, v_{x,3} = v_{y,3} = 10^{-6}$.

Level \ Time	s=1	s=2	s=3	s=4	s=5	s=6
$l = 3$	0.7340	2.1675	4.3383	7.2321	10.8486	15.1875
$l = 4$	1.1258	1.5041	2.1445	3.5727	5.3566	7.4953
$l = 5$	1.3536	2.1216	3.2705	4.7978	6.7008	8.9772
$\max_{(x,y)\in\Omega} \lvert ue(x,y,s\cdot\Delta t)\rvert$	61	121	181	241	301	361

Table 3.3: The error and maximum value of the exact solution for $\Delta t = 60sec$

Remark 3.3.1. *In **Figure 3.3** is illustrated the efficiency of streamline diffusion (SD): the oscillations generated by the multigrid method on the boundaries between the layers where the process takes place, in the case when SD is not used, dissapeared when the SD is used.*

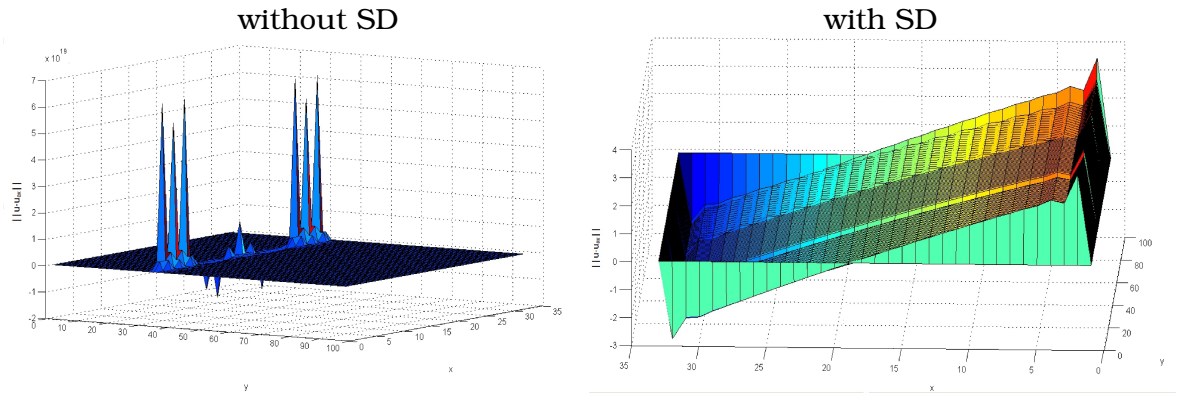


Figure 3.3: The error obtained without the SD method and with SD method at T=240 sec, on the grid with $l = 3$ for dominant convection

3.4 Multigrid method for the nonstationary convection-diffusion problem in three dimensions

The mathematical model for the convection-diffusion process of a substance in a liquid environement in the three-dimensional case is

$$c_t\frac{\partial u}{\partial t} + V_x\frac{\partial u}{\partial x} + V_y\frac{\partial u}{\partial y} + V_z\frac{\partial u}{\partial z} = -\left(D_x\frac{\partial^2 u}{\partial x^2} + D_y\frac{\partial^2 u}{\partial y^2} + D_z\frac{\partial^2 u}{\partial z^2}\right) + \alpha u + f,$$
$$(x, y, z) \in \Omega \subset \mathbf{R}^3, \qquad (3.22)$$

with Dirichlet conditions on the boundaries $u(x, y, z) = u_0(x, y, z), (x, y, z) \in \partial\Omega$. In (3.22) $\overline{V} = V_x\overline{i} + V_y\overline{j} + V_z\overline{k}$ is the convection coefficients vector, $\overline{D} = D_x\overline{i} + D_y\overline{j} + D_z\overline{k}$ the diffusion coefficients vector, α is the reaction coefficient [32], f is a source of possible perturbations. The system of linear equations generated with the second order finte differences method for the space direction and backward Euler discretization method for the time direction is

$$\begin{aligned}
&\left[c_t\frac{1}{h_t} + 2\left(\frac{D_x}{h_x^2} + \frac{D_y}{h_y^2} + \frac{D_z}{h_z^2}\right) - \alpha\right] u_{i,j,k,t} + \\
&+\left(\frac{v_x}{2h_x} - \frac{D_x}{h_x^2}\right) u_{i+1,j,k,t}\left(-\frac{v_x}{2h_x} - \frac{D_x}{h_x^2}\right) u_{i-1,j,k,t} + \\
&+\left(\frac{v_y}{2h_y} - \frac{D_y}{h_y^2}\right) u_{i,j+1,k,t} + \left(-\frac{v_y}{2h_y} - \frac{D_y}{h_y^2}\right) u_{i,j-1,k,t} + \\
&+\left(\frac{v_z}{2h_z} - \frac{D_z}{h_z^2}\right) u_{i,j,k+1,t} + \left(-\frac{v_z}{2h_z} - \frac{D_z}{h_z^2}\right) u_{i,j,k-1,t} = f_{i,j,k} + c_t\frac{1}{h_t}u_{i,j,k,t_0}.
\end{aligned} \qquad (3.23)$$

3.4.1 Numerical results obtained with the multigrid method in the tridimensional case

Example 3.4.1.

The domain where we used the multigrid method for problem 3.22 is a cube with the sides of 10^{-6} and the coefficients of the equation are $c_t = 1, D_x = D_y = D_z = 1 \cdot 10^{-8}, v_x = v_y = v_z = 0, \alpha = 0$ [71], [74]. If we use as initial data $u(x, y, z, t) =$

$u_{ex}(x,y,z,t) = x^2+y^2+z^2+t$, for any $(x,y,z) \in \partial\Omega$, the error obtained is represented in Figure 3.4.

level	max	$\|\|u-u_{ex}\|\|$
l=2	0.5068	0.6194
l=3	0.2722	0.4324
l=4	0.2720	0.3494
l=5	0.2721	0.3067

$max(u_{ex}(l, s \cdot dt)) = 540$,
s = 10 is the number of time steps used

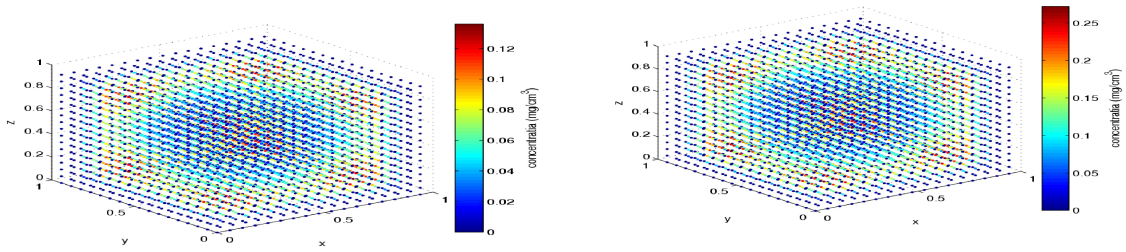
Table 3.4: The error of the multigrid method for problem 3.22

In Table 3.4 we denoted with

$$max = \max_{i=1:N} |u(x_i,y_i,z_i,t_s) - u_{ex}(x_i,y_i,z_i,t_s)|,$$

$$||u-u_{ex}|| = \sum_{i=1}^{N} \left[u(x_i,y_i,z_i,t_s) - u_{ex}(x_i,y_i,z_i,t_s) \right]^2,$$

$t_s = s \cdot h_t$ being the time moment untill we used the multigrid algorithm, $h_t = 60s$.



a. $t = 5 \cdot \Delta t$ b. $t = 10 \cdot \Delta t$

Figure 3.4: The error obtained for the example 3.4.1 for $t_5 = 5 \cdot h_t$ and $t_{10} = 10 \cdot h_t$

3.5 Multigrid method for the diffusion equation in a multilayer medium

The diffusion equation is a classic example of parabolic equation. In this section is designed a multigrid method for such an equation (paper [60]).

3.5.1 Mathematical model

The equation with boundary conditions is

$$\begin{cases} \dfrac{\partial u_i(z,t)}{\partial t} = D(z)\dfrac{\partial u_i(z,t)}{\partial z}, \\ u(0,t) = u_0, \\ u(b,t) = 0 \text{ sau } -D(z)\dfrac{\partial u(z,t)}{\partial z}\Big|_{z=b} = 0, \qquad i = 1,2,...,nd, t \geq 0. \\ \left[\!\left[-D(z)\dfrac{\partial u(z,t)}{\partial z}\right]\!\right]\Big|_{z=z_{0i}} = 0, \end{cases} \tag{3.24}$$

Property 3.5.1. *[60] The system of equations obtained after the discretization with the finite differences method is*

$$\frac{u_{i,t} - u_{i,t-1}}{h_t} = D(z_i)\frac{u_{i+1,t} - u_{i-1,t}}{2h_z},\ i = 1,2,...,N. \tag{3.25}$$

Remark 3.5.1. *[60] În the nodes on the boundaries between different layers of the skin the discretized system of equations is*

$$D(z_i^-)\frac{u_{i,t} - u_{i-1,t}}{h_z} = D(z_i^+)\frac{u_{i+1,t} - u_{i,t}}{h_z},\ i = 1,2,...,nd-1. \tag{3.26}$$

3.5.2 Numerical results

The multigrid method has been used in three directions:

1. If the concentration of the substance is known, the amount of it that reaches a certain level of the skin has been computed (**Table 3.5**)

The concentration has been computed at 9 minutes after the substance had been applied, using time intervals of 60 seconds, c_0 is the initial value of the concentration maintained constant on the skin surface. $D_1 = 1 \cdot 10^{-12}$, $D_2 = 1 \cdot 10^{-10}$, $D_3 = 3 \cdot 10^{-10}$, $c_0 = 2 \cdot 10^{-3}$ are the diffusion coefficients in the first layers of the skin. h is the depth where the concentration (of the substance that reaches that point) is computed. The second and third columns of the table show the values between which the concentration is comprised at the respective depth, for the grid with 4 and 6 levels.

$h(m)$	l=4	l=6
$20 \cdot 10^{-6}$	$[1.0734; 1.0967] \cdot 10^{-3}$	$[1.0563; 1.0622] \cdot 10^{-3}$
$120 \cdot 10^{-6}$	$[0.9551; 0.9566] \cdot 10^{-3}$	$[0.9360; 0.9363] \cdot 10^{-3}$
$200 \cdot 10^{-6}$	$[0.8415; 0.9551] \cdot 10^{-3}$	$[0.7719; 0.7976] \cdot 10^{-3}$
$1000 \cdot 10^{-6}$	$[0.0541; 0.0653] \cdot 10^{-3}$	$[0.0453; 0.0475] \cdot 10^{-3}$
$1600 \cdot 10^{-6}$	$[1.3482; 2.7545] \cdot 10^{-6}$	$[0.6432; 0.9670] \cdot 10^{-6}$

Table 3.5: The values of the concentration at the depth h

2. In order to maintain a required level of concentration of the active substance (to be efficient), the concentration needed on the surface of the skin has been computed. In **Table 3.6**: c represents the concentration needed at a depth equal to h; c_0 is the concentration to be applied on the skin surface; the diffusion coefficients that we worked with are $D_1 = 1 \cdot 10^{-12}$, $D_2 = 1 \cdot 10^{-10}$, $D_3 = 3 \cdot 10^{-10}$ [71], [74]. We used 31 interior nodes in each layer of the skin and 63 nodes on the layer where the concentration was required to be comprised between the given values.

$h(m)$	$c(g/cm^3)$	$c_0(g/cm^3)$
$15 \cdot 10^{-6}$	$1 \cdot 10^{-3} \pm 10^{-6}$	$1.2832 \cdot 10^{-3}$
$100 \cdot 10^{-6}$	$2 \cdot 10^{-4} \pm 10^{-7}$	$1.0578 \cdot 10^{-3}$
$200 \cdot 10^{-6}$	$3 \cdot 10^{-4} \pm 10^{-7}$	$5.5977 \cdot 10^{-3}$
$1000 \cdot 10^{-6}$	$4 \cdot 10^{-4} \pm 10^{-7}$	$14.5703 \cdot 10^{-3}$
$1600 \cdot 10^{-6}$	$5 \cdot 10^{-5} \pm 10^{-7}$	$26 \cdot 10^{-3}$

Table 3.6: The concentration c_0 that has to be kept on the surface in order to get the concentration c at the depth h

3. The way the concentration at a given depth changes in time has been studied, in the case when the substance had been applied for 2 minutes on the surface at a constant concentration (**Fig. 3.5**)

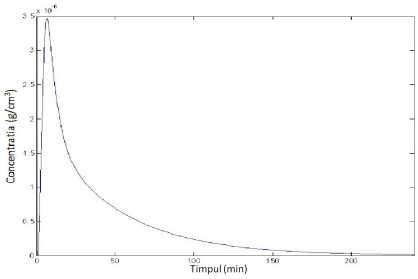


Figure 3.5: The variation of the concentration at the depth of 1000nm, for 6 hours (Nanoparticles have been applied on the surface for 2 minutes)

The multigrid method is rapidly convergent for many problems. The number of iterations needed for the method to be convergent for the studied problem in this case is presented in **Table 3.7**.

$t(min)$	1	2	3	4	5	10	20	50	100	150	200	240
nivel	$\varepsilon = 10^{-8}$											
3	4	6	7	7	7	4	5	4	4	3	2	2
4	3	5	6	6	6	4	4	4	3	3	2	2
5	3	4	5	6	6	4	3	3	3	2	2	2
	$\varepsilon = 10^{-10}$											
3	11	13	14	14	14	12	10	6	6	6	5	4
4	9	12	14	14	14	12	9	7	6	5	4	4
5	5	11	13	13	13	11	8	4	4	4	4	3

Table 3.7: The number of iterations needed for the multigrid method to be convergent

The condition for convergence was that the module of the difference between the values of two successive iterations, at the depth 200nm, to remain smaller than ε.

Chapter 4

The stellar prolongation method

In this chapter we introduce a new method "stellar prolongation" for solving differential equations that uses a system of grids with different grid steps, as it has been described in the papers [56] and [57].

4.1 The model problem

We use as a model problem the convection-diffusion equation

$$\begin{cases} -e\Delta u + au_x = f, & (x,y) \in (0,1)\times(0,1) = \Omega, \\ u = 0, & (x,y) \in \partial\Omega. \end{cases} \tag{4.1}$$

We denote by l the level on wich we want to determine the approximative values of the exact solution and by $n_l = 2^{l+1} - 1$ the number of modes from the grid on this level. If we use one of the discretization methods described in Chapter 1.1, we get the system of liniar equations on every level $l = 0, 1, ...$ This system contains n_l^2 equations and n_l^2 unknowns. First the discretized system is solved using the Gaussian elimination with partial pivoting, on G_{l_0} - a coarse grid on which the system1 has a small number of equations and unknowns. The values for l_0 can be chosen for example $l_0 = 2$ or $l_0 = 3$. So, the exact solution of the problem on the level l_0 is approximated by the values $u_i, i \in \{1, 2, ..., n_{l_0}^2\}$ (**Figure ??**), that only contain the error from the discretization method. Then, the solution obtained on the level l_0 is used for computing the solution on a superior level l. In order to solve the problem on the level l, the grid G_{l_0} has to be further divided. Thus, each domain from the grid, $\Omega_k, k = 1, ..., (n_0+1)^2$, will be splitted into $(n_i+1)^2$ subdomains (**Figure ??**), where $n_i = 2^{l_i+1} - 1$, and $l_i = l - l_0 - 1$.

4.2 The values on the boundaries of the subdomains

4.2.1 Stellar prolongation

In order to determine more accurately the values of the solution on the boundaries of Ω_k, one can use the solutions of the systems obtained discretizing the initial

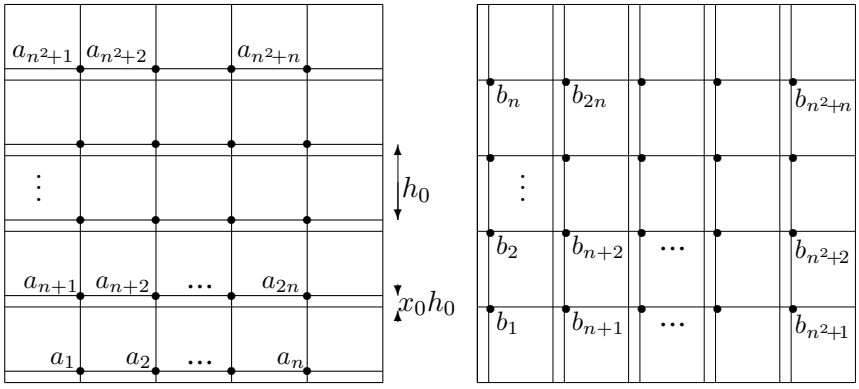


Figure 4.1: The nodes a_i and b_i from the grid G_l

equation in the grid points corresponding to the values a_i and b_i, $i = 1, 2, ..., n^2 + n, n = n_{l_0}$ from **Figure 4.1**.

The values a_k and $b_k, k = 1, 2, ..., n(n+1)$ depend on the vertical or horizontal distance ζ, by which they are translated from the old grid G_0 and will be denoted from now on by $a_k(\zeta)$ and $b_k(\zeta)$, $\zeta = jh, j = 1, ..., n_i$. They are the solutions of the systems

$$Aa = T, \tag{4.2}$$

$$Ba = T, \tag{4.3}$$

where the matrices A and B are given in the following.

4.2.2 Discretization with finite differences

Property 4.2.1. *[56] If the partial derivatives differential equation (4.1) is discretised with the second order finite differences method, the matrix A of system 4.2 will be*

$$A = \begin{bmatrix} C & D & \Theta & ... & \Theta & \Theta \\ S & C & D & ... & \Theta & \Theta \\ \Theta & S & C & ... & \Theta & \Theta \\ \vdots & & & & \ddots & \\ \Theta & \Theta & \Theta & ... & S & C \end{bmatrix} \in M_{(n_0+1)n_0,(n_0+1)n_0}(\mathbb{R}), \tag{4.4}$$

with

$$C = \begin{bmatrix} q_c & q_r & 0 & ... & 0 \\ q_l & q_c & q_r & ... & 0 \\ 0 & q_l & q_c & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & q_c \end{bmatrix}, D = \begin{bmatrix} q_u & 0 & 0 & ... & 0 \\ 0 & q_u & 0 & ... & 0 \\ 0 & 0 & q_u & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & q_u \end{bmatrix}, S = \begin{bmatrix} q_d & 0 & 0 & ... & 0 \\ 0 & q_d & 0 & ... & 0 \\ 0 & 0 & q_d & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & q_d \end{bmatrix},$$

$$C, D, S \in M_{n_0,n_0}(\mathbb{R}) \tag{4.5}$$

The term T in system (4.2) is a vector with the components

$$t_{in_0+j}(\zeta) = h_0^2 f\left(jh_0, ih_0+\zeta\right) - \left[\!\!\left[\begin{array}{ccc} 0 & q_u & 0 \\ q_l & q_c & q_r \\ 0 & q_d & 0 \end{array} \right]\!\!\right] u_{fr}\left(jh_0, ih_0+\zeta\right), \qquad (4.6)$$
$$i = 0, ..., n_0, j = 1, ..., n_0,$$

u_{fr} is a function which is zero inside the domain Ω on which the system is solved and equal to the boundaries values on $\partial\Omega$, h is the grid step on the level l_0 and

$$\left[\!\!\left[\begin{array}{ccc} 0 & q_u & 0 \\ q_l & q_c & q_r \\ 0 & q_d & 0 \end{array} \right]\!\!\right] = \left[\!\!\left[\begin{array}{ccc} 0 & -e\alpha y_0 & 0 \\ -e-\dfrac{ah_0}{2} & 2[e+\alpha(x_0+y_0)] & -e+\dfrac{ah_0}{2} \\ 0 & -e\alpha x_0 & 0 \end{array} \right]\!\!\right].$$

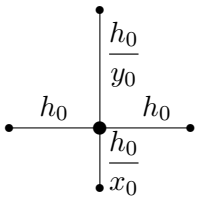


Figure 4.2: The neighbour nodes from the grid corresponding to the points a_k

Property 4.2.2. *[56] The values $b_k(\zeta), k = 1, 2, ..., n_0(n_0+1)$ depend on their horizontal distance ζ from the old grid and are computed solving a system with the matrix B having the form (4.4), but in which*

$$C = \begin{bmatrix} q_c & q_u & 0 & ... & 0 \\ q_d & q_c & q_u & ... & 0 \\ 0 & q_d & q_c & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & q_c \end{bmatrix}, D = \begin{bmatrix} q_r & 0 & 0 & ... & 0 \\ 0 & q_r & 0 & ... & 0 \\ 0 & 0 & q_r & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & q_r \end{bmatrix}, S = \begin{bmatrix} q_l & 0 & 0 & ... & 0 \\ 0 & q_l & 0 & ... & 0 \\ 0 & 0 & q_l & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & q_l \end{bmatrix},$$

where

$$\left[\!\!\left[\begin{array}{ccc} 0 & q_u & 0 \\ q_l & q_c & q_r \\ 0 & q_d & 0 \end{array} \right]\!\!\right] = \left[\!\!\left[\begin{array}{ccc} 0 & -e & 0 \\ -e\alpha x_0 + \alpha h_0 \rho\delta & e[2+\overline{\alpha}(x_0+y_0)] + \alpha\gamma\rho & e\alpha y_0 + \alpha h_0\rho\beta \\ 0 & -e & 0 \end{array} \right]\!\!\right].$$

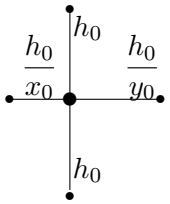


Figure 4.3: The neighbour nodes from the grid corresponding to the points b_k

4.2.3 Discretization with finite element method

Property 4.2.3. *[56] If the discretization of the differential equation (4.1) is made with the finite element method, using the notation*

$$\left[\!\!\left[\begin{array}{ccc} l_1 & l_2 & l_3 \\ l_4 & l_5 & l_6 \\ l_7 & l_8 & l_9 \end{array} \right]\!\!\right] = \left[\!\!\left[\begin{array}{ccc} k_{24}^D & k_{23}^D + k_{14}^C & k_{13}^C \\ k_{34}^A + k_{21}^D & k_{33}^A + k_{44}^B + k_{11}^C + k_{22}^D & k_{43}^B + k_{12}^C \\ k_{31}^A & k_{32}^A + k_{41}^B & k_{42}^B \end{array} \right]\!\!\right],$$

the matrix A in system (4.2) will have the following components

$$C = \begin{bmatrix} l_5 & l_6 & 0 & ... & 0 \\ l_4 & l_5 & l_6 & ... & 0 \\ 0 & l_4 & l_5 & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & l_5 \end{bmatrix}, D = \begin{bmatrix} l_2 & l_3 & 0 & ... & 0 \\ l_1 & l_2 & l_3 & ... & 0 \\ 0 & l_1 & l_2 & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & l_2 \end{bmatrix}, S = \begin{bmatrix} l_8 & l_9 & 0 & ... & 0 \\ l_7 & l_8 & l_9 & ... & 0 \\ 0 & l_7 & l_8 & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & l_8 \end{bmatrix} \quad (4.7)$$

and the right hand side term T has the elements

$$t_{in_0+j}(\zeta) = f_3^A + f_4^B + f_1^C + f_2^D - \left[\!\!\left[\begin{array}{ccc} l_1 & l_2 & l_3 \\ l_4 & l_5 & l_6 \\ l_7 & l_8 & l_9 \end{array} \right]\!\!\right] u_{fr}\left(jh_0, ih_0 + \zeta\right), \quad (4.8)$$
$$i = 0, ..., n_0, j = 1, ..., n_0.$$

Property 4.2.4. *[56] The matrix B of the system (4.3) has*

$$C = \begin{bmatrix} l_5 & l_2 & 0 & ... & 0 \\ l_8 & l_5 & l_2 & ... & 0 \\ 0 & l_8 & l_5 & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & l_5 \end{bmatrix} \in M_{n_0,n_0}(\mathbb{R}), D = \begin{bmatrix} l_6 & l_3 & 0 & ... & 0 \\ l_9 & l_6 & l_3 & ... & 0 \\ 0 & l_9 & l_6 & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & l_6 \end{bmatrix} \in M_{n_0,n_0}(\mathbb{R}),$$
$$S = \begin{bmatrix} l_4 & l_1 & 0 & ... & 0 \\ l_7 & l_4 & l_1 & ... & 0 \\ 0 & l_7 & l_4 & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & l_4 \end{bmatrix} \in M_{n_0,n_0}(\mathbb{R}) \quad (4.9)$$

and in the system (4.3), the matrix T from the right hand side, obtained with finite element method, has

$$t_{in_0+j}(\zeta) = f_3^A + f_4^B + f_1^C + f_2^D - \left[\!\!\left[\begin{array}{ccc} l_1 & l_2 & l_3 \\ l_4 & l_5 & l_6 \\ l_7 & l_8 & l_9 \end{array} \right]\!\!\right] u_{fr}\left(ih_0 + \zeta, jh_0\right),$$
$$i = 0, ..., n_0, j = 1, ..., n_0.$$

4.3 The stellar prolongation method for Robin conditions on the borders

On Γ_1 and Γ_2 (**Figure 4.4**) the boundary conditions for equation (4.1) are

$$e\frac{\partial u}{\partial n} + \xi u = g, (x, y) \in \Gamma_1 \cup \Gamma_2 \quad (4.10)$$

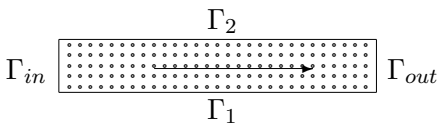


Figure 4.4: The boundaries of the domain for the Robin conditions problem

e is the diffusion coefficient, ξ the permeability of the walls Γ_1, Γ_2 and

$$\frac{\partial}{\partial n} = \frac{\partial}{\partial x}\boldsymbol{n_x} + \frac{\partial}{\partial y}\boldsymbol{n_y}$$

is the derivative along the direction normal to the frontier.

Property 4.3.1. *If Robin conditions are used, the values of the unknown u on the boundaries Γ_1 and Γ_2 are*

$$\begin{aligned} u_{i,1} &= pu_{i,2} + qg_{i,1}, \\ u_{i,n_l+2} &= pu_{i,n_l+1} + qg_{i,n_l+2}, i = 1, 2, ..., n_l, \end{aligned} \tag{4.11}$$

where

$$p = \frac{e}{e + \xi h_y}, q = \frac{h_y}{e + \xi h_y}.$$

Property 4.3.2. *The matrix of the system (4.2), when Robin conditions (4.10) are used, is*

$$A = \begin{bmatrix} C_1 & D & \Theta & ... & \Theta & \Theta \\ S & C & D & ... & \Theta & \Theta \\ \Theta & S & C & ... & \Theta & \Theta \\ \vdots & & & \ddots & & \\ \Theta & \Theta & \Theta & ... & S & C_2 \end{bmatrix} \in M_{(n_0+1)n_0,(n_0+1)n_0}(\mathbb{R}). \tag{4.12}$$

This matrix has the components $C, D, S \in M_{n_0,n_0}(\mathbb{R})$ defined by (4.7) and

$$C_1 = \begin{bmatrix} l_5 + pl_8 & l_6 + pl_9 & 0 & ... & 0 \\ l_4 + pl_7 & l_5 + pl_8 & l_6 + pl_9 & ... & 0 \\ 0 & l_4 + pl_7 & l_5 + pl_8 & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & l_5 + pl_8 \end{bmatrix} \in M_{n_0,n_0}(\mathbb{R}),$$

$$C_2 = \begin{bmatrix} l_5 + pl_2 & l_6 + pl_3 & 0 & ... & 0 \\ l_4 + pl_1 & l_5 + pl_2 & l_6 + pl_3 & ... & 0 \\ 0 & l_4 + pl_1 & l_5 + pl_2 & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & l_5 + pl_2 \end{bmatrix} \in M_{n_0,n_0}(\mathbb{R}).$$

The components of the right hand side term in system (4.2) are

$$t_{in_0+j}(\zeta) = f_3^A + f_4^B + f_1^C + f_2^D - \left[\!\left[\begin{array}{ccc} l_1 & l_2 & l_3 \\ l_4 & l_5 & l_6 \\ l_7 & l_8 & l_9 \end{array} \right]\!\right] \big(u_{fr}(jh_0, ih_0+\zeta) + q \cdot g(jh_0, ih_0+\zeta)\big),$$
$$i = 0, ..., n_0, j = 1, ..., n_0.$$

Here, u_{fr} is a function equal to zero inside Ω, Γ_1 and Γ_2, and with the boundary values on $\Gamma_{in} \cup \Gamma_{out}$, h_0 is the grid step on the level l_0.

Property 4.3.3. *The matrix B from system (4.3) is*

$$B = \begin{bmatrix} C & D & \Theta & ... & \Theta & \Theta \\ S & C & D & ... & \Theta & \Theta \\ \Theta & S & C & ... & \Theta & \Theta \\ \vdots & & & \ddots & & \\ \vdots & & & & \ddots & \\ \Theta & \Theta & \Theta & ... & S & C \end{bmatrix} \in M_{n_0,n_0}(\mathbb{R}),$$

where

$$C = \begin{bmatrix} l_5 + pl_8 & l_2 & 0 & ... & 0 \\ l_8 & l_5 & l_2 & ... & 0 \\ 0 & l_8 & l_5 & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & l_5 + pl_2 \end{bmatrix}, D = \begin{bmatrix} l_6 + pl_9 & l_3 & 0 & ... & 0 \\ l_9 & l_6 & l_3 & ... & 0 \\ 0 & l_9 & l_6 & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & l_6 + pl_3 \end{bmatrix} \in M_{n_0,n_0}(\mathbb{R}), \tag{4.13}$$

$$S = \begin{bmatrix} l_4 + pl_7 & l_1 & 0 & ... & 0 \\ l_7 & l_4 & l_1 & ... & 0 \\ 0 & l_7 & l_4 & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & l_4 + pl_1 \end{bmatrix} \in M_{n_0,n_0}(\mathbb{R}). \tag{4.14}$$

The right hand side term in system (4.3) has the elements

$$t_{in_0+j}(\zeta) = f_3^A + f_4^B + f_1^C + f_2^D - \left[\!\left[\begin{array}{ccc} l_1 & l_2 & l_3 \\ l_4 & l_5 & l_6 \\ l_7 & l_8 & l_9 \end{array} \right]\!\right] \big(u_{fr}\left(ih_0+\zeta, jh_0\right) + q \cdot g\left(ih_0+\zeta, jh_0\right)\big),$$
$$i = 0, ..., n_0, j = 1, ..., n_0.$$

4.4 Applying the stellar prolongation method

4.4.1 The comparison of the numerical results obtained with the multigrid method with those obtained with the stellar prolongation method

For the convection-diffusion problem (4.1), in table 4.1 are given some results for different values of the coefficients of the equation. It is well known that for this

type of problems (when the convection is dominant), the computing of a numerical solution is quite difficult (for example multigrid method is slowly convergent or even divergent) as the ratio between the diffusion and the convection coefficients is increasing. The results are obtained with: multigrid method (MG), stellar prolongation method (SP), with the finite elements method for discretization (SPFEM), stellar prolongation method with second order finite differences, and arithmetic mean prolongation (SPDF2), stellar prolongation with finite elements (SPFEM), and multigrid method for the solutin on subdomains (SPMGM).

1	**SPFEM**	**MG**	**SPDF2**	**SPMGM**
l	a=20	e=10		
3	$7.0338 \cdot 10^{-5}$	$1.1430 \cdot 10^{-5}$	0.0013	$7.0338 \cdot 10^{-5}$
4	$7.0338 \cdot 10^{-5}$	$2.8699 \cdot 10^{-6}$	0.0013	$7.0338 \cdot 10^{-5}$
5	$7.0338 \cdot 10^{-5}$	$7.1829 \cdot 10^{-7}$	0.0013	$1.8970 \cdot 10^{-4}$

2	**SPFEM**	**MG**	**SPDF2**	**SPMGM**
l	a=1/10	e=1		
3	$4.0852 \cdot 10^{-4}$	$2.1411 \cdot 10^{-5}$	0.0023	$4.0852 \cdot 10^{-4}$
4	$4.0852 \cdot 10^{-4}$	$5.3661 \cdot 10^{-6}$	0.0023	$4.0852 \cdot 10^{-4}$
5	$4.0852 \cdot 10^{-4}$	$1.3436 \cdot 10^{-6}$	0.0023	$4.8486 \cdot 10^{-4}$

3	**SPFEM**	**MG**	**SPDF2**	**SPMGM**
l	a=1/10	e=1/1000	ue $\sim 1.69 \cdot 10^{+7}$	
3	517.2537	$4.0200 \cdot 10^{+25}$	$3.1612 \cdot 10^{+4}$	418.5255
4	344.3122	$5.6336 \cdot 10^{+24}$	$3.0705 \cdot 10^{+4}$	465.1508
5	302.1993	$1.7728 \cdot 10^{+23}$	$3.0944 \cdot 10^{+4}$	$2.2814 \cdot 10^{+3}$

Table 4.1: The errors obtained with different methods for the problem 4.1

In the following this method is used for some convection-diffusion problems.

Problem 4.4.1.

$$\begin{cases} -\Delta u + \boldsymbol{\alpha} \bigtriangledown u = 0, & (x,y) \in (0,1) \times (0,1) = \Omega, \\ u = u_{ex}, & (x,y) \in \partial\Omega, \end{cases} , \boldsymbol{\alpha} = (c,d), \qquad u_{ex} = \frac{e^{cx}-1}{e^c-1}\frac{e^{dy}-1}{e^d-1}.$$

The results obtained after applying the multigrid method (MG), or the stellar prolongation with finite element method (SPFEM) for the problem presented before are given in table 4.2.

Remark 4.4.1. *The numerical data obtained show that stellar prolongation method can be more efficient than the multigrid method for convection-diffusion problems when the convection is dominant (the ratio R between the diffusion and the convection coefficients is bigger than 1).*

Problem 4.4.1	**MG**	**SPFEM**	**MG**	**SPFEM**
	$l = 3$		$l = 4$	
R=0.001	$9.7034 \cdot 10^{-5}$	$3.9636 \cdot 10^{-11}$	$1.2372 \cdot 10^{-4}$	$3.9636 \cdot 10^{-11}$
R=1	$6.1850 \cdot 10^{-5}$	$8.3193 \cdot 10^{-5}$	$8.6583 \cdot 10^{-5}$	$8.3193 \cdot 10^{-5}$
R=10	0.0042	0.0226	0.0010	0.0226
R=100	$1.9026 \cdot 10^{+39}$	0.7337	$4.4842 \cdot 10^{+39}$	0.3725
R=200	$1.0186 \cdot 10^{+59}$	1.8342	$7.7763 \cdot 10^{+66}$	0.8146

Table 4.2: The error obtained using the multigrid and stellar prolongation methods for the problem 4.4.1

4.5 Aplications of stellar prolongation method in practical problems

In this section we present two examples of the stellar prollongation method used for a convection-diffusion problem in a bidimensional domain, that models the blood flow in a vessel, then a diffusion problem in an irregular domain.

4.5.1 The stellar prolongation method for a stationary convection-diffusion problem

4.5.2 Model problem

For the study of the flow of a solvent in a blood vessel we use the following equation with partial derivatives

$$\begin{cases} m\Delta c + \boldsymbol{n} \bigtriangledown c + \alpha c = f, & (x,y) \in \Omega, \\ c = g_1, & (x,y) \in \Gamma_{in}, \\ c = g_2, & (x,y) \in \Gamma_{out}, \\ c = \xi g_3, & (x,y) \in \Gamma, \end{cases} \quad \boldsymbol{n} = (n_1, n_2). \tag{4.15}$$

Here $c = c(x,y)$ is the concentration of the solvent in the blood, the first term $m\Delta c$, in equation (2.6) describes the diffusion, the second term, $\boldsymbol{n} \bigtriangledown c$, describes the convection and is formed of two other terms, one along the *Ox* direction, $n_1 \dfrac{\partial c}{\partial x}$ and one along *Oy*, $n_2 \dfrac{\partial c}{\partial y}$; m is the diffusion coefficient, $\boldsymbol{n}$ is a field of speeds, α is the reaction coefficient, f is a term of possible perturbations of the concentration, due for example to chemical reactions. The ratio between the coefficients m and $\boldsymbol{n}$ determines if in the physical process the convection or the diffusion is dominant. The convection is dominant if $\dfrac{|\boldsymbol{n}|}{m} \gg 1$.

Finding the numerical solution for the convection - diffusion problem (4.15) becomes increasingly more difficult as this ratio becomes larger than one (meaning that the convection is dominant in the process), and this is the case studied for model problem (4.15) (the ratio is 10^4).

4.5.3 Numerical results

For the convection-diffusion problem (4.15), the value of the concentration on the domain, determined with the stellar prolongation method, is presented in **Figure 4.5**.

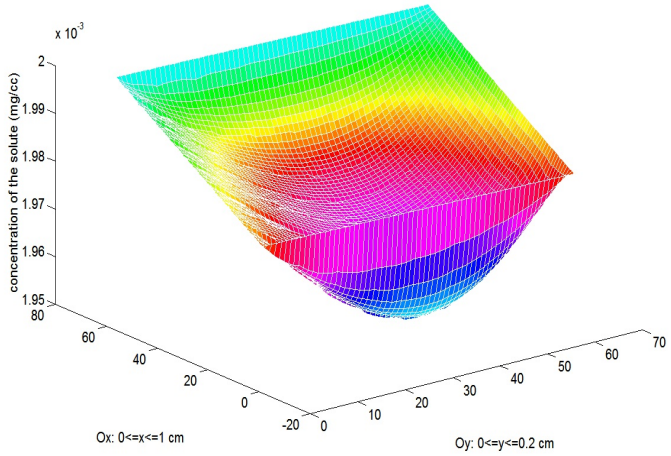


Figure 4.5: The concentration of the solvent on the domain [0;1]×[0;0,2]

4.5.4 The model of a fluid transport through a nonregular domain

The domain from **Figure 4.6** models a blood vessel that has toroidal plaque on the walls.

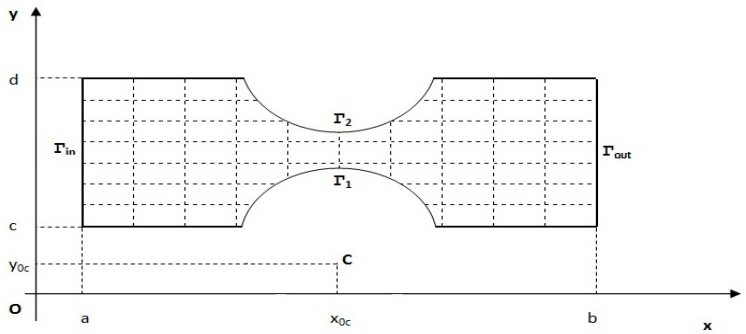


Figure 4.6: The domain Ω

The equation that models the diffusion process in a domain as in **Figure 4.6** is:

$$\begin{cases} -D_1\dfrac{\partial^2 u}{\partial x^2} - D_2\dfrac{\partial^2 u}{\partial y^2} = f, & x, y \in \Omega; \\ u = x^2 + y^2, & x, y \in \partial\Omega. \end{cases}$$

The diffusion coefficients used have been $D_1 = D_2 = 1$ and $f = -4$, in which case the exact solution is $u_{ex} = x^2 + y^2$. For Ω, the center of the circle is $C(x_{0c}, y_{0c})$ with $x_{0c} = 0.5, y_{0c} = -0.2, R = 0.25$ and $a = 0, b = 1, c = 0, d = 0.2$.

For the discretization we used finite differences method of second order, in the interior points of the domain the equation being

$$-D_1\frac{u_{i+1,j}-2u_{i,j}+u_{i-1,j}}{h_x^2}-D_2\frac{u_{i,j+1}-2u_{i,j}+u_{i,j-1}}{h_y^2}=f, \tag{4.16}$$

or

$$\left[\!\!\left[\begin{array}{ccc} 0 & -\frac{D_2}{h_y^2} & 0 \\ -\frac{D_1}{h_x^2} & 2\left(\frac{D_1}{h_x^2}+\frac{D_2}{h_y^2}\right) & -\frac{D_1}{h_x^2} \\ 0 & -\frac{D_2}{h_y^2} & 0 \end{array}\right]\!\!\right] u_l(x_i,y_j)=f_l(x_i,y_j),(x_i,y_j)\in\Omega_l. \tag{4.17}$$

For the points $(x_i,y_j)\in\Omega_l$ near the irregular borders, we used the Shortley-Weller scheme [34], [86]:

$$2\left[\!\!\left[\begin{array}{ccc} 0 & -\frac{D_2}{h_N(h_N+h_S)} & 0 \\ -\frac{D_1}{h_V(h_E+h_V)} & \frac{D_1}{h_Eh_V}+\frac{D_2}{h_Eh_V} & -\frac{D_1}{h_E(h_E+h_V)} \\ 0 & -\frac{D_2}{h_S(h_N+h_S)} & 0 \end{array}\right]\!\!\right] u_l(x_i,y_j)=f_l(x_i,y_j). \tag{4.18}$$

Working with these data, for the known exact solution, we applied stellar prolongation method, with the first grid on the level $l_0=2$ and the finest grid corresponding to level $l=5$, the error $u-u_{ex}$ on the domain Ω is shown in **Figure 4.7**.

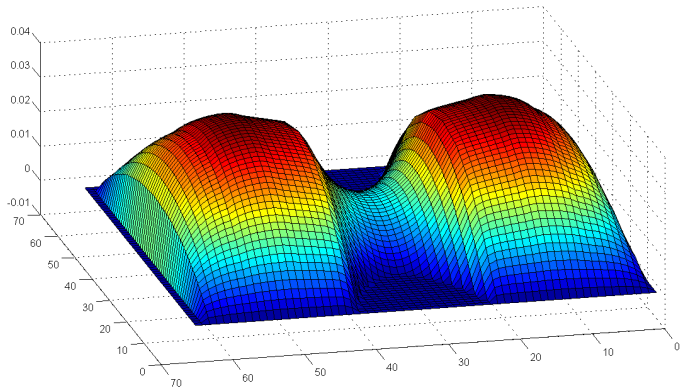


Figure 4.7: The error on the grid corresponding to $l=5$ and $l_0=2$

Bibliography

[1] R. E. BANK, M. BENBOURENANE, *The Hierarchical Basis Multigrid Method for Convection-Diffusion Equations*, Numer. Math., 61 (1992): 7-37.

[2] P. BASTIAN, G. WITTUM, *On Robust and Adaptive Multigrid Methods,* Multigrid Methods IV, Proceedings of the fourth European Multigrid Conference, Birkhäuser, ISNM International Series of Numerical Mathematics, 116, pp. 1-17, Amsterdam, 1994.

[3] S. R. BHATTARAI, R. BAHADUR, S. ARYAL, N. BHATTARAI, S. Y. KIM, H. K. YI, P. H. HWANG, H. Y. KIM, *Hydrophobically Modified Chitosan/Gold Nanoparticles for DNA Delivery,* J. Nanopart. Res. 10 (2008): 151-162.

[4] E.B. BECKER, G.F. CAREY, J.T. ODEN, *Finite Elements. An Introduction*, Texas Institute for Computational Mecanics, UT Austin, 1981.

[5] S. BOCA, M. POTARA, F. TODERAŞ, O. STEPHAN, P. L. BALDECK, S. AŞTILEAN, *Uptake and Biological Effects of Chitosan-capped Gold Nanoparticles on Chinese Hamster Ovary Cells,* Materials Science and Engineering C31 (2011): 184-189.

[6] D. BRAESS, *The Contraction Number of a Multigrid Method for Solving the Poisson Equation*, Numer. Math. 37 (1981): 387-404.

[7] J. H. BRAMBLE, *Multigrid Methods*, Pitman Research Notes in Mathematics, V 294, John Wiley and Sons, 1993.

[8] A. BRANDT, *Multi-level adaptive technique (MLAT) for fast numerical solution to boundary value problems*, Proceedings of the 3rd International Conference on Numerical Methods in Fluid Mechanics, Lecture Notes in Physics, 18 (eds H. Cabannes and R. Temam), pp. 82-89, Springer, Berlin, 1973.

[9] A. BRANDT, *Multi-level adaptive techniques (MLAT). I. The multigrid method*, Research Rep. RC 6026, IBM T.J. Watson Research Center, Yorktown Heights, NY, 1976.

[10] A. BRANDT, Multilevel Adaptive Solutions to Boundary Value Problems, *Mathematics of Computation*, 31 (1977): 333-390.

[11] A. BRANDT, *Multigrid Techniques: 1984 Guide with Applications to Fluid Dynamics*, GMD-Studien Nr. 85, Gesellschaft für Matematik und Datenverarbeitung, St. Augustin, Bonn, 1984.

[12] A. BRANDT, *Guide to Multigrid Development with Applications to Computational Fluid Dynamics*, Lecture Notes in Mathematics, Springer, 1984

[13] W. L. BRIGGS, H. VAN EMDEN, S. MCCORMICK, *A Multigrid Tutorial, Second Edition*, SIAM, Philadelphia, PA, 2000.

[14] W. L. BRIGGS, H. VAN EMDEN, S. MCCORMICK, *A Multigrid Tutorial, Second Edition, part II*, SIAM, Philadelphia, PA, 2000.

[15] BORZI, A. *Introduction to Multigrid Methods*, Lecture Notes in Mathematics, Oxford University Computing Laboratory, 1994.

[16] X. L. CAO, C. CHENG, Y. L. MA, C. S. ZHAO, *Preparation of Silver Nanoparticles with Antimicrobial Activities and the Researches of their Biocompatibilities*, J Mater Sci: Mater Med, 21 (2010): 2861-2868.

[17] D. M. CAUSON, C. G MINGHAM, *Introductory Finite Differences Method*, `http://www.cs.man.ac.uk/~fumie/tmp/introductory-finite-difference-methods-for-pdes.pdf`

[18] T.F. CHAN, W. L. WAN, *Robust Multigrid Methods for Elliptic Linear Systems*, J. Comput. Appl. Math., 123 (2000): 323–352.

[19] L. CHEN, J. XU, *An optimal streamline diffusion finite element method for a singular perturbed problems*, AMS Contemporary Mathematics Series: Recent Advances in Adaptive Computation, 383 (2005): 236–246.

[20] I. CHIOREAN, T. CĂTINAŞ, R. TRÎMBIŢAŞ, *Analiză numerică*, Presa Universitară Clujeană , Cluj Napoca, 2010.

[21] G. COMAN, *Analiză numerică*, Libris, Cluj Napoca, 1995.

[22] G. COMAN, I. CHIOREAN, T. CĂTINAŞ, *Numerical Analysis- An Advanced Course*, Presa Universitară Clujeană, Cluj Napoca, 2007.

[23] E. E. Connor, J. Mwamuka, A. Gole, C. J. Murphy, M.D. Wyatt, *Gold nanoparticles are taken up by human cells but do not cause acute cytotoxicity*, Small, 1, Issue 3 (2005): 325-7.

[24] M. CRIŞAN, C. CATTANI, R. BADEA, P. MITREA, M. FLOREA, D. CRIŞAN, D. MITREA, R. BUCUR, G. CHECICHES, *Modelling Cutaneous Senescence Process*, D. Taniar et al. (Eds.): ICCSA 2010, Part II, LNCS 6017, 215–224, Springer-Verlag Berlin, Heidelberg, 2010.

[25] M. CRIŞAN, C. CATTANI, R. BADEA, R. COSGAREA, S. DUDEA, P. MITREA, M.LUPŞOR, *Complex histological, genetical, ultrasonography and clinical studies in early noninvasive diagnosis of the photoinduced cutaneous senescence and in the photoinduced skin cancers, using computerized imaging, modern biotechnology and mathematical modelling methods*, Automat. Comp. Appl. Math., 18 (2009), No.2 : 231-255,

[26] J.E. DENDRY JR., J.D. MOULTON, *Black Box Multigrid with a Coarsening Factor of Three*, Numerical Linear Algebra With Applications, 17 (2010): 577-598.

[27] J. EBERHARD, G. WITTUM, *A Coarsening Multigrid Method for Flow in Heterogenous Porous Media*, Multiscale Methods in Science and Engineering, Lecture Notes in Computational Science and Engineering, 44 (2005): 113-134.

[28] H.C. ELMAN, D.J. SILVESTER, A.J. WATHEN, *Finite Elements and Fast Iterative Solvers*, Oxford University Press, Oxford, 2005.

[29] H. C. ELMAN, A. RAMAGE, *Fourier Analysis of Multigrid for a Model Two-Dimensional Convection-Diffusion Equation,* `http://www.cs.umd.edu/~elman/papers/elman-ramage-mg.pdf`

[30] R. P. FEDORENKO, *A relaxation method for solving elliptic difference equations*, USSR Comput. Math. Math. Phys. 1 (1962): 1092–1096.

[31] R. P. FEDORENKO, *The speed of convergence of one iterative process*, USSR Comput. Math. Math. Phys., 4 (1964): 227–235.

[32] L. FORMAGGIA, S. PEROTTO, P. ZUNINO, *An anisotropic a-posteriori error estimate for a convection-diffusion problem,* Comput. Visual. Sci., 4 (2001): 99–104.

[33] M. M. GUPTA, J. ZHANG, *High Accuracy Multigrid Solution of the 3D convection-diffusion equation* Applied Mathematics and Computation, 113 (2000): 249-274.

[34] W. HACKBUSH, *Multigrid Method and Applications*, Springer, Berlin, Heidelberg, 1985.

[35] W. HACKBUSH, *Elliptic Differential Equations*, Springer-Verlag, New York, 1992.

[36] P. HANSBO, *The characteristic streamline diffusion method for the time-dependent incompressible Navier–Stokes equations*, Comput. Methods Appl. Mech. Eng. 99 (1992): 171–186.

[37] S. H. HSU,Y. B. CHANG, C. L. TSAI, K. Y. FU, S. H. WANG, H. J. TSENG, *Characterization and Biocompatibility of Chitosan Nanocomposites*, Colloids and Surfaces B: Biointerfaces, 85 (2011): 198-206.

[38] T. HUGHES, A. BROOKS, *A multidimensional upwind scheme with no crosswind diffusion,* Finite element methods for convection dominated flows, ASME, New York, 1979.

[39] C. JOHNSON, U. NAVERT, *An analysis of some finite element methods for advection–diffusion problems*, Analytical and Numerical Approaches to Asymptotic Problems in Analysis, North-Holland Publ., Amsterdam, 1981.

[40] C. JOHNSON, *Finite element methods for convection–diffusion problems*, Computing Methods in Engineering and Applied Sciences, North-Holland Publ., Amsterdam, 1981.

[41] C. JOHNSON, *The characteristic streamline diffusion finite element method*, Math. Appl. Comput. 10 (1991): 229–242.

[42] M. KARALASHVILI, S. GROSS, W. MARQUARDT, A. MHAMDI, A. REUSKEN, *Identification of transport coefficient models in convection-diffusion equations*, SIAM Journal on Scientific Computing, vol. 33, no. 1(2011): 303-327.

[43] U. LANGER, D. PUSCH, *Convergence Analysis of Geometrical Multigrid Methods for Solving Data Sparse Boundary Element Equations*, Comput. Visual Sci., 11 (2008): 181-189.

[44] R. W. LEWIS, P. NITHIARASU, K. N. SEETHARAMU, *Fundamentals of the Finite Element Method for Heat and Fluid Flow*, John Wiley & Sons Ltd, Chichester, 2004.

[45] J. MANDEL, M. BREZINA, P. VANEK, *Energy Optimization of Algebraic Multigrid Bases*, Computing 62 (1999): 205-228, Springer-Verlag.

[46] S. MCCORMICK, *Multigrid Methods*, SIAM, Philadelphia, Pennsylvania, 1987.

[47] U. MEYER YANG, *On long- Range Interpolation Operators for Aggressive Coarsening*, Numerical Linear Algebra With Applications 17 (2010): 453-472.

[48] W. F. MITCHELL, *The hp-multigrid method applied to hp-adaptive refinement of triangular grids* Numer. Linear Algebra Appl. 17 (2010): 211–228.

[49] D. MORIN, *Fourier analysis*, `http://www.people.fas.harvard.edu/~djmorin/waves/Fourier.pdf`

[50] K.W. MORTON, D.F. MAYERS, *Numerical Solution of Partial Differential Equations, An Introduction*, Cambridge University Press, 2005.

[51] A.C. MUREŞAN, Y. NOTAY, *Analysis of aggregation-based multigrid*, SIAM Journal on Scientific Computing, 30 (2008): 1082–1103.

[52] A. NAPOV, Y. NOTAY, *When does two-grid optimality carry over to the V-cycle?* Numerical Linear Algebra With Applications, 17 (2010): 273-290.

[53] A. NAPOV, Y. NOTAY, *Smoothing factor, order of prolongation and actual multigrid convergence,* Numer. Math. 118 (2011): 457–483.

[54] C. R. NĂSTASE, D. J. MAVRIPLIS *Discontinuous Galerkin Methods Using an hp-Multigrid Solver for Inviscid Compressible Flows on Three-dimensional Unstructured Meshes,* 44th AIAA Aerospace Sciences Meeting and Exhibit, January 9–12, 2006, Reno, NV.

[55] D. NEUMANN, *Modeling Transdermal Absorption*, Biotechnology: Pharmaceutical Aspects, Vol. VII, Springer NY, 2008.

[56] G. NUŢ, *A Prolongation Technique for Solving Partial Differential Equations with a Multigrid Method,* Proceedings of Modelling and Development of Intelligent Systems Conference, Lucian Blaga University Press, pp. 105-115, Sibiu, 2011.

[57] G. NUŢ, I. CHIOREAN, M.CRIŞAN, *The Finite Element Method Applied to a Problem of Blood Flow in Vessels,* Computational Mathematical Methods in Medicine, doi:101155/2012/204926, 2012.

[58] G. NUŢ, I. CHIOREAN, P. BLAGA, *Convergence and Error of Some Numerical Methods for Solving a Convection-Diffusion Problem*, Applied Mathematics, Vol. 4 No. 5A (2013): 72-79. doi: 10.4236/am.2013.45A009.

[59] G. NUŢ, *A Two-dimensional Convergence and Error Study for the Time Dependent Convection-diffusion Equation,* Proceedings of 13th International Conference Computational and Mathematical Methods in Science and Engineering, Vol. IV, pp. 1097-1108, Almeria, Spain, 2013.

[60] G. NUŢ, *Mathematical Modelling of the Skin Penetration by Nanoparticles Carried Drugs*, Annals of the Tiberiu Popoviciu Seminar of Functional Equations, Approximation and Convexity, 11 (2013): 97-111.

[61] G. NUŢ, I. CHIOREAN, *The Frequency Spectrum for the Local Fourier Analysis of Multigrid Method in the One-dimensional Case,* Proceedings of the " European Integration- Between Tradition and Modernity"- Conference, Târgu Mureş, 2013 (to appear).

[62] M. OLSHANSKII, A. REUSKEN, *On Robust Multigrid Method for Convection-Diffusion Finite Element Problems*, `http://www.mathcs.emory.edu/~molshan/ftp/pub/proceed_cd.pdf`

[63] M. OLSHANSKII, A. REUSKEN, *Convergence Analysis of a Multigrid method for a Convection- dominated Model Problem*, SIAM Journal on Numerical Analysis, Volume 42 Issue 3 (2004): 1261-1291.

[64] B. OSGOOD, *The Fourier Transform and its Applications,* `http://see.stanford.edu/materials/lsoftaee261/book-fall-07.pdf`

[65] A. PARDHANANI, W. SPOTZ, G. F. CAREY, *A Stable Multigrid Strategy for Convection-Diffusion Using High Order Compact Discretization*, Electronic Transactions on Numerical Analysis, 6 (1997): 211-223.

[66] T. M. PERTMER, M. D. EISENBRAUN, D. MCCABE, S. K. PRAYAGA, J. R. HAYNES, *Gene gun-based nucleic acid immunization: elicitation of humoral and cytotoxic T lymphocyte responses following epidermal delivery of nanogram quantities of DNA,* Vaccine 13 (2004): 1427–1430.

[67] C. PFLAUM, *A Multigrid Conjugate Gradient Method*, Applied Numerical Mahematics, 58 (2008): 1803-1817.

[68] B. PHIPPS, M. CORMIER, B. GALE, W. W. VAN OSDOL, *Transdermal drug delivery,* Encyclopedia of Biomaterials and Biomedical Engineering, 2004.

[69] R. O. POTTS, M. L. FRANCOEUR, *The influence of stratum-corneum morphology on water permeability,* J. Invest. Dermatol. 96 (1991): 495–499.

[70] L. QIAN, X. FENG, Y. HE, *The characteristic finite difference streamline diffusion method for convection-dominated diffusion problems,* Applied Mathematical Modelling 36 (2012): 561–572.

[71] B. AL-QALLAF, D. BHUSAN DAS, D. MORI, Z. CUI, *Modelling Transdermal Delivery of High Molecular Weight Drugs from Microneedle Systems*, Phil. Trans. R. Soc. A, 365 (2007): 2951-2967.

[72] A. RAMAGE, H. C. ELMAN, *Some observations on multigrid convergence for convection–diffusion equations,* Comput. Visual Sci., 10 (2007): 43–56.

[73] A. REUSKEN, *Convergence Analysis of a Multigrid Method for Convection-Diffusion Equations*, Numerische Mathematik, Vol.91, Issue 2 (2002): 323-349.

[74] J. E. RIM, P. M. PINSKY, W. W. V. OSDOL, *Multiscale Modeling Framework of Transdermal Drug Delivery*, Annals of Biomedical Engineering, Vol. 37, No. 6 (2009): 1217-1229.

[75] W. RUGE, K. STÜBEN, *Algebraic Multigrid.* In *Multigrid Method, Frontiers in Applied Mathematics*, SIAM, Philadelphia, 1987.

[76] M. SALA, R.S. TUMINARO, *A new Petrov–Galerkin smoothed aggregation preconditioner for nonsymmetric linear systems*, SIAM Journal on Scientific Computing 31 (2008): 143–166.

[77] B. SEIBOLD, *Performance of Algebraic Multigrid Methods for Non-symmetric Matrices Arising in Particle Methods*, Numer. Linear Algebra Appl., 17 (2010): 433-451.

[78] Y. SHAPIRA, *Matrix –Based Multigrid: Theory and Applications*, Second Edition, Springer, 2008.

[79] M. STOLNITZ, A. N. BASHKATOV, E. A. GENINA, V. TUCHIN, *Mathematical modeling of clearing liquid penetration into the skin,* Saratov Fall Meeting 2006: Optical Technologies in Biophysics and Medicine, Vol. 6535 (2007): 653520.

[80] K. STÜBEN, U. TROTTENBERG, *Multigrid Methods: Fundamental Algoritms, Model Problem Analysis and Applications*, GMD-Studien, 1989.

[81] K. STÜBEN, An introduction to algebraic multigrid, GMD Report 70, Fraunhofer SCAI, Sankt Augustin, Germany, 1999.

[82] M. STYNES, Convection-diffusion problems, SDFEM/SUPG and a priori meshes, *International Journal of Computing Science and Mathematics* 1, Issue 2-4 (2007): 412-431.

[83] A. THEKALE, T. GRADL, K. KLAMROTH, U. RUDE, *Optimizing the Number of Multigrid Cycles in the Full Multigrid Algoritm*, Numerical Linear Algebra With Applications, 17(2010): 199-210.

[84] D. TRIF, *Ecuaţii cu derivate parţiale*, UBB, Cluj, 1993.

[85] D. TRIF, *Metode numerice pentru ecuaţii diferenţiale şi sisteme dinamice*, Transilvania Press, Cluj, 1997.

[86] U. TROTTENBERG, C. OOSTERLEE, A. SCHULLER, *Multigrid*, Elsevier Academic Press, London, 2001.

[87] R. S. VARGA, *Matrix Iterative Analysis*, Springer- Verlag, Berlin, Heidelberg, 2000.

[88] D. VASILEVA, *Numerical Analysis and Its Applications on an Adaptive Semirefinement Multigrid Algorithm for Convection-Diffusion Problems*, Springer-Verlag Berlin, Heidelberg, 2009.

[89] P. WESSELING, *An Introduction to Multigrid Method*, John Wiley & Sons, New York, 1991.

[90] R. WIENANDS, W. JOPPICH, *Practical Fourier Analysis for multigrid methods*, Chapman & Hall/CRC Press, Boca Raton, Florida, 2005.

[91] D. V. M. YI- LO LIN, J. C. JEN, S.H. HSU, I.M. CHIU, *Sciatic Nerve Repair by Microgrooved Nerve Conduits Made of Chitosan-Gold Nanocomposites*, Surgical Neurology, 70, S1 (2008): 9-18.

[92] G. YONGBIN, *Multigrid method and fourth-order compact difference discretization scheme with unequal meshsizes for 3D poisson equation* Journal of Computational Physics, 229 (2010): 6381–6391.

[93] J. ZHANG, *Multigrid Method and Fourth Order Compact Difference Scheme for 2D Poisson Equation with Unequal Meshsize Discretization*, Journal of Computational Physics, 179 (2002): 170-179.