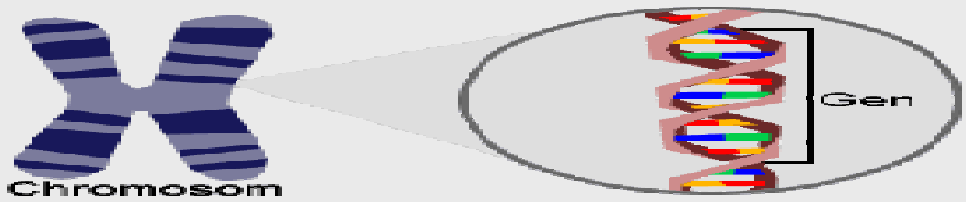


Artificial Intelligence for Solving PDEs (Partial Differential Equations)

Prof. NIKOS E. MASTORAKIS
Technical University of Sofia
Industrial Engineering Department
http://elfe.tu-sofia.bg/elfe/staff.htm ,
http://elfe.tu-sofia.bg/elfe/curriculum4.htm and
http://elfe.tu-sofia.bg/elfe/curriculum3.htm

http://www.wseas.org/mastorakis/CV_Mastorakis.doc

also, except the Technical University of Sofia, Bulgaria

Prof. NIKOS E. MASTORAKIS is
Honorary Professor, University of Cluj, ROMANIA,
http://outstanding.wseas.us

Honorary Professor, Budapest Tech, Budapest, HUNGARY,
http://www.youtube.com/watch?v=MsuEGbbFRz8

Prize of Excellence from Romanian Academy of Science, Bucharest, ROMANIA
http://outstanding.wseas.us

Honorary Diploma from Georgian National Academy of Sciences (GNAS), GEORGIA
http://www.science.org.ge/english.html

Professor at ASEI (Military Institutes of University Education), Hellenic Naval Academy, GREECE
http://www.hna.gr mastor@wseas.org

The first part of our lecture will focus on the application of Genetic Algorithms with Nelder-Mead Optimization for the Finite Elements Methods applied on Non-linear Problems.

Some examples will be given regarding problems in Fluid Mechanics

$$\Delta_p u := \operatorname{div}\left(|\nabla u|^{p-2} \nabla u \right)$$

ABSTRACT: The “p” Laplacian operator, or the “p” Laplace operator, is a quasilinear elliptic partial differential operator of second order. The “p” Laplacian equation is a generalization of the Partial Differential Equation of Laplace Equation and in this lecture, we present a way of its solution using Finite Elements. Our method of Finite Elements leads to an Optimization Problem that can be solved by an appropriate combination of Genetic Algorithms and Nelder-Mead . Our method is illustrated by a numerical example. Other methods for the solution of other equations that contain the “p” Laplacian operator are also discussed.

Many nonlinear problems in physics and mechanics are formulated in equations that contain the “p” Laplacian, where the p-Laplacian operator is defined as follows. See the next Equation:

$$\Delta_p u := \operatorname{div}\left(\left| \nabla u \right|^{p-2} \nabla u \right)$$

In a recent paper, Bognar presented a very interesting numerical and analytic investigation of problems of fluid mechanics that are described with PDEs containing the p-Laplacian operator. Previous publications include reaction-diffusion problems, non-Newtonian fluid flows , fluid flows through certain types of porous media , the Lane-Emden equations for equilibrium configurations of spherically symmetric gaseous stellar objects, singular solutions for the Emden-Fowler equation and the Einstein-Yang-Mills equation, the existence and nonexistence of black hole solutions, nonlinear elasticity, glaciology and petroleum extraction

It is clear that for

$$p = 2$$

we have

$$\Delta_p = \Delta \, .$$

The study of the "p"- Laplacian equation started more than thirty years ago. In recent years, rapid development has been achieved for the study of equation involving operator Delta "P" and a vast literature has appeared on the theory of quasilinear differential equations.). Recently Strikwerda summarized many Finite Difference Schemes for Partial Differential Equations

Also, Bognar had studied the equation of turbulent filtration in porous media with the following equation

$$\theta \frac{\partial \rho}{\partial t} = c^{\alpha} \lambda \operatorname{div}\left(\left| \nabla \rho^{n} \right|^{p-2} \nabla \rho^{n} \right), \qquad \textbf{(1)}$$

Where the constants satisfy these inequalities

$$\theta > 0 \quad n > 0 \quad p > 1 \quad np > 1$$

If we scale out the constants in the previous equation, we derive the Equation (2)

$$\frac{\partial u}{\partial t} = \Delta_{p}\left(u^{n}\right) \qquad \textbf{(2)}$$

Note that a particular case n=1 of (2) is the non-Newtonian filtration equation

$$\frac{\partial u}{\partial t} = \Delta_p u \qquad (3)$$

which is also called evolution p-Laplacian equation.

The case that $p > 1 + 1/n$ is called the slow diffusion a and the case $p < 1 + 1/n$ is called fast diffusion.

Also Bognar studied the equation

$$\frac{\partial u}{\partial t} = \operatorname{div}\left(\left|\nabla u\right|^{p-2} \nabla u\right) + \lambda u^{q},$$

Where p>1, q>0 and λ is some constant in which the nonlinear term

$$\lambda u^{q}$$

describes the nonlinear source in the diffusion process, called "heat source" if $\lambda > 0$ while it is called and "cold source" if $\lambda < 0$

Just as the Newtonian equation the appearance of nonlinear sources will exert a great influence to the properties of solutions and the influence of "heat source" and "cold source" is completely different.

In another Study, an attempt is made by the author to solve the equation (2), (1) and (4) using various numerical schemes.

In this lecture, we will solve the boundary value problem

$$\operatorname{div}\left(\left|\nabla u\right|^{p-2}\nabla u\right)=0$$

where u is known on the boundary of our domain using Variational Techniques of Finite elements.

- **The Problem is reduced to an Optimization problem that can be solved by Genetic Algorithms with Nelder-Mead.**

- **An early paper of the author with the title “Solving Differential Equations via Genetic Algorithms” was presented in a conference in 1996.**

- **Actually, the author presented in 1996 the solution of ODE and PDE using Genetic Algorithms optimization, while the author use the same method to solve various problems in some other publications**

Main Results:

Our main results are as follows

We start solving the boundary value problem of the following form

$$\operatorname{div}\left(\left|\nabla u\right|^{p-2}\nabla u\right)=0 \qquad \textbf{(4)}$$

where *u* is a known function on the boundary of our domain.

As one can see, the solution of this "p"-Laplacian equation with Dirichlet boundary conditions in a domain Ω is the minimiser of the energy functional of the following equation

$$J(u)=\int\left|\nabla u\right|^{p}dv \qquad \textbf{(5)}$$

We consider that *u* is written as a linear combination of our known basis' functions with unknown coefficients.

$$u = \sum_n \lambda_n f_n \qquad u = \sum_n f_n$$

So, we have the following minimization problem

$$\min \int \left| \nabla \left(\sum_n f_n \right) \right|^p dv$$

One can select a triangular mesh and appropriate functions

$$f_n = a_n x + b_n y + c_n$$

that have non-zero value only in the "*n-th*" triangle

- **These our finite elements**

So, in a triangular mesh,

we can have

$$f_n = a_n x + b_n y + c_n$$

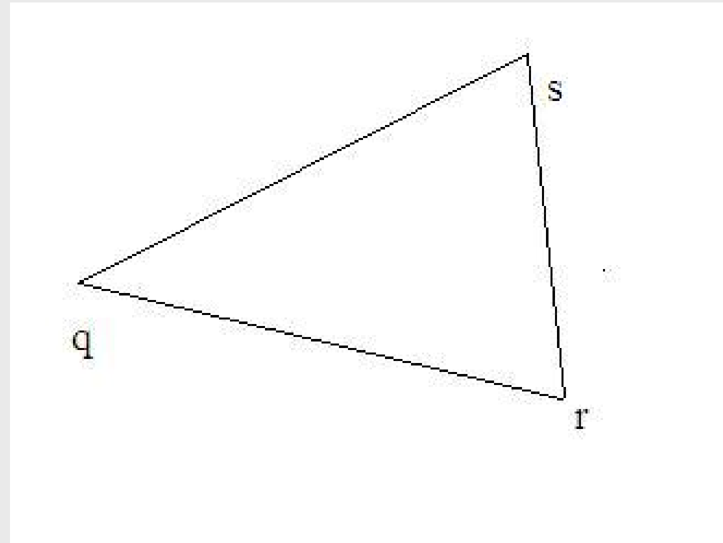


Fig.1 A triangle in a 2-D mesh

for the n-th triangle. To avoid to write continuity conditions on the common vertices of the triangles of the mesh, one can find that in the n-th triangle of the points “s”, “q” and “r” (see Figure 1)

$$u_s = a_n x_s + b_n y_s + c_n \qquad (7.1)$$

$$u_q = a_n x_q + b_n y_q + c_n \qquad (7.2)$$

$$u_r = a_n x_r + b_n y_r + c_n \qquad (7.3)$$

Actually the Figure 1 is a triangle in our 2-D mesh

And now we have the Equations (7.1), (7.2) and (7.3)

There three equations can be solved with respect to

$$a_n, b_n, c_n$$

as follows

$$a_n = \frac{\begin{vmatrix} u_s & y_s & 1 \\ u_q & y_q & 1 \\ u_r & y_r & 1 \end{vmatrix}}{D} \qquad b_n = \frac{\begin{vmatrix} x_s & u_s & 1 \\ x_q & u_q & 1 \\ x_r & u_r & 1 \end{vmatrix}}{D}$$

$$c_n = \frac{\begin{vmatrix} x_s & y_s & u_s \\ x_q & y_q & u_q \\ x_r & y_r & u_r \end{vmatrix}}{D}$$

These are the equation (8.1), (8.2) and (8.3)

These are the Equations (8.1), (8.2) and (8.3) where “D” is given as follows

$$D = \begin{vmatrix} x_s & y_s & 1 \\ x_q & y_q & 1 \\ x_r & y_r & 1 \end{vmatrix}$$

Note that two times D is the algebraic area of the triangle.

So, from the minimization problem that the following equation describes

$$\min \int \left| \nabla \left(\sum_n f_n \right) \right|^p dv$$

we find the equivalent minimization problem of the following equation

$$\min \int \left| \phi(u_n) \right|^p dv \qquad \textbf{(9)}$$

which is minimization of Us, Uq, Ur

Note that $\phi(u_n)$ **is the function that we find after replacing**

$$a_n = \frac{\begin{vmatrix} u_s & y_s & 1 \\ u_q & y_q & 1 \\ u_r & y_r & 1 \end{vmatrix}}{D} \qquad b_n = \frac{\begin{vmatrix} x_s & u_s & 1 \\ x_q & u_q & 1 \\ x_r & u_r & 1 \end{vmatrix}}{D} \qquad c_n = \frac{\begin{vmatrix} x_s & y_s & u_s \\ x_q & y_q & u_q \\ x_r & y_r & u_r \end{vmatrix}}{D}$$

The Equation (9) can be solved now by a variety of techniques.

- The author uses the Method of Genetic Algorithms with the so-called method of Nelder and Meade for Non-linear Problems.
- The same optimization scheme Genetic Algorithms with Nelder-Meade method has recently applied by the author with great success

Before proceeding in the solution of the problem, some background on "G-A" i.e. Genetic Algorithms and Nelder-Mead is necessary.

In many papers, we have proposed a hybrid method that includes

- **a) Genetic Algorithm for finding rather the neiborhood of the global minimum than the global minimum itself and**
- **b) Nelder-Mead algorithm to find the exact point of the global minimum itself.**

So, with this Hybrid method of Genetic Algorithm + Nelder-Mead we combine the advantages of both methods, that are

- **A) the convergence to the global minimum via the genetic algorithm**

plus

- **B) the high accuracy of the Nelder-Mead method.**

We emphasize here that

- **If we use only a Genetic Algorithm then we have the problem of low accuracy.**

- **If we use only Nelder-Mead, then we have the problem of the possible convergence to a local (not to the global) minimum.**

These disadvantages are removed in the case of our Hybrid method that combines Genetic Algorithm with Nelder-Mead method.

- We recall the following definitions from the Genetic Algorithms literature:

- **Fitness** function is the objective function we want to minimize.

- **Population size** specifies how many individuals there are in each generation. We can use various Fitness Scaling Options (rank, proportional, top, shift linear, etc), as well as various Selection Options (like Stochastic uniform, Remainder, Uniform, Roulette, Tournament).

Fitness Scaling Options: We can use scaling functions. A Scaling function specifies the function that performs the scaling. A scaling function converts raw fitness scores returned by the fitness function to values in a range that is suitable for the selection function.

We have the following options:

- **Rank Scaling Option:** scales the raw scores based on the rank of each individual, rather than its score. The rank of an individual is its position in the sorted scores. The rank of the fittest individual is 1, the next fittest is 2 and so on. Rank fitness scaling removes the effect of the spread of the raw scores.

and the options:

- **Proportional Scaling Option:** **The Proportional Scaling makes the expectation proportional to the raw fitness score. This strategy has weaknesses when raw scores are not in a "good" range.**

- **Top Scaling Option:** **The Top Scaling scales the individuals with the highest fitness values equally.**

and the options:

- **Shift linear Scaling Option:** The shift linear scaling option scales the raw scores so that the expectation of the fittest individual is equal to a constant, which you can specify as Maximum survival rate, multiplied by the average score.

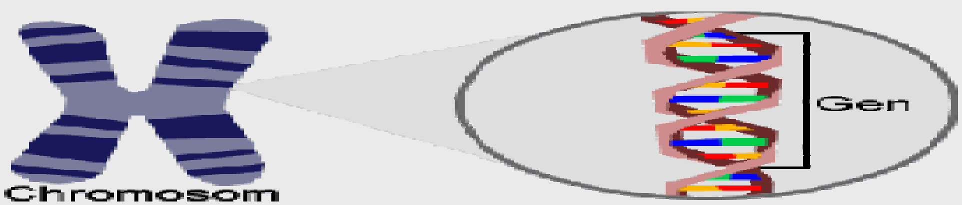
We can have also option in our Reproduction in order to determine how the genetic algorithm creates children at each new generation.

For example,

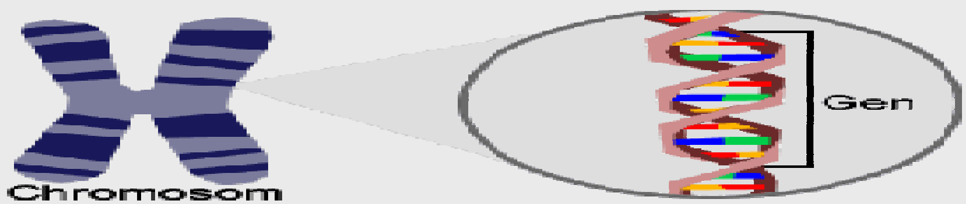
- **Elite Counter** specifies the number of individuals that are guaranteed to survive to the next generation.

- **Crossover** combines two individuals, or parents, to form a new individual, or child, for the next generation.

- **Crossover fraction** specifies the fraction of the next generation, other than elite individuals, that are produced by crossover.



- **Scattered Crossover:** Scattered Crossover creates a random binary vector. It then selects the genes where the vector is a 1 from the first parent, and the genes where the vector is a 0 from the second parent, and combines the genes to form the child.



- **Mutation:** **Mutation makes small random changes in the individuals in the population, which provide genetic diversity and enable the GA to search a broader space. Gaussian Mutation: We call that the Mutation is Gaussian if the Mutation adds a random number to each vector entry of an individual. This random number is taken from a Gaussian distribution centered on zero. The variance of this distribution can be controlled with two parameters. The Scale parameter determines the variance at the first generation. The Shrink parameter controls how variance shrinks as generations go by.** ***If the Shrink parameter is 0, the variance is constant. If the Shrink parameter is 1, the variance shrinks to 0 linearly as the last generation is reached.***

- **Migration** is the movement of individuals between subpopulations (the best individuals from one subpopulation replace the worst individuals in another subpopulation). We can control how migration occurs by the following three parameters.

- **Direction of Migration:** **Migration can take place in one direction or two. In the so-called "Forward migration" the nth subpopulation migrates into the (n+1)'th subpopulation. while in the so-called "Both directions Migration", the nth subpopulation migrates into both the (n-1)th and the (n+1)th subpopulation.**

- **Migration wraps at the ends of the subpopulations. That is, the last subpopulation migrates into the first, and the first may migrate into the last. To prevent wrapping, specify a subpopulation of size zero.**

- Fraction of Migration is the number of the individuals that we move between the subpopulations.

  So, Fraction of Migration is the fraction of the smaller of the two subpopulations that moves.

  If individuals migrate from a subpopulation of 50 individuals into a population of 100 individuals and Fraction is 0.1, 5 individuals (0.1 * 50) migrate. Individuals that migrate from one subpopulation to another are copied.

  They are not removed from the source subpopulation. Interval of Migration counts how many generations pass between migrations.

- The Nelder-Mead simplex algorithm appeared in 1965 and is now one of the most widely used methods for nonlinear unconstrained optimization.

- The Nelder-Mead method attempts to minimize a scalar-valued nonlinear function of n real variables using only function values, without any derivative information (explicit or implicit).

- The Nelder-Mead method thus falls in the general class of direct search methods. The method is described as follows:

- Let *f(x)* be the function for minimization where *x* is a vector in *n* real variables.

- We select *n+1* initial points for *x* and we follow the steps:

- **Step 1. Order:**

Order the n+1 vertices to satisfy

$f(x1) \le f(x2) \le \ldots \le f(xn+1)$,

using the tie-breaking rules given below.

- **Step 2. Reflect.** **Compute the reflection point xr from**

$$x_r = \bar{x} + \rho(\bar{x} - x_{n+1}) = (1+\rho)\bar{x} - \rho x_{n+1}$$

where

$$\bar{x} = \sum_{i=1}^{n} x_i / n$$

is the centroid of the n best points (all vertices except for xn+1). Evaluate fr=f(xr).

If f1 ≤ fr < fn accept the reflected point xr and terminate the iteration.

- **Step 3.** **Expand. If fr < f1 , calculate the expansion point xe,**

$$x_e = \bar{x} + \chi(x_r - \bar{x}) = \bar{x} + \rho\chi(\bar{x} - x_{n+1}) = (1 + \rho\chi)\bar{x} - \rho\chi x_{n+1}$$

and evaluate fe=f(xe). If fe < fr, accept xe and terminate the iteration; otherwise (if fe ≥ fr), accept xr and terminate the iteration.

- **Step 4. Contract.** **If fr ≥ fn, perform a contraction between and the better of xn+1 and xr.**

$$x_{cc} = \bar{x} - \gamma(\bar{x} - x_{n+1}) = (1-\gamma)\bar{x} + \gamma x_{n+1}$$

and evaluate fcc = f(xcc)

Outside.

If fn ≤ fr < fn+1 (i.e. xr is strictly better than xn+1), perform an outside contraction: calculate   and evaluate fc = f(xc). If fc ≤ fr, accept xc and terminate the iteration; otherwise, go to step 5 (perform a shrink).

Inside. **If fr ≥ fn+1, perform an inside contraction: calculate**

$$x_{cc} = \bar{x} - \gamma(\bar{x} - x_{n+1}) = (1-\gamma)\bar{x} + \gamma x_{n+1}$$

and evaluate fcc = f(xcc). If fcc < fn+1, accept xcc and terminate the iteration; otherwise, go to step 5 (perform a shrink).

Step 5.

Perform a shrink step. Evaluate f at the n points

$$v_i = x_1 + \sigma\,(x_i - x_1),\; i = 2, \ldots, n+1.$$

The (unordered) vertices of the simplex at the next iteration consist of $x_1, v_2, \ldots, v_{n+1}$.

After this preparation, we are ready to solve the minimization problem of our functional of the Equation (9) as a minimization problem.

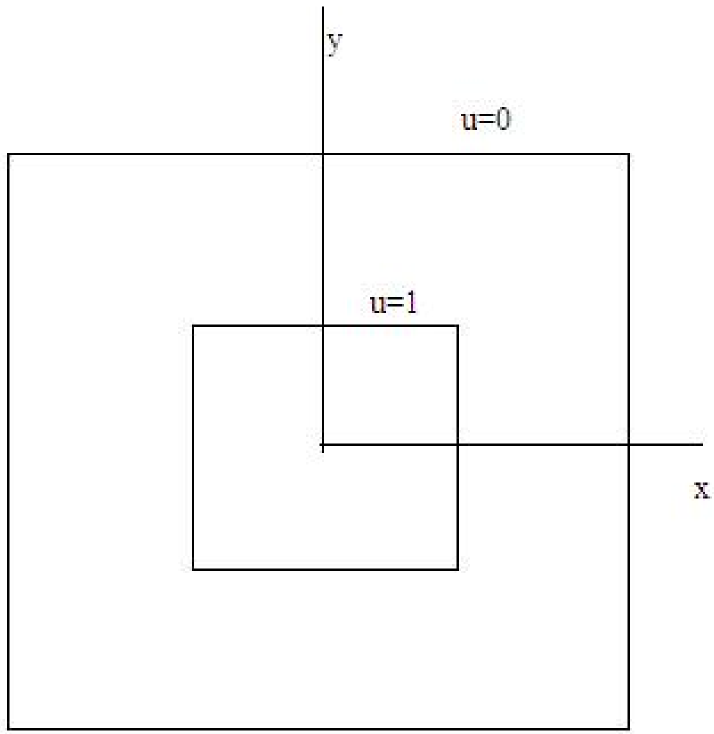
- **The minimization is achieved by using Genetic Algorithms (GA) and the method of Nelder-Mead exactly as we described previously.**
- **We can use the MATLAB software package**

In the next numerical example our GENETIC ALGORITHM has the following Parameters:

- **Population type:**
  Double Vector Population size: 30
- **Creation function: Uniform**
- **Fitness scaling: Rank**
- **Selection function: roulette**
- **Reproduction: 6 – Crossover fraction 0.8**
- **Mutation: Gaussian – Scale 1.0,**
- **Shrink 1.0**
- **Crossover: Scattered**
- **Migration: Both – fraction 0.2, interval: 20**
- **Stopping criteria: 50 generations**

We present now the following Numerical Example

$$\mathrm{div}\left(\left|\nabla u\right|^{p-2}\nabla u\right)=0$$



First consider now the following problem of the Figure 2 in this Cartesian domain with *u* equal to zero in the external boundary and *u* equal to 1 in the internal boundary.

u=0 $x=\pm 2, -2\le y\le 2$

$$u\in[0,2]\times[0,2]-[0,1]\times[0,1]$$

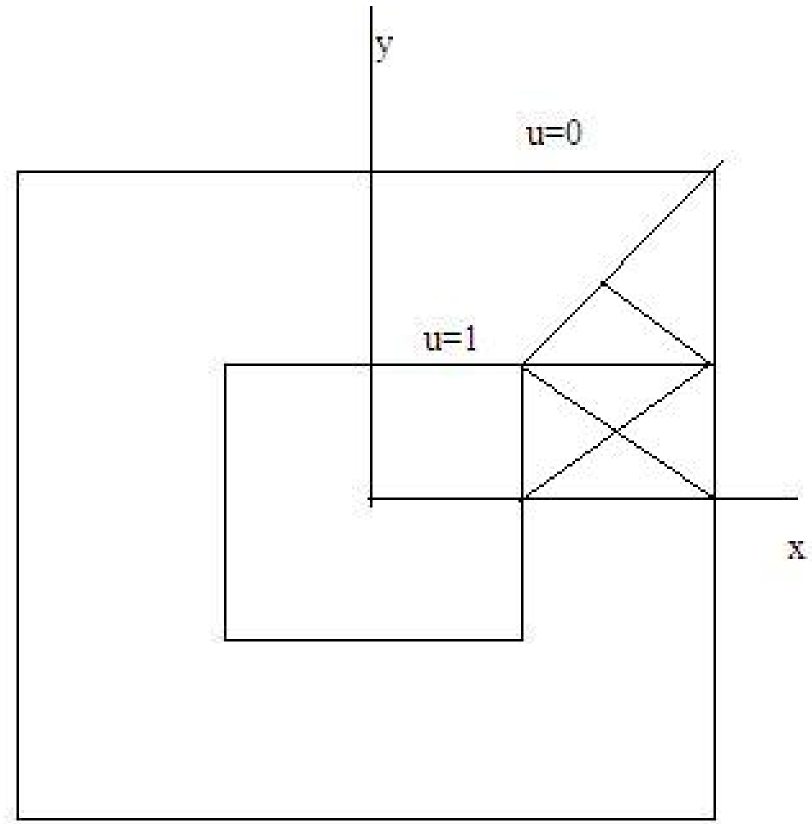
u=1 $y=\pm 2, -2\le x\le 2$

← **Fig.2**

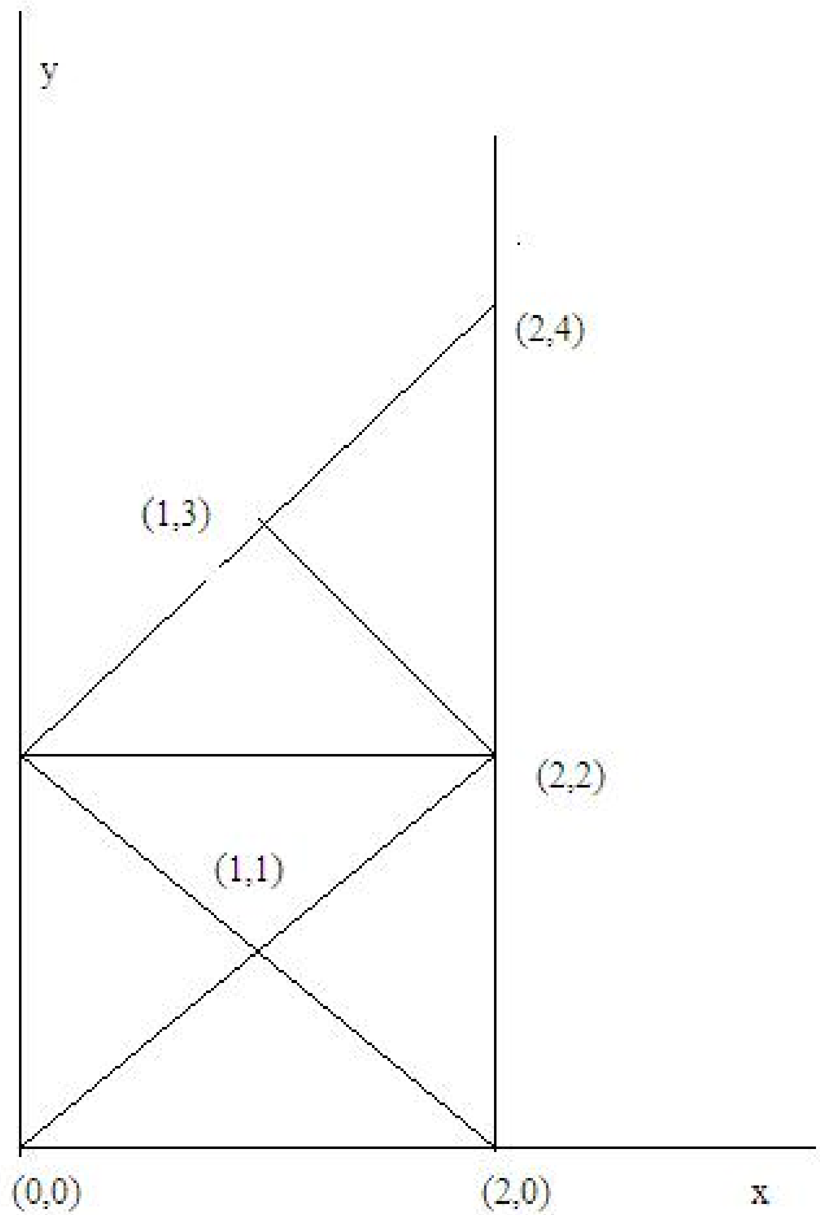
$$x = \pm 1, -1 \le y \le 1$$

$$y = \pm 1, -1 \le x \le 1$$

See the Figure 3 $\rightarrow$



Due to symmetry, we can split the domain in 8 same trapezoids (trapezia). It is sufficient to solve our problem in one of them with the boundary conditions “u” equal to 0 in the external boundary and “u” equal to 1 in the internal boundary.



Taking one of these trapezoids and splitting it into 6 triangles like in Figure 3, we have in some enlargement the following Figure,

← This is the Figure 4

We consider as $u_1, u_2, u_3, u_4, u_5, u_6, u_7$

the value of the u at the points

$$(0,0),(2,0),(2,2),(2,4),(1,3),(0,2),(1,1)$$

$$u_1 = u(0,0),$$

$$u_2 = u(2,0),$$

$$u_3 = u(2,2),$$

$$u_4 = u(2,4),$$

$$u_5 = u(1,3),$$

$$u_6 = u(0,2),$$

$$u_7 = u(1,1)$$

Then by considering

$$u_1 = u(0,0),$$
$$u_2 = u(2,0),$$
$$u_3 = u(2,2),$$
$$u_4 = u(2,4),$$
$$u_5 = u(1,3),$$
$$u_6 = u(0,2),$$
$$u_7 = u(1,1)$$

$$u_s = a_n x_s + b_n y_s + c_n$$
$$u_q = a_n x_q + b_n y_q + c_n$$
$$u_r = a_n x_r + b_n y_r + c_n$$

and now we have the Equations (7.1), (7.2) and (7.3)

In each triangle, the three equations can be solved with respect to a_n, b_n, c_n

as follows

$$a_n = \frac{\begin{vmatrix} u_s & y_s & 1 \\ u_q & y_q & 1 \\ u_r & y_r & 1 \end{vmatrix}}{D}$$

$$b_n = \frac{\begin{vmatrix} x_s & u_s & 1 \\ x_q & u_q & 1 \\ x_r & u_r & 1 \end{vmatrix}}{D}$$

$$c_n = \frac{\begin{vmatrix} x_s & y_s & u_s \\ x_q & y_q & u_q \\ x_r & y_r & u_r \end{vmatrix}}{D}$$

We have, considering also that u1 and u6 are equal to 1 and u2, u3, u4 are equal to 0. So, after some algebraic manipulation we find that we have to minimize the quantity *I* with respect to u5 and u7

Where I is given as follows

$$I = \left|2u_5\right|^p + \left|\sqrt{1^2 + (1-2u_5)^2}\right|^p + \left|\sqrt{1^2 + (1-2u_7)^2}\right|^p +$$

$$+\left|(2-2u_7)\right|^p + \left|\sqrt{1^2 + (1-2u_7)^2}\right|^p + \left|(2u_7)\right|^p$$

With respect to

$$u_5, u_7$$

In order to find the global minimum of I we use GA

- **Population type: Double Vector Population**
- **Population size: 30**
- **Creation function: Uniform**
- **Fitness scaling: Rank**
- **Selection function: roulette**
- **Reproduction: 6**
- **Crossover fraction 0.8**
- **Mutation: Gaussian – Scale 1.0, Shrink 1.0**
- **Crossover: Scattered**
- **Migration: Both – fraction 0.2, interval: 20**
- **Stopping criteria: 50 generations)**

and continue with Nelder-Mead

So we find the following results for different values of “p”

p	u_5	u_7	I
2	0.2500	0.5000	5.5000
3	0.3145	0.5000	5.4623
4	0.3471	0.5000	5.4280
5	0.3678	0.5000	5.3994
6	0.3824	0.5000	5.3754
7	0.3935	0.5000	5.3550
8	0.4024	0.5000	5.3373
10	0.4155	0.5000	5.3082
20	0.4468	0.5000	5.2246
50	0.4721	0.5000	5.1375
200	0.4903	0.5000	5.0582

Solution of the equations (2), (3) and (4)

We remind the problems:

$$\theta\frac{\partial\rho}{\partial t}=c^{\alpha}\lambda\,\mathrm{div}\left(\left|\nabla\rho^{n}\right|^{p-2}\nabla\rho^{n}\right), \qquad (1)$$

If we scale out the constants in (1), we derive

$$\frac{\partial u}{\partial t}=\Delta_{p}\left(u^{n}\right)$$

where a particular case of (2) is the non-Newtonian filtration equation

$$\frac{\partial u}{\partial t}=\Delta_{p}u$$

and

$$\frac{\partial u}{\partial t}=\mathrm{div}\left(\left|\nabla u\right|^{p-2}\nabla u\right)+\lambda u^{q}, \qquad (4)$$

Consider that u can be written as

$u = \sum_n \lambda_n(t) f_n$ or $u = \sum_n f_n(t)$ where λ_n have been incorporated to $f_n(t)$

In this "dynamic" case, in a triangular mesh of $\mathbb{R}^2$ we can have $f_n = a_n(t)x + b_n(t)y + c_n(t)$ triangle.

$$u_s = a_n(t)x_s + b_n(t)y_s + c_n(t) \qquad (7.1)$$

$$u_q = a_n(t)x_q + b_n(t)y_q + c_n(t) \qquad (7.2)$$

$$u_r = a_n(t)x_r + b_n(t)y_r + c_n(t) \qquad (7.3)$$

Of course, we can use higher degree polynomials like quadratic or cubic.
For quadratic:

$$u_s = a_n(t)x_s + b_n(t)y_s + c_n(t) + d_n(t)x_s^2$$

$$+ e_n(t)d_s^2 + h_n(t)x_s y_s$$

$$u_q = a_n(t)x_q + b_n(t)y_q + c_n(t) + d_n(t)x_q^2$$

$$+ e_n(t)d_q^2 + h_n(t)x_q y_q$$

$$u_r = a_n(t)x_r + b_n(t)y_r + c_n(t) + d_n(t)x_r^2$$
$$+e_n(t)d_r^2 + h_n(t)x_r y_r$$

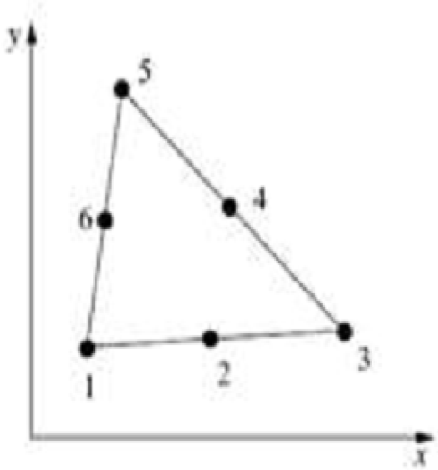


Fig.5

We express $a_n(t), b_n(t), c_n(t), d_n(t), e_n(t), h_n(t)$
with respect not only u in vertices, but also in a node along the midside of each edge. See F

Finally using the so-called **collocation method** or **least square method** or **the met**
([35]÷[40]) we can obtain a system of non-linear Ordinary Differential Equations th
a variety of methods (Runge – Kutta etc…).

Part I -- Synopsis

We have examined the boundary value problem $\operatorname{div}\left(|\nabla u|^{p-2}\nabla u\right)=0$ where u is a kno
boundary of our domain using Variational Principle (Finite elements). The Proble
Optimization problem that can be solved by Genetic Algorithms plus Nelder-Mea
paper of the author with the title "Solving Differential Equations via Genetic
presented in [1] while the author use the same method to solve various problems in [2

With the Hybrid method of Genetic Algorithm + Nelder-Mead we have combined
both methods, that are **a)** the convergence to the global minimum (genetic algorithn
accuracy of the Nelder-Mead method.

both methods, that are **a)** the convergence to the global minimum (genetic algorithm) plus accuracy of the Nelder-Mead method.

Also, we have discussed briefly the solution of $\frac{\partial u}{\partial t} = \Delta_p \left(u^n\right)$

$$\frac{\partial u}{\partial t} = \Delta_p u$$

and

$$\frac{\partial u}{\partial t} = \operatorname{div}\left(\left|\nabla u\right|^{p-2} \nabla u\right) + \lambda u^q,$$

using the so-called collocation method or least square method or the method of moments.

With the Hybrid method of Genetic Algorithm + Nelder-Mead we have combined the advantages of both methods, that are

a) the convergence to the global minimum (genetic algorithm) plus

b) the high accuracy

Another problem that we will examine now is the Numerical Solution of the Schrodinger-Maxwell equations (with a general nonlinear term) via Finite Elements and Genetic Algorithms with Nelder-Mead

Recently, the existence of a nontrivial solution to the nonlinear Schrodinger-Maxwell equations in R^3, assuming on the nonlinearity the general hypotheses introduced by Berestycki & Lions has been proved. In this lecture the Numerical Solution of the system of Partial Differential Equations of Schrodinger-Maxwell equations (with a general nonlinear term) via Finite Elements and Genetic Algorithms with Nelder-Mead is proposed. The method of Finite Elements and Genetic Algorithms with Nelder-Mead that has been proposed by the author recently is also used.

1 Introduction

Recently many authors have examined the following system of the non-linear Partial Differen
Equations (PDEs) in $\mathbb{R}^3$

$$-\nabla^2 u + q\phi u = g(u) \quad (1)$$

$$-\nabla^2 \phi = qu^2 \quad (2)$$

with $g(.)$ being a known function.

The system of (1) and (2) is called:

Schrodinger-Maxwell equations. This system of Equations arises in many mathematical phy
contexts, such as in quantum electrodynamics, in nonlinear
optics, in nano-mechanics and in plasma physics.

The greatest part of the literature focuses on the study of the previous system for the very spe
nonlinearity $g(u) = -u + |u|^{p-1} u$ and existence, nonexistence and multiplicity results are provi
in many papers for this particular problem (see [18]÷[28]).

In [29], Azzollini, D'Avenia and Pomponio that a solution of a boundary problem of (1) and (2) yie

Recently, Azzollini, D'Avenia and Pomponio proved that a solution of a boundary problem of (1) and (2) yields the minimization of some functional.

In this lecture, we solve the problem with the method of finite elements

In this lecture we will solve the boundary value problem of (1) and (2) where g is known using Variational Techniques (Finite elements).

First we will produce the appropriate functional for minimization. After finding this functional, the solution of (1) and (2) with the necessary boundary conditions can be easily reduced to an Optimization problem that can be solved by Genetic Algorithms with Nelder-Mead.

An early paper of the author with the title “Solving Differential Equations via Genetic Algorithms” was presented in 1996

Now, we examine the Variational Formulation of (1) and (2) and Finite Elements Approach with GA

2 Variational Formulation of (1) and (2) and Finite Elements Approach with GA

Consider that our functional is functional of

u,ϕ , i.e. $I = I(u,\phi)$

Let the "point" of u_0,ϕ_0 that minimize the I(u,ϕ). Then for another point u,ϕ w

$u = u_0 + \varepsilon_1 u_1, \phi = \phi_0 + \varepsilon_2 \phi_1$

So, we must have the first order conditions

$\frac{\partial I(u,\phi)}{\partial \varepsilon_1} = 0$ and $\frac{\partial I(u,\phi)}{\partial \varepsilon_2} = 0$

Working first for (1) we can formulate:

$$I = \frac{1}{2}\iiint_V (\nabla u)^2 dv + \frac{1}{2}\iiint_V q\phi u^2 dv - \iiint_V G(u) dv + B(\phi)$$

with $G(u)=\int g(u)du$ and $B(\phi)$ a function in ϕ

It is easy to verify by replacing $u=u_0+\varepsilon_1 u_1$ that

the condition $\dfrac{\partial I(u,\phi)}{\partial \varepsilon_1}=0$ leads to

$\iiint_V (\nabla u_0)(\nabla u_1)dv+\iiint_V q\phi u_0 u_1 dv-\iiint_V g(u_0)u_1 dv=0$ Now by applying the Green's first ide

$$\oiint u_1(\nabla u \cdot \bar{n})+\iiint_V(-\nabla^2 u_0)u_1 dv+$$

$$+\iiint_V q\phi u_0 u_1 dv-\iiint_V g(u_0)u_1 dv=0$$

Considering appropriate u we can have $\oiint u_1(\nabla u \cdot \bar{n})=0$ which means

$\iiint_V(-\nabla^2 u_0)u_1 dv+\iiint_V q\phi u_0 u_1 dv-\iiint_V g(u_0)u_1 dv=0$ or

$$\iiint_V (-\nabla^2 u_0 + q\phi u_0 - g(u_0))u_1 dv = 0$$

But u_1 is arbitrary which implies

$-\nabla^2 u_0 + q\phi u_0 - g(u_0) = 0$ i.e. we have (1)

Working analogously with (2) we could have

$$I = \frac{1}{2}\iiint_V (\nabla\phi)^2 dv - \iiint_V q\phi u^2 dv + C(u) \qquad (4)$$

with $C(u)$ a function in u

We must compromise (3) and (4). To this end we multiply the right hand member of (4) with coefficient -1/2 and finally we propose the functional

$$I = \frac{1}{2}\iiint_V (\nabla u)^2 dv - \frac{1}{4}\iiint_V (\nabla\phi)^2 dv +$$

$$+\frac{1}{2}\iiint_V q\phi u^2 dv - \iiint_V G(u) dv$$

So, the solution of the problem of Schrodinger-Maxwell equations

$$-\nabla^2 u + q\phi u = g(u) \qquad (1)$$

$$-\nabla^2 \phi = qu^2 \qquad (2)$$

leads to

$$\min_{u,\phi} I$$

where

$$I = \frac{1}{2}\iiint_V (\nabla u)^2 dv - \frac{1}{4}\iiint_V (\nabla \phi)^2 dv +$$

$$+\frac{1}{2}\iiint_V q\phi u^2 dv - \iiint_V G(u)dv$$

We consider that *u* is written as a linear combination of our known basis' functions with unknown coefficients.

$$u = \sum_n \lambda_n f_n \qquad u = \sum_n f_n$$

So, we have the following minimization problem

Minimize I

$$I = \frac{1}{2}\iiint_V (\nabla u)^2\, dv - \frac{1}{4}\iiint_V (\nabla \phi)^2\, dv +$$

$$+\frac{1}{2}\iiint_V q\phi u^2\, dv - \iiint_V G(u)\, dv$$

One can select a triangular mesh and appropriate functions $f_n = a_n x + b_n y + c_n$

that have non-zero value only in the "*n-th*" triangle

- **These our finite elements**

So, in a triangular mesh,

we can have

$$f_n = a_n x + b_n y + c_n$$

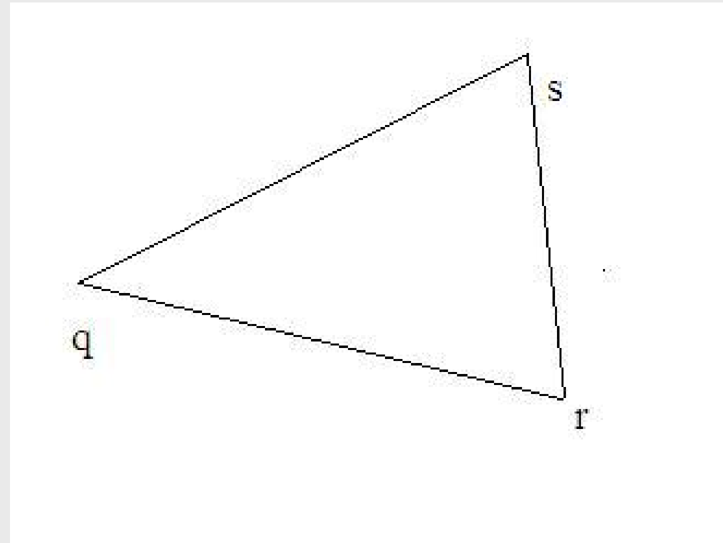


Fig.1 A triangle in a 2-D mesh

for the n-th triangle. To avoid to write continuity conditions on the common vertices of the triangles of the mesh, one can find that in the n-th triangle of the points "s", "q" and "r" (see Figure 1)

- $$u_s = a_n x_s + b_n y_s + c_n \quad (7.1)$$

$$u_q = a_n x_q + b_n y_q + c_n \quad (7.2)$$

$$u_r = a_n x_r + b_n y_r + c_n \quad (7.3)$$

Actually the Figure 1 is a triangle in our 2-D mesh

And now we have the Equations (7.1), (7.2) and (7.3)

There three equations can be solved with respect to

$$a_n, b_n, c_n$$

as follows

$$a_n = \frac{\begin{vmatrix} u_s & y_s & 1 \\ u_q & y_q & 1 \\ u_r & y_r & 1 \end{vmatrix}}{D} \qquad b_n = \frac{\begin{vmatrix} x_s & u_s & 1 \\ x_q & u_q & 1 \\ x_r & u_r & 1 \end{vmatrix}}{D}$$

$$c_n = \frac{\begin{vmatrix} x_s & y_s & u_s \\ x_q & y_q & u_q \\ x_r & y_r & u_r \end{vmatrix}}{D}$$

These are the equation (8.1), (8.2) and (8.3)

These are the Equations (8.1), (8.2) and (8.3) where “D” is given as follows

$$D = \begin{vmatrix} x_s & y_s & 1 \\ x_q & y_q & 1 \\ x_r & y_r & 1 \end{vmatrix}$$

Note that two times D is the algebraic area of the triangle.

$$\min_{u,\phi} I$$

where

$$I = \frac{1}{2}\iiint_V (\nabla u)^2 dv - \frac{1}{4}\iiint_V (\nabla \phi)^2 dv +$$

$$+\frac{1}{2}\iiint_V q\phi u^2 dv - \iiint_V G(u) dv$$

we find the equivalent minimization problem

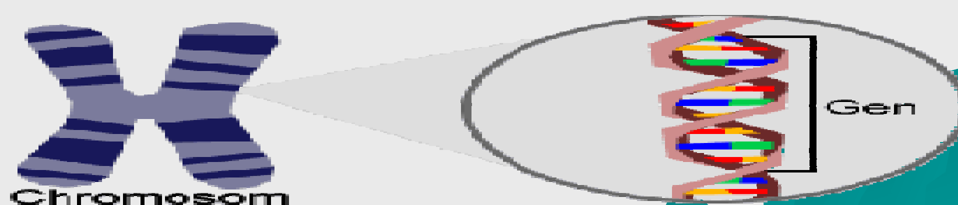
$$\min \int W(a_n, b_n, c_n, \tilde{a}_n, \tilde{b}_n, \tilde{c}_n) dv \quad (9)$$

where $W(a_n, b_n, c_n, \tilde{a}_n, \tilde{b}_n, \tilde{c}_n)$ is the function that we find after replacing $f_n = a_n x + b$

$\tilde{f}_n = \tilde{a}_n x + \tilde{b}_n y + \tilde{c}_n$ ...

The Equation (9) can be solved now by a variety of techniques.

- The author uses the Method of Genetic Algorithms with the so-called method of Nelder and Meade for Non-linear Problems.
- The same optimization scheme Genetic Algorithms with Nelder-Meade method has recently applied by the author with great success



Before proceeding in the solution of the problem, some background on “G-A” i.e. Genetic Algorithms and Nelder-Mead is necessary.

In many papers, we have proposed a hybrid method that includes

- a) Genetic Algorithm for finding rather the neiborhood of the global minimum than the global minimum itself and
- b) Nelder-Mead algorithm to find the exact point of the global minimum itself.

So, with this Hybrid method of Genetic Algorithm + Nelder-Mead we combine the advantages of both methods, that are

- A) the convergence to the global minimum via the genetic algorithm

plus

- B) the high accuracy of the Nelder-Mead method.

In order to find the global minimum of I we use GA

- **Population type: Double Vector Population**
- **Population size: 30**
- **Creation function: Uniform**
- **Fitness scaling: Rank**
- **Selection function: roulette**
- **Reproduction: 6**
- **Crossover fraction 0.8**
- **Mutation: Gaussian – Scale 1.0, Shrink 1.0**
- **Crossover: Scattered**
- **Migration: Both – fraction 0.2, interval: 20**
- **Stopping criteria: 50 generations)**

and continue with Nelder-Mead

So we find the following results for different values of "p"

So, the problem can be solved now by a variety of techniques. The author uses Genetic Algorithms with Nelder-Meade for Non-linear Problems as in [2], [3], [4], [5], [6], [7], [8].
The same optimization scheme: Genetic Algorithms with Nelder-Meade is also applied for (19).

References:

1. Nikos E. Mastorakis, "Solving Differential Equations via Genetic Algorithms", Proceedings of the Circuits, Systems and Computers '96, (CSC'96), Piraeus, Greece, July 15-17, 1996, 3rd Volume: Appendix, pp.733-737
2. Nikos E. Mastorakis, "On the solution of ill-conditioned systems of Linear and Non-Linear Equations via Genetic Algorithms (GAs) and Nelder-Mead Simplex search", 6th WSEAS International Conference on EVOLUTIONARY COMPUTING (EC 2005), Lisbon, Portugal, June 16-18, 2005.

3. Nikos E. Mastorakis, “Genetic Algorithms and Nelder-Mead Method for the Solution of Boundary Value Problems with the Collocation Method”, WSEAS Transactions on Information Science and Applications, Issue 11, Volume 2, 2005, pp. 2016-2020.

4. Nikos E. Mastorakis, “On the Solution of Ill-Conditioned Systems of Linear and Non-Linear Equations via Genetic Algorithms (GAs) and Nelder-Mead Simplex Search”, WSEAS Transactions on Information Science and Applications, Issue 5, Volume 2, 2005, pp. 460-466.

5. Nikos Mastorakis, “Genetic Algorithms and Nelder-Mead Method for the Solution of Boundary Value Problems with the Collocation Method”, 5th WSEAS International Conference on SIMULATION, MODELING AND OPTIMIZATION (SMO '05), Corfu Island, Greece, August 17-19, 2005.

6. Nikos E. Mastorakis, “Solving Non-linear Equations via Genetic Algorithm”, WSEAS Transactions on Information Science and Applications, Issue 5, Volume 2, 2005, pp. 455-459.

7. Nikos E. Mastorakis, “The Singular Value Decomposition (SVD) in Tensors (Multidimensional Arrays) as an Optimization Problem. Solution via Genetic Algorithms and method of Nelder-Mead”, 6th WSEAS Int. Conf. on SYSTEMS THEORY AND SCIENTIFIC COMPUTATION (ISTASC'06), Elounda, Agios Nikolaos, Crete Island, Greece, August 21-23, 2006.

8. Nikos E. Mastorakis, "Unstable Ordinary Differential Equations: Solution via Genetic Algorithms and the method of Nelder-Mead", The 6th WSEAS International Conference on SYSTEMS THEORY AND SCIENTIFIC COMPUTATION, Elounda, Agios Nikolaos, Crete Island, Greece, August 18-20, 2006.

9. Nikos E. Mastorakis, "Unstable Ordinary Differential Equations: Solution via Genetic Algorithms and the Method of Nelder-Mead", WSEAS TRANSACTIONS on MATHEMATICS, Issue 12, Volume 5, December 2006, pp. 1276-1281.

10. Nikos E. Mastorakis, "An Extended Crank-Nicholson Method and its Applications in the Solution of Partial Differential Equations: 1-D and 3-D Conduction Equations", WSEAS TRANSACTIONS on MATHEMATICS, Issue 1, Volume 6, January 2007, pp. 215-224.

11. Saeed-Reza Sabbagh-Yazdi, Behzad Saeedifard, Nikos E. Mastorakis, "Accurate and Efficient Numerical Solution for Trans-Critical Steady Flow in a Channel with Variable Geometry", WSEAS TRANSACTIONS on APPLIED and THEORETICAL MECHANICS, Issue 1, Volume 2, January 2007, pp. 1-10.

12. Saeed-Reza Sabbagh-Yazdi, Mohammad Zounemat-Kermani, Nikos E. Mastorakis, "Velocity Profile over Spillway by Finite Volume Solution of Slopping Depth Averaged Flow", WSEAS TRANSACTIONS on APPLIED and THEORETICAL MECHANICS, Issue 3, Volume 2, March 2007, pp. 85.

13. Iurie Caraus and Nikos E. Mastorakis, “Convergence of the Collocation Methods for Singular Integrodifferential Equations in Lebesgue Spaces”, WSEAS TRANSACTIONS on MATHEMATICS, Issue 11, Volume 6, November 2007, pp. 859-864.

14. Iurie Caraus, Nikos E. Mastorakis, “The Stability of Collocation Methods for Approximate Solution of Singular Integro-Differential Equations”, WSEAS TRANSACTIONS on MATHEMATICS, Issue 4, Volume 7, April 2008, pp. 121-129.

15. Xu Gen Qi, Nikos E. Mastorakis, “Spectral distribution of a star-shaped coupled network”, WSEAS TRANSACTIONS on APPLIED and THEORETICAL MECHANICS , Issue 4, Volume 3, April 2008.

16. Iurie Caraus, Nikos E. Mastorakis, “Direct Methods for Numerical Solution of Singular Integro-Differentiale Quations in Classical (case $\gamma \neq 0$)”, 10th WSEAS International Conference on MATHEMATICAL and COMPUTATIONAL METHODS in SCIENCE and ENGINEERING (MACMESE'08), Bucharest, Romania, November 7-9, 2008.

17. Gabriella Bognar, Numerical and Numerical and Analytic Investigation of Some Nonlinear Problems in Fluid Mechanics, COMPUTER and SIMULATION in MODERN SCIENCE, Vol.II, WSEAS Press, pp.172-179, 2008

18. Astrita G., Marrucci G., Principles of Non-Newtonian Fluid Mechanics, McGraw-Hill, New York, NY, USA, 1974.
19. Volquer R.E., Nonlinear flow in porous media by finite elements, ASCE Proc., J. Hydraulics Division Proc. Am. Soc. Civil Eng., 95 (1969), 2093-2114
20. Ahmed N., Sunada D.K., Nonlinear flow in porous media, J. Hydraulics Division Proc. Am. Soc. Civil Eng., 95 (1969), 1847-1857.
21. Peebles P.J.E., Star distribution near a collapsed object, Astrophysical Journal, Vol. 178, (1972),. 371-376.
22. Carelman T., Probl`emes math´ematiques dans la th´eorie cin´etique de gas, Almquist-Wiksells, Uppsala, 1957.
23. Bartnik R., McKinnon J., Particle-like solutions of the Einstein-Yang-Mills equations. Phys. Rev. Lett. 61 (1988), 141-144
24. Otani M., A remark on certain nonlinear elliptic equations. *Proc. Fac. Sci. Tokai Univ.* 19 (1984), 23--28.
25. Pelissier, M.-C., Reynaud, L., Étude d'un modèle mathématique d'ecoulement de glacier, C. R. Acad. Sci., Paris, Sér. A 279 (1974), 531-534. (French)
26. Schoenauer M., A monodimensional model for fracturing, In A. Fasano and M. Primicerio (editors): Free Boundary Problems, Theory Applications, Pitman Research Notes in Mathematics 79, Vol. II., London, 701-711 (1983).
27 John C. Strikwerda, Finite Difference Schemes and Partial Differential Equations, SIAM, 2004

28. Hans Petter Langtangen Computational Partial Differential Equations: Numerical Methods and Diffpack Programming, Springer, 2003

29. W. L. Wood, Introduction to Numerical Methods for Water Resources, Oxford University Press, 1993

30. Daniel R. Lynch, Numerical Partial Differential Equations for Environmental Scientists and Engineers: A First Practical Course, Springer, 2005

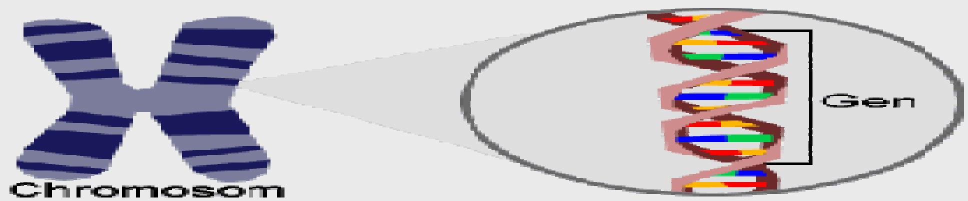
31. Nikos E. Mastorakis, “Numerical Schemes for Non-linear Problems in Fluid Mechanics”, Proceedings of the 4th IASME/WSEAS International Conference on CONTINUUM MECHANICS, Cambridge, UK, February 24-26, 2009, pp.56-61

32. Evans, Lawrence C. , A New Proof of Local C1,α Regularity for Solutions of Certain Degenerate Elliptic P.D.E.", Journal of Differential Equations 45: 356-373, 1982

32. Lewis, John L. (1977). Capacitary functions in convex rings, Archive for Rational Mechanics and Analysis 66: 201–224, 1977

33. Lagarias, J.C., J. A. Reeds, M. H. Wright, and P. E. Wright, "Convergence Properties of the Nelder-Mead Simplex Method in Low Dimensions," SIAM Journal of Optimization, Vol. 9 Number 1, pp. 112-147, 1998

34. J. A. Nelder and R. Mead, “A simplex method for function minimization”, Computer Journal, 7 , 308-313, 1965
35. F. H. Walters, L. R. Parker, S. L. Morgan, and S. N. Deming, Sequential Simplex Optimization, CRC Press, Boca Raton, FL, 1991
36.A. Ern, J.L. Guermond, Theory and practice of finite elements, Springer, 2004, ISBN 0-3872-0574-8
37. S. Brenner, R. L. Scott, The Mathematical Theory of Finite Element Methods, 2nd edition, Springer, 2005, ISBN 0-3879-5451-1
38. P. G. Ciarlet, The Finite Element Method for Elliptic Problems, North-Holland, 1978, ISBN 0-4448-5028-7
39 Y. Saad, Iterative Methods for Sparse Linear Systems, 2nd edition, SIAM, 2003, ISBN 0-8987-1534-2
40. J.J.Conor and C.A.Brebbia, Finite Element Techniques for Fluid Flow, Butterworth, London, 1976



MANY THANKS