

Fluid dynamics of expanding and contracting tube: a case study of methods of series summation

O. D. MAKINDE and M. O. ADEBAYO
Applied Mathematics Department, University of the North,
Private Bag X1106, Sovenga 0727, South Africa

Abstract. The fluid dynamics of expanding and contracting tube is investigated mathematically using both the Navier-Stokes and continuity equations for an incompressible viscous fluid. The problem admits similarity solution, thereby reducing the unsteady Navier-Stokes equations to a parameter dependent fourth order nonlinear ordinary differential equation. Analytical solutions are constructed for the problem using perturbation technique together with a new special type of Hermite-Padé approximants and important properties of the overall structure of the flow are discussed. The model is appropriate to simulate wind tunnel tests on rheological phenomenon in physiological systems.

Keywords: Expanding and contracting tube, Hermite-Padé approximants, bifurcation.

2000 Mathematics Subject Classification: 76E25.

1. INTRODUCTION

Fluid flow through an expanding and contracting tube is a complex problem due to the interaction between the tube-wall and the flowing fluid, Heil (1997). Expanding and contracting tubes are easily deformed by negative transmural pressure owing to marked reduction of rigidity. Thus, they show a considerable nonlinearity and reveal various complicated phenomena. It is usually used to simulate biological flows such as blood flow in arteries or veins, flow of urine in urethras and airflow in the bronchial airways. These investigations are very useful for the study and prediction of many diseases, as the lung disease (asthma and emphysema), or the cardiovascular diseases (heart stroke). The major research goal remains, the full understanding of the flow structure and the mechanism driving this flow. Many previous theoretical works on flow in collapsible tubes concentrated on the development and analysis of simpler models, by reducing the spatial dimension of the problem, which involve a number of ad-hoc assumptions e.g. Conrad (1969), Grotberg (1971), Flaherty et al. (1972), Cowley (1982), Bonis & Ribreau (1987) etc. Experimental example of the work that have been done on collapsible tube includes the one performed with finite-length elastic tubes whose upstream and downstream ends are held open (i.e. Starling-resistor, Brower & Scholten 1975, Bertram 1986). Inside a pressure chamber, thin-walled elastic tube (made of latex rubber) is mounted on two rigid tubes. Fluid (liquid or gas) typically water or air respectively is driven through the tube, either by applying a controlled pressure-drop between the ends of the rigid tubes or by controlling the flow rate. If the external pressure exceeds the fluid pressure by a sufficiently large amount, the tube buckles non-axisymmetrically, which then leads to a nonlinear relation between pressure-drop and flow rate. At sufficiently large Reynolds numbers, the system produces self-excited oscillations, and exhibits hysteresis in transitions between dynamical states, multiple modes of oscillations (each having distinct frequency range), rich and complex nonlinear dynamics (Bertram et al. 1990). The inertia and resistance of the fluid in the supporting rigid tubes have an important influence on the system's overall dynamics. This experiment forms the basis for most recent theoretical investigations due to its three-dimensional nature. Meanwhile, Bertram & Pedley (1982), Bertram & Raymond (1991) investigated two-dimensional channel theoretical model with one wall of the channel been replaced by a membrane under longitudinal tension, viscous flow is driven through the channel by an imposed pressure-drop. The variation between the external pressure and the internal flow determine the deformation of the membrane. The dynamics of the problem is described by nonlinear ODE's whose numerical solutions exhibit oscillatory behaviour reminiscent of that observed in experiments. Despite the difficulties of producing two-dimensional flows experimentally, this system still attracted considerable theoretical attentions. Since it avoid the complications of three-dimensional flows found in the Starling-resistor, while still exhibiting phenomena such as flow limitation and self-excited oscillations.

Meanwhile, mathematical model of physical phenomena often results in nonlinear equations for some unknown function. Usually the problems cannot be solved exactly. The solutions of these nonlinear systems are dominated by their singularities: physically, a real singularity controls the local behaviour of a solution. There is a long tradition in applied mathematics to solve nonlinear problems by expansion in powers of some “small” perturbation parameter. The advantage of this approach is that it reduces the original nonlinear problem to a sequence of linear problems (Makinde, 1999). However, it is not always possible to find an unlimited number of terms of power series. Often it is possible to obtain a finite number of terms of that series and these may contain a remarkable amount of information. One can reveal the solution behaviour near the critical points by analysing partial sum (Makinde, 2001). Over the last quarter century, highly specialised techniques have been developed to improved the series summation and also used to extract the required information of the singularities from a finite number of series coefficients. The most frequently used methods include Domb-Sykes(1957), Shafer (1974), Hunter and Guerrieri (1980), Sergeev (1986), Drazin and Tourigny (1996), Sergeev and Goodson (1998), Makinde et al. (2002), etc.

In this paper, we investigate the flow of a viscous incompressible fluid in a collapsible tube. A special type of Hermite-Padé approximants technique is presented and utilised to analyse the flow structure. The chief merit of this new method is its ability to reveal the dominant singularity in the flow field together with solution branches of the underlying problem in addition to the one represented by the original series. In Sections 2 & 3, we establish the mathematical formulation for the problem. With the benefit of twenty years of advances in computing hardware, we are able to find many terms of the solution series as presented in Section 4. The methods used to sum the series are described in Section 5. In Section 6, we discuss the pertinent results.

2. MATHEMATICAL FORMULATION

The problem under consideration is that of transient flow of a viscous incompressible fluid in a collapsible tube. Take a cylindrical polar coordinate system (r, θ, z) where oz lies along the centre of the tube, r is the distance measured radially and θ is the azimuthal angle. Let u and v be the velocity components in the directions of z and r increasing respectively. It is assumed that the tube’s wall is at $r = a_0\sqrt{(1 - \alpha t)}$, where α is a constant of dimension $[T^{-1}]$ which characterizes unsteadiness in the flow field, a_0 is the characteristic radius of the tube at time $t = 0$ as shown in fig. 1;

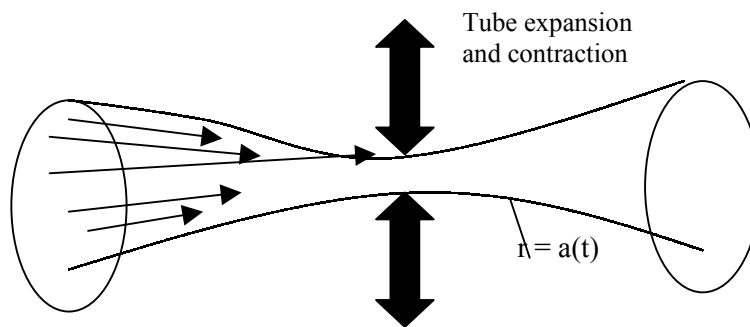


Fig. 1. Schematic diagram of the problem.

Then, for axisymmetric unsteady viscous incompressible flow, the Navier-Stokes equations are

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial z} + v \frac{\partial u}{\partial r} = -\frac{1}{\rho} \frac{\partial P}{\partial z} + \nu \nabla^2 u, \quad (2.01)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial z} + v \frac{\partial v}{\partial r} = -\frac{1}{\rho} \frac{\partial P}{\partial r} + \nu (\nabla^2 v - \frac{v}{r^2}), \quad (2.02)$$

where $\nabla^2 = \partial^2 / \partial r^2 + \partial / r \partial r + \partial^2 / \partial z^2$, P is the pressure, ρ the density and ν the kinematic viscosity of the fluid. The equation of continuity is

$$\frac{\partial}{\partial r}(rv) + r \frac{\partial u}{\partial z} = 0. \quad (2.03)$$

The appropriate boundary conditions are

Regularity of solution along z -axis, i.e.

$$\frac{\partial u}{\partial r} = 0, \quad v = 0, \quad \text{on } r = 0. \quad (2.04)$$

The axial and normal velocities at the wall are prescribed as

$$u = 0, \quad v = \frac{da}{dt}, \quad \text{on } r = a(t). \quad (2.05)$$

We introduce the stream-function Ψ and vorticity ω in the following manner:

$$u = \frac{1}{r} \frac{\partial \Psi}{\partial r} \quad \text{and} \quad v = -\frac{1}{r} \frac{\partial \Psi}{\partial z}, \quad (2.06)$$

$$\omega = \frac{\partial v}{\partial z} - \frac{\partial u}{\partial r} = -\frac{1}{r} \frac{\partial^2 \Psi}{\partial z^2} - \frac{1}{r} \frac{\partial^2 \Psi}{\partial r^2} + \frac{1}{r^2} \frac{\partial \Psi}{\partial r}. \quad (2.07)$$

Eliminating pressure P from (2.01) and (2.02) by using (2.06) and (2.07) we get

$$\frac{\partial \omega}{\partial t} + \frac{1}{r} \frac{\partial(\Psi, \omega)}{\partial(r, z)} + \frac{\omega}{r^2} \frac{\partial \Psi}{\partial z} = \nu \left[\nabla^2 \omega - \frac{\omega}{r^2} \right], \quad \omega = -\nabla^2 \Psi, \quad (2.08)$$

$$\frac{\partial \Psi}{\partial r} = 0, \quad \frac{\partial \Psi}{\partial z} = -a \frac{da}{dt}, \quad \text{on } r = a(t), \quad (2.09)$$

$$\frac{\partial}{\partial r} \left(\frac{1}{r} \frac{\partial \Psi}{\partial r} \right) = 0, \quad \frac{\partial \Psi}{\partial z} = 0, \quad \text{on } r = 0. \quad (2.10)$$

We introduce the following transformation;

$$\eta = \frac{r}{a_0 \sqrt{1 - \alpha t}}, \quad \Psi = \frac{a_0^2 \alpha z F(\eta)}{2}, \quad \omega = -\frac{\alpha z G(\eta)}{2 a_0 (\sqrt{1 - \alpha t})^3}. \quad (2.11)$$

Substituting equation (2.11) into equations (2.08)-(2.10), we obtain

$$\frac{d}{d\eta} \left[\frac{1}{\eta} \frac{d(\eta G)}{d\eta} \right] = R \left[\frac{G}{\eta} \frac{dF}{d\eta} - F \frac{d}{d\eta} \left(\frac{G}{\eta} \right) + \eta \frac{dG}{d\eta} + 3G \right], \quad G = \frac{d}{d\eta} \left(\frac{1}{\eta} \frac{dF}{d\eta} \right), \quad (2.12)$$

$$\frac{dF}{d\eta} = 0, \quad F = 1, \quad \text{on } \eta = 1, \quad (2.13)$$

$$\frac{d}{d\eta} \left(\frac{1}{\eta} \frac{dF}{d\eta} \right) = 0, \quad F = 0, \quad \text{on} \quad \eta = 0, \quad (2.14)$$

where $R = a_0^2 \alpha / 2\omega$ is the local Reynolds number ($R > 0$ represents contraction while $R < 0$ represents expansion of the tube's wall).

3. METHOD OF SOLUTION

To solve equations (2.12)-(2.14), it is convenient to take a power series expansion in the flow local Reynolds number R , i.e.,

$$F = \sum_{i=0}^{\infty} F_i R^i, \quad G = \sum_{i=0}^{\infty} G_i R^i, \quad (3.01)$$

substitute (3.01) into equations (2.12)–(2.14) and collecting the coefficients of like powers of R , we obtain the following;

Zeroth Order:

$$\frac{d}{d\eta} \left(\frac{1}{\eta} \frac{d(\eta G_0)}{d\eta} \right) = 0, \quad G_0 = \frac{d}{d\eta} \left(\frac{1}{\eta} \frac{dF_0}{d\eta} \right), \quad (3.02)$$

$$\frac{dF_0}{d\eta} = 0, \quad F_0 = 1, \quad \text{on} \quad \eta = 1, \quad (3.03)$$

$$\frac{d}{d\eta} \left(\frac{1}{\eta} \frac{dF_0}{d\eta} \right) = 0, \quad F_0 = 0, \quad \text{on} \quad \eta = 0, \quad (3.04)$$

Higher Order ($n \geq 1$):

$$\frac{d}{d\eta} \left(\frac{1}{\eta} \frac{d(\eta G_n)}{d\eta} \right) = R \left[\sum_{i=0}^{n-1} \left(\frac{G_i}{\eta} \frac{dF_{n-i-1}}{d\eta} - F_i \frac{d}{d\eta} \left(\frac{G_{n-i-1}}{\eta} \right) \right) + \eta \frac{dG_{n-1}}{d\eta} + 3G_{n-1} \right], \quad G_n = \frac{d}{d\eta} \left(\frac{1}{\eta} \frac{dF_n}{d\eta} \right), \quad (3.05)$$

$$\frac{dF_n}{d\eta} = 0, \quad F_n = 0, \quad \text{on} \quad \eta = 1, \quad (3.06)$$

$$\frac{d}{d\eta} \left(\frac{1}{\eta} \frac{dF_n}{d\eta} \right) = 0, \quad F_n = 0, \quad \text{on} \quad \eta = 0. \quad (3.07)$$

We have written a MAPLE program that calculates successively the coefficients of the solution series. In outline, it consists of the following segments:

- (1) Declaration of arrays for the solution series coefficients e.g. $F = \text{array}(0..43)$, $G = \text{array}(0..43)$.
- (2) Input the leading order term and their derivatives i.e. F_0 , G_0 .
- (3) Using a MAPLE loop procedure, iterate to solve equations (3.05)-(3.07) for the higher order terms i.e. F_n , G_n , $n=1,2,3,\dots$
- (4) Compute the skin friction and axial pressure gradient coefficients.

Details of the MAPLE program can be found in the appendix. Some of the solution stream-function and vorticity are then given as follows:

$$F(\eta) = (2\eta^2 - \eta^4) + \frac{R\eta^2}{36}(\eta^2 - 1)^2(\eta^2 - 10) + \frac{R^2\eta^2}{1940400}(\eta^2 - 1)^2(2\eta^6 - 10\eta^4 + 596\eta^2 - 1057) + O(R^3), \quad (3.08)$$

$$G(\eta) = -8\eta + \frac{2Rr}{3}(2\eta^4 - 12\eta^3 + 7) - \frac{R^2\eta}{135}(3\eta^8 - 105\eta^6 + 480\eta^4 - 705\eta^2 + 271) + O(R^3). \quad (3.09)$$

The wall skin friction is given by

$$\tau_w = -\mu \frac{\partial u}{\partial r} = -\frac{\alpha\mu z}{2a_0(\sqrt{1-\alpha t})^3} \frac{d}{d\eta} \left(\frac{1}{\eta} \frac{dF}{d\eta} \right) \quad \text{at } \eta = 1, \quad (3.10)$$

where μ is the coefficient of dynamic viscosity. From the axial component of the Navier-Stokes equations, the pressure drop in the longitudinal direction can be obtained. Let

$$\frac{\partial P}{\partial z} = \frac{\mu\alpha z A}{2a_0^2(1-\alpha t)^2}, \quad (3.11)$$

we substitute (2.11) together with (3.11) into (1.01) and obtain

$$A = \frac{1}{\eta} \frac{d}{d\eta} \left[\eta \frac{d}{d\eta} \left(\frac{1}{\eta} \frac{dF}{d\eta} \right) \right] - R \left(\left(\frac{1}{\eta} \frac{dF}{d\eta} \right)^2 - \frac{F}{\eta} \frac{d}{d\eta} \left(\frac{1}{\eta} \frac{dF}{d\eta} \right) + \eta \frac{d}{d\eta} \left(\frac{1}{\eta} \frac{dF}{d\eta} \right) + \frac{2}{\eta} \frac{dF}{d\eta} \right). \quad (3.12)$$

4. COMPUTER EXTENDED SERIES SOLUTION

In order to investigate the flow structure at moderately large Reynolds numbers, we expand $H=G(\eta)$ at $\eta=1$ and A representing wall skin friction and axial pressure gradient parameters respectively, in powers of the Reynolds number R , i.e.

$$H = 8 + 2R - \frac{56R^2}{135} + \frac{3389R^3}{22680} - \frac{510494R^4}{81648000} + \frac{1213633R^5}{143700480} + \dots \quad (4.01)$$

$$A = 8 + \frac{38R}{3} - \frac{152R^2}{135} + \frac{3287R^3}{7560} - \frac{15420829R^4}{81648000} + \frac{133921837R^5}{1539648000} + \dots \quad (4.02)$$

We compute the first 44 coefficients of the above series as shown in Table (4.01) below. The signs of the coefficients alternate after the second term and are monotonically decreasing in magnitude. Hence, the convergence of the series may be limited by a singularity on the negative real axis.

Table (4.01): Computation showing the coefficients of skin friction (H) and the axial pressure gradient (A).

I	H[I]	A[I]
0	8.00000000000000000000	8.00000000000000000000
1	2.00000000000000000000	12.66666666666666666666
2	-0.41481481481481481481	-1.12592592592592592592
3	0.1494268077601410934744	0.4347883597883597883597
4	-0.0625238707622966882226	-0.1888696477562218302959

5	0.0281519333361540151663	0.0869821134441119009020
6	-0.0132724706996553826849	-0.0416211775640870658507
7	0.0064643388160512792644	0.0204834857891982133353
8	-0.0032265990912916592118	-0.0103027895546493893152
9	0.0016417802495310663793	0.0052732235273090776156
10	-0.0008484311187308859129	-0.0027377428203082558834
11	0.0004440820648382163006	0.0014383673576706599757
12	-0.0002349402466731163546	-0.0007633259858988316029
13	0.0001254305567430137767	0.0004085881358779705208
14	-0.0000674917367212586422	-0.0002203406416769250305
15	0.0000365642147638740717	0.0001195992765265208744
16	-0.000019927763852042002	-0.0000652907605394189572
17	0.0000109183831428787684	0.0000358248604550732079
18	-0.60104252693295131x10 ⁻⁵	-0.0000197465699469006895
19	0.33226732632656363x10 ⁻⁵	0.0000109288257549187453
20	-0.18438486165334343x10 ⁻⁵	-0.6070983316920430x10 ⁻⁵
21	0.10267460443666857 x10 ⁻⁵	0.33837697772319878 x10 ⁻⁵
22	-0.57354398864899106 x10 ⁻⁶	-0.18917804092095402 x10 ⁻⁵
23	0.32130638137606970 x10 ⁻⁶	0.10606127724960538 x10 ⁻⁵
24	-0.18047552471895858 x10 ⁻⁶	-0.59615721680683764 x10 ⁻⁶
25	0.10161859240932710 x10 ⁻⁶	0.33588889421710805 x10 ⁻⁶
26	-0.5734608688398468 x10 ⁻⁷	-0.18966368143603949 x10 ⁻⁶
27	0.32429391236793735 x10 ⁻⁷	0.10731428830386769 x10 ⁻⁶
28	-0.18374451829805262 x10 ⁻⁷	-0.60835082276469294 x10 ⁻⁷
29	0.10429735365447369 x10 ⁻⁷	0.345476396215183357x10 ⁻⁷
30	-0.59301252514417988 x10 ⁻⁸	-0.19651710560692073 x10 ⁻⁷
31	0.33770651565496140 x10 ⁻⁸	0.111957897005567141x10 ⁻⁷
32	-0.1926006091458700 x10 ⁻⁸	-0.63876388090257912x10 ⁻⁸
33	0.10999677996265670 x10 ⁻⁸	0.364938537339216432x10 ⁻⁸
34	-0.62902978347529542 x10 ⁻⁹	-0.20876544368137609x10 ⁻⁸
35	0.36016306705767121 x10 ⁻⁹	0.119570938332980531x10 ⁻⁸
36	-0.20645924643932868 x10 ⁻⁹	-0.68563344512564036x10 ⁻⁹
37	0.11848122398245690 x10 ⁻⁹	0.393578321549529340x10 ⁻⁹
38	-0.68064353071254785 x10 ⁻¹⁰	-0.22616151458953450x10 ⁻⁹
39	0.39140086061786793 x10 ⁻¹⁰	0.130086427859861979x10 ⁻⁹
40	-0.22528604394664139 x10 ⁻¹⁰	-0.74894526488293995x10 ⁻¹⁰
41	0.12978885564204897 x10 ⁻¹⁰	0.431572279542049867x10 ⁻¹⁰
42	-0.74836362617043105 x10 ⁻¹¹	-0.24889966117313476x10 ⁻¹⁰
43	0.43185993904897869 x10 ⁻¹¹	0.143663202118368715x10 ⁻¹⁰

5. SERIES IMPROVEMENT TECHNIQUE: (A NEW APPROACH)

The main tool of this paper is a simple technique of series summation based on the generalization of Padé approximants and may be described as follows. Let us suppose that the partial sum

$$U_{N-1}(\lambda) = \sum_{i=0}^{N-1} a_i \lambda^i = U(\lambda) + O(\lambda^N) \text{ as } \lambda \rightarrow 0, \quad (5.01)$$

is given. We are concerned with the bifurcation study by analytic continuation as well as the dominant behaviour of the solution by using partial sum (5.01). We expect that the accuracy of the critical parameters

will ensure the accuracy of the solution. It is well known that the dominant behaviour of a solution of a linear ODE can often be written as Guttman (1989)

$$U(\lambda) \approx \begin{cases} K(\lambda_C - \lambda)^\alpha & \text{for } \alpha \neq 0, 1, 2, \dots \\ K(\lambda_C - \lambda)^\alpha \ln|\lambda_C - \lambda| & \text{for } \alpha = 0, 1, 2, \dots \end{cases} \quad \text{as } \lambda \rightarrow \lambda_C \quad (5.02)$$

where K is some constant and λ_c is the critical point with the exponent α . However, we shall make the simplest hypothesis in the contest of nonlinear problems by assuming the $U(\lambda)$ is the local representation of an algebraic function of λ . Therefore, we seek an expression of the form

$$F_d(\lambda, U_{N-1}) = A_{0N}(\lambda) + A_{1N}^d(\lambda)U^{(1)} + A_{2N}^d(\lambda)U^{(2)} + A_{3N}^d(\lambda)U^{(3)}, \quad (5.03)$$

such that

$$A_{0N}(\lambda)=1, \quad A_{iN}(\lambda) = \sum_{j=1}^{d+i} b_{ij} \lambda^{j-1}, \quad (5.04)$$

and

$$F_d(\lambda, U) = O(\lambda^{N+1}) \quad \text{as } \lambda \rightarrow 0, \quad (5.05)$$

where $d \geq 1$, $i = 1, 2, 3$. The condition (5.04) normalises the F_d and ensures that the order of series A_{iN} increases as i and d increase in value. There are thus $3(2+d)$ undetermined coefficients b_{ij} in the expression (5.03). The requirement (5.05) reduces the problem to a system of N linear equations for the unknown coefficients of F_d . The entries of the underlying matrix depend only on the N given coefficients a_i . Henceforth, we shall take

$$N=3(2+d), \quad (5.06)$$

so that the number of equations equals the number of unknowns. Equation (5.05) is a new special type of Hermite-Padé approximants. Both the algebraic and differential approximants form of equation (5.05) are considered. For instance, we let

$$U^{(1)}=U, \quad U^{(2)}=U^2, \quad U^{(3)}=U^3, \quad (5.07)$$

and obtain a cubic Padé approximant. This gives an extension of the idea of quadratic Padé approximants by Shafer (1974) and Sergeev (1986). Furthermore, Sergeev and Goodson (1998), Drazin and Tourigny (1996) had also suggested a similar form of higher order algebraic approximants. Generally, this enables us to obtain solution branches of the underlying problem in addition to the one represented by the original series. In the same manner, we let

$$U^{(1)}=U, \quad U^{(2)}=DU, \quad U^{(3)}=D^2U, \quad (5.08)$$

in equation (5.04), where D is the differential operator given by $D=d/d\lambda$. This leads to a second order differential approximants. It is an extension of the integral approximants idea by Hunter and Baker (1979) and enables us to obtain the dominant singularity in the flow field i.e. by equating the coefficient $A_{3N}(\lambda)$ in the equation (5.05) to zero. The critical exponent α_N can easily be found by using Newton's polygon algorithm. However, it is well known that, in the case of algebraic equations, the only singularities that are structurally stable are simple turning points. Hence, in practice, one almost invariably obtains $\alpha_N = 1/2$. If we assume a singularity of algebraic type as in equation (5.02), then the exponent may be approximated by

$$\alpha_N = 1 - \frac{A_{2N}(\lambda_{CN})}{DA_{3N}(\lambda_{CN})}. \quad (5.09)$$

Using the above procedure, we performed series summation and improvement study on the solution series obtained in Table (4.01). Our results show the dominant singularity in the flow field to be $R_c = -1.6739367347720807$ (which corresponds to the radius of convergence and the turning point in the flow field) with the critical exponent $\alpha_c = 0.5$ as shown in the table (5.01). We also noticed that $A \sim A_{-1}/R$ and $H \sim H_{-1}/R$ as $R \rightarrow 0$ on the secondary solution branch where $A_{-1} \approx -280.98050$ and $H_{-1} \approx -67.6702067$. It is noteworthy to mention that the wall shear stress $H \rightarrow 0$ as $R \rightarrow -1.6431409627402$, i.e., separation and possible flow reversal occur due to tube wall expansion.

Table (5.01): Computation showing the dominant singularity, secondary solution branch asymptotic behaviour and critical exponent.

d	N	R_c	A_{-1}	H_{-1}	α_c
1	9	-1.6863617034262282	-----	-----	0.331632240
3	15	-1.6739700125855259	-276.48020	67.3394184	0.498634106
5	21	-1.6739367218509650	-281.11885	67.6661825	0.498726578
7	27	-1.6739367371687654	-280.98120	67.6702443	0.500000000
9	33	-1.6739367347718036	-280.98046	67.6702065	0.500000000
11	39	-1.6739367347720807	-280.98050	67.6702067	0.500000000
12	42	-1.6739367347720807	-280.98050	67.6702067	0.500000000

6. GRAPHICAL RESULTS AND DISCUSSION

In fig.6.01, we observed that the fluid axial velocity profile is parabolic with maximum value at centerline and minimum at the plates. It is interesting to note that the fluid axial velocity generally decreases with an increase in tube contraction due to the strong influence of the negative transmural pressure owing to marked reduction of rigidity (i.e. $R > 0$). In reality, this is possible, since contraction brings about a reduction in the tube's cross-sectional area, hence, decreasing the amount of flow passing through the compressed region. Table 5.01, shows the convergence of the dominant singularity R_c in the flow field together with its corresponding exponent α_c , as well as the asymptotic behaviour of wall skin friction and pressure gradient as the flow Reynolds number tends to zero. It is noteworthy also that R_c is the bifurcation point and lies in the negative real axis of the flow Reynolds number R , i.e. the region representing tube's contraction. This critical value of Reynolds number enables the biomedical engineers to determine accurately the maximum expansion of the tube walls due to the variation in the tube's external and internal pressure i.e. $a_0 = \sqrt{2\nu R_c / \alpha}$.

Figs.6.02 and 6.03 show the sketch of bifurcation diagrams for the skin friction and fluid axial pressure gradient parameters. For tube's contraction i.e. $R > 0$, only one solution branch exist i.e. type I. This is the primary solution branch and it shows that the wall skin friction and fluid axial pressure gradient increase with increase in R . In the expansion region, i.e. $R < 0$, two solution branches are identified (i.e. type I, II). A simple turning point with exponent $\alpha_c = 0.5$ exists between type I and type II solution branches i.e. R_c . We observed that the type II solution is physically unreasonable but mathematically interesting. It is interesting to note that the turning point here also corresponds to the dominant singularity in the flow field.

Finally, in this paper, we have proposed a new form of series summation and improvement technique based on the generalizations of Padé approximants, i.e. a special type of Hermite-Padé approximant. We have applied this method to investigate the problem of squeezing flow in parallel plates viscometer with great success. The chief novelty of this procedure is its ability to reveal the dominant singularities together with solution branches of the underlying nonlinear problem in addition to the branch represented locally by the original series. Generally, we have found that this new method is very competitive. However, we have not yet developed a theory that would explain its strengths and limitations and so we have relied on intelligent numerical investigation.

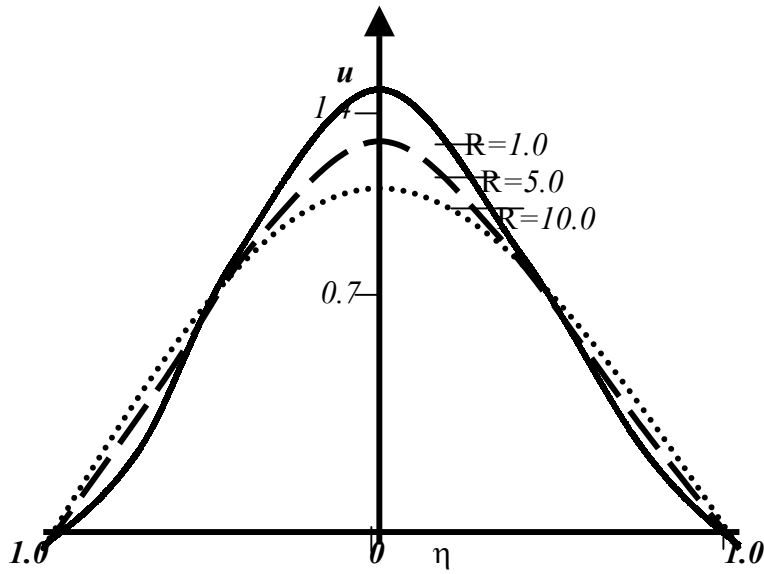


Fig. 6.01 Axial velocity profile during tube contraction ($R>0$).

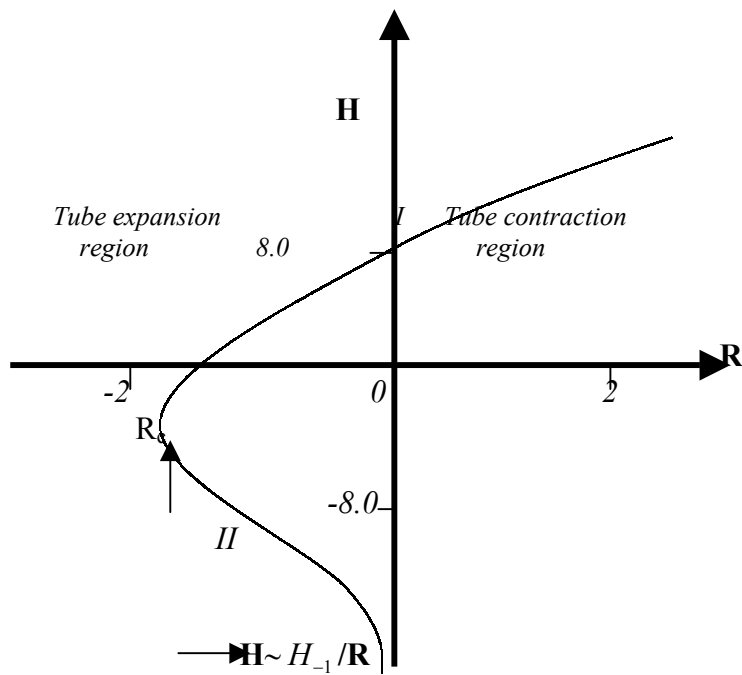


Fig. 6.02 A sketch of bifurcation diagram for skin friction.

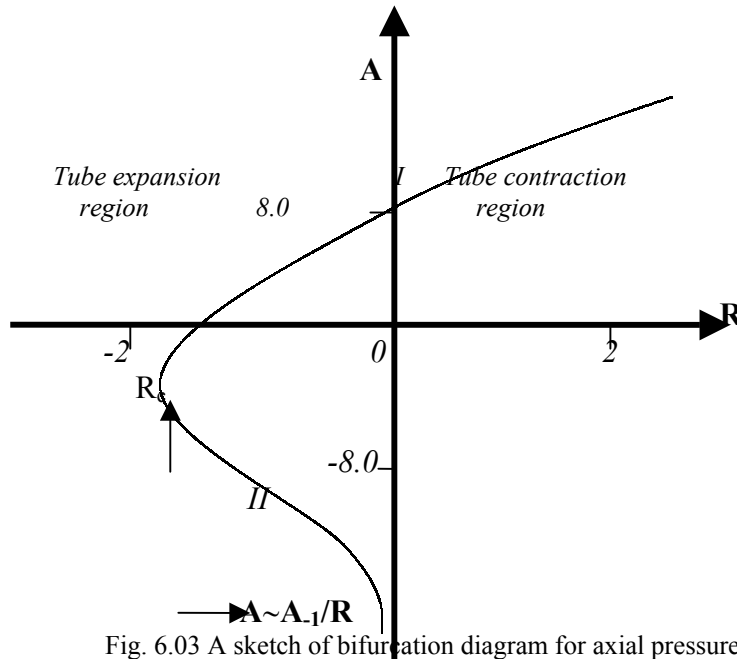


Fig. 6.03 A sketch of bifurcation diagram for axial pressure gradient.

APPENDIX

A1: The Maple procedure to solve the equations (3.05) to (3.07).

Here we declare the arrays to store the computed results

```
F:=array(0..34): G:=array(0..34): Fr:=array(0..34): Gr:=array(0..34):
```

Here we input the zero order solution F[0] and G[0].

```
F[0]:= (2*r^2-r^4): G[0]:= -8*r:
```

```
Fr[0]:= diff(F[0],r): Gr[0]:= diff(G[0],r):
```

This computes the higher order terms i.e. n>0.

```
for n from 1 by 1 to 43 do
```

```
  A1:=normal(1/r*sum(g[i]*fr[n-i-1]+f[i]*(g[n-i-1]/r-gr[n-i-1]),i=0..n1)):
  A2:=normal(r*gr[n-1]+3*g[n-1]):
```

```
  A:=R*(A1+A2):
```

```
  A1:=0:A2:=0: g11:=normal(r*(int(A,r)+K)):
  A:=0: g1:=normal(int(g11,r)/r):
```

```
  g11:=0: f11:=normal(r*(int(g1,r)+M)):
  f1:=normal(int(f11,r)):
```

```
  r:=1: K:=normal(solve(f11=0,K)): M:=normal(solve(f1=0,M)):
  r:='r':
```

```
  f11:=0: F[n]:=normal(f1):
```

```
  f1:=0: G[n]:=normal(g1):
```

```
  g1:=0: Fr[n]:=normal(diff(F[n],r)): Gr[n]:=normal(diff(G[n],r)):
  K:='K': M:='M': print(F[n]): print(G[n]):
```

Here we compute the wall skin friction coefficients.

```
print(evalf(sub(r=1,G[n]))):
```

```
od: quit();
```

REFERENCE

- [1] Bertram, C. D., Unstable equilibrium behaviour in collapsible tubes, *J. Biomech.* **19** (1986), 61-69.

- [2] Bertram, C. D., and Pedley, T. J., A mathematical model of unsteady collapsible tube behaviour, *J. Biomech.* **15** (1982), 39-50.
- [3] Bertram, C. D., and Raymond, C. J., Measurements of wave speed and compliance in a collapsible tube during self-excited oscillations: A test of the choking hypothesis, *Med. Biol. Eng. Comput.* **29** (1991), 493-500.
- [4] Bertram, C. D., Raymond, C. J., and Pedley, T. J., Mapping of instabilities for flow through collapsible tubes of differing length, *J. Fluids. Struct.* **4** (1990), 125-153.
- [5] Bertram, C. D., Raymond, C. J., and Pedley, T. J., Application of nonlinear dynamics concepts to the analysis of self-excited oscillations of a collapsible tube conveying a fluid, *J. Fluids. Struct.* **5** (1991), 391-287.
- [6] Bonis, M., and Ribreau, C., Etude de quelques proprietes de l'ecoulement dans une conduite collabable, *La Houille Blanche* **3-4** (1987), 165-173.
- [7] Brower, R. W., and Scholten, C., Experimental evidence on the mechanism for the instability of flow in collapsible vessels, *Med. Biol. Engng* **13** (1975), 839-845.
- [8] Conrad, W. A., Pressure-flow relationship in collapsible tubes, *IEEE Trans. Bio-Med. Engng BME-16* (1969), 284-295.
- [9] Cowley, S. J., Elastic jumps in fluid-filled elastic tubes, *J. Fluid Mech.* **116** (1982), 459-473.
- [10] Cowley, S. J., On the wavetrains associated with elastic jumps on fluid-filled elastic tubes, *Q. J. Mech. Appl. Math.* **36** (1983), 289-312.
- [11] Domb, C and Sykes, M. F., On the susceptibility of a ferromagnetic above the Curie point, *Proc. R. Soc. London, Ser. A*, **240** (1957), 214-228.
- [12] Drazin, P. G., and Tourigny, Y., Numerical study of bifurcations by analytic continuation of a function defined by a power series, *SIAM J. Appl. Math.* **56** (1996), 1-18.
- [13] Elad, D., Kamm, R. D., and Shapiro, A. H., Choking phenomena in a lung-like model, *ASME J. Biomech. Engng* **109** (1987), 1-9.
- [14] Flaherty, J. E., Keller, J. B., and Rubinow, S. I., Post buckling behaviour of elastic tubes and rings with opposite sides in contact, *SIAM J. Appl. Math.* **23** (1972), 446 – 455.
- [15] Grotberg, J. B., Pulmonary flow and transport phenomena, *Ann. Rev. Fluid Mech.* **26** (1971), 529-571.
- [16] Guttamann, A. J., Asymptotic analysis of power-series expansions, Phase Transitions and Critical Phenomena, *C. Domb and J. K. Lebowitz, eds.* Academic Press, New York, (1989) pp. 1-234.
- [17] Hunter, C., and Guerrieri, B., Deducing the properties of singularities of functions from their Taylor series coefficients, *SIAM J. Appl. Math.*, **39** (1980), 248-263.
- [18] Hunter, D. L. and Baker G. A., Methods of series analysis III: Integral approximant methods, *Phys. Rev. B* **19** (1979), 3808-3821.
- [19] Heil, M., Stokes flow in collapsible tubes-computational and experiment, *J. Fluid Mech.*, **353** (1997), 285-312.
- [20] Makinde, O. D., Extending the utility of perturbation series in problems of laminar flow in a porous pipe and a diverging channel, *Jour. of Austral. Math. Soc. Ser. B* **41** (1999), 118-128.
- [21] Makinde, O. D., Heat and mass transfer in a pipe with moving surface: Effects of viscosity variation and energy dissipation, *Quaestiones Mathematicae*, **24** (2001), 97-108.
- [22] Makinde, O. D., Motsumi, T. G., and Ramollo, M. P., Squeezing flow between parallel plates: A bifurcation study, *Far East Jour. Appl. Math.*, **9**, 2 (2002), 81-94.
- [23] Shafer, R. E., On quadratic approximation, *SIAM J. Numer. Anal.*, **11** (1974), 447-460.
- [24] Sergeev, A. V. A recursive algorithm for Padé-Hermite approximations, *U.S.S.R. Comput. Math. Phys.* **26** (1986), 17-22.
- [25] Sergeev, A. V. and Goodson, A. Z. Summation of asymptotic expansions of multiple-valued functions using algebraic approximants: Application to anharmonic oscillators, *J. Phys., A: Math. Gen.* **31** (1998), 4301-4317.

Project of a hybrid-heuristical algorithm Progetto di un algoritmo ibrido-euristico

L. Marra

Abstract. Starting with the analysis of some algorithms of CLP we introduce the concept of metabox and establish an algorithm which provides good results for volumetric reasons, for the weight balancing and for the management of the boarding and unloading. A special attention is drawn to the fragility of the objects contained in the package. This is the crucial problem for the senders. It is implied in the most transports.

Sommario. Partendo dall'analisi di alcuni algoritmi del CLP si introduce il concetto di metascatola e si mette a punto un algoritmo che oltre a fornire dei buoni risultati per quanto riguarda la resa volumetrica, raggiunge buoni risultati per quanto riguarda il bilanciamento del peso e la gestione delle operazioni di carico e scarico. Particolare attenzione viene posta poi al problema della fragilità degli oggetti contenuti nel pacco, problema cruciale per gli spedizionieri e che coinvolge la maggior parte dei trasporti.

1. INTRODUZIONE

Si affronta il progetto e lo sviluppo di un algoritmo per la soluzione del problema relativo al caricamento di un container (CLP). Le basi di tale algoritmo sono euristiche e si sono fatte delle scelte con il dichiarato obiettivo di rispondere contemporaneamente ad alcune delle diverse problematiche presentate dal CLP quali, entrando nello specifico, la resa volumetrica (AVU¹), la stabilità, il bilanciamento del peso ed un'agevole gestione delle operazioni di carico e scarico. Si sono attinte idee dai modelli di vari studiosi che si è cercato poi di migliorare ed è quindi per questo motivo che l'algoritmo che s'introdurrà, oltre che essere euristico, è stato chiamato anche ibrido.

I modelli presi in considerazione sono quelli di Bischoff e Davies [25], quello di Gehring-Menschner-Meyer [11] ed infine quello di Bischoff-Ratliff [19] per cui l'algoritmo proposto non è del tutto innovativo ma ha fondamentalmente l'intento di sperimentare l'efficienza di una serie di regole euristiche costituenti un sistema di caricamento che mira al conseguimento di diversi scopi, in particolare quello di una buona performance volumetrica. Inoltre, giacché una delle caratteristiche adottate è quella di concepire il caricamento del container per sezioni verticali (muri) indipendenti, il metodo si presterebbe anche a rispondere ad esigenze del tipo multi-drop² ma ciò andrebbe ovviamente a discapito di un buon riempimento complessivo del container stesso.

Ad ogni modo l'algoritmo è stato tradotto in un software (in linguaggio Matlab[®]) ed è stato fornito di un'interfaccia grafica del cui concepimento teorico si parlerà in un prossimo lavoro.

1.2 Obiettivi ed approccio risolutivo dell'algoritmo

Riprendendo quanto affermato, lo sviluppo dell'algoritmo ibrido-euristico per la risoluzione del CLP si propone il raggiungimento contemporaneo dei seguenti obiettivi:

- A. Ottenimento di una buona efficienza volumetrica.
- B. Sistemazione geometrica delle scatole in modo da garantire un'adeguata stabilità dell'intero carico.
- C. Agevole manovrabilità nelle operazioni di carico e scarico.
- D. Possibilità di gestione di carichi del tipo multi-drop.
- E. Possibilità di operazioni di *post-processing* mirate a realizzare una distribuzione bilanciata del peso all'interno del container.

¹ Acronimo di *Average Volume Utilization* (ossia utilizzazione media del volume)

² Trattasi dell'ipotesi di un carico che è suddiviso in più di una destinazione

I punti C, D ed E si possono concretizzare pensando ad un caricamento del container per sezioni (muri) indipendenti. Ora, tali sezioni potrebbero immaginarsi tanto in senso orizzontale quanto in senso verticale; ai fini, però, del perseguimento dell'altro obiettivo B, è ovvio far ricadere la scelta su un riempimento per sezioni verticali e ciò in virtù dei risultati negativi prodotti dall'analisi di stabilità del metodo GMM che prevede al contrario la sistemazione del carico per strati orizzontali.

Riguardo infine al punto A si è deciso di realizzare la costruzione di ogni singolo muro attraverso l'impiego di "metascatole" che possono essere definite come un insieme di scatole appartenenti ad una stessa tipologia (quindi scatole omogenee) e raggruppate tra loro spazialmente in modo da apparire come un'unica grande scatola. Va da sé che, al limite, per metascatola si può anche intendere il singolo pacco! Il motivo di una tale scelta risiede ovviamente nel tentativo di rendere la strategia di caricamento migliore da un punto di vista dell'occupazione del container e ciò sulla base di due considerazioni empiriche: la prima scaturisce dalla natura stessa del CLP per cui, essendo il problema di fondo quello di colmare uno spazio assegnato con un numero di oggetti, più piccoli, di varie dimensioni, è evidente che si otterrà una sistemazione tanto

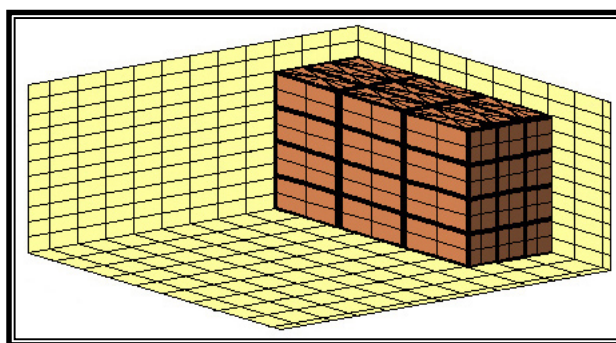


Figura 1. Esempio di metascatola costituita da 36 scatole omogenee, ossia appartenenti ad una stessa tipologia

migliore quanto più sarà efficiente la combinazione di tali oggetti nel processo di allocazione ed è quindi altrettanto evidente che, in tal senso, non esiste combinazione più vantaggiosa di quella di affiancare tra loro oggetti di identiche dimensioni. La seconda considerazione guarda, invece, ad un altro fatto oggettivo vale a dire che una metascatola offre, agli eventuali altri pacchi che potrebbero trovare posto su di essa, una superficie d'appoggio certamente superiore a quella offerta da una scatola singola.

Chiaramente un simile modo di agire si discosta da quanto previsto dal metodo BR di Bischoff-Ratcliff che considera sempre il concetto di muro (sezione verticale del container), ma con la differenza che esso è formato da colonne o subcolonne aventi come base un'unica scatola. D'altra parte, in quel metodo, la strategia impiegata è votata alla gestione di carichi multi-drop per i quali, nel processo di riempimento del singolo muro, ove verranno assegnati oggetti aventi tutti la stessa destinazione, in effetti, il numero di tipologie e la relativa numerosità delle scatole da piazzare sono piuttosto limitati. Ma questa è una possibilità che, di fatto, non è esclusa a priori dall'algoritmo ibrido-euristico che, per come definisce la metascatola, arriva a contemplare anche, come caso limite, situazioni come quelle considerate in BR. In altre parole, considerata una generica tipologia di pacchi, la filosofia è quella di costruire metascatole grandi se la disponibilità di pacchi è grande, e metascatole più piccole (colonne, subcolonne o addirittura un unico pezzo) se viceversa la disponibilità è più piccola.

Quello che in realtà si riduce, pensando ad un riempimento del container tramite metascatole, è la possibilità di effettuare operazioni di *swapping*³ tra le parti indipendenti del carico per il bilanciamento del peso. E' evidente, infatti, che quanto più piccole sono le parti in cui viene frazionato l'intero carico, tanto maggiori saranno le possibilità di scambiare tra di loro le varie parti per una più omogenea distribuzione delle masse. E sotto questo aspetto il metodo BR è certamente più performante, poiché prevede, oltre alla suddivisione del carico complessivo in muri, la suddivisione dei muri in colonne isolate. Tuttavia va anche detto che questo elemento di vantaggio si manifesta essenzialmente quando si ha a che fare con carichi

³ Sostituzione, scambio

debolmente eterogenei, mentre per carichi più eterogenei la differenza diventa più lieve giacché, a parità di volume totale del carico, un maggior numero di tipologie vuol dire mediamente un minor numero di scatole per tipologia e quindi l'impossibilità di costruire metascatole relativamente grandi.

Rispetto al metodo BR, viceversa, la costituzione di metascatole garantisce una migliore stabilità e ciò grazie ad un impaccamento che è compatto proprio perché formato da scatole appartenenti ad una stessa tipologia. Stavolta, esattamente al contrario di quanto visto poc'anzi, gli effetti positivi della stabilità sono più presenti nell'ipotesi di carichi debolmente eterogenei e meno in quella di carichi maggiormente eterogenei.

In definitiva il layout di carico risultante dalle scelte operate, è frutto di un problema di taglio tridimensionale, essendo costituito da una serie d'unità ciascuna descrivibile sempre come un set ghigliottinabile: la prima regione d'indagine è costituita dal volume del container che è suddivisibile in muri, quindi ogni muro risulterà composto da metascatole, ed infine ogni metascatola da scatole di tipo omogeneo. L'organizzazione finale si presenta quindi come una struttura gerarchica che si può tranquillamente accostare a quella tipica dei frattali.

Si conclude il paragrafo ricordando la natura euristica dell'algoritmo realizzato, quindi sottolineando la consapevolezza riguardo alle varie penalità che una tale scelta comporta. Fra tutte, l'appena ricordata suddivisione del carico in una sequenza di set ghigliottinabili che vuol dire frammentazione a più livelli dello spazio da colmare e, in definitiva, una ridotta capacità in termini di utilizzazione del volume.

1.3 Procedura risolutiva generale

Anzitutto viene fissato un sistema di riferimento cartesiano solidale con il container come quello rappresentato in figura 2.6. L'origine coincide con il vertice inferiore sinistro del *back-end*⁴ del container mentre gli assi x , y , z coincidono rispettivamente con la sua lunghezza (o profondità), larghezza ed altezza.

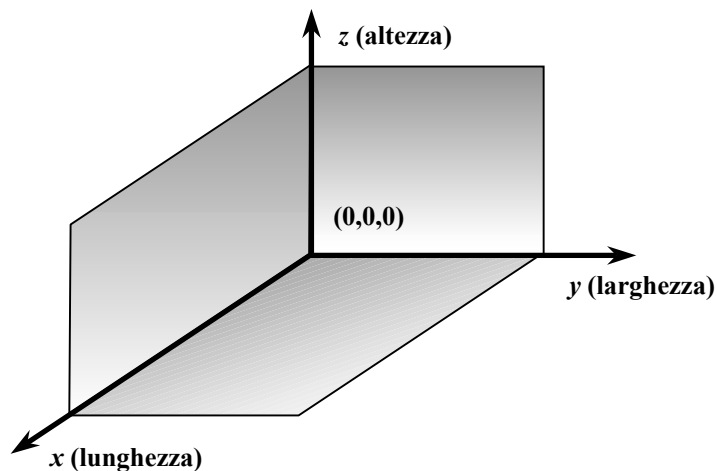


Figura 2.: Sistema di riferimento adottato. L'origine coincide con il vertice in basso a sinistra del *back-end* del container

L'idea di base, quindi, è quella di riempire il container per sezioni (muri) verticali cominciando dal fondo fino ad arrivare al limite del suo lato accessibile. Per quanto riguarda le dimensioni delle sezioni, esse hanno tutte larghezza ed altezza coincidenti con le rispettive dimensioni del container, mentre la profondità è variabile

⁴ Ossia la parete verticale del container opposta rispetto a quella d'accesso

dipendendo dalla metascatola che per prima viene allocata nell'operazione di costruzione del muro. In pratica viene utilizzata la definizione per cui la profondità di ogni muro coincide con la dimensione della metascatola (per tale ragione indicata con l'acronimo di LDB⁵) che, dopo il posizionamento, risulta parallela alla lunghezza del container. Il caricamento avrà termine con la costruzione dell'ultimo muro che è possibile individuare grazie all'utilizzo di una variabile RD⁶ che conserva, aggiornandolo alla fine della costruzione di ogni muro, il valore della profondità residua del container ancora disponibile per l'alloggiamento di ulteriori pacchi.

In realtà, nella definizione di profondità delle varie sezioni verticali, esiste un'eccezione che riguarda proprio l'ultimo muro. La procedura risolutiva infatti, dopo la costruzione del muro *i*-simo, effettua sempre un confronto tra la profondità residua del container e quella che dovrebbe essere la profondità del muro successivo (*i*+1-simo): se dal confronto risulta che c'è ancora spazio disponibile, allora si mantiene il muro *i*-simo e viene avviata la costruzione dell'*i*+1-simo; viceversa, se lo spazio non è sufficiente, allora il muro *i*-simo viene smantellato e la profondità residua del container (qual era prima della edificazione dello stesso muro *i*-simo) viene fissata come profondità dell'ultimo muro. Pertanto, per i primi muri, la profondità è fissata dalla prima metascatola allocata, per l'ultimo è invece fissata dalla profondità residua del container.

Ritornando per un attimo al tema su dimensioni e sistemi di riferimento, è importante notare che tutti gli oggetti considerati, siano essi muri, scatole, metascatole o spazi da allocare, avendo in comune la medesima forma geometrica (e cioè quella di un parallelepipedo), sono tutti individuati in uno stesso modo e cioè attraverso le coordinate del loro vertice più prossimo all'origine del container; in particolare, però, per le metascatole e per le singole scatole il riferimento al sistema cartesiano di figura 2. è indiretto nel senso che, almeno in un primo momento, la loro posizione non è definita rispetto all'origine del container, bensì rispetto a quella del proprio muro d'appartenenza (ossia rispetto al punto del muro definito con la sigla LLBC⁷ che serve per individuarne, a sua volta, la posizione).

Entrando ancor di più nel merito delle modalità di realizzazione di un singolo muro, va detto che il principio fondamentale al quale si fa riferimento è quello che, da un lato individua degli spazi da colmare, mentre dall'altro ricerca, all'interno del sotto-set di scatole ancora da caricare, quelle tipologie che meglio si adattano al loro riempimento. Il procedimento è pertanto di tipo iterativo. La generica iterazione è infatti innescata al piazzamento di ogni metascatola, operazione questa che ha due importanti conseguenze: la prima è che si aggiorna la lista delle scatole non ancora caricate, la seconda è che lo spazio ove è stata effettuata l'allocazione risulta partizionato in spazi più piccoli. Prima però di considerare tale frammentazione, è opportuno distinguere il caso della collocazione della prima metascatola del muro da quella di una qualsiasi altra metascatola utilizzata per il suo completamento. Infatti nel primo caso lo spazio risulta suddiviso in tre sottospazi e cioè uno occupato, che è quello dove è sistemata la metascatola, e altri due liberi situati superiormente e lateralmente alla metascatola stessa. Nel secondo caso, invece, i sottospazi generati sono quattro: uno occupato dalla metascatola e altri tre liberi, vale a dire i due del caso precedente più un terzo che si sviluppa frontalmente alla metascatola fino a raggiungere il limite della profondità del muro (e che in effetti non può esistere nel primo caso essendo proprio la prima metascatola del muro a fissarne la profondità).

Ad ogni modo, ciascuno degli spazi via via prodotti è rappresentato con un vettore riga le cui componenti ne contengono le informazioni relative alle dimensioni, alle coordinate e, in alcuni frangenti, relative al volume. Inoltre essi sono successivamente impilati in una lista del tutto simile a quella prevista dal metodo GMM, gestita con tecnica LIFO⁸ per cui si tenterà sempre di colmare per primo l'ultimo spazio entrato in elenco. Ovviamente si farà in modo che ad entrare per ultimi nella lista siano gli spazi che, secondo prefissati criteri, hanno priorità maggiore nel processo di riempimento. Un tentativo di allocazione di uno spazio può produrre due risultati: un successo o un insuccesso. Nel primo caso ci sarà produzione di ulteriori tre spazi liberi, nel secondo caso lo spazio verrà semplicemente eliminato dalla lista. Il discorso verrà ripreso ed approfondito ulteriormente nei prossimi paragrafi ove si parlerà più dettagliatamente dell'iter seguito per la realizzazione di un muro.

Si preferisce, invece, fare adesso alcune considerazioni riguardo al set di scatole che costituiscono l'ordine di carico. Innanzitutto, come da prassi adottata dagli altri algoritmi analizzati nei capitoli precedenti, sono prese in considerazione solo scatole non deformabili con dimensioni rettangolari. Quindi in

⁵ Acronimo di *Layer Determining Box*

⁶ Acronimo di *Residual depth*

⁷ Acronimo di *Low Left Bottom Corner*

⁸ Acronimo di *Larst In First Out*

Figura 3.: Foglio di calcolo Excel[→] ove sono raccolte le informazioni sull'ordine di carico. Ogni riga

	1	2	3	4	5	6	7	8	9
1		Lungh	Largh	Alt	Vol unit	Disp	Vinc		
2	Tipologia 1	60	50	45	135000	36	1		587
3	Tipologia 2	46	45	43	89010	0	0		233
4	Tipologia 3	98	44	36	155232	0	1		220
5	Tipologia 4	100	89	31	275900	0	1		Dim Container
6	Tipologia 5	108	76	47	385776	12	1		
7	Tipologia 6	75	36	28	75600	0	1		
8	Tipologia 7	108	82	56	495936	0	0		
9	Tipologia 8	45	41	24	44280	16	0		
10	Tipologia 9	103	48	38	187872	0	1		
11	Tipologia 10	74	62	46	211048	0	0		
12	Tipologia 11	96	56	44	236544	20	1		
13	Tipologia 12	117	64	42	314496	0	1		
14	Tipologia 13	73	71	50	259150	0	0		
15	Tipologia 14	61	48	47	137616	0	0		
16	Tipologia 15	91	49	27	120393	8	1		
17	Tipologia 16	78	77	28	168168	0	1		
18	Tipologia 17	119	78	34	315588	0	1		
19	Tipologia 18	106	78	63	520884	40	0		
20	Tipologia 19	82	68	62	345712	0	0		
21	Tipologia 20	100	73	53	386900	0	0		
22						132			
23						Totale			
24						Scatole			
25									
26									

corrisponde ad una tipologia (nell'esempio ne sono considerate 20) rispetto alla quale, nelle varie colonne, sono riportati nell'ordine i seguenti dati: dimensioni della scatola, volume unitario, numero di scatole (numerosità della tipologia), eventuale vincolo sul posizionamento in altezza. Infine, in un riquadro più piccolo sulla destra, vengono inseriti i dati relativi alle dimensioni del container.

un foglio di calcolo Excel[→] (riportato in figura 3.6), insieme alle dimensioni del container dove verranno collocate, vengono inseriti tutti i dati che le concernono, ovvero dimensioni, volume unitario, numerosità ed eventuale presenza di un vincolo sul posizionamento in altezza⁹. Quest'ultimo, in particolare, è un dato molto importante; infatti, essendo la logica dell'algoritmo quella di piazzare le scatole in modo da ottenere la più efficiente utilizzazione possibile del volume del container, è lampante che esso, nella ricerca della collocazione migliore, valuti tutte le loro orientazioni¹⁰. Con la presenza di un vincolo, invece, che obbliga tra l'altro a orientare la relativa scatola in modo che la dimensione vincolata coincida con la sua altezza¹¹, le possibilità delle orientazioni si riducono notevolmente, per l'esattezza a due che sono quelle corrispondenti alle possibili permutazioni delle dimensioni di base. Naturalmente, questo discorso sulle orientazioni fatto per una scatola, rimane assolutamente valido anche per una metascatola se si considera che, per definizione, essa è ottenuta affiancando e sovrapponendo scatole della stessa tipologia ed orientate tutte nello stesso modo.

⁹ Tale informazione è affidata ad un indicatore booleano che è posto a "1" quando il vincolo sussiste, altrimenti a "0"

¹⁰ In assenza di vincoli una scatola può essere collocata in 6 modi possibili ossia due (corrispondenti alle possibili permutazioni delle due dimensioni di base) per ciascuno dei tre criteri (corrispondenti alle 3 dimensioni) con cui si può fissare l'altezza

¹¹ E' difatti per questo che si parla di vincolo sul posizionamento in altezza!

L'ultima rilevante osservazione sulla procedura risolutiva generale dell'algoritmo, riguarda un'operazione di pre-processamento effettuata proprio sulla lista dell'ordine di carico e che, pertanto, precede l'avvio della routine relativa alla costruzione dei muri. Una volta acquisiti i dati dal foglio di calcolo Excel[®],
 Figura 4.: Flowchart illustrante la procedura risolutiva generale proposta dall'algoritmo ibrido-*euristico*

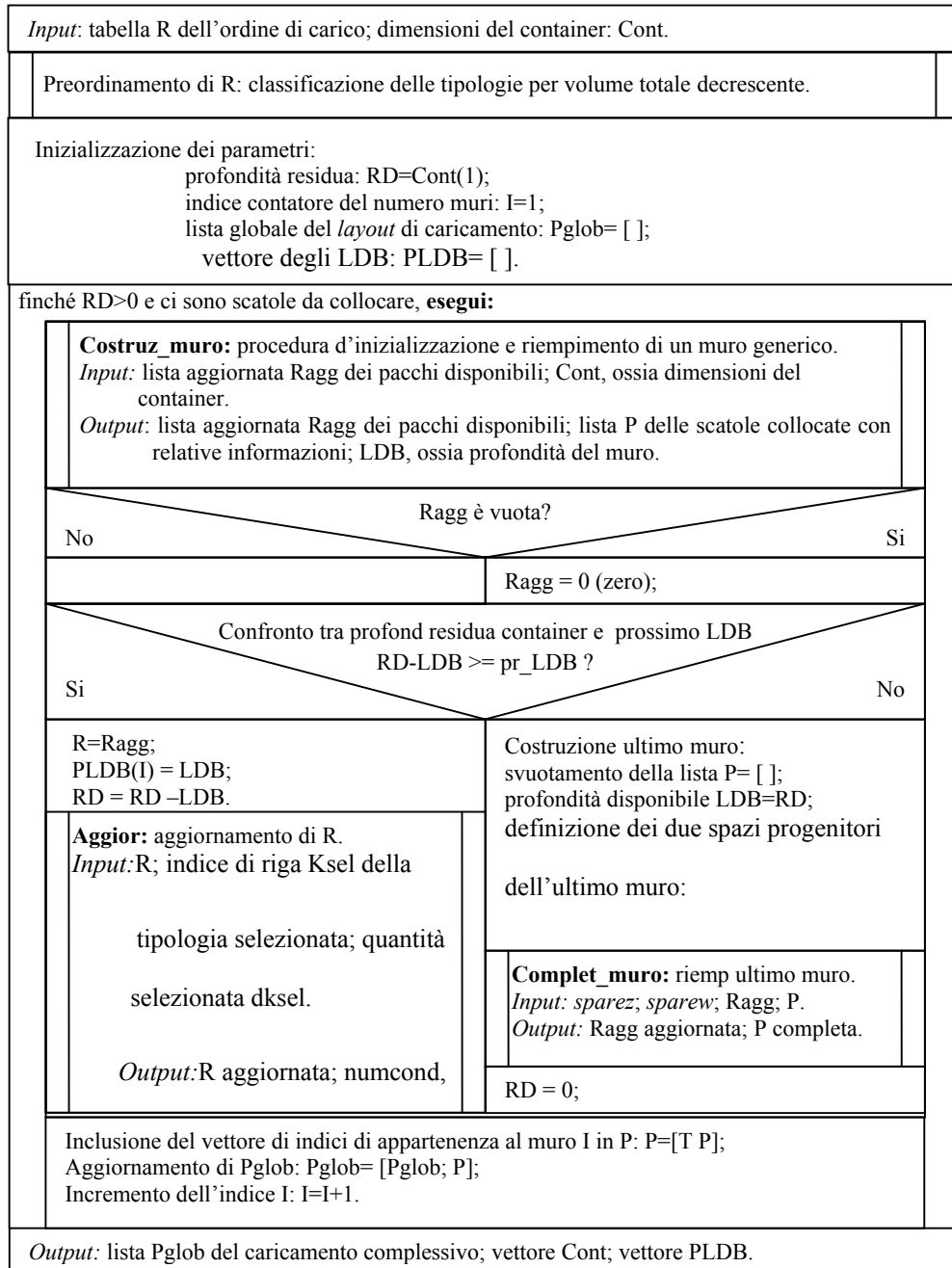


Figura 5.: Flowchart della procedura di costruzione dei muri "Costruz_muro"

Input: lista corrente Ragg delle scatole ancora da caricare; vettore Cont delle dimensioni del container.	
Iniz_muro: procedura d’inizializzazione della costruzione di un muro. Individuazione della I metascatola che fissa la profondità del muro. Input: lista Ragg delle disponibilità residue; vettore Cont delle dimensioni del container. Output: LDB, ossia profondità del muro; Ksel, ossia indice che, in Ragg, individua la tipologia con cui si realizza la I metascatola; dksel, ossia la quantità di scatole impiegate; matrice P contenente l’elenco, una per una, delle scatole impiegate; sparez e sparew, ossia i due spazi progenitori.	
Aggior: procedura d’aggiornamento e controllo della lista Ragg. Input: lista Ragg delle disponibilità; indice di riga Ksel; dksel, ossia scatole da sottrarre dalla lista Ragg. Output: lista Ragg aggiornata; numcond, ossia indicatore booleano che viene settato a “1” se la Ragg si svuotata in seguito all’aggiornamento.	
numcond = =0 ?	
Si	No
Completo_muro: procedura di completamento del riempimento del muro. Input: sparez; sparew; lista Ragg; lista P delle	
Output: lista aggiornata Ragg; lista P di tutte le scatole utilizzate per la realizzazione del muro; LDB, ossia profondità del muro appena costruito.	

essi vengono, infatti, inseriti in una matrice¹² che viene immediatamente elaborata e riorganizzata in modo tale che la prima riga corrisponda alla tipologia a volume totale¹³ maggiore, mentre l’ultima a quella a volume totale minore. In altri termini viene operata una riclassificazione delle tipologie disponibili per volume totale decrescente e, ancora una volta, la ragione che spinge verso questa scelta è da ricercarsi nell’obiettivo di massimizzare la resa volumetrica del caricamento. Il ragionamento che si fa è il seguente: se è vero che è opportuno pensare alla formazione di metascatole¹⁴ e se è vero che un oggetto tanto più è voluminoso, tanto più è difficile sperare di collocarlo andando avanti nel processo di riempimento del container, allora risulta logico cercare di collocare per prima le metascatole più grandi che, verosimilmente, saranno appunto quelle creabili con le scatole della tipologia a volume totale maggiore.

A conclusione del paragrafo, in figura 4., si riporta il flowchart che descrive, abbastanza analiticamente, la procedura risolutiva (del problema del CLP) proposta dall’algoritmo ibrido-euristico. La lettura dei paragrafi successivi aiutano a perfezionarne la comprensione.

1.4 Realizzazione dei muri: procedura “Costruz_muro”

L’algoritmo ibrido-euristico affida l’operazione di costruzione dei muri alla routine “Costruz_muro”, di cui si può apprezzare il flowchart funzionale nella figura 5. Ebbene tale routine, come si evince dallo stesso flowchart, si avvale a sua volta di altre due subroutine ciascuna delle quali si occupa, in pratica, di una delle due fasi attraverso cui si compie la realizzazione di un muro. Il riempimento di una generica sezione verticale del container, in effetti, avviene in due passi successivi:

I passo: inizializzazione del muro attraverso la collocazione di un’opportuna metascatola che ne fissa la profondità. Di questa fase si occupa la subroutine “Iniz_muro”;

Il passo: completamento del muro inizializzato attraverso la collocazione di un’opportuna serie di metascatole che dovranno tentare di colmare al meglio gli spazi generati dalla prima metascatola. Di quest’altra fase si occupa, invece, la subroutine “Completo_muro”. Ora, poiché le due procedure

¹² Denominata R e che ha tante righe quante sono le tipologie in gioco

¹³ Per volume totale di una tipologia s’intende il prodotto del volume unitario della singola scatola per il numero di scatole che la compongono

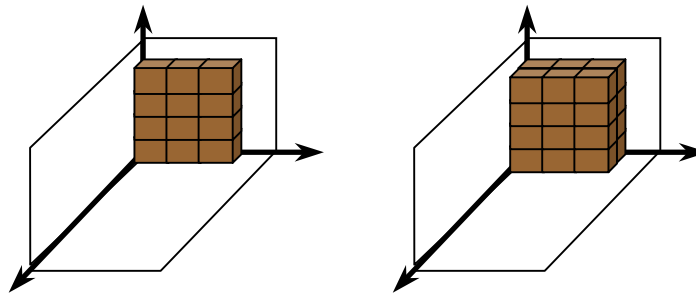
¹⁴ Cfr. par. 2

d'inizializzazione e di completamento di un muro introdotte hanno in comune solo l'obiettivo, che è quello solito dell'occupazione ottimale del volume del container, mentre per il resto agiscono con tecniche e/o regole empiriche piuttosto differenti, si è preferito affrontarne la trattazione separatamente.

1.4.1 Inizializzazione di un muro

Nella fase d'inizializzazione di un muro, poiché non si pongono a priori limitazioni sulle dimensioni della prima metascatola (le cui misure, d'altra parte, dipenderanno dalla numerosità delle scatole ancora da caricare), teoricamente si ha disposizione tutto lo spazio residuo che resta da occupare (nel caso di costruzione del primo muro, addirittura l'intero container!). E' per tale ragione che, in questo primo stadio operativo, si cercherà sempre di piazzare per prima la più grossa metascatola che, in base delle disponibilità delle varie tipologie, si è in grado di realizzare. Pertanto la regola empirica dominante che si utilizza in questa sede, come già accennato precedentemente, è quella che guarda al principio secondo cui tanto più grande è un oggetto,

Figura 6.: Esempio di "unità logiche": a sinistra il caso di una parete (formata da 3 colonne ognuna composta da 4 pacchi), a destra quello di due pareti.



tanto più si abbassano le probabilità di sistemarlo andando avanti nel processo di caricamento.

Detto ciò, è evidente che la procedura d'inizializzazione di un muro creerà la sua metascatola attingendo alla tipologia con volume totale maggiore e che, per il pre-processamento effettuato dall'algoritmo subito dopo l'acquisizione dei dati dal foglio di calcolo Excel, coincide con la prima in testa alla lista di carico. Il problema successivo diventa quindi quello di organizzare le scatole selezionate in modo da ottenere la più grossa metascatola possibile il che è funzione sia delle dimensioni della singola scatola, sia della presenza o meno di un vincolo sul posizionamento in altezza (che vuol dire, rispettivamente un numero minore o maggiore di soluzioni esplorabili), sia infine delle dimensioni dello spazio da colmare¹⁵. Il modo di agire generale prevede che, fissata una dimensione come altezza, si affidi ad un'altra routine ("Permutaz_1") il compito di individuare quale sarà la più grossa *unità logica* realizzabile, scegliendola tra le soluzioni ottenibili dalla permutazione delle due dimensioni di base della scatola.

Per *unità logiche* s'intendono delle metascatole che derivano da un raggruppamento di pacchi che abbia almeno una dimensione finale prossima a quelle dello spazio in questione. In pratica i raggruppamenti di cui si parla sono due: *colonna* e *parete*. La prima è l'insieme del massimo numero di pacchi che si possono disporre uno sopra l'altro fino a colmare l'altezza dello spazio; la seconda è il numero massimo di colonne¹⁶ che possono essere affiancate tra di loro fino a colmare la larghezza dello spazio. Quindi una colonna è colmante rispetto ad una dimensione (altezza), mentre una parete lo è rispetto a due dimensioni (altezza e larghezza). Ebbene per *unità logiche* s'intenderà o un multiplo intero di pareti o un multiplo intero di colonne (tali comunque da non arrivare a formare una parete).

Introdotta queste definizioni, si passa ora in rassegna l'iter risolutivo completo della procedura d'inizializzazione che si sviluppa attraverso le seguenti tappe:

¹⁵ Si ricorda che una generica sezione verticale da colmare ha sempre larghezza ed altezza coincidenti con quelle del container, mentre profondità (prima che venga fissata dal posizionamento della prima metascatola) coincidente con la sua lunghezza residua (RD).

¹⁶ Al limite potrebbe anche essere una!

1. Viene fissata come altezza una delle dimensioni della scatola¹⁷.
2. In base al numero delle scatole disponibili, si verifica anzitutto se si possono costruire unità logiche formate da un numero intero di pareti considerando, allo scopo, le soluzioni offerte dalle due possibili permutazioni delle dimensioni di base. Nell'ipotesi in cui si possono costruire pareti (almeno una) sia ponendo le scatole in un verso, sia ponendole nel verso ottenuto ruotandole, rispetto ad un'asse verticale, di 90°, si preferirà l'orientazione che realizza l'unità logica col maggior numero di scatole (il che vuol dire, in pratica, metascatola più voluminosa). Nella particolare eventualità in cui, con entrambe le orientazioni, si realizzano unità logiche utilizzando lo stesso numero di scatole, allora si opterà per quella che le dispone con la più piccola dimensione (di base) parallela alla lunghezza del container¹⁸. Nel caso in cui, invece, il numero delle scatole è insufficiente per formare anche solo una parete (comunque si orientino le dimensioni di base), allora si passa al punto successivo.
3. Verificato che non c'è il numero di pacchi sufficiente per costruire neppure una parete, la procedura prosegue valutando se esiste la possibilità di costruire unità logiche formate da colonne (almeno una). In caso affermativo, si sceglie di orientare le scatole ponendo, stavolta, parallelamente alla lunghezza del container, la dimensione (di base) maggiore. La giustificazione risiede nella volontà (quando si hanno almeno due colonne) di offrire, agli eventuali pacchi che dovessero posizionarsi su tale metascatola, una superficie d'appoggio con dimensioni più omogenee¹⁹. In caso negativo, quando cioè non si può costruire neppure una colonna, si passa invece al punto successivo.
4. Verificato che non c'è il numero di pacchi sufficiente per costruire né una parete né una colonna, infine la procedura piazzerà semplicemente quelli disponibili (organizzandoli in subcolonne) e lo farà orientandoli così come descritto al punto 2.

Comunque sia, seguendo le istruzioni riportate nei quattro passi precedenti, che nell'insieme costituiscono la procedura "Permutaz_1", rimane individuata una certa metascatola (formata da pareti, da colonne o da subcolonne). A questo punto, se la tipologia cui ci si sta riferendo ha il vincolo sul posizionamento in altezza, allora tale metascatola sarà effettivamente la prima del muro in costruzione, altrimenti si dovrà ripetere per altre due volte (considerando a turno, come altezza, una delle altre due dimensioni) la suddetta procedura. In quest'ultima ipotesi la prima metascatola del muro dovrà essere scelta tra le tre individuate dalla routine "Permutaz_1" (una per ciascuna delle tre volte che viene applicata). Il compito di scegliere spetta alla procedura "Iniz_muro" che, ovviamente, opterà per quella formata dal maggior numero di pacchi.

Con la creazione ed il piazzamento della prima metascatola ha termine la fase di inizializzazione del muro. Infatti, dopo tale operazione, restano individuate completamente le sue caratteristiche e cioè la profondità e due spazi liberi contigui alla metascatola, indicati come sparez e sparew, delle cui tecniche di riempimento (per il completamento del muro) si parlerà nel paragrafo seguente.

¹⁷ Inizialmente si lascia come altezza la dimensione così come riportata nel *datasheet* di Excel, ossia la più piccola delle tre (infatti, al di là delle orientazioni con cui verranno posizionate all'interno del container, lunghezza, larghezza e altezza sono definite rispettivamente come la dimensione maggiore, media e minore della scatola). Tale scelta si effettua in previsione del fatto che, se la tipologia considerata ha il vincolo sul posizionamento in altezza, rimane l'unica opzione possibile.

¹⁸ Si adotta, quindi, un criterio che è conservativo rispetto alla profondità del container. La filosofia è quella di rendere disponibile un volume maggiore per la costruzione dei muri successivi

¹⁹ Ovviamente il criterio evocato è intuitivo e non è quindi detto che corrisponda, sempre, alla soluzione migliore

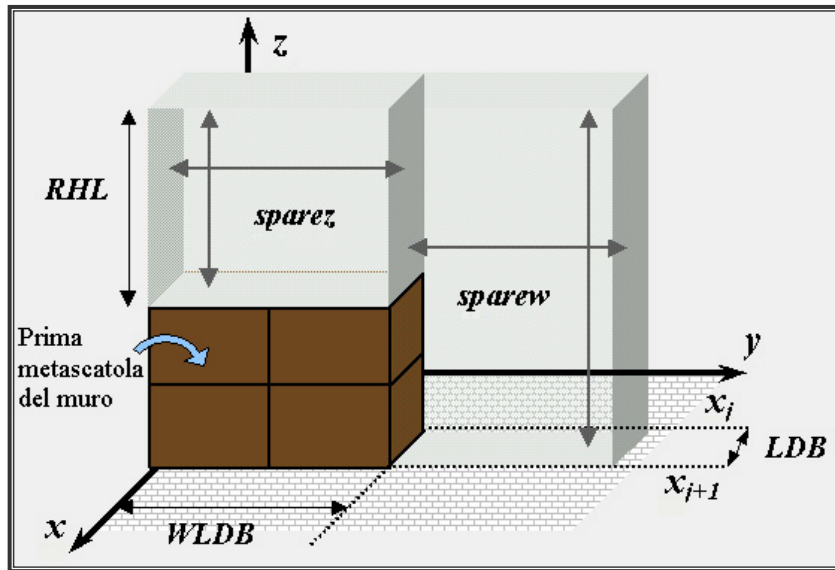


Figura 7.: Spazi progenitori “sparez” e “sparew” generati in seguito al piazzamento della prima metascatola. RHL, WLDB e LDB sono le grandezze attraverso cui tali spazi vengono definiti. In particolare con LDB si indica anche la profondità del muro.

1.4.2 Completamento di un muro

Il completamento della costruzione di un muro si perfeziona con il riempimento degli spazi progenitori sparez e sparew rappresentati in figura 7.6. Indicate con x_j e x_{j+1} le coordinate estreme, relative all’asse x , della profondità del generico muro j -mo, tali spazi sono analiticamente descrivibili come i luoghi dei punti:

$$\text{spare } z = \{(x, y, z) : x_j \leq x \leq x_{j+1}, 0 \leq y \leq \text{WLDB}, \text{HLDB} \leq z \leq \text{Hcont}\}$$

$$\text{spare } w = \{(x, y, z) : x_j \leq x \leq x_{j+1}, \text{WLDB} \leq y \leq \text{Wcont}, 0 \leq z \leq \text{Hcont}\}$$

ove i valori WLDB ed HLDB indicano rispettivamente larghezza ed altezza della metascatola, mentre Hcont e Wcont indicano altezza e larghezza del container. La procedura che l’algoritmo ibrido-euristico deputa all’adempimento di questa seconda fase della costruzione di un muro è la “Completo_muro”. Essa inizia con l’inserire gli spazi sparez e sparew (ovvero due vettori riga che ne contengono le informazioni relative a dimensioni, coordinate e volume) in una lista L, quindi applica una procedura ricorsiva di riempimenti degli spazi che, per costruzione, prima si occupa dello spazio superiore e poi di quello laterale.

Pertanto, complessivamente, sono avviati due cicli, uno per ogni spazio progenitore. Per ciascuno di tali cicli viene dapprima inizializzata una lista temporanea W (contenente inizialmente il solo spazio progenitore considerato, ma cui si aggiungono di volta in volta quelli prodotti dalla collocazione di ogni metascatola), dunque si evoca una subroutine (la “Riemp_spaz”) la cui funzione è quella di individuare, tra tutte le tipologie che hanno ancora pacchi da piazzare, quella che è in grado di formare la metascatola che colmi al meglio lo spazio in esame; spazio che è quello progenitore alla prima iterazione, uno generico della lista W per le iterazioni successive e fino ad esaurimento della lista stessa.

In effetti la collocazione di una metascatoia all'interno di uno spazio generico implica la generazione di ulteriori tre spazi (posti superiormente, frontalmente e lateralmente rispetto alla metascatoia piazzata) le cui

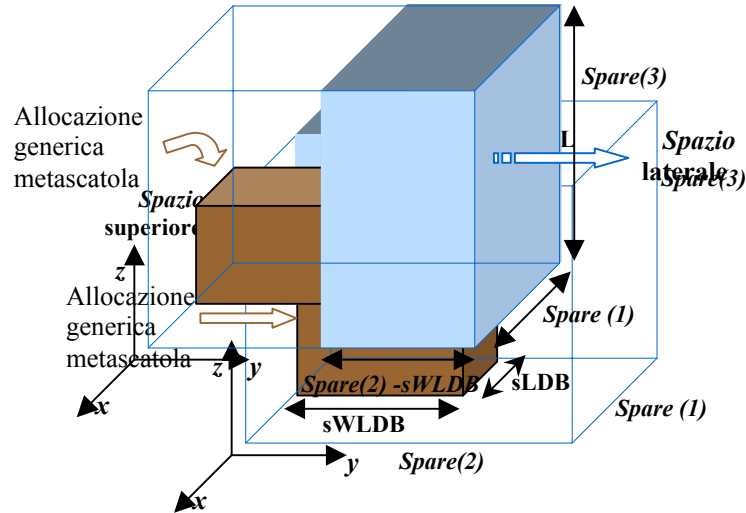


Figura 8.: Spazio superiore sovrastante la metascatoia piazzata per il riempimento di un generico spazio d'indagine (Spare) di dimensioni spare(i), $i = 1, 2, 3$.
caratteristiche geometriche sono rappresentate nelle figg. 8., 9. e 10..

Figura 9.: Spazio generato lateralmente al piazzamento

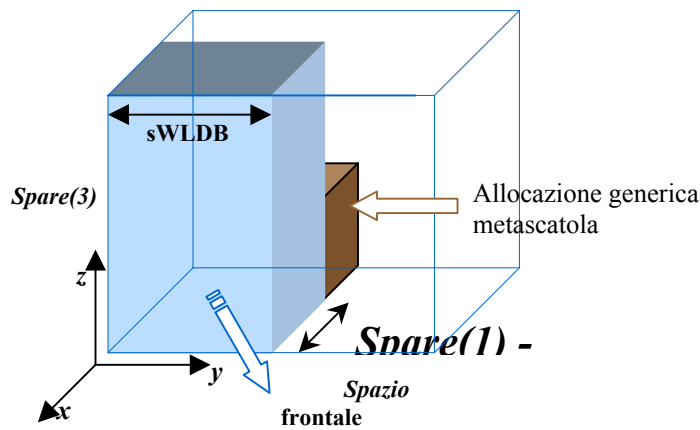


Figura 10.: Spazio generato frontalmente al piazzamento

Ad ogni iterazione tali spazi vengono piazzati in cima alla lista temporanea²⁰ da cui, frattanto, viene anche eliminato lo spazio generante. Il ciclo ha termine quando sono stati indagati tutti gli spazi della lista oppure quando, ad un certo momento, non ci sono più scatole da collocare. L'ultima osservazione riguarda la necessità di stabilire l'ordine (priorità) con il quale devono essere indagati gli spazi presenti in W; ebbene, dopo il piazzamento di ogni metascatoia, la routine prima cercherà di colmare lo spazio superiore quindi gli

²⁰ Gestita con tecnica LIFO

altri due (laterale e frontale) dando la precedenza a quello con volume inferiore. Il ragionamento adottato è il seguente: uno spazio indagato per primo può, ai fini del suo riempimento, contare su una numerosità di scatole (appartenenti alle varie tipologie) superiore rispetto a quella su cui può invece contare uno spazio indagato successivamente e quindi, almeno teoricamente, ha maggiore probabilità di realizzare una migliore performance volumetrica. Con le priorità stabilite, pertanto, si vogliono dare maggiori chance dapprima allo spazio superiore (e ciò corrisponde ad una intenzione di rendere il più stabile possibile l'incolonnamento di metascatole) poi, dei restanti, a quello più piccolo (che avendo intuitivamente più difficoltà di essere colmato rispetto a quello più grande, è giusto che sia quello che, tra i due, possa fare affidamento su un numero di scatole maggiore).

Si entra infine nel merito delle caratteristiche e dei principi euristici su cui si fonda la procedura "Riemp_spaz" che, fissato uno spazio d'indagine preso dalla lista temporanea W (all'interno di uno dei cicli di riempimento degli spazi progenitori), ha il compito di ricercare la metascatola che meglio lo colmi. Rispetto alla procedura ("Iniz_muro") analizzata in occasione della descrizione della fase d'inizializzazione di un muro, e che aveva invece il compito di individuare la prima metascatola in assoluto da collocare al suo interno, la "Riemp_spaz" ha un'analogia ed una differenza fondamentali. La differenza è che mentre la prima ricerca la sua metascatola all'interno di un'unica tipologia²¹, la seconda lo fa investigando all'interno di tutte quelle che, in quel momento, hanno disponibilità non nulla di pacchi. Tale diversità non dovrebbe stupire perché diverse sono anche le esigenze nei due casi d'inizializzazione di un muro e di riempimento di uno spazio generico. Si è infatti già detto che quando si avvia la costruzione di un muro, teoricamente, si può anche sfruttare tutto il volume del container ancora disponibile e ciò giustifica, appunto, il tentativo di piazzare prioritariamente le scatole della tipologia a volume totale maggiore; nel secondo caso, invece, si deve pensare che l'algoritmo è chiamato a riempire spazi di varie grandezze e poiché ogni volta può essere diversa la tipologia che per caratteristiche geometriche meglio si adatta a colmare lo spazio in esame, ogni volta sarà pertanto anche giusto guardare alle soluzioni offerte da tutte le tipologie!

L'aspetto che, invece, accomuna le due procedure è quello di valutare, laddove non esista il vincolo sul posizionamento in altezza, anche le rese volumetriche che si avrebbero considerando tutte le possibili orientazioni delle scatole. In questa ipotesi la procedura "Riemp_spaz" fissa, a turno, come altezza delle scatole una delle loro tre dimensioni e, ogni volta, incarica un'altra routine ("Permutaz_2") di individuare la più grossa unità logica tra quelle ottenibili dalla permutazione delle due dimensioni di base; quindi delle tre soluzioni ottenute (una per ciascuna delle volte che evoca la routine "Permutaz_2") sceglie quella che rappresenta la metascatola più voluminosa.

Fermo restando le definizioni di unità logiche date in occasione della descrizione della fase d'inizializzazione di un muro²², immaginandole però riferite, ogni volta, alle dimensioni dello spazio analizzato piuttosto che a quelle del container, si elencano di seguito i passi attraverso cui si sviluppa la procedura "Permutaz_2":

1. Viene fissata come altezza una delle dimensioni delle scatole²³.
2. Si riorganizza la lista delle tipologie con disponibilità non nulle, per volumi unitari decrescenti²⁴ (la ragione di questa operazione verrà chiarita in seguito).
3. In base al numero delle scatole disponibili, si verifica anzitutto se si possono costruire unità logiche formate da un numero intero di pareti considerando, allo scopo, e per ciascuna tipologia, le soluzioni offerte dalle due possibili permutazioni delle dimensioni di base. Nell'ipotesi in cui esista più di una tipologia in grado di costruire almeno una parete allora si opterà naturalmente per quella che crea la metascatola più grande. Nell'ipotesi in cui, invece, non esista alcuna di siffatte tipologie, si passa al punto successivo.
4. Verificato che non esiste alcuna tipologia con un numero di pacchi sufficiente per costruire almeno una parete, la procedura prosegue valutando se esiste la possibilità di costruire unità logiche formate da

²¹ Ovvero quella a volume totale maggiore al momento dell'inizializzazione del muro in esame

²² Cfr. par. 6.4.1

²³ Cfr. nota n° 17

²⁴ Dopo tale riorganizzazione in testa alla lista ci sarà la tipologia le cui scatole hanno, singolarmente, volume superiore rispetto a quello delle scatole di tutte le altre tipologie

colonne (almeno una). In questo caso, dapprima si cerca una soluzione (metascatola) fissando l'orientazione delle scatole in modo che la dimensione (di base) maggiore sia parallela alla lunghezza del container²⁵, e solo nell'evenienza di un insuccesso, si passa a cercarne una dopo aver cambiato l'orientazione delle scatole. In entrambe le ipotesi, tra eventuali più tipologie rispondenti ai requisiti richiesti, si sceglie quella più *performante*. Se, invece, non si può costruire neppure una colonna in nessun modo e con nessuna tipologia, si passa al punto successivo.

5. Verificato che nessuna tipologia possiede un numero di pacchi sufficiente per costruire né una parete né una colonna, infine la procedura sceglie comunque quella in grado di realizzare la metascatola (che sarà una subcolonna o al limite un'unica scatola) più grande. Riguardo alle orientazioni delle dimensioni di base dei pacchi, valgono le stesse considerazioni fatte nel punto precedente.
6. Per le tipologie non soggette a vincolo, si cambia la dimensione fissata come altezza e si ripetono le operazioni dei punti 3,4 e 5 (quelle che, realmente, rappresentano la procedura "Permutaz_2").

Qualunque sia il tipo di metascatola (insieme di una o più pareti, insieme di una o più colonne, subcolonna) che la procedura ricerca, può sempre accadere che quella più voluminosa possa essere ottenuta contemporaneamente da due (o più!) tipologie. In questa poco probabile, ma pur sempre possibile, evenienza viene in soccorso il più volte evocato principio che suggerisce di collocare per primi gli oggetti più ingombranti, per cui, alla fine, si preferirà la tipologia che possiede le scatole a volume unitario maggiore. Per fare un esempio: se una tipologia T1 forma una metascatola di volume V utilizzando due scatole, mentre un'altra tipologia T2 forma una metascatola, diversa dalla prima ma sempre di volume V, utilizzando però dieci scatole, allora l'algoritmo sceglierebbe la metascatola creata da T1 perché, evidentemente, le sue singole scatole sono più voluminose di quelle di T2!

Si può ora capire lo scopo della riorganizzazione delle tipologie prevista dal punto 2 della *routine*: consentire in ogni momento all'algoritmo di individuare più facilmente la tipologia che contempla i pacchi a maggior volume unitario. L'individuazione avverrebbe, infatti, guardando semplicemente alla testa della lista.

Il paragrafo, così come la trattazione dell'intero metodo euristico proposto, si conclude con due osservazioni. La prima è che dopo la costruzione di un muro viene naturalmente aggiornato l'elenco delle scatole ancora da caricare e poiché, verosimilmente, la tipologia a volume totale maggiore non sarà più la prima in cima alla lista, in previsione dell'inizializzazione del muro successivo viene rieffettuata l'operazione di ordinamento per volumi totali decrescenti.

La seconda osservazione riprende un'affermazione fatta nel terzo paragrafo e cioè che la profondità dell'ultimo muro viene fissata, dopo il controllo di una certa condizione, da quella residua del container. Ciò ha un'importante conseguenza ai fini dell'iter da seguire per il suo riempimento. In effetti, in questo caso, non esiste una fase d'inizializzazione ma viene direttamente applicata quella di completamento: è come se si avesse uno spazio progenitore nullo (per esempio lo *sparez*) ed un altro (per esempio lo *sparew*) coincidente con l'ultima sezione verticale da colmare del container. Tale particolarità può anche essere apprezzata nel *flowchart* di figura 4.6 ove risulta collocata all'interno della strategia risolutiva generale dell'algoritmo ibrido- euristico.

²⁵ La giustificazione risiede nella volontà (quando si hanno almeno due colonne) di offrire, agli eventuali pacchi che dovessero posizionarsi su tale metascatola, una superficie d'appoggio con dimensioni più omogenee.

Applications of nonlinear dynamics to communications security

A. Șerbanescu, P. Ciotirnae, D. Andrei

Military Technical Academy Bucharest, Romania
Bd. G. Coșbuc 81-83 sector 5 Bucharest, Romania
Phone: +4021.335.46.60 ext.148

E-mail: serbal@mta.ro, ciotirnae@mta.ro, dandrei@mta.ro

Abstract. The fact that numerical sequences with chaotic behavior are very suitable for use in secure communications is proved. More improved results in use of them are found. An efficient data chaotic encryption algorithm and its hardware architecture design are proposed. Based on a chaotic system, an unpredictable binary sequence is generated and used to encrypt and decrypt the data message. The simulation results show that the proposed algorithm possesses the features of data security and simplicity in the required operations. The method possesses the desirable properties for high security data and voice-multiplexed transmissions. Moreover, the encrypted complex data signal is using a multi route through the network in order to increase the data security.

1. INTRODUCTION

In recent years there are several international centers of studies which are interested in solving the problem of a suitable use of nonlinear dynamical systems with chaotic behavior in secure communication scheme.

The basic ideas can be classified into three major groups: *individual data channel encryption*, *multiplexed data channel encryption* and *their combining form*. The *individual data channel encryption* algorithms scramble the original information and form a complex signal which is sent through the public or insecure channel. On the other hand, the *inverse transformation* algorithms transform the encrypted data value of the original signal. For example, a voice channel is partitioned into subbands and scrambled to transform it into an unintelligible noiselike signal. In this paper, a new chaotic encryption algorithm is proposed and its architecture is minutely described.

2. METHOD FOR CRYPTOGRAPHY WITH CHAOTICAL SIGNAL

Consider a chaotic system described by the Cauchy problem for

$$\dot{x} = Ax + bf(x) + c \quad (1)$$

where: $x \in \mathfrak{R}^n$, $A \in \mathfrak{R}^{n \times n}$, $b \in \mathfrak{R}^{n \times 1}$, $c \in \mathfrak{R}^{n \times 1}$ and $f : \mathfrak{R}^n \rightarrow \mathfrak{R}^n$

The transmitting part

For the information signal $i(t)$, the crypto signal is

$$s(t) = s(p(t) \cdot K(t)) \quad (2)$$

where $s(\cdot)$ is a generic encryption function and $K(t)$ in the key.

If a symmetric algorithm is adopted the information signal is decrypted as:

$$i(t) = d(s(t) \cdot K(t)) \quad (3)$$

where $d(\cdot)$ is the decryption function.

For the key we assume that:

$$K(t) = K(x(t)) \quad (4)$$

The resulting signal occurs in an equation, generating a dynamical system, described by

$$\dot{x} = Ax + bf(x) + c + bs(t) \quad (5)$$

The receiving part

For the receiver the most important objective is the **synchronization**. This goal is achieved by designing the decrypter as a nonlinear observer for the state of the encrypter.

The decrypter also occurs in a dynamical system described by

$$\dot{y} = Ay + bf(y) + c + g(h = r(y)) \quad (6)$$

where: $g : \mathfrak{R} \rightarrow \mathfrak{R}^n$ is a suitable chosen nonlinear function.

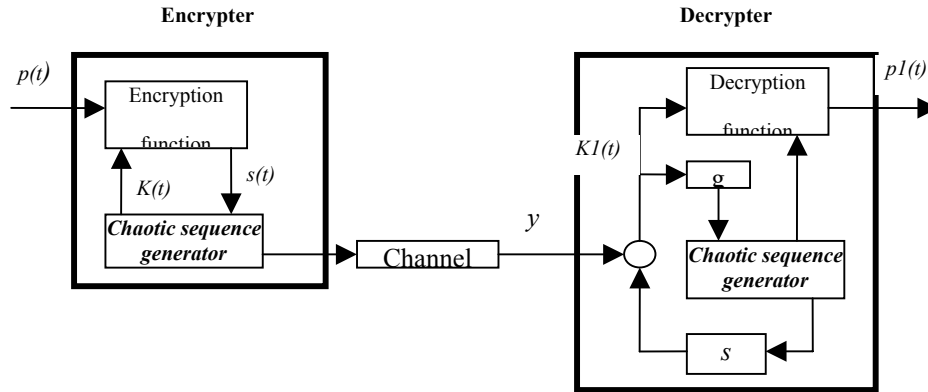


Fig. 1. Bloc representation for encryption method.

We must design the eq. (6) so that y converges to the state of the x as $t \rightarrow \infty$. We define the error signal as

$$e(t) = \|y(t) - x(t)\| \quad (7)$$

Therefore our intention is to have $e(t) \rightarrow 0$ as $t \rightarrow \infty$. If this is achieved for any initial condition $y(0), x(0)$ we say that the synchronization error system has a globally asymptotically stable equilibrium point for $e = 0$.

A graphical representation of the method proposed is presented in fig. 1.

Here is the proof for the assertion that the receiving signal is extract from coded signal with the $K1(t)$ key.

$$K1(t) = K(y(t)) \quad (8)$$

Let $s1(t) = h - r(y)$ be the crypto signal retrieved by the decrypter and let

$$i1(t) = d(s1(t), K1(t)) \text{ be the information signal obtained by the receiver.}$$

If (6) is a global observer of the (5), it follows

$$s1(t) \rightarrow s(t) \quad (9)$$

because $y \rightarrow x$ as $t \rightarrow \infty$ for any initial condition.

As consequence $K(y(t)) \rightarrow K(x(t))$ so $K1(t) \rightarrow K(t)$ and the $s1(t) \rightarrow s(t)$. Finally, the comparison between (3) and (8) confirm that the assertion (9) is correct.

3. DATA TRANSMISSIONS ON EUROCOM MULTIPLE CHANNELS

Necessary conditions for military data transmissions are the assurances of integrity and protection of data. For these goals classical channel methods for protection can be used. Instead of these we use a hybrid method witch allows classical methods and involves new ones.

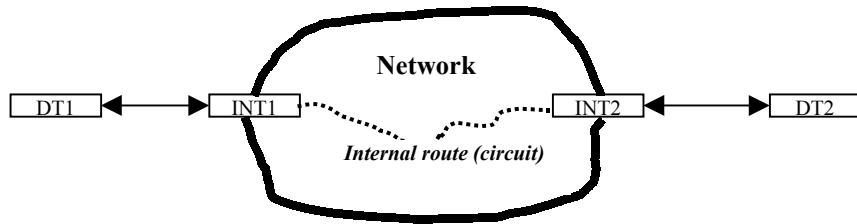


Fig. 2. Data transmission on single circuit.

In fig. 2. a classical transmission between two digital terminals (DT) through specific interface (INT) is described. On demand the network creates a single internal route. This route is exposed to jamming and interception.

We use a method which involves a multiple route. It is achieved a high degree of robustness against jamming and interception. For this it is necessary to create a special interface (M.R. int) which is capable to split the original source of information and to provide the network interface with a set of a multiple independent source of data.

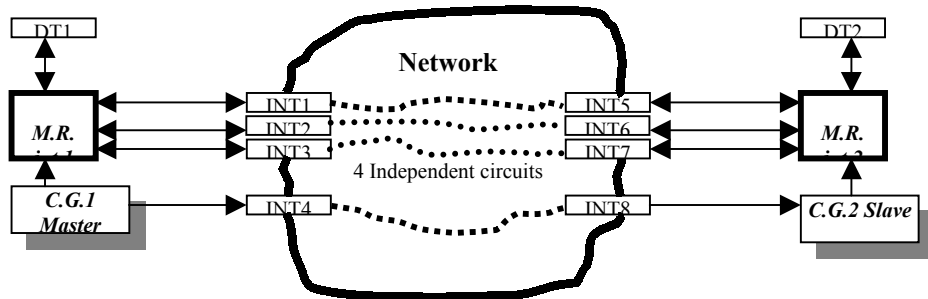


Fig. 3. Data transmission on multiple circuits.

The method is presented in fig. 3. In order to increase the “power” of the method splitting the message a chaotic numerical signal generators (C.G.) are involved.

The chaotic sequence is generated externally. In addition it is sent to the receiver through a separate channel (route).

The procedure of splitting the original message is represented in fig. 4. The method uses a chaotic sequence in order to determine the number of bits to route to the first circuit and so on.

The chaotic sequence is converted in hex numbers grouping sets of 4 bits, and it is validate if the number obtained is $7 \leq x \leq 13$. The receiver creates the number in the same manner.

Example: Let be the chaotical sequence to
 0101 0101 1100 0110 1010 1101 0110 0101 1001 1001 0101 0110
 Hex conversion is:
 -05- -05- -12- -06- -10- -13- -06- -05- -09- -09- -05- -06-

After validation with conditions $7 \leq x \leq 13$ we obtain a random sequence in the domain [7,13].

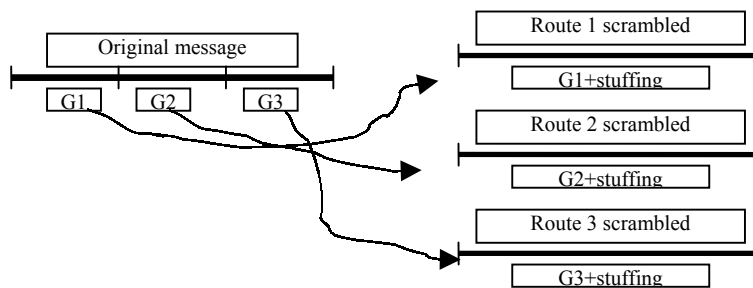


Fig. 4. Splitting the information on multiple circuits.

4. COMPARISON BETWEEN CLASSICAL AND CHAOTIC CRYPTOGRAPHY

Classical cryptography works on discrete values and in discrete time while in chaotic cryptography continuous-value systems that may operate in continuous or discrete time are used (Table 1).

<i>Classical cryptography</i>	<i>Chaotic cryptography</i>
- integer values on finite fields	- continuous values using fixed or floating point representation
- algebraic methods	- analytic methods
- digital realization by integer arithmetic	- digital realization by non integer arithmetic

Table 1. Comparison between classical and chaotic cryptography.

Chaotic maps and cryptographic algorithms (or more generally maps defined on finite fields) have also some similar properties: sensitivity to initial conditions and parameters, random like behavior and unstable orbits with long periods, depending upon the precision of the numerical implementation.

For example a block encryption algorithm can be re-written as a discrete time dynamical system

$$x_{n+1} = F(x_n) \quad (10)$$

where the initial condition x_0 is the plain text to be encrypted, and the final state x_k is the cipher text. Due to the fact that the chaotic map can be expressed by a similar equation, the property of the map being chaotic implies spreading out the influence of a single plaintext digit over many cipher text digits. In other words, any set of initial conditions of a chaotic map will eventually spread over the whole phase space as the system evolves.

5. TYPES OF CRYPTOGRAPHIC ALGORITHMS OR CIPHERS

The purpose of a cipher is to take unencrypted data (the plaintext) in order to produce an encrypted version of it called the cipher text. There are one major classes of ciphers: stream ciphers and block ciphers.

5.1. Chaotic stream cipher

The simplest example of a stream cipher is a bit of data is “xor-ed” with another bit of a stream generated by a pseudorandom generator. To decrypt the message the same pseudorandom bit stream is generated and the inverse operation (another “xor”) is done.

Our chaotic stream cipher is based on a chaotic pseudorandom generator with the main part represented by a discrete time chaotic map known as “Piecewise Affine Markov Map” (PWAM).

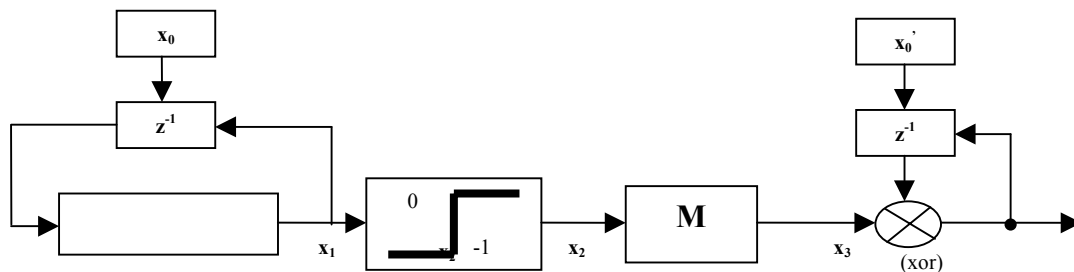


Fig. 6. Chaotic pseudorandom generator.

The chaotic map is defined by the following expression:

$$f_{M1}(x) = \begin{cases} B(D - |x|) & , |x| < D \\ B(|x| - 2D) & , |x| \geq D \end{cases} \quad (11)$$

where $B=3, D=1, x_0=0.1$.

The above function is defined on the real number domain such that the initial condition (x_0) and the output variable (x_i) belong to the interval $(-3,3)$. Keeping in mind the fact that the above chaotic map must be implemented on a finite precision machine we represent the real numbers x_0, x_i in floating point single precision arithmetic (sign 1 bit, exponent 8 bits, mantissa 23 bits).

The second block stands for a selection logic so that only the values that lie in the subinterval $(-1,0)$ are selected. This must be done in order to select a subinterval with a good distribution of generated values by the PWAM, because some of the chaotic properties are lost after imposing the precision of the numerical representation.

M is a mapping block described by the equation

$$x_3 = \left[255 \cdot \frac{x_2 - a}{b - a} \right], \quad (12)$$

where $[]$ stands for integer part function;

$a=-1, b=0$;

$x_0', x_3 \in \{0, 1, \dots, 255\}$ are integer numbers represented on 8 bits, $x_0'=255$.

Thus each of the single precision real numbers (x_2) is transformed into an 8 bit integer in order to generate the pseudorandom bit stream. The last transformation is applied on 8 bits once to ensure good statistical properties of the generated bit stream.

The encryption key is represented in our implementation by the initial condition of PWAM x_0 (32 bits) and the initial condition of the final “xor” operation x_0' (8 bits), that is a 40 bits key length, is used. As an example we present here the results obtained with the frequency test and the discrete Fourier transform test for a stream of 80000 bits length (figs. 7, and 8).

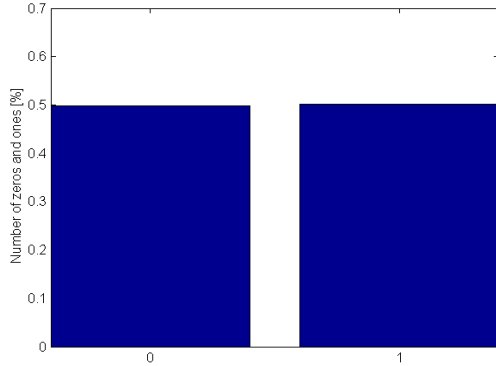


Fig. 7. Frequency test

(nb. of zeros: 39890; nb. of ones: 40110)

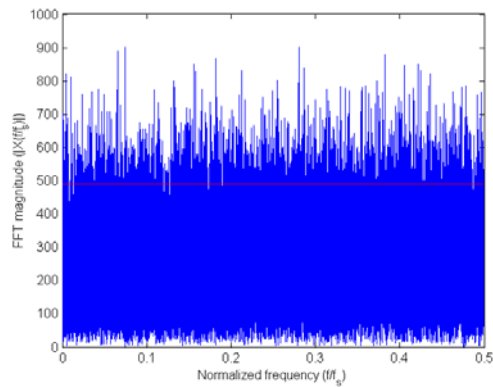


Fig. 8. Discrete Fourier transform test.

We have done several statistical tests and our generator has passed all such that we have concluded that the bit stream generated with the above structure has very good statistical properties and therefore it can be used in a stream cipher.

5.2 Chaotic block cipher (Generalized Chaotic Encryption Method)

A block cipher operates on blocks of data that are larger than one bit.

Here we present a chaos based secret key algorithm using the simple one-dimensional logistic map:

$$x_{n+1} = r \cdot x_n \cdot (1 - x_n) \quad (13)$$

where $x_n \in (0,1)$ and $r \in (3.7,4)$ is the control parameter chosen such that (13) has a chaotic behavior.

The message to be transmitted is a text composed by some alphabet and we associate portions (e-intervals) of the attractor ((0, 1) domain) with alphabet characters. The cipher text of some character (plaintext) is the number of iterations applied in equation (13) to make its trajectory, departing from an initial condition x_0 , reaches an e-interval associated with that character. If the attractor is divided into S intervals or sets, where S is also the character number of the alphabet, then each e-interval has the form

$$I_n = [x_{\min} + (n - 1) \cdot e, x_{\min} + n \cdot e] \tag{14}$$

where $n \in \{1, 2, \dots, S\}$, $e = (x_{\max} - x_{\min})/S$ and $[x_{\min}, x_{\max}]$ represents the whole attractor or a portion of it.

As far as the number of iterations needed to reach a certain interval I_n is concerning we have numerically checked that this number can become quite large for certain initial conditions. Thus, in order to ensure a reasonably number of iterations in any situation, we have decided to divide first the whole attractor into f groups and then to divide each group into S e-intervals (fig. 9).

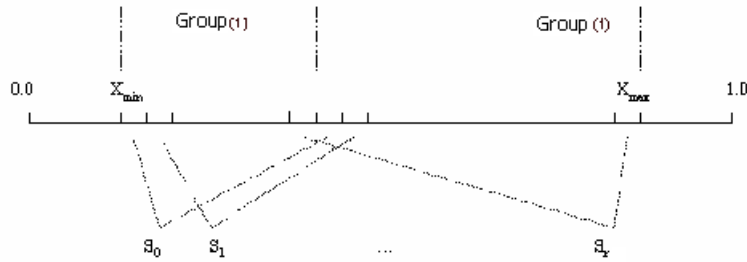


Fig. 9. Generalized association method.

A logic diagram of the above described generalized chaotic encryption method is given in fig. 10.

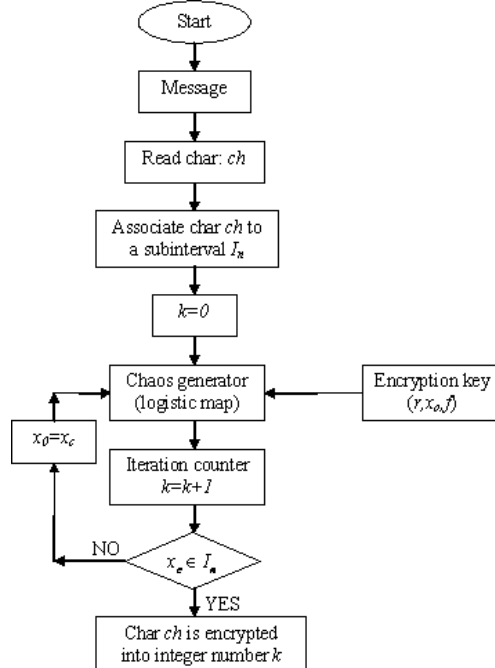


Fig. 10. Generalized chaotic encryption method.

To decrypt the message, at the receiver side equation, (4) is iterated with the same initial condition as much times as indicated by the cipher text.

6. CONCLUSIONS

In this paper, an efficient algorithm and its architecture have been proposed. The associated architecture possesses the following desirable features: high level of security, real-time processing capability, medium

hardware cost, and low power consumption. Therefore, it is very suitable for military and private network implementation. Besides, the simulation results have indicated that the algorithm can easily make images to be in a chaotic state very soon.

The estimated processing speed achieves 2MHz, which can meet the requirement of first order real-time data/voice multiplexed encryption applications.

REFERENCES

- [1] Baranovski, A.L., Daems, D., Design of 1-D chaotic maps with prescribed statistical properties. *Int. Journal of Bifurcations and Chaos*, **5**, 6 (1995), 1585-1598
- [2] Blum, L, Schub M., A simple unpredictable pseudo-random number generator, *SIAM Journal of Computing*. **15**, 2 (1986), 364-383.
- [3] Cernák, J. Digital generators of chaos. *Physics Letters A*. **214**, (1996), 151-160.
- [4] Couture, R and L'Ecuyer, P., Distribution properties of multiply-with-carry random number generators, *Mathematics of Computation*, **66** (1997), 591.
- [5] Couture, R and L'Ecuyer, P., Guest Editors' Introduction. *ACM transactions on Modeling and Computer Simulation*, **8**, 1 (1998), 1-2.
- [6] Entacher, Karl., Bad Subsequences of well-known linear congruential pseudorandom number generators. *ACM transactions on Modeling and Computer Simulation*, **8**, 1 (1998), 61-70.
- [7] W. J. Rugh., *Nonlinear System Theory, TheVolterra/Wiener Approach*. The Johns Hopkins University Press, 1981.
- [8] D. D. Weiner, J. F. Spina., *Sinusoidal analysis and modeling of weakly nonlinear circuits*, Van Nostrand Reinhold, 1980.
- [9] A. Abel, A. Bauer, K. Kelber, and W. Schwarz., Chaotic codes for CDMA applications, in *Proc. ECCTD*, vol. 1, Budapest, 1997. 306–311
- [10] A. Abel, M. Götzt, and W. Schwarz. Statistical analysis of chaotic communication schemes,. in *Proc. ISCAS*, Monterey, 1998.
- [11] S. S. Haykin., *Digital communications*. John Wiley & Sons, New York, 1988.
- [12] M. K. Simon, J. K. Omura, R. A. Scholtz, and B. K. Levitt, *Spread spectrum communications handbook*, McGraw-Hill, New York, 1994.

Hydrodynamics of dispersed, multiphase flows by wave breaking

Ovidiu Tanasescu
SDB Bucuresti

Horatiu Tanasescu
ICEPRONAV Galati

Abstract. Our days CFD is routinely used in a huge variety of applications. However, there are some areas in which CFD has serious limitations and active research is necessary to overcome them. Such a great limitation of CFD is represented by the dispersed, multiphase flows. Multiphase flows are commonly met in practice, and consequently their simulations is of great interest. In the same way like turbulent flows, multiphase flows (which may also be turbulent in one or more phases) can correspond to solutions of the equations of motion, and direct numerical simulation can be applied to them. However, in many practical problems, multiphase flows require additional special modeling. Still these additional models, tend to ignore or simplify a lot of important details of the flow, such as droplet or particle shape and their impact on interphase mass, momentum and energy transport. Our practical target is ship hydrodynamics wave breaking, tackled with very recently in the scientific world. Physically speaking bow breaking waves dissipate energy and affect forward hydrodynamic resistance of ship hull as a whole. Subsequent spray formation and air entrainment can be important for ship wake configuration. Additionally, from the safety point of view, breaking ocean waves interacting with ships are of main importance for ship capsizing. The global theme aims are: nonlinear three-dimensional water waves, wind-wave-turbulence and ocean waves – ship hull wave system interaction. Some subareas to be considered are: large amplitude and breaking waves, deterministic versus statistical models, stability of water waves, the role of turbulence in ocean dynamics and the implications of forcing and dissipation. The concerns of the present paper, which is the first one in a series, are focusing on:

- a 3D Galerkin weighted residuals finite element method for surface ships wave making resistance computation;
- experimental-analytical-numerical genesis and dynamics modeling of bow wave breaking influence on surface ships wave making resistance.

1. INTRODUCTION

Waves are a source of delight. They also cause and produce enormous destructions and drawbacks. We need to understand how they form and how they propagate, and find ways to harness their energy safely. Mathematic holds the key to this understanding. There are fascinating mathematical problems associated with water waves of great interest for both pure and applied mathematicians, and the water wave equations have spreaded and amplified everywhere in mathematics, as for example the D. J. Korteweg and G. de Vries equation (1895): if u is the particle displacement and the Ox is the axis in travelling direction, we have

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = -b \frac{\partial^3 u}{\partial x^3}$$

where t is the time, and b is a positive constant. Of all various types of wave motion that occur naturally (created by the wind), artificially (generated by structure-water interaction) or both, surface water waves are not merely the most easily observed but of great scientific importance because of their special impact in ship hydrodynamics and not only. As we already mentioned in the abstract, the final theoretical target of this long cycle of papers, is hydrodynamics of dispersed, multiphase flows by wave breaking, with the practical

application in ship hydrodynamics: the influence of bow waves on wave resistance and turbulent wakes, the action of breaking waves on moving ships, not to mention the capsizing of ships in heavy seas.

The wave resistance of surface ships is simulated by using field finite element method – Galerkin formulation. A theoretical – physical, analytical and numerical – three-dimensional model for free surface potential flow around surface ship hulls is considered. The spatial flow domain unstructured discretisation is performed by using curved boundaries hexahedron finite elements having twenty nodes, between parallel planes with the ship's central longitudinal plane). The special characteristic of the method is, in the authors opinion, that in considering the whole bulk of the fluid, the naval architecture potential flow problems could be discretised much more accurately. Moreover, having in view future developments, (unlike the boundary element method – Rankine sources – presently used for wave resistance computation almost everywhere in the world), the method described in this paper is not limited to the case of the potential flow, but it may be modified such that to include viscosity, in this way having also the possibility to take into account its influence on stern wave making. Ship bow breaking waves are investigated too.

The complex, competitive, rational process of selecting of a hull form that which is the best of all possible designs within a prescribed objective function (minimum forward hydrodynamic resistance) and a given set of constraints (of geometrical nature and practical design) is technically challenging. Until recently, the only way to evaluate ship's resistance performance was through experiment. The past ten years have, however, seen major achievements in a good numerical free surface potential flow prediction, inviscid flow models remaining the most important for naval architecture despite the recently increased abilities to predict the viscous flow in the boundary layer and wake. Application to inviscid flows are wave resistance computations, seakeeping, manoeuvring and propeller flows. From all applications to inviscid flows, the present work is focusing on the wave resistance problem only.

2. PHYSICAL AND ANALYTICAL MODEL

Let us consider a ship hull model, piercing the free surface, moving steady horizontally in still water of infinite depth with a constant velocity U_∞ (upstream). We formulate the wave resistance problem in a Cartesian coordinate system fixed to the ship model (time-independent flow). The X-Y plane is at the design draft, the X axis is positive toward the stern and the vertical axis Z is positive upward.

Assumptions:

- fluid is ideal (inviscid);
- fluid (water) is considered incompressible;
- motion is irrotational;

Further down will try that all conditions of the physical model to be converted into analytical equations if possible.

By virtue of above mentioned assumptions, it is convenient to introduce the potential function $\phi(X, Y, Z)$, or velocity potential, such that, $\vec{U} = \text{grad}\phi = \nabla\phi$. ϕ 's partial derivative in any direction gives the velocity component in that direction, $\phi_X = u$, $\phi_Y = v$, $\phi_Z = w$. The continuity equation (conservation of mass) for the steady flow of an incompressible fluid reads $\text{div}\vec{U} = \nabla \cdot \vec{U} = 0$ in the case of potential flow becomes $\text{div}U = \nabla \cdot \nabla\phi = \nabla^2\phi = \Delta\phi(X, Y, Z) = 0$, i.e. the Laplace's equation. Thus the potential ϕ satisfies the Laplace equation (i.e. it's a harmonic function). This is an elliptic partial differential equation. In view of the existence and uniqueness of its solution it must be supplemented with boundary conditions at all boundaries of the computational domain (the wetted hull surface, the water free surface and at infinity). Laplace equation together with the boundary conditions determine completely the motion.

2.1 Boundary conditions for the completely velocity potential ϕ

- on wetted hull surface: water does not penetrate the hull (Neumann condition), $n \cdot U = n \cdot \nabla\phi = 0$ (n is the unit vector normal to the hull) or

$$\phi_n = 0$$

- on the water free surface; physical nature of the water free surface requires two boundary conditions:

a. kinematical condition: water does not penetrate the water surface; thus at the surface $Z=\zeta$,

$$\nabla\phi \cdot \nabla\zeta = \phi_z \quad (1)$$

or

$$\phi_x \cdot \zeta_x + \phi_y \cdot \zeta_y - \phi_z = 0; \quad (2)$$

b. dynamical condition: pressure on the water free surface must be atmospheric and independent of the position on it,

$$\frac{1}{2}(\nabla\phi)^2 + g \cdot \zeta = \frac{1}{2}U_\infty^2, \quad (3)$$

or

$$g\zeta + \frac{1}{2}(\nabla\phi \cdot \nabla\phi) - U_\infty^2 = 0; \quad (4)$$

- decay (or regularity) condition: far away from the ship model, the flow is unperturbed and parallel, having a speed equal to ship model speed U_∞ , i.e.,

$$\lim_{(X^2+Y^2+Z^2) \rightarrow \infty} \nabla\phi = (U_\infty, 0, 0);$$

- radiation condition: waves generated by the ship propagate only downstream (no upstream waves); this condition is introduced by the numerical method requiring a special treatment;

- equilibrium condition: the ship hull must be in equilibrium.

Combining of kinematical and dynamical boundary conditions (the equations (1) and (3)) and eliminating the unknown wave elevation ζ we are lead to the following non-linear free surface boundary condition,

$$\frac{1}{2}\nabla\phi \cdot \nabla(\nabla\phi)^2 + g\phi_z = 0$$

The exact problem formulated above is non-linear, since the free surface boundary condition itself is non-linear and should be exactly satisfied on the wavy surface $Z=\zeta(X,Y)$, which is unknown, and must be computed as a part of the solution. Thus, numerical methods, which have been applied to solve the problem usually entail some kind of linearization procedure.

The unknown pressures p on the hull and wavy surface $Z=\zeta(X,Y)$ will generate a potential ϕ and a wave elevation ζ , which fulfill the boundary conditions (2) and (4). Thus, for a mathematical consistent linearization we can introduce the following functions:

$$K(p, \zeta) = \phi_x \cdot \zeta_x + \phi_y \cdot \zeta_y - \phi_z = 0 \quad (5)$$

$$D(p, \zeta) = \zeta + \frac{1}{2g}[(\phi_x^2 + \phi_y^2 + \phi_z^2) - U_\infty^2] = 0 \quad (6)$$

By introducing small perturbations δp , according to the classical Michell's theory, small $\delta\phi$ and $\delta\zeta$ are induced,

$$p = p' + \delta p \rightarrow \phi = \phi' + \delta\phi, \quad \zeta = \zeta' + \delta\zeta$$

where: ϕ - the exact solution; ϕ' - an approximate solution; $\delta\phi$ - the potential difference; ' - denotes a quantity from the immediate former iteration.

Expanding (5) and (6) in the first order Taylor series, we have

$$K(p, \zeta) \approx K(p', \zeta') + \Delta K(p, \zeta') + \Delta K(p', \zeta) \approx K(p', \zeta') + \frac{\partial}{\partial p} K(p, \zeta') \delta p + \frac{\partial}{\partial \zeta} K(p', \zeta') \delta \zeta \approx 0$$

where

$$\begin{aligned} K(p', \zeta') &= \phi_x \zeta'_x + \phi_y \zeta'_y - \phi_z, \quad \Delta K(p, \zeta') = \delta \phi_x \zeta'_x + \delta \phi_y \zeta'_y - \delta \phi_z \\ \Delta K(p', \zeta) &= \phi'_x \delta \zeta_x + \phi'_y \delta \zeta_y + (\phi'_{xz} \zeta'_x + \phi'_{yz} \zeta'_y - \phi'_{zz}) \delta \zeta \end{aligned}$$

and

$$D(p, \zeta) \approx D(p', \zeta') + \Delta D(p, \zeta') + \Delta D(p', \zeta) \approx D(p', \zeta') + \frac{\partial}{\partial p} D(p, \zeta') \delta p + \frac{\partial}{\partial \zeta} D(p', \zeta') \delta \zeta \approx 0$$

where

$$D(p', \zeta') = \zeta' - \frac{1}{2} [U_\infty^2 - (\phi'_x + \phi'_y + \phi'_z)],$$

$$\Delta D(p, \zeta') = \frac{1}{g} (\phi'_x \delta \phi_x + \phi'_y \delta \phi_y + \phi'_z \delta \phi_z),$$

$$\Delta D(p', \zeta) = \delta \zeta + \frac{1}{g} (\phi'_x \phi'_{xz} + \phi'_y \phi'_{yz} + \phi'_z \phi'_{zz}) \delta \zeta.$$

ϕ'_z and the second order derivatives in the Z direction (ϕ'_{xz} , ϕ'_{yz} , ϕ'_{zz}) being relative small are expected to go to zero and can be neglected. Thus the free surface boundary conditions become:

$$\phi_x \zeta'_x + \phi_y \zeta'_y - \phi_z + \phi'_x \delta \zeta_x + \phi'_y \delta \zeta_y = 0 \quad (7)$$

and

$$\delta \zeta = \frac{1}{2g} [U_\infty^2 + \phi_x'^2 + \phi_y'^2 - 2(\phi'_x \phi_x + \phi'_y \phi_y)] - \zeta' \quad (8)$$

By insertion of (7) in (8) the final mathematically consistent linearized boundary condition is obtained.

3. NUMERICAL MODEL

The Galerkin weighted residuals basically consists in:

$$\iiint_D \text{weighting function} \times \text{differential equation} = 0$$

Applying it to Laplace equation

$$\Delta \phi = 0$$

we have

$$\iiint_D N_i \left(\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} + \frac{\partial^2 \phi}{\partial z^2} \right) dx dy dz = 0,$$

where D is the spatial flow domain. Integrating by parts in order to reduce the order of the equation

$$\iiint_D \left(\frac{\partial N_i}{\partial x} \cdot \frac{\partial \phi}{\partial x} + \frac{\partial N_i}{\partial y} \cdot \frac{\partial \phi}{\partial y} + \frac{\partial N_i}{\partial z} \cdot \frac{\partial \phi}{\partial z} \right) dx dy dz - \iint_S N_i \cdot \frac{\partial \phi}{\partial n} dS = 0 \quad (9)$$

$i=1, \dots, m$; where m - the total number of nodes;

Taking into account that within the flow domain the surface integrals cancels at element interfaces, without loss of generality (4) can be rewritten, as follows

$$\iiint_D \left(\frac{\partial N_i}{\partial x} \cdot \frac{\partial \phi}{\partial x} + \frac{\partial N_i}{\partial y} \cdot \frac{\partial \phi}{\partial y} + \frac{\partial N_i}{\partial z} \cdot \frac{\partial \phi}{\partial z} \right) dx dy dz = 0$$

Introducing trial functions to depict the spatial variation of ϕ , for each element, we have

$$\iiint_{D^e} \left(\frac{\partial N_i}{\partial x} \cdot \sum_{j=1}^{20} \frac{\partial N_j}{\partial x} \phi_j + \frac{\partial N_i}{\partial y} \cdot \sum_{j=1}^{20} \frac{\partial N_j}{\partial y} \phi_j + \frac{\partial N_i}{\partial z} \cdot \sum_{j=1}^{20} \frac{\partial N_j}{\partial z} \phi_j \right) dx dy dz .$$

or in matrix form

$$A^e \cdot \phi^e$$

in which

$$a_{ij} = \iiint_{D^e} \left(\frac{\partial N_i}{\partial x} \cdot \frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y} \cdot \frac{\partial N_j}{\partial y} + \frac{\partial N_i}{\partial z} \cdot \frac{\partial N_j}{\partial z} \right) dx dy dz ,$$

leading to a 20x20 symmetric matrix $a_{ij} = a_{ji}$

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{120} \\ a_{21} & a_{22} & \dots & a_{220} \\ \vdots & & & \\ a_{201} & \dots & \dots & a_{2020} \end{bmatrix} \cdot \begin{Bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_{20} \end{Bmatrix}$$

$$\begin{Bmatrix} \frac{\partial N_i}{\partial x} \\ \frac{\partial N_i}{\partial y} \\ \frac{\partial N_i}{\partial z} \end{Bmatrix} = \begin{bmatrix} \hat{J}_{11} & \hat{J}_{12} & \hat{J}_{13} \\ \hat{J}_{21} & \hat{J}_{22} & \hat{J}_{23} \\ \hat{J}_{31} & \hat{J}_{32} & \hat{J}_{33} \end{bmatrix} \cdot \begin{Bmatrix} \frac{\partial N_i}{\partial \xi} \\ \frac{\partial N_i}{\partial \eta} \\ \frac{\partial N_i}{\partial \zeta} \end{Bmatrix} ,$$

$$\iiint_{D^e} dx dy dz = \int_{-1}^{+1} \int_{-1}^{+1} \int_{-1}^{+1} |J| \cdot d\xi d\eta d\zeta ,$$

$$\int_{-1}^{+1} \int_{-1}^{+1} \int_{-1}^{+1} F(\xi, \eta, \zeta) d\xi d\eta d\zeta = I ,$$

$$I = \sum_{i=1}^3 \sum_{j=1}^3 \sum_{k=1}^3 a_i a_j a_k F(\xi_i, \eta_j, \zeta_k) .$$

3.1 Practical succession of the main steps necessary for elemental matrix coefficients evaluation (3D)

- a. Evaluation of the shape functions N;
 b. Evaluation of the local derivatives of the shape functions;
 c. Evaluation of the Jacobian matrix J;
 d. Evaluation of the determinant of the Jacobian Matrix det J;
 e. Evaluation of the inverse of the Jacobian $|J|^{-1}$;

f. Evaluation of weighted volume:

The value of the weighted volume at each Gauss point is required in order to carry out the numerical integration, since from equations (11) and (12) we deduce

$$\iiint_{D^e} F(x, y, z) dx dy dz = \int_{-1}^{+1} \int_{-1}^{+1} \int_{-1}^{+1} F(\xi, \eta, \zeta) |J| d\xi d\eta d\zeta = \sum_{i=1}^3 \sum_{j=1}^3 \sum_{k=1}^3 a_i a_j a_k F(\xi_i, \eta_j, \zeta_k) |J|.$$

Therefore each of the terms $a_i a_j a_k |J|$ is computed at each Gauss point; later these terms can, be multiplied by the function value and summed over each Gauss point to perform the integration;

g. Evaluation of the global derivative of the shape functions $\partial N / \partial x$, $\partial N / \partial y$ and $\partial N / \partial z$

Having evaluated the local derivatives of the shape functions $\partial N / \partial \xi$, $\partial N / \partial \eta$ and $\partial N / \partial \zeta$ together with $\partial \xi / \partial x$, $\partial \eta / \partial x$, $\partial \zeta / \partial x$, $\partial \xi / \partial y$, $\partial \eta / \partial y$, $\partial \zeta / \partial y$ and $\partial \xi / \partial z$, $\partial \eta / \partial z$, $\partial \zeta / \partial z$ we can calculate now the global derivatives of the shape functions of the form $\partial N / \partial x$, $\partial N / \partial y$, $\partial N / \partial z$ using relations

$$\begin{aligned} \frac{\partial N}{\partial x} &= \frac{\partial N}{\partial \xi} \cdot \frac{\partial \xi}{\partial x} + \frac{\partial N}{\partial \eta} \cdot \frac{\partial \eta}{\partial x} + \frac{\partial N}{\partial \zeta} \cdot \frac{\partial \zeta}{\partial x} \\ \frac{\partial N}{\partial y} &= \frac{\partial N}{\partial \xi} \cdot \frac{\partial \xi}{\partial y} + \frac{\partial N}{\partial \eta} \cdot \frac{\partial \eta}{\partial y} + \frac{\partial N}{\partial \zeta} \cdot \frac{\partial \zeta}{\partial y} \\ \frac{\partial N}{\partial z} &= \frac{\partial N}{\partial \xi} \cdot \frac{\partial \xi}{\partial z} + \frac{\partial N}{\partial \eta} \cdot \frac{\partial \eta}{\partial z} + \frac{\partial N}{\partial \zeta} \cdot \frac{\partial \zeta}{\partial z} \end{aligned}$$

h. Evaluation of matrix coefficients a_{ij}

The coefficients, as given by equation (10), can now be calculated since

$$a_{IJ} = \sum_{i=1}^3 \sum_{j=1}^3 \sum_{k=1}^3 \left(\frac{\partial N_I}{\partial x} \cdot \frac{\partial N_J}{\partial x} + \frac{\partial N_I}{\partial y} \cdot \frac{\partial N_J}{\partial y} + \frac{\partial N_I}{\partial z} \cdot \frac{\partial N_J}{\partial z} \right) \cdot |J|_{(\xi_i, \eta_j, \zeta_k)} a_i a_j a_k$$

in which the capitals I, J denote nodal numbers and i, j, k Gauss point numbers.

Once the governing equation and boundary conditions have been defined, the procedure is conceptually straightforward. It is highly dependent on the way in which both the geometry of the flow domain and spatial variation of the variables is defined. First the code is executed in single-body Neumann condition, at zero Froude number. In the next step, the linear solution is computed. Finally, non-linear effects on the free surface are taken into account by an iterative procedure (an Euler-Lagrange approach), always linearizing the free surface boundary conditions on the basis of the immediate former solution, until convergence is reached. Simultaneously, the finite elements are moved correspondingly so that the grid is adapted each step to follow the free surface. The infinity condition is fulfilled by using semi-infinite elements. For radiation condition a special treatment is applied.

4. WAVE PATTERN ANALYSIS

The wave-pattern analysis is a valuable diagnostic tool, particularly when used with a measurement of the momentum defect in the viscous-wake region. This technique has led to the discovery of an additional drag

component associated with wave breaking, and to a better understanding of the bulbous bows of full form big ships with low Froude number.

In fine high-speed vessels a bulbous bow promotes beneficial interference between the waves generated at different points along the length of the hull. Thus, for such vessels, the bow bulb reduces the wave resistance. Originally, bulbous bows of a similar form were fitted to certain ship types on the basis of experimental measurements indicating significant reductions in the total drag, but these reductions often exceeded the total estimated resistance.

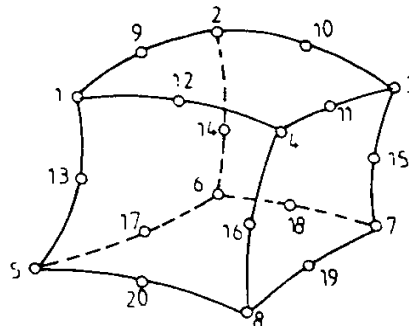
This apparent paradox has been reconciled by careful experimental measurements of the wave-energy flux and of momentum defect in the wake due to the viscous form drag. The latter measurement has revealed the existence of momentum associated with the breaking waves. Thus the wave-breaking resistance follows from the breaking of waves near the ship, predominantly at the bow. In this manner, the energy lost is convected downstream in the form of large-scale turbulence or eddies.

For full form giant ships, the bulbous bow is effective in reducing the magnitude of bow wave and thereby in avoiding wave breaking. For predicting the total drag, one may argue that it makes little difference whether the energy is wasted in wave radiation or in wave breaking, but only through an understanding of the mechanisms involved can the total drag be reduced intelligently and systematically. In this context we consider that using both a spatial moving boundary f.e.m. and an gaining insight into how waves break is the best solution.

4.1. Wave resistance computation

Once the field of velocities computed, the wave resistance can be obtained as follows

$$R_w = \rho U_\infty^2 \iint_{S_w} c_p n_x dx, \text{ where } c_p = \frac{p - p_\infty}{0.5 \cdot \rho U_\infty^2} = 1 - \left(\frac{U}{U_\infty} \right)^2.$$



Hexahedron finite element
with curved boundaries
(see location of nodes)

5. METHODOLOGY OF THE EXPERIMENTAL PART

Three types of breaking waves must be investigated: plunging breaker; spilling breaker; breaker that start as a small scale plunging breaker and evolves further similar to a spilling breaker.

The surface geometry is measured by means of image analysis (spatial geometry), in addition to ordinary wave gauges (temporal geometry). The surface is described by a set of parameters, derived to best describe features of the wave geometry that are important to the physics in the breaking event. In addition to measuring the geometry of the free surface the kinematics is measured by using Particle Image Velocimetry. The measurements is conducted first in a wave laboratory and then in a towing tank laboratory.

6. CONCLUSION

It is hoped that, in this way, a new more accurate teoretical-numerical method for ships wave making resistance prediction will be found.

REFERENCES

- [1] Bai, K.J., Kim, J.W., Kim, Y.H., Numerical computations for a nonlinear free surface flow problem.
- [2] Batchelor, G.K., An introduction to fluid dynamics, Cambridge University Press, 1991.
- [3] Bertram, V., Practical Ship Hydrodynamics, Butterworth-Heinemann, 2000.
- [4] Chung, T.J., Finite element analysis in fluid dynamics”, McGraw-Hill, New York, 1978.
- [5] Dawson, C.W., A practical computer method for solving ship-wave problem, 2-nd International Conference on Numerical Ship Hydrodynamics, University of California, Berkley, 1977.
- [6] Eggers, K., Sharma, S., Ward, L., An assessment of some experimental methods for determining the wave making characteristics of a ship form, Society of Naval Architects and Marine Engineers Transactions, **75**: 112-157.
- [7] Fletcher, C.A.J., Computational techniques for fluid dynamics, Springer, Berlin, 1987.
- [8] Jensen, G., Bertram, V., Söding, H., Ship wave-resistance computations, 5-th International Conference on Numerical Ship Hydrodynamics, 1990.
- [9] Jensen, G., Söding, H., Rankine source methods for numerical solutions of steady wave resistance problem, 17-th Symposium on Naval Hydrodynamics, Hague, 1988.
- [10] Jensen, P.S., On the numerical radiation condition in the steady-state ship wave problem, Journal of Ship Research, **31**, 1(1987), 14-22.
- [11] Larsson, L., Broberg, L., Kim, K.J., Zhang, D.H., New viscous and inviscid CFD Techniques for Ship Flows, 5th International Conference on Numerical Ship Hydrodynamics, Hiroshima, 1989.
- [12] Longuet-Higgins, M.S., Progress toward understanding how waves break, 21-st Symposium on Naval Hydrodynamics, 1997.
- [13] Pierson, J.W., Jean-Pierre Azed, Monte carlo simulations of nonlinear wave records with implications for models of breaking waves, Journal of Ship Research, **43**, 2(1999), 121-134.
- [14] Tuck, E.O., Scullen, D.C., Lazauskas, L., Wave patterns and minimum wave resistance for high-speed vessels , 24-th Symposium on Naval Hydrodynamics, Fukuoka, Japan, 8-13 July, 2002.
- [15] Tulin, M.P., Landrini, M., Breaking waves in the ocean and around ships, 23-rd Symposium on Naval Hydrodynamics, 2001.
- [16] Washizu, K., Some applications of finite element techniques to nonlinear free surface fluid flow problems, Osaka, Japan, 1981.

Mathematics modelling of main process of pollutants transport

Modelarea matematica a principalelor procese de transport al poluantilor. Metode de rezolvare si interpretare a solutiilor

Turcan Radu*, Turcan Viorica**

* Univ din Oradea, Fac.de Stiinte, Departamentul de Matematica

** Colegiul Tehnic «Constantin Brancusi» - Oradea

Rezumat. Prezenta lucrare continua articolul [5] si prezinta în sectiunea 1 principalele metode analitice, si numerice pentru studiul miscarii uni si bidimensionale, in medii poroase. Am analizat in principal doua dintre aceste metode si anume: metoda diferentelor finite – pentru miscari unidimensionale si metoda elementelor finite in cazul dispersiei bidimensionale, deoarece metodele numerice au o mai mare aplicabilitate, inclusiv in cazurile in care domeniul de miscare este complex. Ambele metode presupun o discretizare pe intregul domeniu de miscare.

In sectiunea 2 este prezentat sub forma generala un program performant, corespunzator modelarii curgerii apelor subterane si a poluantilor prin acvifere.

1. METODE DE REZOLVARE ȘI INTERPRETARE A SOLUȚIILOR

1.1. Metoda analitică pentru mișcări unidimensionale și bidimensionale

Ecuatia de transfer a poluanților în medii poroase are forma [4]

$$\frac{\partial C}{\partial t} + \frac{1}{R} \nabla \bar{u} C - \nabla \left(\frac{\bar{D}}{R} \nabla C \right) + \lambda C - \frac{q^* \rho_p C_p}{\rho m_e R} - \frac{\sigma^*}{\rho m_e R} = 0 \quad (1)$$

1.1.1. Mișcarea unidimensională

În acest caz, ecuația (1), devine:

$$\frac{\partial C}{\partial t} + \frac{u}{R} \cdot \frac{\partial C}{\partial x} - \frac{D_C}{R} \cdot \frac{\partial^2 C}{\partial x^2} + \lambda C = 0, \quad (2)$$

unde: $C(x;t)$ = concentrația poluantului la distanța x de locul injectării și la timpul t , de momentul injectării; D_L = coeficientul de dispersie longitudinală; t = timpul; u = viteza de mișcare reală a apei prin mediul poros; λ = rata de descompunere.

Pentru injectia de poluant, de tip impuls, într-un acvifer infinit la ambele capete, i.e. $x \in (-\infty; \infty)$, la $x = 0$ și $t = 0$; în acvifer se introduce o masă de poluant ΔM , ceea ce matematic se exprima prin conditia initiala

$$C_0(x_0;0) = \frac{\Delta M}{m_e T b R} \cdot \delta(x) \quad (3)$$

respective conditiile la mari distante

$$C(\pm\infty; t) = 0 \quad (4)$$

unde: $C_0(x_0;0)$ = concentrația poluantului în momentul injectării; ΔM = masa de poluant, în momentul inițial; m_e = porozitatea adimensională efectivă, T = grosimea stratului de acvifer, b = lățimea; R = factor de întârziere; $\delta(x)$ = funcția lui Dirac.

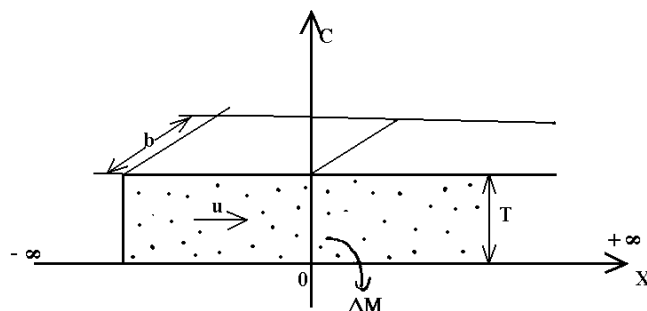


Fig.7.

Soluția analitică a problemei (2)-(4) a fost dată de Crank, în 1950, și are forma

$$C(x;t) = \frac{\Delta M}{2bTm_e \sqrt{\pi D_L t / R}} \exp\left\{-\frac{(x - ut / R)^2}{4D_L t / R}\right\} \exp\{-\lambda t\}$$

unde $\Delta M \cdot e^{-\lambda t} = \int_{-\infty}^{\infty} bTm_e RC(x;t) dx$.

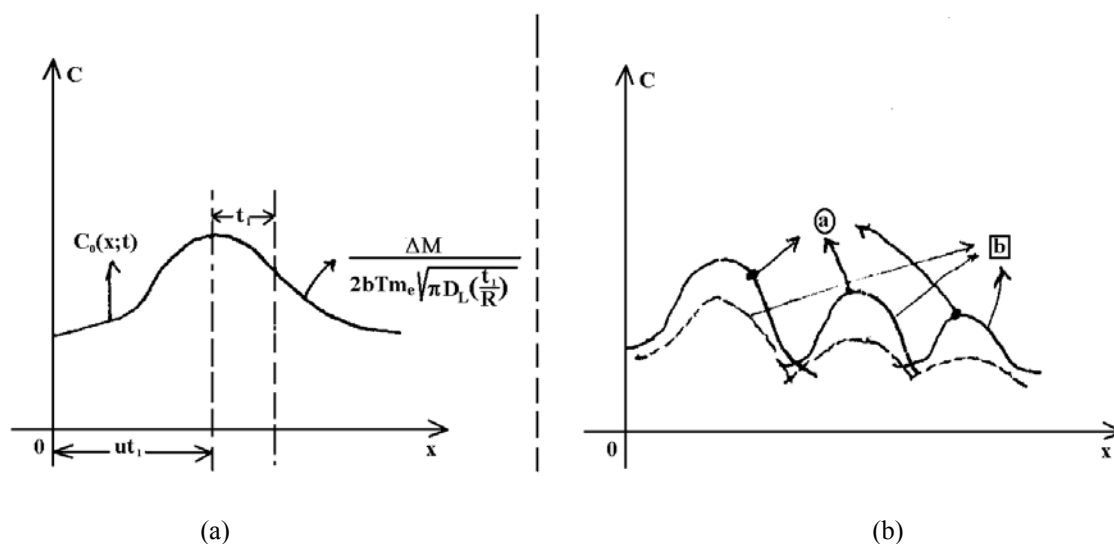


Fig.1.

Evoluția în timp și spațiu a concentrației poluantului, în stratul acvifer infinit și în ambele sensuri, când poluantul este de tip conservativ (i.e. $\lambda=0$) sau de tip neconservativ (i.e. $\lambda>0$), (vezi fig.9).

1.1.2. Mișcarea bidimensională

Pentru mișcarea dimensională, ecuația (1) devine

$$\frac{\partial C}{\partial t} + \frac{u}{R} \cdot \frac{\partial C}{\partial x} - \frac{D_L}{R} \cdot \frac{\partial^2 C}{\partial x^2} - \frac{D_T}{R} \cdot \frac{\partial^2 C}{\partial y^2} = 0.$$

Sa consideram pentru injecția punctuală, cu poluant de tip impuls. In acest caz, în punctul de coordonate $x=0$ și $y=0$ se efectuează injecția de poluant într-un acvifer constand dintr-un domeniu plan, infinit și omogen. Condițiile la limită, la mari distante și cele initiale sunt respectiv

$$C^{(0) \rightarrow \lambda}(x=0; y=0; t) = \frac{\Delta M}{m_e \cdot T \cdot R} \cdot \delta(x) \delta(z)$$

$$\begin{cases} C^{(0)}(x = \pm\infty; y = \pm\infty; t) = 0 \\ C^{(0)}(x = 0; y = 0; t = 0) = 0 \end{cases}$$

iar condiția de normare este se scrie ca

$$\int_{x=-\infty}^{+\infty} \int_{y=-\infty}^{+\infty} m_e T R C^0(x; y; t) dx dy = \Delta M$$

Soluția acestei probleme a fost dată de Csanady (1981) și are forma

$$C^{(0)}(x, y; t) = \frac{\Delta M}{4\pi m_e T \cdot \sqrt{D_L \cdot D_L \cdot t}} \cdot \exp\left[-\frac{(x - ut/R)}{4D_L t/R} - \frac{y^2}{4D_T t/R}\right]$$

1.2. Metode numerice

1.2.1. Metoda diferențelor finite pentru mișcări unidimensionale

Fie ecuația cu derivate parțiale, a celei mai simple mișcări: dispersia unidimensională (într-un curent uniform):

$$\frac{\partial C}{\partial t} = u \cdot \frac{\partial C}{\partial x} + D \cdot \frac{\partial^2 C}{\partial x^2}$$

Metoda diferențelor finite constă în discretizarea intervalului unidimensional spațial $x \in [0; \infty)$ și a celui temporal $t \in [0; \infty)$ și înlocuirea ecuației cu derivate parțiale cu o ecuație algebrică, în valorile nodale spațio-temporale ale concentrației.

Prima variantă a metodei diferențelor finite este metoda explicită, care constă în determinarea concentrației C la momentul $t + \Delta t$ în funcție de valorile la t , ale acesteia în nodurile spațiale (vezi fig. 2).

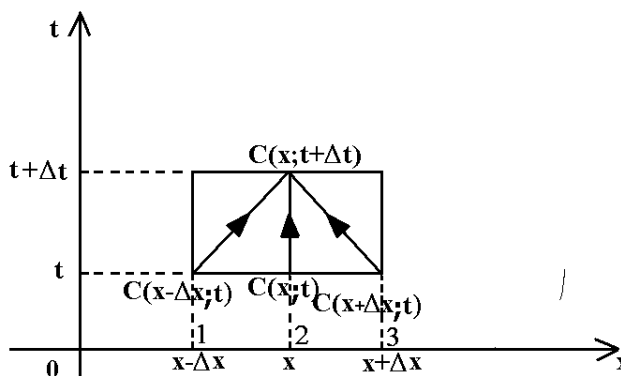


Fig. 2.

Pentru scrierea relațiilor cu diferențe finite, există mai multe posibilități: DIF – progresiv; DIF – central; DIF – regresiv.

Aceste 3 variante se referă la tipul aproximării derivatelor parțiale de ordinul I. Astfel, în cazul DIF – progresiv, din dezvoltarea

$$C(x + \Delta x; t) = C(x; t) + \Delta x \cdot \frac{\partial C}{\partial x} + \frac{\Delta x^2}{2!} \cdot \frac{\partial^2 C}{\partial x^2} + \dots \quad (5)$$

rezultă

$$\frac{\partial C(x;t)}{\partial x} = \frac{C(x+\Delta x;t) - C(x;t)}{\Delta x} - \frac{1}{2} \Delta x \cdot \frac{\partial^2 C}{\partial x^2} + 0 \cdot \left(\Delta x^2; \frac{\partial^3 C}{\partial x^3} \right)$$

În cazul în care avem DIF – regresiv, din dezvoltarea

$$C(x-\Delta x;t) = C(x;t) - \Delta x \cdot \frac{\partial C}{\partial x} + \frac{\Delta x^2}{2!} \cdot \frac{\partial^2 C}{\partial x^2} + \dots \quad (6)$$

rezultă:

$$\frac{\partial C(x;t)}{\partial x} = \frac{C(x;t) - C(x-\Delta x;t)}{\Delta x} + \frac{1}{2} \Delta x \cdot \frac{\partial^2 C}{\partial x^2} + 0 \cdot \left(\Delta x^2; \frac{\partial^3 C}{\partial x^3} \right)$$

iar cazul DIF – central se obține din relațiile (5) și (6), prin scăderea lor, i.e.

$$\frac{\partial C(x;t)}{\partial x} = \frac{C(x+\Delta x;t) - C(x-\Delta x;t)}{2\Delta x} + 0 \cdot \left(\Delta x^2; \frac{\partial^3 C}{\partial x^3} \right).$$

Pentru derivata a II-a se poate utiliza numai forma centrală, din relațiile (5) și (6) rezultând, prin adunare,

$$\frac{\partial^2 C}{\partial x^2} = \frac{C(x-\Delta x;t) - 2C(x;t) + C(x+\Delta x;t)}{\Delta x^2} + 0 \cdot \left(\Delta x^2; \frac{\partial^4 C}{\partial x^4} \right)$$

Pentru discretizarea temporală, în variantă explicită, se poate folosi numai DIF – progresiv

$$C(x;t+\Delta t) = C(x;t) + \Delta t \cdot \frac{\partial C}{\partial t} + \frac{\Delta t^2}{2!} \cdot \frac{\partial^2 C}{\partial t^2} + \dots$$

rezultând

$$\frac{\partial C(x;t)}{\partial t} = \frac{C(x;t+\Delta t) - C(x;t)}{\Delta t} - \frac{1}{2} \Delta t \cdot \frac{\partial^2 C}{\partial t^2} + 0 \cdot \left(\Delta t^2; \frac{\partial^3 C}{\partial t^3} \right).$$

1.2.2. Metoda elementelor finite (în cazul dispersiei bidimensionale)

Fie dispersia bidimensională simplă, i.e. mișcarea convectivă este uniformă, cu viteza constantă u , iar dispersia este bidimensională, având coeficientul de dispersie de forma

$$D_{i,j} = \begin{bmatrix} D_{xx} & 0 \\ 0 & D_{yy} \end{bmatrix} = \begin{bmatrix} \alpha_L u & 0 \\ 0 & \alpha_T u \end{bmatrix}$$

În general $\alpha_L \neq \alpha_T$, i.e. dispersia longitudinală α_L este diferită de dispersia transversală α_T .

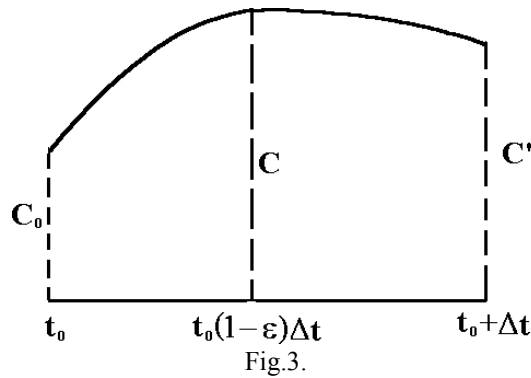
Ecuția generală a transportului poluanților într-un acvifer bidimensional are forma

$$\frac{\partial C}{\partial t} = -u \frac{\partial C}{\partial x} + \frac{\partial}{\partial x} \left(D_{xx} \frac{\partial C}{\partial x} \right) + \frac{\partial}{\partial y} \left(D_{yy} \frac{\partial C}{\partial y} \right). \quad (7)$$

Derivata parțială a lui C în raport cu timpul se poate exprima într-o formă mai convenabilă (care să cuprindă câteva forme de interpolare) pornind de la formula de interpolare pentru valoarea medie:

$$C = \varepsilon C_0 + (1-\varepsilon) C'. \quad (8)$$

Se observă că pentru: $\varepsilon = 1/2$, rezultă $C = \frac{C_0 + C'}{2}$, i.e. avem o interpolare constantă, cu valoarea medie a concentrației (fig. 3).



Pentru concentrația C , exprimată prin formula (8), avem câteva cazuri particulare:

- 1) pentru $\varepsilon = 1 \Rightarrow C = C_0$ adevărată pentru $t = t_0$;
- 2) pentru $\varepsilon = 0 \Rightarrow C = C'$ adevărată pentru $t = t_0 + \Delta t$;
- 3) pentru $\varepsilon = \text{arbitrar} \Rightarrow C = \varepsilon C_0 + (1 - \varepsilon)C'$ adevărată pentru $t = t_0 + (1 - \varepsilon)\Delta t$.

Atunci derivata lui C în raport cu timpul devine

$$\frac{\partial C}{\partial t} = \frac{C - C_0}{(1 - \varepsilon)\Delta t} \quad (9)$$

Înlocuind (9) în (7) obținem

$$\frac{C - C_0}{(1 - \varepsilon)\Delta t} = -u \frac{\partial C}{\partial x} + \frac{\partial}{\partial x} \left(D_{xx} \frac{\partial C}{\partial x} \right) + \frac{\partial}{\partial y} \left(D_{yy} \frac{\partial C}{\partial y} \right),$$

sau, echivalent

$$\frac{\partial}{\partial x} \left(D_{xx} \frac{\partial C}{\partial x} \right) + \frac{\partial}{\partial y} \left(D_{yy} \frac{\partial C}{\partial y} \right) - u \frac{\partial C}{\partial x} - \frac{C - C_0}{(1 - \varepsilon)\Delta t} = 0,$$

i.e. ecuația generală de transport a poluanților printr-un acvifer bidimensional.

2. ASM (AQUIFER SIMULATION MODEL)

2.1. Descriere generală

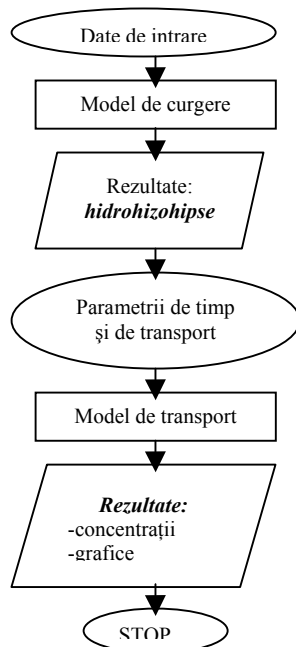
2.1.1 Performanțele programului

Probleme pe care le rezolvă sunt: previziuni hidrogeologice; interpretarea datelor de pompare la puțuri; evaluarea testelor de pompare; simularea curgerii apelor subterane; interpretarea datelor de concentrație; simularea transportului poluanților în ape subterane.

2.1.2 Descrierea generală a ASM

ASM este un program ce simulează curgerea apelor subterane și transportul poluanților folosind metoda diferențelor finite în varianta sa celulară, i.e. toate mărimile fizice (hidraulice) sunt considerate în centrul celulei.

2.1.3. Schema bloc a programului



2.2. ASM – date de intrare pentru parametrii de curgere

- Tipul regimului de curgere permanent sau nepermanent;
- tipul acviferului: cu nivel liber sau sub presiune;
- mărimea acviferului: lungimea și latimea;
- mărimea celulelor: Δx (m), Δy (m);
- numărul celulelor: după axa Ox: N_x , după axa Oy: N_y ;
- grosimea acviferului: 2 m;
- caracterizarea acviferului prin: permeabilitatea [K(m/sec)] la cel cu nivel liber; transmisivitatea [T(m²/sec)] la cel sub presiune; acvifer omogen (K sau T = constante); neomogen (K sau T = variabile);
- numărul și poziția puțurilor de: extragere (-m³/sec), injecție (+m³/sec);
- alimentarea acviferului din precipitații cu debite constante sau variabile;
- condiții la limită (înălțimi piezometrice, cunoscute pe frontiera domeniului);
- coeficienții de înmagazinare pentru celulele acviferului:
 - 0 → pentru celulele la care nu se cunoaște înălțimea piezometrică
 - 1 → pentru celulele situate pe frontiera domeniului

Cu aceste date de intrare, modulul de curgere poate fi rulat și se obțin:

- înălțimile piezometrice în toate celulele din domeniu și
- bilanțul apei (intrări, ieșiri de volume de apă)

2.3. ASM – date de intrare pentru parametrii de transport

- porozitatea efectivă (0,18 ÷ 0,30);
- factor de întârziere: R = 1 (fără întârziere) și R > 1 (cu întârziere);
- număr de particule, folosite în simulare;
- dispersivitatea longitudinală (1% din lungimea modelului);
- dispersivitatea transversală (1/10 din dispersivitatea longitudinală);
- tipul injecției (permanentă sau instantanee);

- rata de injecție (pentru injecția permanentă a unui poluant);
- masa de injecție (pentru injecția instantanee);
- poziția surselor de poluare (în care celulă se află);
- timpul total de simulare (în zile);
- pasul de timp (în zile).

Cu aceste date de intrare (pentru parametrii de transport și de timp) modulul de transport poate fi rulat, obținând rezultate ale simulării transportului poluanților în acvifer.

Rezultatele oferite de modulul de transport sunt: concentrații finale (după timpul total de simulare); variația concentrațiilor în puțuri de observație (în maxim 3 celule) stabilite anterior de utilizator; distanțele de la puțuri, pentru care timpul de transport convectiv are o valoare dată (pentru 20 de zile sau 50 de zile); traseul parcurs de o particulă de poluant dintr-un anumit punct, până la cel mai apropiat puț de extracție.

2.4. Alte precizări pentru ASM

Programul rulează sub MS – DOS; fișierul principal este: asm.exe; datele de intrare se salvează într-un fișier cu nume ales de utilizator; datele de ieșire (înălțimi piezometrice în celule) sunt create de program; rezultatele se pot vizualiza pe monitor; se pot capta (cu programe grafice).

BIBLIOGRAFIE

- [1]. David, I., Grundwasserhydraulik, Vieweg, Wiesbaden, 1998.
- [2]. Gheorghita, St. I., Metode matematice in hidrogazodinamica subterana, Ed. Academiei, Bucuresti, 1966.
- [3]. David, I., Sumalan, I., Carabet, A., Nitusca, A., Transportul poluantilor prin medii fluide, Universitatea „Politehnica” din Timisoara, 1966.
- [4]. Turcan, R., Turcan, V. Modelarea curgerii apei si a transportului poluantilor prin acvifere – metode matematice, Conferinta Nationala de Analiza Matematica si Aplicatii, Universitatea „Babes-Bolyai”, Cluj-Napoca (8-9 nov 2002).
- [5]. Turcan R., Turcan V., The transport in aquifers, în acest volum

Integrity in distributed database

Mădălina VĂLEANU
 Department of Medical Informatics and Biostatistics,
 Universitatea de Medicina si Farmacie „Iuliu Hatieganu” Cluj-Napoca

Grigor MOLDOVAN
 Department of Computer Science,
 Universitatea Babes-Bolyai Cluj-Napoca

Abstract. In this paper a proper terminology for discretionary models databases is defined and some integrity types for these models are specified. A modality for maintaining integrity of distributed databases is presented too.

Keywords: database, distributed database, database integrity, distributed database integrity, integrity level

1. INTRODUCTION

The database is now an integral part of our day-to-day life that often we are not aware using one. We can consider a database to be a collection of logically related data and a description of this data designed to meet the information needs of an organization [6,7].

The Database Management System (DBMS) is software that enables users to define, create and maintain the database and provides controlled access to this database. Data and database administration are the roles generally associated with the management and control of a DBMS and its data.

The Data Administrator (DA) is responsible for the management of the data resource including database planning, development and maintenance of standards, policies and procedures, and conceptual/logical database design.

The Database Administrator (DBA) is responsible for the physical realization of the database, including physical database design and implementation, security and integrity control, maintenance of the operational system, and ensuring satisfactory performance for the application and users.

Database integrity refers to the validity and consistency of stored data. Integrity is usually expressed in terms of constraints, which are consistency rules that the database is not permitted to violate. Constraints may apply to data items within a single record or they may apply to relationships between records. For example, an integrity constraint could state that an employee's salary cannot be greater than 10.000 \$ or that the branch number contained in the employee's record, representing the branch that the employee works at, must correspond to an existing branch office. Again, integration allows the DBA to define, and the DBMS to enforce, integrity constraints.

Ensuring *integrity of information* means preventing/detecting/deterring the improper modification of information (Sandhu and Jajodia, 1990). For example, in a military environment the target coordinates of a missile should not be improperly modified. Also, in commercial environments, data integrity is a relevant aspect: the good working of an organization depends on correct operations a correct and coherent data. For example, an employee should not be able to modify his or her own salary, or improperly alter data regarding an electronic payment.

Integrity of the database concerns database protection from unauthorized access that could modify the contents of data, as well as from errors, viruses, sabotage or failures in the system that could damage stored data. This kind of protection is partly carried out by the DBMS through proper system controls, and various backup and recovery procedures, and partly through ad hoc security procedures.

Operational integrity of data aims to ensure the logical consistency of data in a database during concurrent transactions. The *concurrency manager* is the DBMS subsystem that fulfils this requirement.

The concurrency manager ensures the serializability properties of transactions. Serializability means that the outcome of a concurrent run of a set of transactions is the same as the one produced by a strict sequence of these transactions. Isolation means mutual independence among transactions, thus avoiding "domino effects", where an "abort transaction" causes other transactions to abort in a cascade.

The problem of ensuring that concurrent access to the same data item by different transaction does not lead to data inconsistency is a commonly solved through *locking* techniques.

Lock and unlock techniques consist, respectively, in blocking data items for the time needed to execute an operation, and in releasing the item once the operation has been completed. In this way, a transaction can lock a data item, making it inaccessible to other transactions. The item is accessible again at release time.

Semantic integrity of data is to ensure the logical consistency of modified data by controlling data values in the allowed range. Restrictions on data values (for example, attribute values in a relational database) are expressed as *integrity constraints*.

Constraints can be defined for the whole database (conditions defining the correct state of a database), or for transactions (conditions to be verified in order to execute a modification to the database).

2. DISCRETIONARY MODELS AND INTEGRITY

Each system subject is assigned three integrity levels [3,4,15]:

- *Absolute Integrity Level (AIL)* is the integrity level given to the data contained in the object and the subject upon its creation. It is fixed for the whole life cycle of the subject. Typically, this level will be the integrity level of the user on behalf of whom the subject is acting;
- *Read Integrity Level (RIL)* is the lowest integrity level from which the subject is allowed to read;
- *Write Integrity Level (WIL)* is the highest integrity level to which the subject is allowed to write.

Each object is assigned three integrity levels:

- *Absolute Integrity Level (AIL)* is the integrity level of the data contained in the object and the subject. It is fixed for the whole life cycle of the object;
- *Migration Integrity Level (MIL)* is the lowest integrity level to which data in the object may flow;
- *Corruption Integrity Level (CIL)* is the highest level from which data may flow into the object.

The TCB model

The TCB model defines multilevel relations [10]. A multilevel relation is defined as any relation where there exist classification attributes C_i for each attribute A_i and a classification attribute for each tuple. Formally, a multilevel relation is represented by a schema $R(A_1, C_1, \dots, A_n, C_n, T_C)$. An attribute A_i corresponding to C_i of a multilevel relation is single level iff, C_i is defined on a domain represented by a single class in the access class lattice; otherwise it is multilevel. A multilevel relation is single level if all attributes are single-level and of the same class.

Each tuple in a multilevel relation has the form $\langle a_1|c_1, \dots, a_n|c_n, t \rangle$, where $a_i|c_i$ indicates the value and the classification of attribute i respectively. Element t indicates the access class associated with the tuple: that is, the access class of the information in (or encoded in) the tuple.

Properties:

- *Database class integrity:* the access class of a relation schema must dominate the access class of the name of the database to which it belongs;
- *Visible data property:* the access of the relation schema must be dominated by the access class of the lowest data that can be stored in the relation. The greatest lower bound of the range of access classes specified for an attribute must dominate the access class of the relation schema;
- *View class integrity:* the access class of a view definition must dominate the access class of any relation or view named in the view definition;
- *Multilevel entity integrity:* let AK be the set of data attributes forming the primary key of a relation R . All classification attributes C_i corresponding to data attributes $A_j \in AK$ have the same value within any given tuple of R , and this class is dominated by the value of each classification attribute C_j whose data attribute $A_j \notin AK$. No tuple in an instance of R can have null values for any of the primary key attributes;

- *Multilevel referential integrity*: no tuple in a relation can have a non-null secondary key unless a tuple exist in the referenced relation with the corresponding primary key. Within a tuple, the access class of each element comprising a secondary key must be the same (that is, the secondary key attributes must be uniformly classified) and must dominate the access class of the primary key element(s) in the tuple referenced.

The Jajodia and Sandhu model

The Jajodia and Sandhu model is a model for the application of mandatory policies in relational database system.

A multilevel relation consists of the following parts:

- a state-independent multilevel relation scheme $R(A_1, C_1, \dots, A_n, C_n, TC)$, where each A_i is a data attribute defined over domain D_i , each C_i is a classification attribute for A_i and TC is the tuple-class attribute;
- a collection of state-dependent relation instances $R_c(A_1, C_1, \dots, A_n, C_n, TC)$, one for each access class c .

Properties:

Entity integrity: Let AK be the apparent key of a relation R . A multilevel relation R satisfies entity integrity if, and only if, for all instances R_c of R and $t \in R_c$

$$(1) A_i \in AK \Rightarrow t[A_i] \neq null ,$$

$$(2) A_i, A_j \in AK \Rightarrow t[C_i] = t[C_j], \text{i.e. } AK \text{ is uniformly classified, and}$$

$$(3) A_i \notin AK, t[C_i] \geq t[C_{AK}] \text{ (where } C_{AK} \text{ is defined as the classification of the apparent key).}$$

Null integrity: a multilevel relation R satisfies null integrity if for each instance R_c of R both the following conditions are satisfied:

(1) for all $t \in R_c, t[A_i] = null \Rightarrow t[C_i] = t[C_{AK}]$, that is null values are classified at the level of the key;

(2) R_c is subsumption free in the sense that it does not contain two distinct tuples such that one subsumes the other.

Polyinstantiation integrity property: a multilevel relation R satisfies polyinstantiation integrity iff, for ever R_c , for all $A_i:AK, C_{AK}, C_i \rightarrow A_i$.

3. LOGICAL DATABASE DESIGN METHODOLOGY FOR THE RELATIONAL MODEL

The logical database design methodology for the relational model need two steps:

- build and validate local data model for each user view;
- build and validate global logical data model.

Integrity constraints are the constraints that we wish to impose in order to protect the database from becoming inconsistent. Note that, although DBMS control on integrity may or may not exist, this is not the question here. At this stage, we are concerned only with high-level design, that is specifying what integrity constraints are required, irrespective of how this might be achieved. Having identified the integrity constraints, we will have a local logical data model that is a complete and accurate representation of a user view. If necessary, we could produce a physical database design from the local logical data model, for example, to prototype the system for the user.[12,14]

We consider five types of integrity constraints:

- ✓ required data;
- ✓ attribute domain constraints;
- ✓ entity integrity;
- ✓ referential integrity;
- ✓ enterprise constraints.

Required data

Some attributes must always contain a valid value; in other words, they are not allowed to hold nulls. For example, every member of a staff has an associated job position (such as Manager, Assistants).

These constraints should have been identified when we documented the attributes in the data dictionary.

Attribute domain constraints.

Every attribute has a domain, which is a set of values that are legal. For example, the sex of a member of staff is either 'M' or 'F', so the domain of the Sex attribute is a single character string of 'M' or 'F'.

These constraints should have been identified when we choose the attribute domains for the data model

Entity integrity

The primary key of an entity cannot hold nulls. For example, each occurrence of the staff relation must have a value for the primary key attribute, namely Staff_No.

These constraints should have been considered when we identified the primary keys for each entity type.

Referential integrity

A foreign key links each occurrence in the child relation to the occurrence in the parent relation containing the matching candidate key value. Referential integrity means that, if the foreign key contains a value, that value must refer to an existing occurrence in the parent relation.

Enterprise constraints

Finally, we consider constraints known as enterprise constraints, sometimes called business rules. Updates to entities may be constrained by enterprise rules governing the 'real world' transaction that are represented by the updates.

4. DISTRIBUTED DATABASE

In a distributed DBMS, we need in addition a *global transaction manager* or *transaction coordinator* at each site, to coordinate the execution of both the global and local transactions initiated at that site [2,5]. Inter-site communication is still through the *Data Communications* component (transaction manager at different sites do not communicate directly with each other).

When a transaction is submitted at some site, the transaction manager at that site breaks it up into a collection of one or more subtransactions that execute at different sites, submits them to transaction manager at the other site, and coordinates their activity [14].

We can consider aspects of concurrency control and recovery that require additional attention because of data distribution. There are many concurrency control protocols. Here appear the following issues[13]:

- **Distributed Concurrency Control:** how can locks for objects stored across several sites be managed? How can deadlocks be detected in a distributed database?
- **Distributed Recovery:** transaction atomicity must be ensured – when transaction commits, all its actions, across all the sites that it executes at, must persist. Similarly, when a transaction aborts, none of its action must be allowed to persist.

Distributed Concurrency Control

The choice of technique for implementing asynchronous replication determined *which* objects are to be locked. *When* locks are obtained and released is determined by the concurrency protocol. We now consider how lock and unlock requests are implemented in a distributed environment.

Lock management can be distributed across sites in many ways:

- *centralized:* a single site is in charge of handling lock and unlock requests for all objects.
- *primary copy:* one copy of each objects is designated as the primary copy. All requests to lock or unlock a copy of this object are handled by the lock manager at the site where the primary copy is stored.
- *fully distributed:* requests to lock or unlock a copy of an object stored at a site are handled by the lock manager at the site where the copy is stored.

Distributed Recovery

Recovery in a distributed DBMS is more complicated than in a centralized DBMS for the following reasons:

- new kinds of failure can arise, namely failure of communication link and failure of a remote site at which a subtransaction is executing;
- either all transactions of a given transaction must commit, or none must commit, and this property must be guaranteed despite any combination of site and link failures. This guarantee is achieved using a **commit protocol**.

Three-phase commit (3PC) is non-blocking for site failures, except in the event of the failure of all sites. Communication failure can, however, result in different sites reaching different decisions, thereby violating the atomicity of global transactions. The protocol requires that:

- no network partitioning can occur;
- at least one site must always be available;
- at most K sites can fail simultaneously (system is classified as K -resilient).

The 3PC protocol imposes a significant additional cost during normal execution, and requires that communication link failures do not lead to a network partition (wherein some sites cannot reach some other sites through any part) in order to ensure freedom from blocking

Maintaining integrity

Time	P_1	P_2
T_1	Begin_transaction	Begin_transaction
T_2	$Bal_x = Bal_x - 60$	$Bal_x = Bal_x - 50$
T_3	Write (Bal_x)	Write (Bal_x)
T_4	commit	commit

Successfully completed update operations by users in different partitions can easily violate integrity constraints, as illustrated in previous figure. Assume that a bank places a constraint on a customer account (with balance Bal_x) that it cannot go below 0. In partition P_1 , a transaction has withdrawn 60 from the account and in partition P_2 a transaction has withdrawn 50 from the same account. Assuming at the start, both partitions have 100 in Bal_x , then on completion one has 40 in Bal_x and the other has 50. Importantly, neither has violated the integrity constraints. However, when the partitions recover and the transactions are both fully implemented, the balance of the account will be -10, and the integrity constraints will have been violated.

Processing in a partitioned network involves a trade-off in availability and correctness. Absolute correctness is easiest to provide if no processing of replicated data is allowed during partitioning. On the other hand, availability is maximized if no restrictions are placed on the processing of replicated data during partitioning [12].

In general, it is not possible to design a non-blocking atomic commit protocol for arbitrarily partitioned networks. Since recovery and concurrency control are so closely related, the recovery techniques that will be used following network partitioning will depend on the particular concurrency control strategy being used. Methods are classified as pessimistic or optimistic.

Pessimistic protocols

Pessimistic protocols choose consistency of the database over availability, and would therefore not allow transactions to execute in a partition if no guarantee the consistency can be maintained. The protocol uses a pessimistic concurrency control algorithms. Recovery using this approach is much more straightforward, since updates would have been confined to a single, distinguished partition. Recovery or reconnection of the network simply involves propagating all the updates to every other site.

Optimistic protocols

Optimistic protocols, on the other hand, choose availability of the database at the expense of consistency, and use an optimistic approach to concurrency control, in which updates are allowed to proceed independently in the various partitions. Therefore, when sites recover, inconsistencies are likely.

To determine whether inconsistencies exist, precedence graphs can be used to keep track of dependencies among data. While the network is partitioned, updates proceed without restriction, and precedence graphs are maintained by each partition. When the network has recovered, the precedence graphs

for all partitions are combined. Inconsistencies are indicated if there is a cycle in the graph. The resolution of inconsistencies depends upon the semantics of the transactions, and thus it is generally not possible for the recovery to re-establish consistency without intervention.

5. CONCLUSIONS

For distributed databases are keep the classical (usual) integrity types and as a request can be added the integrity types referred to global transactions (concurrent) in inter-site communication.

REFERENCES

- [1] Bell, D., Grimson, J., Distributed database system, Addison-Wesley, 1992.
- [2] Bobak, A., Distributed and multi-database systems , Bantam Books, 1993.
- [3] Cartell, R., Object data management , Addison –Wesley – ACM Press Books, 1992.
- [4] Castrano, S., Fugini, M., Martella, G., Samarati, P., Database security, Addison-Wesley, 1999.
- [5] Ceri, S., Pelagatti, G., Distributed satabases. Principles and systems, International Student Edition, McGraw Hill, 1985.
- [6] Connolly, T. M., Begg, C. E., Database systems, Addison-Wesley, 1999.
- [7] Date, C. J., An introduction to database systems , Addison –Wesley – ACM Press Books, 1982.
- [8] Fleming, C., B. von Halle, Handbook of relational database design, Adison-Wesley, 1989.
- [9] Franklin, M., Concurrency control and recovery in Handbook of computer science, A.B. Tucker , CRC Press, 1996.
- [10] Gilula, M., The set model for database and informational systems, Addison–Wesley – ACM Press Books, 1999.
- [11] Gollmann, D., Computer security, John Wiley&Sons, 2000.
- [12] Hoffer, J, George, J., Valacich, J., Modern system analysis & design ,3-rd ed., Prentice Hall, 2002.
- [13] Ramakrishnan, R., Database management system, Mc Garw-Hill, 1998.
- [14] Ozsu, M., Valduriez, P., Principles of distributed database systems, Prentice-Hall, 1991.
- [15] Sandhu, R. S., Jajodia, S., Integrity mechanism in database management systems in Proc. 13th National Computer Security Conference, Oct. 1990.
- [16] Vossen, G., Date Models, Database languages and database management systems, Addison–Wesley, 1991.
- [17] Widom, J, Ceri, S., Active database system , Morgan Kaufmann, 1996.
- [18] Zaniolo, C., Ceri, S., Faloutsos, C., Snodgrass, R, Subrahmanianv, Zicari, R., Advanced database systems, Morgan Kaufmann, 1997.

Some applications of the Perov's fixed point theorem

Ioan Dzitac, Simona Dzitac, Horea Oros,
University of Oradea, idzitac@uoradea.ro

Abstract. In this paper we will present an application of the Perov's fixed point theorem in the canonical n -Banach spaces and one consequence in the \mathbf{R}^n space. A sequential simulation program in C++ for parallel asynchronous method RFAIM [Dzi 00] is presented for a numerical example of the [RCM 75].

Key words: fixed point theorem, numerical method, parallel asynchronous iterations, RFAIM method

1. PRELIMINARIES RESULTS

In this paper we shall denote \mathbf{R} the real number set and \mathbf{R}^n the natural ordonated real linear space: $\mathbf{R}^n := (\mathbf{R}^n, \mathbf{R}, \mathbf{x} + \mathbf{y}, \lambda \mathbf{x}, \mathbf{x} \leq \mathbf{y})$, where, for $\mathbf{x} = (x_1, x_2, \dots, x_n) \in \mathbf{R}^n$, $\mathbf{y} = (y_1, y_2, \dots, y_n) \in \mathbf{R}^n$ and $\lambda \in \mathbf{R}$, we have:

- (1.1) $\mathbf{x} = \mathbf{y} \Leftrightarrow x_i = y_i, \quad i = 1..n;$
- (1.2) $\mathbf{x} \leq \mathbf{y} \Leftrightarrow x_i \leq y_i, \quad i = 1..n;$
- (1.3) $\mathbf{x} + \mathbf{y} = (x_1 + y_1, x_2 + y_2, \dots, x_n + y_n);$
- (1.4) $\lambda \mathbf{x} = (\lambda x_1, \lambda x_2, \dots, \lambda x_n);$
- (1.5) $\mathbf{x} \leq \mathbf{y} \Leftrightarrow x_i \leq y_i, \quad i = 1..n.$

Definition 1.1 [n -metric complete space]: Let X a nonempty set. A vector function $d_n : X \times X \rightarrow \mathbf{R}^n$ is called n -metric on X , if the following conditions are satisfied:

- (1.6) $d_n(\mathbf{x}, \mathbf{y}) \geq 0_{\mathbf{R}^n}, \quad \forall \mathbf{x}, \mathbf{y} \in X;$
 $d_n(\mathbf{x}, \mathbf{y}) = 0_{\mathbf{R}^n} \Leftrightarrow \mathbf{x} = \mathbf{y},$

where $0_{\mathbf{R}^n}$ is the null element from \mathbf{R}^n

- (1.7) $d_n(\mathbf{x}, \mathbf{y}) = d_n(\mathbf{y}, \mathbf{x}), \quad \forall \mathbf{x}, \mathbf{y} \in X;$
- (1.8) $d_n(\mathbf{x}, \mathbf{y}) = d_n(\mathbf{x}, \mathbf{z}) + d_n(\mathbf{y}, \mathbf{z}), \quad \forall \mathbf{x}, \mathbf{y}, \mathbf{z} \in X.$

The couple (X, d_n) is named n -metric space. A n -metric space in which every fundamental sequence is convergent is called n -metric complete space.

Remarks:

- 1) A 1 -metrical space is a regular metric space, equipped with a scalar metric.
- 2) \mathbf{R}^n is n -metrical complete space, in raport with te n - vector metric: $d_n : \mathbf{R}^n \times \mathbf{R}^n \rightarrow \mathbf{R}^n,$

$$(1.9) \quad d_n(\mathbf{x}, \mathbf{y}) = (|x_1 - y_1|, |x_2 - y_2|, \dots, |x_n - y_n|).$$

Definition 1.2 [n -normed space]: Let $(X, \mathbf{R}, \mathbf{x} + \mathbf{y}, \lambda \mathbf{x})$ be a real vector space. A vector function $p_n(\mathbf{x}) : X \rightarrow \mathbf{R}^n, \quad \forall \mathbf{x} \in X$ is called a n -norm on X (or a *vector norm*), if the following conditions are satisfied:

- (1.10) $p_n(\mathbf{x}) \geq 0_{\mathbf{R}^n}, \quad \forall \mathbf{x} \in X,$
 $p_n(\mathbf{x}) = 0_{\mathbf{R}^n} \Leftrightarrow \mathbf{x} = 0_X;$
- (1.11) $p_n(\mathbf{x} + \mathbf{y}) \leq p_n(\mathbf{x}) + p_n(\mathbf{y}), \quad \forall \mathbf{x}, \mathbf{y} \in X;$
- (1.12) $p_n(\lambda \mathbf{x}) = |\lambda| p_n(\mathbf{x}), \quad \forall \mathbf{x} \in X, \quad \forall \lambda \in \mathbf{R}.$

The couple (X, p_n) is called n -normed space (or *normed space with the vector norm*).

Remark: A 1 -normed space is a regular normed space, equipped with a scalar norm.

Definition 1.3 [n -metric induce by a n -norm]: Let (X, p_n) a n -norm space. If the vectorial function $d_n : X \times X \rightarrow \mathbf{R}^n,$

$$(1.13) \quad d_n(x, y) = p_n(x - y), \quad \forall x, y \in X,$$

is a n -metric on X , then the function d_n is called the n -metric induce by the n -norm p_n .

Remark: The vector norm: $p_n: \mathbf{R}^n \rightarrow \mathbf{R}^n$,

$$(1.14) \quad p_n(x) = (|x_1|, |x_2|, \dots, |x_n|), \quad \forall x \in \mathbf{R}^n.$$

induces the n -metric given by relation (1.9).

Definition 1.4 [Banach n -space]: A n -normed space (X, p_n) which is complete with respect to the n -metric induce by the n -norm p_n is called n -Banach space.

Remark. The normed space \mathbf{R}^n with n -norm (1.14) is a n -Banach space.

Definition 1.5 [Product of Banach spaces]: Let the Banach spaces (X_i, p_i) , $i=1..n$, where $p_i: X_i \rightarrow [0, \infty)$ are the scalar norms on the real vector spaces X_i . The product

$$(1.15) \quad X = \prod_{i=1}^n X_i = X_1 \times X_2 \times \dots \times X_n$$

with $x = (x_1, x_2, \dots, x_n) \in X$, $x_i \in X_i$, $i = 1..n$.

Theorem 1.1: Let X be a product of Banach spaces

$$X = \prod_{i=1}^n X_i = X_1 \times X_2 \times \dots \times X_n,$$

and the scalar norms $p_i: X_i \rightarrow [0, \infty)$ on real Banach spaces X_i .

The vectorial function $\|x\|_{\rightarrow}: X \rightarrow \mathbf{R}^n$,

$$(1.16) \quad \|x\|_{\rightarrow} = (p_1(x_1), p_2(x_2), \dots, p_n(x_n)), \quad \forall x = (x_1, x_2, \dots, x_n) \in X,$$

is a n -norm on X

Proof:

1) Because X_i are real vector space and X their product shall be a real vector space, if we define for $x = (x_1, x_2, \dots, x_n) \in X$, $y = (y_1, y_2, \dots, y_n) \in X$ și $\lambda \in \mathbf{R}$,

$$(1.17) \quad x + y = (x_1 + y_1, x_2 + y_2, \dots, x_n + y_n), \quad \forall x, y \in X;$$

$$(1.18) \quad \lambda x = (\lambda x_1, \lambda x_2, \dots, \lambda x_n), \quad \forall x \in X, \quad \forall \lambda \in \mathbf{R}.$$

2) For all $i=1..n$ we have the following proprieties of the norms:

$$(1.19) \quad p_i(x_i) \geq 0_{\mathbf{R}}, \quad \forall x_i \in X_i,$$

$$p_i(x_i) = 0_{\mathbf{R}} \Leftrightarrow x_i = 0_{X_i}, \quad ;$$

$$(1.20) \quad p_i(x_i + y_i) \leq p_i(x_i) + p_i(y_i), \quad \forall x_i, y_i \in X_i;$$

$$(1.21) \quad p_i(\lambda x_i) = |\lambda| p_i(x_i), \quad \forall x_i \in X, \quad \forall \lambda \in \mathbf{R}.$$

It verifies immediately the proprieties:

$$(1.22) \quad \|x\|_{\rightarrow} \geq 0_{\mathbf{R}^n}, \quad \forall x \in X, \quad \|x\|_{\rightarrow} = 0_{\mathbf{R}^n} \Leftrightarrow x = 0_X;$$

and

$$(1.23) \quad \|\lambda x\|_{\rightarrow} = |\lambda| p_n(x), \quad \forall x \in X, \quad \forall \lambda \in \mathbf{R}.$$

Further we have:

$$\begin{aligned} \|x + y\|_{\rightarrow} &= (p_1(x_1 + y_1), p_2(x_2 + y_2), \dots, p_n(x_n + y_n)) \leq \\ &(p_1(x_1) + p_1(y_1), p_2(x_2) + p_2(y_2), \dots, p_n(x_n) + p_n(y_n)) = \\ &(p_1(x_1), p_2(x_2), \dots, p_n(x_n)) + (p_1(y_1), p_2(y_2), \dots, p_n(y_n)) = \|x\|_{\rightarrow} + \|y\|_{\rightarrow}. \end{aligned}$$

Therefore:

(1.24)
$$\|x + y\|_{\rightarrow} \leq \|x\|_{\rightarrow} + \|y\|_{\rightarrow}.$$

The relations (1.22), (1.23) and (1.24) shows that the vector norm $\|x\|_{\rightarrow}$ is a n -norm on the product of Banach spaces X .

Definition 1.6 [The canonical vector norm]:

Let $X = \prod_{i=1}^n X_i = X_1 \times X_2 \times \dots \times X_n$ be a product of Banach spaces equipped with the n -norm:

(1.25)
$$\begin{aligned} \|x\|_{\rightarrow} : X \rightarrow \mathbf{R}^n, \\ \|x\|_{\rightarrow} = (p_1(x_1), p_2(x_2), \dots, p_n(x_n)), \quad \forall x = (x_1, x_2, \dots, x_n) \in X. \end{aligned}$$

where $p_i : X_i \rightarrow [0, \infty)$ are the scalar norms on the real Banach spaces X_i . This vector norm is called the *canonical vector norm* on the product of Banach spaces X .

Theorem 1.2: The product of Banach spaces :

$$X = \prod_{i=1}^n X_i = X_1 \times X_2 \times \dots \times X_n,$$

with the canonical vector norm given by (1.25), is a n -Banach space.

Proof.

We showed in theorem 1.1 that the product of Banach spaces X is a n -normed space in raport with the canonical vectorial norm defined by the relations (1.25). We shall show now that X is complete in raport with the n -metric induced by te canonical norm, i.e. X is a n -Banach space .

Let $d_n : X \times X \rightarrow \mathbf{R}^n$ be a n -metric induced by the canonical vectorial norm:

(1.26)
$$d_n(x, y) = \|x - y\|_{\rightarrow}, \quad \forall x, y \in X.$$

We consider a fundamental sequence of elements from X ,

$$(x_i^m)_{m>0}, \quad \text{where } x^m = (x_1^m, x_2^m, \dots, x_n^m) \in X, \quad x_i^m \in X_i, \quad i = 1..n.$$

Therefore, $(\forall) \varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n) > 0_{\mathbf{R}^n}$, exists $k = k(\varepsilon)$, $k \in \mathbf{N}^*$, such as: $d_n(x^{1+p}, x^p) < \varepsilon$, $(\forall) p > k$ and 1 natural. I.e. $d_n(x^{1+p}, x^p) = \|x^{1+p} - x^p\|_{\rightarrow} < \varepsilon$, :from where:

$$p_i(x_i^{1+p} - x_i^p) < \varepsilon_i, \quad i = 1..n,$$

which shows that the sequence $(x_i^m)_{m>0}$ is fundamental in X_i .

As X_i is complete (being a Banach space) implies the fact that exists $a_i \in X_i$, such as $x_i^m \rightarrow a_i$, for m that tends to infinite, $(\forall) i = 1..n$: $x^m \rightarrow a$, $a = (a_1, a_2, \dots, a_n) \in X$ and therefore this sequence is convergent in X , i.e. X is a n -Banach space, being complete in raport with the n -metric induced by the canonical vector norm.

Definition 1.7 [n-Banach canonical space]: We shall call a *n-Banach canonical space*, the *n-Banach space* $(X, \|\cdot\|_{\rightarrow})$, defined on the product of Banach spaces:

$$X = \prod_{i=1}^n X_i = X_1 \times X_2 \times \dots \times X_n,$$

equipped with the canonical vectorial norm: $\|\cdot\|_{\rightarrow} : X \rightarrow \mathbb{R}^n$,

$$\|\cdot\|_{\rightarrow} = (p_1(x_1), p_2(x_2), \dots, p_n(x_n)), \quad \forall x = (x_1, x_2, \dots, x_n) \in X.$$

Definition 1.8 [n-euclidean norm on \mathbb{R}^n]: The canonical norm:

$$(1.27) \quad \|\cdot\|_{\rightarrow} : \mathbb{R}^n \rightarrow \mathbb{R}^n, \quad \|\cdot\|_{\rightarrow} = (|x_1|, |x_2|, \dots, |x_n|), \quad x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n,$$

we shall name it *n-euclidean norm on \mathbb{R}^n*

Theorem 1.3: The space \mathbb{R}^n is a *n-canonical Banach space* in rapport with the *n-euclidean norm*.

Proof.

Let \mathbb{R} be a vector real space, equipped with the norm $\|x\| = |x|$. This norm induces the Euclid's metric, in rapport with which \mathbb{R} is a complete metrical space, i.e. is a Banach space.

So, the product of Banach spaces $(\mathbb{R}^n, \|\cdot\|_{\rightarrow})$ shall be a canonical *n-Banach space*, in accordance with theorem 1.2 and definition 1.7.

2. PEROV'S FIXED POINT THEOREM FOR CANONICAL CONTRACTIONS

The fixed points set of the map $f : X \rightarrow X$, is denoted by:

$$(2.1) \quad F_f = \{x \mid f(x) = x, x \in X\}.$$

Definition 2.1 [The spectral raze of a real matrix]: Let M be the square matrix of *n* order with real numbers. We shall call the *spectral raze* of the matrix M , the real number denoted by $\rho(M)$:

$$(2.2) \quad \rho(M) = \max |\lambda_k|, \quad k=1 \dots n,$$

where λ_k are the proper values of matrix M .

Definition 2.2 [A matrix converging to zero]: A square *n*-order and non-negative M matrix is called *convergent to zero* if „the power matrix” $M^m = \underbrace{M \cdot M \dots M}_{m \text{ factors}}$ is convergent to zero (denoted by $M \rightarrow O$,

convergent to the null matrix O , where *m* tends to infinite.

Theorem 2.1: Let M be a square *n*-order, real and non-negative matrix. The following assertions are equivalent:

$$(2.3) \quad M^m \rightarrow O;$$

$$(2.4) \quad \rho(M) < 1;$$

$$(2.5) \quad \det(E - M) \neq 0;$$

$$(2.6) \quad (E - M)^{-1} = E + M + M^2 + \dots,$$

where E is the unit *n*-order matrix.

The proof of this theorem is found in [Rus 79], on page 35-37 (with the corresponding terminological adaptation).

Definition 2.3 [The lipschitzian map in n-metric]: Let (X, d_n) be a n-metric space. A map $f: X \rightarrow X$ is called lipschitzian in a n-metric on X, if there exists a matrix L, square of n-order and non-negative, so that the Lipschitz condition is satisfied:

$$(2.7) \quad d_n(f(x), f(y)) \leq L d_n(x, y), \quad \forall x, y \in X.$$

The L matrix is called the Lipschitz matrix.

Definition 2.4 [The contracting map in n-metric]: A lipschitzian map $f: X \rightarrow X$ in n-metric is called contraction on X, if the map admits a Lipschitz matrix K (in this case K named the contraction matrix), of spectral raze smaller then 1.

Theorem 2.2 [Perov]: Let (X, d_n) be a n-metric complete space and a contracting map $f: X \rightarrow X$, with the contraction matrix K. We have:

$$(2.8) \quad F_f = \{x^*\};$$

(2.9) The iterative sequence $x^m = f^m(x^0)$, $\forall x^0 \in X$, $m \in \mathbb{N}^*$ is convergent to a unique fixed point x^* of f;

$$(2.10) \quad \text{The estimation: } d_n(x^m, x^*) \leq K^m (E-K)^{-1} d_n(x^1, x^0);$$

(2.11) If $g: X \rightarrow X$ is the approximation of f, i.e. exists in \mathbb{R}^n a $\varepsilon > 0$ such that

$$d_n(f(x), g(y)) < \varepsilon, \quad \forall x, y \in X,$$

then, for the iterative sequence $y^m = g^m(x^0)$, we have the estimation:

$$d_n(y^m, x^*) \leq \varepsilon (E-K)^{-1} + K^m (E-K)^{-1} d_n(x^1, x^0).$$

The proof of this theorem can be found in [Rus 79, page 39] (with the corresponding terminological adaptations and notations).

Definition 2.5 [The canonical contraction]: Let $(X, \|x\|_{\rightarrow})$ be a n-Banach canonical space with the canonical vectorial norm $\|x\|_{\rightarrow}$. A map $f: X \rightarrow X$ is named a canonical contraction on X, if there exists a non-negative matrix K with the spectral ray smaller then 1, such that:

$$(2.12) \quad \|f(x) - f(y)\|_{\rightarrow} \leq K \|x - y\|_{\rightarrow}, \quad \forall x, y \in X.$$

Theorem 2.3: Let $(X, \|x\|_{\rightarrow})$ a n-Banach canonical space and $f: X \rightarrow X$ a canonical contraction on X, with the contraction matrix K.

In those conditions we have:

$$(2.13) \quad F_f = \{x^*\};$$

(2.14) The sequence $x^m = f(x^{m-1})$, $\forall x^0 \in X$, $m \in \mathbb{N}^*$ is convergent to x^* , the unique fixed point of f;

(2.15) It takes place the estimation:

$$\|x^m - x^*\|_{\rightarrow} \leq K^m (E-K)^{-1} \|x^1 - x^0\|_{\rightarrow};$$

(2.16) If $g: X \rightarrow X$ is an approximant of f, i.e. exists in \mathbb{R}^n a $\varepsilon > 0$ such as $\|f(x) - g(x)\|_{\rightarrow} < \varepsilon$, $\forall x \in X$, then for the sequence we have the estimation:

$$\|y^m - x^*\|_{\rightarrow} \leq \varepsilon (E-K)^{-1} + K^m (E-K)^{-1} \|x^1 - x^0\|_{\rightarrow}.$$

Proof.

According to the theorem 1.2 we have that X is a n-metric complete space in raport with the n-metric induced by the canonical vector norm. Because the map f is a canonical contraction on X, the conclusions of our theorem results from Perov's theorem.

Theorem 2.4: Let $X = \mathbb{R}^n$ and $f: X \rightarrow X$ be a canonical contraction on X with respect to the canonical vectorial norm:

$$\|x\|_{\rightarrow}: \mathbb{R}^n \rightarrow \mathbb{R}^n, \|x\|_{\rightarrow} = (|x_1|, |x_2|, \dots, |x_n|), x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n,$$

with the contraction matrix K.

In those conditions we have:

(2.17) $F_f = \{x^*\}$;

(2.18) The sequence $x^m = f(x^{m-1})$, $\forall x^0 \in X$, $m \in \mathbb{N}^*$ is convergent to x^* , the unique point of f ;

(2.19) It takes place the estimation:

$$\|x^m - x^*\|_{\rightarrow} \leq K^m (E-K)^{-1} \|x^1 - x^0\|_{\rightarrow};$$

(2.20) If $g: X \rightarrow X$ is an approximation of f , i.e. exists in \mathbb{R}^n an ε with positive components, such that $\|f(x) - g(x)\|_{\rightarrow} < \varepsilon$, $\forall x \in X$, then for the sequence $y^m = g(y^{m-1})$, $m = 1, 2, \dots$, $y^0 = x^0 \in X$ we have the estimation:

$$\|y^m - x^*\|_{\rightarrow} \leq \varepsilon (E-K)^{-1} + K^m (E-K)^{-1} \|x^1 - x^0\|_{\rightarrow}.$$

Proof.

Applying the result of theorem 1.2 and theorem 2.3, the conclusions of our theorem results immediatly from Perov's theorem.

3. NUMERICAL EXAMPLE

(3.1) In [RCM 73] it's gave the following example of ten nonlinear equations system, with ten real unknown:

$$x[i] = f_i(x[1], x[2], x[3], x[4], x[5], x[6], x[7], x[8], x[9], x[10]), \quad i=1..10,$$

$$x[1] = f_1(x) = (x[1] + x[2]) / 4 + x[3] / 20$$

$$x[2] = f_2(x) = (\cos(x[1] + x[2] + x[4] - x[5] - x[8] - x[9])) / 6.6$$

$$x[3] = f_3(x) = 0.5 * \cos(x[1] / 2 + x[3] / 3 - 2 * x[5] / 3) + x[7] / 5$$

$$x[4] = f_4(x) = 0.2 * (\sin(2 * x[3]) + \cos(x[1] + x[4])) + x[5] / 6$$

$$x[5] = f_5(x) = (\exp(-x[1] * x[1]) + \exp(-x[8] * x[8])) / 2$$

$$x[6] = f_6(x) = (x[1] - x[2] + x[8] + x[9] + 3 * x[10]) / 7.2$$

$$x[7] = f_7(x) = \exp(-x[1]) + x[9] / 11$$

$$x[8] = f_8(x) = (\cos(x[1] + x[3] + x[6] + x[9])) / 5$$

$$x[9] = f_9(x) = (\sin(x[2] + x[4] + x[5] + x[8])) / 5$$

$$x[10] = f_{10}(x) = (\exp(-x[3] * x[3]) + \exp(-x[8] * x[8])) / 2 - x[10] / 11.$$

Let $f = (f_1, \dots, f_{10})$. It's showed that the $f: \mathbb{R}^{10} \rightarrow \mathbb{R}^{10}$ operator admits in private, the K contraction matrix, with the subunit spectral ray, $\rho(K) = 0.7868\dots$, where K is gave by the following table:

0,2500	0,2500	0,05000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
0,1520	0,1520	0,0000	0,1520	0,1520	0,0000	0,0000	0,1520	0,1520	0,0000
0,2500	0,0000	0,1670	0,0000	0,3340	0,2000	0,2000	0,0000	0,0000	0,0000
0,2000	0,0000	0,4000	0,2000	0,1670	0,0000	0,0000	0,0000	0,0000	0,0000
0,4289	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,4289	0,0000	0,0000
0,1390	0,1390	0,0000	0,0000	0,0000	0,0000	0,0000	0,1390	0,1390	0,4170
0,8579	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0910	0,0000
0,2000	0,0000	0,2000	0,0000	0,0000	0,2000	0,0000	0,0000	0,2000	0,0000
0,0000	0,2000	0,0000	0,2000	0,2000	0,0000	0,0000	0,2000	0,0000	0,0000
0,0000	0,0000	0,4289	0,0000	0,0000	0,0000	0,0000	0,42890	0,0000	0,0910

Program RFAIM_SEQ_SIM (based of RFAIM method [Dzi00])

```
//Program RFAIM_SEQ_SIM
//Sequential simulation of the asynchronous parallelism RFAIM

# include <conio.h>
# include <iostream.h>
# include <iomanip.h>
```

```

#include <math.h>
#include <stdlib.h>

enum bool {false, true};
long double eps = 1.0e-9;
long double intervale
[] [2]={{0.0,1.0},{0.0,1.0},{0.0,1.0},{0.0,1.0},{0.0,1.0},
{0.0,1.0},{0.0,1.0},{0.0, 0.2},{0.0,0.2},{0.0,1.0}};

bool finish (long double [ ], long double [ ], int);
void tipar (long double [ ], int);
void shuffle (unsigned [ ], int);
void genereaza (long double [ ], long double [ ], long double [ ], int);

long double f1 (long double [ ]);
long double f2 (long double [ ]);
long double f3 (long double [ ]);
long double f4 (long double [ ]);
long double f5 (long double [ ]);
long double f6 (long double [ ]);
long double f7 (long double [ ]);
long double g8 (long double [ ]);
long double f9 (long double [ ]);
long double f10 (long double [ ]);

int matherr (struct exception *e)
{
    switch (e->type)
    {
        case DOMAIN:
            cout << "Argumentul " << e->arg1<<"este eronat in"<<e->name;
            break;
        case OVERFLOW:
            cout << "Argumentul " << e->arg1<<"este prea mare in functia
"<<e->name;
            break;
        case TLOSS:
            cout << "Argumentul " << e->arg1<<"este prea mare in functia
"<<e->name;
            break;
        default:
            cout << "eroare \ n " ;
    }
    exit (-1);
}

int main ( )
{
    clrscr ( ) ;
    //long double
x1[]={0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0};
    //long double
x2[]={0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0};
    //long double
x3[]={0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0};
    long double
x1[]={1.0,2.0,3.0,4.0,5.0,6.0,7.0,8.0,9.0,10.0};
    long double
x2[]={1.0,2.0,3.0,4.0,5.0,6.0,7.0,8.0,9.0,10.0};
    long double
x3[]={1.0,2.0,3.0,4.0,5.0,6.0,7.0,8.0,9.0,10.0};
    unsigned nr_comp=sizeof(x1)/sizeof(x1[0]),i,contor=0;
    long double *x=x2;
    long double (*pf [ ]) , int) =
        {f1, f2, f3, f4, f5, f6, f7, f8, f9, f10} ;
    unsigned tab [ ] = {0, 1, 1, 2, 3, 4, 5, 6, 7, 8, 9};
    genereaza (x1, x2, x3, nr_comp);
    do {
        contor++ ;

```

```

        shuffle (tab, nr_comp);
        for (i = 0 ; i < nr_comp ; i++)
        {
            x1[i] = x2[i] ;
        }
        for (i = 0 ; i < nr_comp ; i++)
        {
            x2[tab[i]] = pf[tab [i] (x3) ;
            x3[tab[i]] = x2[tab[i] ] ;
        }
    } while (!finish (x1, x2, nr_comp) ) ;

    cout <<"Am obtinut solutia dupa "<<contor>> "iterari simple"<< endl ;
    tipar (x2, nr_comp) ;
    return 0 ;
}
long double f1 (long double x[ ], int i)
{
    //long double ret = ((x[0]+x[1]) / 4 + x[2] / 20) ;
    long double ret = (x[1]) / 3 + x[2] / 15
    if (ret >intervale [i] [0] && ret< intervale [i] [1] )
        return ret ;
    else
        return intervale [i] [0] + (intervale [i] [1] - intervale [i] [0] * (long
double ) rand ( ) / RAND_MAX ;
}
long double f2 (long double x[ ], int i)
{
    long double
ret=((cos(x[0]+x[1]+x[3]-x[4]-x[7]-x[8]))/6.6);
if (x[i]>intervale [i] [0] && x[0]<intervale [i] [1] )
    return ret ;
    else
        return intervale [i] [0] + (intervale [i] [1] - intervale [i] [0] * (long double )
rand ( ) / RAND_MAX ;
}
long double f3 (long double x[ ], int i)
{
    long double ret = (0.5*cos (x[0] / 2 +x[2] / 3- 2*x[4] / 3 + x[6] / 5.0) ;
if (x[i] > intervale [i] [0] && x[0] < intervale [i] [1] )
    return ret ;
    else
        return intervale [i] [0] + (intervale [i] [1] - intervale [i] [0] * (long double )
rand ( ) / RAND_MAX ;
}
long double f4 (long double x[ ], int i)
{
    long double ret = (0.2*sin (2.0*x[2])+ 0.2*cos (x[0]) + x[3] + x[4] / 6) ;
if (x[i] > intervale [i] [0] && x[0] < intervale [i] [1] )
    return ret ;
    else
        return intervale [i] [0] + (intervale [i] [1] - intervale [i] [0] * (long double )
rand ( ) / RAND_MAX ;
}
long double f5 (long double x[ ], int i)
{
    long double ret=((expl(-x[0]*x[0]+expl(-x[7]*x[7])))/2
if (x[i]>intervale [i] [0] && x[0]<intervale [i] [1])
    return ret ;
    else
        return intervale [i] [0] + (intervale [i] [1] - intervale [i] [0] * (long double )
rand ( ) / RAND_MAX ;
}
long double f6 (long double x[ ], int i)

```

```

{
    long double ret=(x[0]-x[1]+x[7]+x[8]+3*x[9])/7.2;
if (x[i]>intervale [i] [0] && x[0]<intervale [i] [1] )
    return ret ;
    else
        return intervale [i] [0] + (intervale [i] [1] - intervale [i] [0] * (long double )
rand ( ) / RAND_MAX ;
}

long double f7 (long double x[ ], int i)
{
    long double ret = (expl (-x[0])+x[8] /11 ;
if (x[i] > intervale [i] [0] && x[0] < intervale [i] [1] )
    return ret ;
    else
        return intervale [i] [0] + (intervale [i] [1] - intervale [i] [0] * (long double )
rand ( )/RAND_MAX;
}

long double f8 (long double x[ ], int i)
{
    //long double ret=((cos(x[0]+x[2]+x[5]+x[8]))/5);
    long double ret = asinl (5*x[8]))-(x[1]+x[3]+x[4];
if (x[i]>intervale [i] [0] && x[0]<intervale [i] [1] )
    return ret ;
    else
        return intervale [i] [0] + (intervale [i] [1] - intervale [i] [0] * (long double )
rand ( ) / RAND_MAX ;
}

long double f9 (long double x[ ], int i)
{
    //long double ret = ((sin (x[1] + x[3] + x[4] +x[7])) / 5 ;
    long double ret = acosl (5*x[7]) - x[0] - x[2] - x[5];
if (x[i]>intervale [i] [0] && x[0]<intervale [i] [1])
    return ret ;
    else
        return intervale [i] [0] + (intervale [i] [1] - intervale [i] [0] * (long double )
rand ( )/RAND_MAX;
}

long double f10 (long double x[ ], int i)
{
//long double ret = (expl(-x[2]* x[2]+ expl(-x[7]* x[7])) /2 - x[9] /11 ;
long double ret = 11*(expl(-x[2]*x[2])+expl(-x[7]*x[7]))/24 ;
if (x[i] > intervale [i] [0] && x[0] < intervale [i] [1] )
    return ret ;
    else
        return intervale [i] [0] + (intervale [i] [1] - intervale [i] [0] * (long double )
rand ( ) / RAND_MAX ;
}

bool finish (long double x1[ ], long double x2 [ ], int nr)
{
    bool ret = true ;
    int i ;
    for (i = 0 ; i < nr ; i++)
    {
        if ( fabsl (x1[i] - x2[i]) > eps)
        {
            ret = false ;
            break ;
        }
    }
    return ret ;
}

void tipar (long double x2[ ], int nr_comp)

```

```

{
    int i ;
    for (i = 0 ; i < nr_comp ; i++)
    {
        cout .setf (ios::fixed) ;
        cout << "x[" << setw(2) << (i+1)<<" ] = " << setprecision (10) << x2[i] << endl ;
    }
}
void shuffle (unsigned tab [ ], int n)
{
    int i, j ;
    bool ok;
    randomize ();
    for (i = 0 ; i < n ; i++)
    {
        do{
            ok = true;
            tab [i] = random (10);
            for (j = 0 ; j < i; j++)
                if (tab [j] == tab [i])
                {
                    ok = false;
                    break;
                }
            } while (! ok);
        }
    }
}
void genereaza (long double x1[ ], long double x2[ ], long double x3[ ],int nr_comp)
{
    for (int i = 0; i < nr_comp; i++)
    {
        x1[i] = (long double) rand ( ) / RAND_MAX;
        if ( i == 7 || i == 8)
            x1[i] / = 5;
        x3[i] = x2[i] = x1[i];
    }
}

```

Numerical solution (the fixed point value) to be obtain after 10-12 vector iterations:

```

x[1]= 0.0853306672
x[2]= 0.1263607698
x[3]= 0.6466737693
x[4]= 0.5198021063
x[5]= 0.9768165122
x[6]= 0.3503781711
x[7]= 0.9308854467
x[8]= 0.1997436445
x[9]= 0.1394451735
x[10]= 0.7415213722

```

REFERENCES

- [Dzi 00] Dzitac, I., Random-filtered asynchronous iterative methods, Bul. St. Univ. Baia Mare, Ser. B, Matematică – Informatică, Vol. XVI (2000), Nr. 1, pag. 17-24, ISSN 1222- 1201.
- [RCM 75] Robert, F., Charnay, M., Musy, F., Iterations chaotique serie-parallele pour des equations nonlineaire de point fixe. Aplikace Matematiky 20 (1975), p.1-38 (French)
- [Rus 79] RUS I.A.:O Principles and applications of the fixed point, Ed. Dacia, Cluj-Napoca, 1979. (Romanian)

Properties of the solutions of a parabolic equation of non divergent type

Aripov, M., Sadullaeva, Sh. A.
National University of Uzbekistan
E-mail: m.aripov@nuuz.uzsci.net
shahloaz@yahoo.com

Abstract. The properties of the weak solutions of an initial value problem is investigated for a quasilinear parabolic equation of non divergent type. Various similarity equations are constructed by an algorithm of nonlinear splitting. On the basis of some of these estimates solutions and fronts are obtained. The analogue of Fujita-Samarskii condition on a global solvability of an initial value problem is obtained.

Keywords: parabolic equation, not divergent type, asymptotic, weak solution.

MSC 2000 Subject classification: 35K20, 35K55, 35K65.

1. INTRODUCTION

Consider, in a domain $Q = \{(t, x) : t > 0, x \in \mathbb{R}^N\}$, the following Cauchy problem

$$Au \equiv -\frac{\partial u}{\partial t} + u^p \nabla \left(|x|^m u^\sigma \nabla u \right) + \varepsilon \gamma(t) u^\beta = 0 \quad (1)$$

$$|x|^m u_0^{p+\sigma} \nabla u_0 \in C(0, \infty), \sup u_0(x) < \infty, x \in \mathbb{R}^N \quad (2)$$

where $m > 0$, $\sigma > 0$, $p > 0$, $\beta \geq 1$, are given numbers, $\varepsilon = \pm 1$, $0 < \gamma(t) \in C(0, \infty)$, $\nabla(\cdot) = \text{grad}(\cdot)$.

The equation (1) describes various physical processes. It is of a degenerate type. Namely, when $u = 0$ the equation (1) degenerates into a first order equation. For this reason (1) can not have a classical solution. Consequently we shall study the weak solution which has the property $u \geq 0$ and $|x|^m u^{p+\sigma} \nabla u \in C(Q)$.

In the non degenerated case ($p=0$, when $m=\sigma=0$, $\gamma(t)=1$, $\varepsilon=+1$) this problem was considered first by Fujita ($\varepsilon=+1$) [1], and the conditions of global solvability $(\beta-1)N > 2$ where obtained. In [2] more general conditions of global solvability for the divergent case of the equation (1) were obtained while in [5-9] the same results were obtained for equations more general than equation of divergent type. The problem (1) - (2) began to be studied intensively rather recently (see [4] and references).

The present work is devoted to investigation of the properties of the solutions of the problem (1) - (2). One manner of construction of the various similarity equations by algorithm of nonlinear splitting [4-6] is provided.

2. THE SIMILARITY SOLUTIONS; APPROXIMATION SIMILARITY SOLUTIONS

We consider at first more simple case when $\gamma(t) = 0$ in (1)

Lemma 1. Equation (1) has a similarity solution of the following form

$$u(t, x) = f(\xi), \quad \xi = \frac{\phi(x)}{(T+t)^{1/2}}, \quad \phi(x) = \frac{2}{2-m} |x|^{\frac{2-m}{2}}$$

$$f^p \left[\xi^{1-s} \frac{d}{d\xi} \left(\xi^{s-1} f^\sigma \frac{df}{d\xi} \right) \right] + \frac{\xi}{2} \frac{df}{d\xi} = 0 \quad (3)$$

where $T > 0$, and

$$\bar{f}(\xi) = \left(c - \frac{p+\sigma}{4} \xi^2 \right)_+^{1/p+\sigma}, \quad (m)_+ = \max(0, m), \quad s = \frac{2N}{2-m} \quad (4)$$

($c > 0$ is constant, $0 < p < 1$) is a function which, in $|\xi| < 2\sqrt{\frac{c}{p+\sigma}}$, satisfies the equation

$$\bar{f}^p \left(\xi^{1-s} \frac{d}{d\xi} \left(\xi^{s-1} \bar{f}^\sigma \frac{d\bar{f}}{d\xi} \right) \right) + \frac{\xi}{2} \frac{d\bar{f}}{d\xi} + \frac{s}{2(1-p)} \bar{f} = 0. \quad (5)$$

$\bar{f}(\xi)$ is an upper solution of a problem (1),(2), if

$$u_0(x) \leq \bar{f}(\xi|_{t=0}), \quad x \in R^N.$$

It is easy to see that $\xi^{s-1} f^{p+\sigma} \nabla f \in C(0, \infty)$.

We denote by $D = \{(t, x) : t > 0, |x| < l(t)\}$ that part of a domain Q , where a function $l(t)$ is continuous, i.e. $l(t) \in C(0, \infty)$.

Let us construct other similarity solutions.

Lemma 2. The equation (1) has the following similarity type solution

$$u(t, x) = (T \pm t)^{-\alpha} f(\xi)$$

$$\text{where } \xi = \begin{cases} \frac{\phi(x)}{[\tau(t)]^{1/2}} & \text{if } 1 - \alpha(p + \sigma) \neq 0 \\ \sum_{i=1}^{N_i} \alpha_i x_i - c_1 \ln(T \pm t), \quad a = \sum_{i=1}^N a_i^2 \neq 0 & \text{if } 1 - \alpha(p + \sigma) = 0, m = 0 \end{cases}$$

$$\tau(t) = \begin{cases} \pm \frac{(T \pm t)^{1-\alpha(p+\sigma)}}{1-\alpha(p+\sigma)} + c_2, & \text{if } 1 - \alpha(p + \sigma) \neq 0 \\ \pm \ln(T \pm t), & \text{if } 1 - \alpha(p + \sigma) = 0 \end{cases}$$

and the function $\bar{f}(\xi)$ satisfy the equations

$$f^p \left(\xi^{1-s} \frac{d}{d\xi} \left(\xi^{s-1} f^\sigma \frac{df}{d\xi} \right) \right) + \frac{\xi}{2} \frac{df}{d\xi} \pm \frac{\alpha}{1-\alpha(p+\sigma)} f = 0, \text{ if } 1 - \alpha(p + \sigma) \neq 0$$

(6)

and

$$\alpha f^p \frac{d}{d\xi} \left(f^\sigma \frac{df}{d\xi} \right) \pm c_1 \frac{df}{d\xi} \pm \alpha \cdot f = 0, \text{ if } 1 - \alpha(p + \sigma) = 0,$$

respectively.

We notice that the equation (1) has the weak solution

$$u(t, x) = (T + t)^{-\alpha} f(\xi), \quad \bar{f}(\xi) = \left(c - \frac{p+\sigma}{4} \xi^2 \right)_+^{1/p+\sigma}, \quad \xi = \frac{\phi(x)}{(T+t)^{1/2}}, \quad \phi(x) = \frac{2}{2-m} |x|^{2-m/2}$$

$$\alpha = \frac{N}{(2-m)(1-p) + N(p+\sigma)}, \quad (0 < p < 1), \quad \tau(t) = \frac{1}{1-\alpha(p+\sigma)} (T+t)^{1-\alpha(p+\sigma)}.$$

On basis of these preliminaries we shall prove the theorem on the global solvability of the problem (1) - (2) for $\gamma(t) \neq 0$.

3. GLOBAL SOLVABILITY AND ESTIMATION OF THE SOLUTIONS.

Theorem 1. Let, in (1) $\varepsilon = -1$, $0 < p < 1$, $m < 2$,

$$[\bar{u}(t)]^{\beta-(p+1)} \gamma(t) \tau(t) \leq \frac{N}{(1-p)(2-m)}. \quad (7)$$

Then the problem (1) - (2) has a global solution, for which, in Q , the estimation

$$u(t, x) \leq z_+(t, x)$$

holds

Proof. Let us construct the upper solution $z_+(t, x)$ by method of nonlinear splitting [5-7], according to which first we find the solution of the simple equation

$$\frac{d\bar{u}}{dt} = -\gamma(t)\bar{u}^\beta, \text{ i.e. } \bar{u}(t) = (T + (\beta - 1) \int_0^t \gamma(t) dt)^{\frac{1}{\beta-1}} \quad (8)$$

and then look for the solution $u(t, x)$ of equation (1) in the form

$$u(t, x) = \bar{u}(t)W(\tau(t), x), \quad (9)$$

where

$$\tau(t) = \int [\bar{u}(t)]^{p+\sigma} dt = \begin{cases} \frac{(\beta - 1)(T + (\beta - 1) \cdot t)^{\frac{\beta-(p+\sigma+1)}{\beta-1}}}{\beta - (p + \sigma + 1)} & , \text{if } \beta \neq p + \sigma + 1 \\ \frac{1}{(p + \sigma)} \ln(T + (p + \sigma) \cdot t) & , \text{if } \beta = p + \sigma + 1 \end{cases}$$

Then for $W(\tau(t), x)$ from (1) taking into account (9), we obtain the equation

$$\frac{\partial W}{\partial \tau} = W^p \nabla (|x|^m W^\sigma \nabla W) + [\bar{u}(t)]^{\beta-(p+\sigma+1)} \gamma(t) (W \pm W^\beta). \quad (10)$$

By Lemma 1 the main part of equation (10) is similarity. Hence, putting

$$W(\tau, x) = f(\xi), \quad \xi = \frac{\phi(x)}{\tau^{1/2}}, \quad \phi(x) = \frac{2}{2-m} |x|^{2-m/2} \quad (11)$$

in (10), simple algebra lead to the approximately similarity equation.

Then substituting (12) in (11), we obtain the approximation selfsimilar equation

$$f^p \left(\xi^{1-s} \frac{d}{d\xi} \left(\xi^{s-1} f^\sigma \frac{df}{d\xi} \right) \right) + \frac{\xi}{2} \frac{df}{d\xi} + [\bar{u}(t)]^{\beta-(p+\sigma+1)} \gamma(t) \tau(t) (f \pm f^\beta) = 0$$

(12)

where $s = 2N(2 - m)$.

Remark. Let $\gamma(t)$ be a function of Hardy class. Then it is easy to see that

$\gamma(t) \tau(t) [\bar{u}(t)]^{\beta-(p+1)} \sim c$ as $t \sim +\infty$, where c is some constant. Indeed, by Hardy theorem we have

$$\tau(t) = \int_0^t [\bar{u}(t)]^{p+\sigma} dt \sim [\bar{u}(t)]^{p+\sigma} \sim [\gamma(t) t]^{\frac{p+\sigma}{\beta-1}} t, \quad t \sim \infty$$

$$\left[\bar{u}(t)\right]^{\beta-(p+\sigma+1)} = \left[T + \int_0^t \gamma(t) dt\right]^{\frac{\beta-(p+\sigma+1)}{\beta-1}} \sim [t\gamma(t)]^{\frac{\beta-(p+\sigma+1)}{\beta-1}}, \quad \text{for } t \sim \infty.$$

Hence (12) is an asymptotic similarity equation for a function $\gamma(t)$ of Hardy class.

Theorem 2. Let the conditions of Theorem 1 hold, let $\varepsilon = +1$ and let

$$\left(c^{\frac{1}{\beta-1}} + 1\right) [u(t)]^{\beta-(\sigma+p+1)} \gamma(t) \tau(t) \leq \frac{N}{(2-m)(1-p)}, \quad t > 0, \text{ where}$$

$$c^{\frac{1}{\beta-1}} = \max \bar{f}^{\frac{1}{\beta-1}}(\xi)$$

Then the problem (1)-(2) has a global solution for which, in Q , the estimate $u(t, x) \leq z_+(t, x)$ holds.

Theorem 3. Let in (1) be $1 - \alpha(p + \sigma) = 0$, $m=0$, $N=1$ $\gamma(t) = 1$, where the number a is defined by

(6). Then, for the free boundary of solution we have the asymptotic expansion $\sum_{i=1}^{N_a} a_i x_i \approx c_1 \ln t$, where a_i ($i = 1, N$). and $c_1 > 0$ - are constants.

Proof. In a domain $|\xi| < \frac{c}{p + \sigma}$, where $\xi = \sum_{i=1}^{N_a} a_i x_i - c_1 \ln t$, consider a function

$\tilde{f}(\xi) = A\xi^{\frac{1}{p}}$, where $A^p = \pm pc_1/a(p-1)$. The solution of the second equation of (6) is looked in the form $f = \tilde{f} \cdot y(\eta)$, where $\eta = -\ln \xi$. Then the obtained equation for $y(\eta)$ has a solution, which tends to the constant $c_3 > 0$ as $\xi \rightarrow 0$, ($\eta \rightarrow +\infty$), because $f = \tilde{f} \cdot c_3$. That finishes the proof of Theorem 3.

Theorem 4. Let be the condition $[\bar{u}(t)]^{\beta-(p+\sigma+1)} \gamma(t) \tau(t) \leq C$ $t > 0$, where C is a constant. Then in a neighborhood of a front a finite solution of the equation (12) has the asymptotic

$$f(\xi) \approx \bar{f}(\xi) = a \left(c - \frac{p + \sigma}{4} \xi^2 \right)_+^{\frac{1}{p+\sigma}}$$

where $a, c > 0$ are some constants.

This asymptotic behaviour reflects well the physics of the studied process. In particular, as p and $\sigma \rightarrow 0$ (linear case) this formula shows that front of temperature tends to infinity. In linear case the asymptotic

$$f(\xi) \approx \bar{f}(\xi) = \exp\left(-\frac{\xi^2}{4}\right)$$

of the solution was obtained by A. Samarskii in the monograph [1].

4. SOME COROLLARIES.

In particular case of equation (1) when $\gamma(t) = 1$ we have the following Corollaries

Corollary 1. Let $\varepsilon = -1$, $N > 1$, $0 < p < 1$, $\beta > \sigma + p + 1 + \frac{(2-m)(1-p)}{N}$, $u_0(x) \leq z_+(0, x)$, $x \in \mathbb{R}^N$,

where $z_+(t, x)$ is the above defined function

$$\bar{u}(t) = (T + (\beta - 1)t)^{-\frac{1}{\beta-1}}, \quad \xi = \frac{\phi(x)}{[\tau(t)]^{1/2}}, \quad \tau(t) = \frac{1}{\beta - (1 + p + \sigma)} (T + (\beta - 1)t)^{\frac{\beta - (1 + p + \sigma)}{\beta - 1}},$$

$$\beta - (1 + p + \sigma) \neq 0$$

Then for the solution $u(t, x)$ of (1)-(2) the estimate $u(t, x) \leq z(t, x)$ is holds in Q .

Corollary 2. Let, in (1), $\varepsilon = -1$, $m = -2(N-1)$, $N > 1$, $u_0(x) \leq z_+(0, x)$, $x \in \mathbb{R}^N$. Then, for a front $x_f(t)$ is valid the asymptotic expansion $|x_f(t)| \approx a(\ln t)^N$, $t > 1$, where $a > 0$ holds.

This result was early were obtained by J.Lions, M.Bertsch, R.Kersner.

5. LOCALIZATION OF THE SOLUTION IN THE CASE $\beta = 1$

In the following we establish a condition of localization of the solution by constructing an exact solution.

Choosing $u(t, x) = e^{-\int_0^t \gamma(r) dr} w(\tau(t), x)$, $\tau(t) = \int_0^t e^{-(p+\sigma)\int_0^\eta \gamma(\eta) d\eta} d\eta$ reduce the equation (1) to the

form

$$\frac{\partial w}{\partial t} = w^p \nabla \left(|x|^m w^\sigma \nabla w \right)$$

which, as shown in the above, has a solution

$$w(\tau, x) = [T + \tau(t)]^{-\alpha} \bar{f}(\xi), \quad \alpha = \frac{N}{(2 - m)(1 - p) + (p + \sigma)N}$$

where $T > 0$ is a constant,

$$\bar{f}(\xi) = A \left(a - \frac{p + \sigma}{4} \xi^2 \right)_+^{\frac{1}{\sigma + p}}, \quad A^p = \frac{1}{1 - p}, \quad \xi = \phi(x) / (\eta(\tau))^{1/2},$$

$$\phi(x) = \frac{2}{2 - m} |x|^{\frac{2 - m}{2}}, \quad \eta(\tau) = \frac{1}{1 - (p + \sigma)} (T + \tau(t))^{1 - \alpha(p + \sigma)}.$$

The condition $\tau(\infty) < +\infty$ is the condition of localization of the solution of equation (1).

Let $N = 1$, $m = 0$ and $\tau(\infty) < +\infty$. Then the equation (1) has a temperature wave type localized solution

$$u(t, x) = \begin{cases} A(c\tau - |x|)^{\frac{1}{p + \sigma}}, & 0 \leq x \leq c\tau \\ 0, & |x| \geq c\tau \end{cases}$$

where $A^p = \frac{cp}{1 - p}$. In the multidimensional case, when $m = -2(N-1)$, $N \geq 2$, equation (1) also has a

temperature wave type weak solution

$$u(t, x) = A \left(c\tau(t) - \frac{|x|^N}{N} \right)^{\frac{1}{p + \sigma}}, \quad A^{p + \sigma} = \frac{c(p + \sigma)}{1 - p}, \quad 0 < p < 1$$

where $\tau(t)$ is the above defined function. In this case the temperature wave is concentrated only in a domain

$$|x| \leq Nc\tau^{\frac{1}{N}} < \infty, \quad N \geq 2,$$

where the constant c is a velocity of a temperature wave. This is a new nonlinear effect.

6. STRONG ABSORPTION CASE ($0 < \beta < 1$).

A. A. Samarskii [2] notice that the case $0 < \beta < 1$ is very difficult and it is not studied enough. In this case we prove the following theorem about global solvability of the solution of problem (1)-(2).

Theorem 5. Let $u_0 \leq T^{-1/(\beta-1)} \bar{f}(\xi)_{t=0}$, $x \in \mathbb{R}^N$, $0 < p < 1$, $m < 2$,

$$\text{where } \bar{f}(\xi) = \left(a - \frac{\sigma + p}{4} \xi^2 \right)_+^{\frac{1}{\sigma+p}}, \quad \xi = \frac{2}{2-m} |x|^{\frac{2-m}{2}} / (\tau(t))^{1/2}, \quad a > 0 \text{ is a constant,}$$

$$\tau(t) = \int \left[T + (1-\beta) \int \gamma(t) dt \right]^{\frac{\sigma+p}{1-\beta}} dt \quad \text{and}$$

$$\gamma(t) \tau(t) \left[T + (1-\beta) \int \gamma(t) dt \right]^{\frac{\beta-2+p+\sigma}{1-\beta}} \leq \frac{N}{(1-p)(2-m)}, \quad t > 0$$

Then the problem (1)-(2) has a global solution for which in Q the estimate $u(t, x) = \left[T + (1-\beta) \int \gamma(t) dt \right]^{-1/(1-\beta)} \bar{f}(\xi)$, $T > 0$ is valid.

Corollary 3. Let $\gamma(t) = \gamma = \text{const}$. Then the condition of global solvability is

$$2N - [\sigma + p + (2-m)(1-p)] \leq \beta < 2 - (p + \sigma).$$

REFERENCES

- [1] Fujita B. H. On the blowing up of solutions to the Cauchy problem for $u_t = \Delta u + u^{1+\alpha}$, J. Fac. Sci. Univ. Tokio, 1966., Sect. IA. **13** 1(1966), 109-127.
- [2] Samarskii, A. A., Kurdyumov, S. P., Galaktionov, V. A., Mikhajlov, A.P., Blow up in a problem for quasilinear parabolic equations, Moscow, Science 1987.
- [3] Galaktionov, V. A., Kurdyumov, S. P., Mikhajlov, A.P., Samarskii, A. A., On the unbounded solutions of a Cauchy problem for the parabolic equation $u_t = \nabla(u^\sigma \nabla u) + u^\beta$, RAS, USSR, 252 6(1980), 1362-1364.
- [4] Yuxiang Li, Weibing Deng, Chunhong Xie. Global Existence and nonexistence for degenerate parabolic systems. Proc. Am. Math. Soc. ISO (2002). 3661-3670.
- [5] Aripov, M., Asymptotics of a solution of the non-Newtonian polytropic filtration equations. ZAMM, suppl.3 (2000), **80**. 767-768
- [6] Aripov, M., Methods of the standard equations for the solution of nonlinear boundary problems. Fan.Tashkent. 1988.
- [7] Aripov, M., Approximate self-similar approach for solving quasilinear parabolic equation, in "Experimentation, Modeling, Computation in Flow Turbulence and Combustion." Willey & Sons, New York, 1997. vol.2, p. 19-26.
- [8] Aripov, M., Sadullaeva. Sh.A, Vichislitel'naya Tekhnologiya, 4 (2003), **8**, 72-78.
- [9] Aripov, M., Kabiljanova. F.A., Khaydarov, A., Vichislitel'naya Tekhnologiya, 4(2003), **8**, 79-84.

Modelarea agregatelor hidroenergetice prin metoda elementelor finite

Mircea Vereş
Universitatea din Oradea

Abstract. The paper presents problematics of modelling the operation of energy equipment (machines) by the finite element method.

1. NOȚIUNI TEORETICE

1.1. Analiza statică

Problemele statice sunt acelea în care încărcările și condițiile de frontieră nu se schimbă. Ecuațiile care descriu aceste tipuri de probleme, sunt de forma:

$$K \cdot U = F, \quad (1)$$

în care: K este matricea de rigiditate, F sunt încărcările exterioare și U sunt deplasările rezultate în urma aplicării încărcărilor.

1.2. Analiza dinamică

Modelul matematic al unei probleme de analiză dinamică are forma:

$$M\ddot{u} + C\dot{u} + Ku = F(t), \quad (2)$$

în care: K este matricea de rigiditate, F sunt încărcările exterioare, M este matricea maselor, C matricea de amortizare. Dacă numărul de noduri ale unui element este „ndf” (number of degrees of freedom), deplasarea în interiorul fiecărui element poate fi exprimată prin funcția de formă:

$$\begin{aligned} u &= N^T \cdot u_K, \\ N^T &= [N_1; N_2; \dots; N_{ndf}], \\ u_K &= [u_1; u_2; \dots; u_{ndf}]^T. \end{aligned} \quad (3)$$

Matricea maselor, pentru o anumită aplicație, poate fi calculată folosind funcțiile de formă și densitatea materialului:

$$\begin{aligned} M &= \int_{-1}^1 \int_{-1}^1 N_0^T \cdot \rho \cdot N_0 \cdot r \det(I) d\xi dn, \\ N_0 &= [N^T \dots N^T]. \end{aligned} \quad (4)$$

Pentru determinarea matricelor amortizărilor se presupune că aceasta este o combinație liniară între matricea de rigiditate și cea a maselor:

$$C = \alpha \cdot M + \beta \cdot K \quad (5)$$

Valori ridicate ale coeficientului α vor amortiza frecvențele joase, iar valori mari ale coeficientului β vor amortiza frecvențele înalte.

1.3. Determinarea frecvențelor proprii

Frecvențele proprii sunt definite ca soluții ale ecuației (2) în care lipsește termenul de amortizare:

$$M\ddot{u} + Cu = 0, \quad (6)$$

$$u = \acute{u} \cdot e^{i\omega t}. \quad (7)$$

Substituind (7) în ecuația (6) se obține expresia în ω :

$$(-\omega^2 M + K)u = 0. \quad (8)$$

Prin rezolvarea ecuației se determină frecvențele proprii:

$$f = \frac{\omega}{2\pi}. \quad (9)$$

Modurile proprii vor avea forma:

$$\Phi^T M \Phi = I. \quad (10)$$

1.4. Analiza răspunsului în frecvență

Răspunsul în frecvență se studiază în cazul soluției staționare pentru încărcări cu forțe de excitație armonice. Toate încărcările armonice vor excita sistemul simultan, cu aceleași frecvențe. În acest sens, cea mai folosită metodă de analiză a răspunsului în frecvență este metoda descompunerii modale.

Pornind de la ecuația dinamică:

$$M\ddot{u} + C\dot{u} + Ku = p(t), \quad (11)$$

deplasările nodale sunt exprimate în funcție de modurile proprii φ_l în cazul analizei frecvențelor proprii și în funcție de coordonatele generalizate q_l .

$$u = \sum q_l(t) \cdot \varphi_l(x, y) \quad (12)$$

Modurile proprii posedă următoarele proprietăți:

$$\begin{aligned} \Phi^T M \Phi &= I, \\ \Phi^T K \Phi &= \text{diag}(\omega_l^2). \end{aligned} \quad (13)$$

Pentru a descompune termenul de amortizare al ecuației, se presupune ortogonalitatea matricei de amortizare dată de relația:

$$\Phi^T C \Phi = \text{diag}(2\xi_l \omega_l), \quad (14)$$

în care ξ_l este coeficientul de amortizare pentru modul l.

După înlocuiri și transformări succesive se obține relația coordonatelor generalizate:

$$\ddot{q}_l + 2\xi_l \omega_l \dot{q}_l + \omega_l^2 q_l = \varphi_l^T p(t) \quad (15)$$

Folosind exprimarea în domeniul complex a încărcărilor armonice, se obține:

$$p(t) = \dot{I}_{\text{mag}}[p \cdot e^{i\Omega t}] \quad (16)$$

Substituind (16) în ecuația (15), rezultă:

$$\ddot{q}_l + 2\xi_l \omega_l \dot{q}_l + \omega_l^2 q_l = \varphi_l^T \cdot \bar{p} \cdot e^{i\Omega t}, \quad (17)$$

cu soluția staționară:

$$q_l(t) = \bar{q}_l \cdot e^{i\Omega t}, \quad (18)$$

din care rezultă:

$$(-\Omega^2 + 2\xi_l \cdot \omega_l \cdot \Omega_i + \omega_l^2) \cdot \bar{q}_l \cdot e^{i\Omega t} = \varphi_l^T \cdot \bar{p} \cdot e^{i\Omega t} . \quad (19)$$

Din relația (19) se obțin coordonatele generalizate q_l cu ajutorul cărora se calculează deplasările, folosind transformarea de coordonate:

$$\bar{u} = \bar{\phi} \cdot \bar{q} , \quad (20)$$

Pentru a exprima partea reală a soluției complexe a răspunsului în frecvență, se utilizează relațiile:

$$\begin{aligned} u(t) &= I_{mag}(\bar{u} \cdot e^{i\Omega t}) , \\ u(t) &= I_{mag}(\bar{u}) \cdot \cos(\Omega t) + \text{Real}(\bar{u}) \cdot \sin \Omega t , \\ u(t) &= |\bar{u}| \cdot \sin(\Omega t + \gamma) , \end{aligned} \quad (21)$$

$$\gamma = a \tan \frac{I_{mag}(\bar{u})}{\text{Real}(\bar{u})} .$$

2. DETERMINAREA MODURILOR PROPRII DE VIBRAȚIE PENTRU HIDROAGREGATUL CU PUTEREA P=9MW

2.1. Parametrii modelării

Parametrii utilizați în cadrul realizării modelului dinamic sunt :

- - Material :
- **modulul lui Young : $2 \cdot 10^{11} \text{N/m}^2$;**
- **Coefficientul lui Poisson: 0,266.**
- - Structura modelului :
- **numărul de noduri : 1853;**
- **numărul de elemente : 7328;**
- **numărul de grade de libertate : 5778;**
- **numărul relațiilor cinematice : 420**
- - Coordonatele centrului de greutate :
- **Xg : 585,7 mm;**
- **Yg : -585,3 mm;**
- **Zg : -713,6 mm.**
- - Tensorul de inerție în origine :
- 2.450e+009 7.140e+006 9.616e+006
- 7.140e+006 6.749e+010 -4.198e+006
- 9.616e+006 -4.198e+006 6.750e+010

2.2. Rezultatele modelării

Rezultatele simulării au fost obținute luând în considerare următorii parametri :

- numărul de moduri proprii calculate : 10;
- numărul de iterații :44;
- toleranța valorilor proprii obținute : $2.089 \cdot 10^{-4}$

Modurile proprii rezultate sunt prezentate în tabelul 1, iar coeficienții de formă proprie de vibrație sunt prezentați în tabelul 2, unde: Tx,Ty,Tz,Rx,Ry,Rz reprezintă încărcările și reacțiunile din lagărele hidroagregatului.

Tabelul 1. Caracterizarea modurilor proprii de vibrație

Marcarea (numerotarea) MPV	Frecvența proprie [Hz]	Caracterizarea MPV după domeniul de :	
		frecvență	deformare
1	3,3692	foarte mică	Vibrații axiale ale ansamblului rotor-arbore
2	138,61	mică	Vibrații radiale în LR și LT; încovoierea ușoară a arborelui
3	143,66	mică	Vibrații radiale în LR și LT; încovoierea puternică a arborelui între LR și RT
4	447,36	medie	Vibrații radiale mari în LRA, și mai mici în LR și LT; încovoierea arborelui în LRA
5	462,55	medie	Vibrații pronunțate în LRA cu încovoierea arborelui în LRA și foarte reduse în LR și LT
6	528,82	medie	Vibrații reduse în LR și mai mari în LT, încovoierea arborelui între LR și LT
7	935,75	mare	Vibrații longitudinale, reduse radial în toate lagărele
8	961,26	mare	Vibrații puternice în LR și LT pe direcția radială cu încovoierea arborelui între LR și LT
9	1075,3	foarte mare	Vibrații longitudinale mari și radiale în LR și LT, încovoierea arborelui între LR și LT
10	1093,0	foarte mare	Vibrații foarte puternice în LR și LT pe direcție radială cu încovoierea arborelui între LR și LT

MPV – mod propriu de vibrație; LRA – lagăr radial-axial; LR – lagăr radial; LT – lagăr turbină

Tabelul 2. Valorile încărcărilor și a reacțiunilor din lagăre

Mod propriu	Frecvența [Hz]	Tx [%]	Ty [%]	Tz [%]	Rx [%]	Ry [%]	Rz [%]
foarte mică	3,3692	0.00	0.14	0.59	99.6	0.56	0.08
mică	138,61	0.00	7.90	79.86	0.23	68.84	6.78
	143,66	0.00	79.98	7.74	0.03	6.66	59.03
Medie	447,36	0.01	0.00	0.00	0.04	7.79	0.87
	462,55	0.00	0.00	0.00	0.00	0.82	7.81
	528,82	0.00	0.00	0.00	0.00	0.07	0.08
mare	935,75	0.14	0.45	5.20	0.00	3.30	0.29
	961,26	0.02	5.18	0.44	0.00	0.27	3.32
foarte mare	1075,3	87.15	0.00	0.01	0.00	0.00	0.00
	1093,0	0.10	0.03	0.02	0.00	0.01	0.01

În figurile următoare sunt sugerate deformațiile ansamblului la frecvențele proprii calculate.

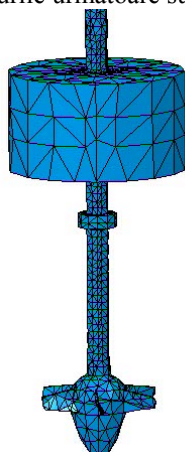


Fig. 1. Mod.1 Deformația arborelui la frecvența de 3,3692 Hz.

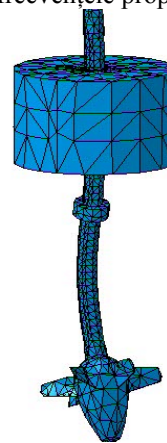


Fig. 2. Mod.2 Deformația arborelui la frecvența de 138,61 Hz

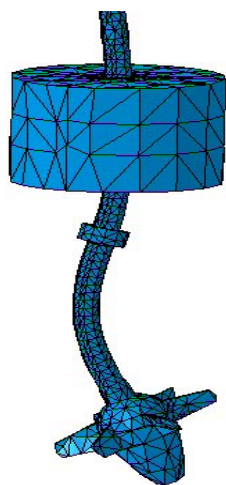


Fig. 3. Mod.3 Deformația arborelui la frecvența de 143,66 Hz.

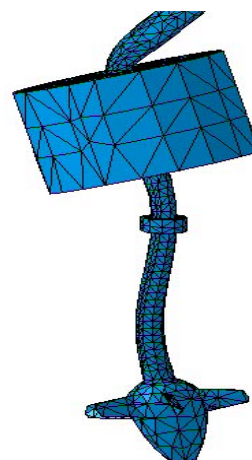


Fig. 4. Mod.4 Deformația arborelui la frecvența de 447,36 Hz

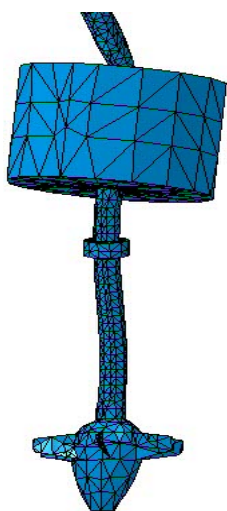


Fig.5. Mod.5 Deformația arborelui la frecvența de 462,55 Hz

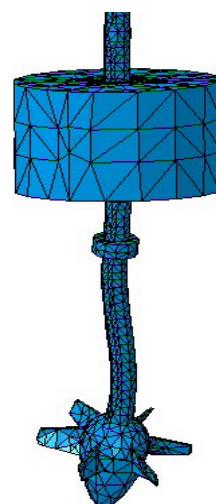


Fig. 6. Mod.6 Deformația arborelui la frecvența de 528,82 Hz

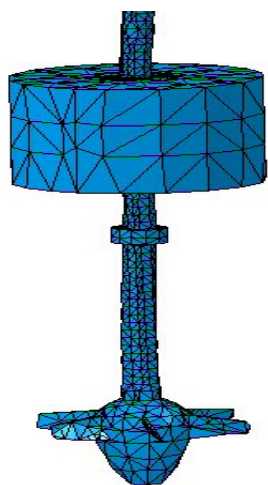


Fig. 7. Mod.7 Deformația arborelui la frecvența de 935,75 Hz.

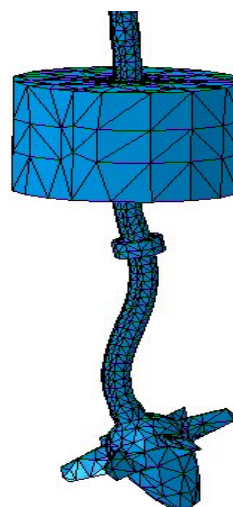


Fig. 8. Mod.8 Deformația arborelui la frecvența de 961,26 Hz

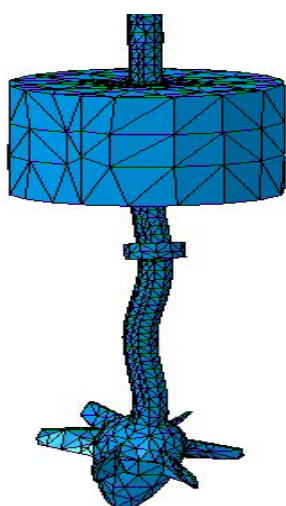


Fig. 9. Mod.9 Deformația arborelui la frecvența de 1075,3 Hz

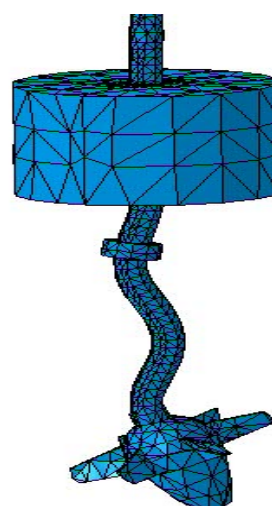


Fig. 10. Mod.10 Deformația arborelui la frecvența de 1093,0 Hz

3. CONCLUZII

1. Modelarea funcționării sistemului experimental pentru cazul utilizării unor lagăre cu alunecare și respectiv, a unor lagăre cu sustentație hidrostatică, permite, în cazul obținerii unui model corect, predicția funcționării sistemului și în alte condiții decât cele testate experimental. Cu ajutorul modelelor se poate stabili răspunsul sistemului în situații limită ce pot duce la defecte majore.

2. Pentru simulare se impune utilizarea unui soft adecvat, cum este programul de simulare “System Identification Toolbox” din cadrul modelului de programare MATLAB al firmei Math Works, și a unor

semnale de excitație sinusoidale generate, de asemenea, pentru un program de calculator; în corelație cu amplitudinea și frecvența excitației reale.

3. Pentru determinarea modurilor proprii de vibrație (MPV) ale unui agregat hidroenergetic se va face uz de modelul dinamic și de un program de calcul adecvat (CATIA).

4. Se apreciază că, pentru studiul efectelor fenomenelor vibraționale asupra gradului de deformare al ansamblului rotor-lagăre (ARL) din structura unui hidroagregat sunt acoperitoare un număr de 10 MPV, corespunzătoare la 10 valori ale frecvenței proprii, aparținând la 5 domenii (foarte mică, mică, medie, mare, foarte mare).

4. BIBLIOGRAFIE

- [1] Felea, I., Coroiu, N., *Fiabilitatea și mentenanța echipamentelor electrice*, Ed. Tehnică, București, 2002.
- [2] Felea, I., *Ingineria fiabilității în electroenergetică*, Ed. Did. și Ped., București, 1996.
- [3] Felea, I., Vereș, M., Bendea, G., Boja, I., *Considerații privind diagnoza tehnică a echipamentelor electrice*, Analele Universității din Oradea, Fascicula Energetică, nr. 9, Oradea, 2003.
- [4] Gârbea, D., *Analiză cu elemente finite*, Ed. Tehnică, București, 1990
- [5] Vereș M., *Considerații privind hidrodinamica lagărelor radiale și axiale*, Analele Universității din Oradea, Fascicula Energetică, Nr.8 vol. I, pag. 271-274, Oradea, 2002.
- [6] Vereș, M., *Cercetări vizând identificarea principiilor și metodelor de diagnoză tehnică a echipamentelor hidroenergetice*. Referat nr.2 de doctorat, Universitatea din Oradea, 2000.
- [7] Vereș, M., *Cercetări privind diagnoza tehnică a echipamentelor hidroenergetice*, Teză de doctorat, Universitatea din Oradea, 2003.

A computer package for analysis of variance

Maria PÂRV

University of Agricultural Sciences and Veterinary Medicine
 Faculty of Veterinary Medicine
 Calea Mănăştur 3-5, RO 3400 Cluj-Napoca, Romania
 e-mail: maria_parv@yahoo.com

Abstract. This paper presents a computer package for analysis of variance, running under Microsoft Windows. The programs POLIFACT, PATRLAT, and DREPTLAT have a common user interface for entering data and displaying the results. Based mainly on the general algorithm presented in [3] and [4], POLIFACT is an ANOVA-type program, performing analysis of variance for a randomised complete polyfactorial design (see [1], [2], [5] and [7]). The other two programs in the package do computations in two particular situations, where the experimental designs are latin squares and latin rectangles [6]. Besides the usual output of ANOVA, the programs produce additional information like comparison tables based on limit differences DL and Duncan tests.

1. TERMINOLOGY

The analysis of variance (or *dispersional analysis*, [7], AV for short) is a statistical method for the analysis of observational data which depend on several factors acting simultaneously, in order to establish the most important factor and to estimate their influence. This method was introduced in the agricultural research by R.A. Fisher and is based on the additive property of variance: it can be written under the form of a sum in which each term represents the variance due to one of the factors considered.

Suppose that Y is a random real variable whose values depend on p factors x_1, x_2, \dots, x_p , qualitative or quantitative. Every factor has a finite number of values, called *levels*. The cardinal of the set of values of the factor j ($1 \leq j \leq p$), i.e. the *number of levels* of j^{th} factor, is denoted by J_j . The p factors considered, x_1, x_2, \dots, x_p have the number of levels J_1, J_2, \dots, J_p .

An important component of AV deals with experiment planning; various experimental plans or schemes use the terms *block* and *cell*. A *cell* is a p -uple (i_1, i_2, \dots, i_p) , where i_j , $1 \leq j \leq p$, is a fixed level of j^{th} factor, $1 \leq i_j \leq J_j$. There are in all $J_1 \cdot J_2 \cdot \dots \cdot J_p$ cells for a so-called *complete* classification or experimental scheme. If not all cells are considered, we have an *incomplete* classification or experimental scheme.

A system of cells is called *block*; sometimes, the notions of cell and block are synonymous. For example, all the cells corresponding to the same level of the first factor belong to a *big block*; there are in all J_1 big blocks. Every big block is divided into J_2 *middle-sized blocks*; every such a intermediate block corresponds to a separate level of the second factor, and so on. In this setting, the cell is the *small block*; all experiments performed in the same cell (called *repetitions*) have in common the same combination of factor levels. An experimental scheme is *balanced* if the same number of repetitions are performed in each cell.

Usual application domains of AV include experimental sciences: chemistry, biology, psychology, sociology, medicine, but especially agricultural science.

2. NOTATION AND FORMULAE

In what follows, AV for a complete and balanced experimental plan is considered, with p factors and r repetitions. The total number of observations, N , is computed by

$$N = r \cdot \prod_{i=1}^p J_i . \quad (1)$$

The observational data are represented in a $(p+1)$ -dimensional table, denoted by X . In every cell (i_1, i_2, \dots, i_p) r experiments (repetitions) are performed; the measured value of the j^{th} experiment is denoted by $x_{i_1 i_2 \dots i_p j}$, with $1 \leq i_k \leq J_k$, $1 \leq k \leq p$, and $1 \leq j \leq r$.

2.1. Effects and interactions

In order to better understand the notions presented further down, we shall consider $p = 4$; the names of factors are A, B, C, and D, while the repetitions (considered as a *fictitious* factor) are denoted by R. For these five factors, the summation indices and their levels will be i, j, k, l, m , and I, J, K, L, M , respectively. The observational data will be x_{ijklm} , with $1 \leq i \leq I, 1 \leq j \leq J, 1 \leq k \leq K, 1 \leq l \leq L$, and $1 \leq m \leq M$.

In order to further simplify the notations, we shall specify at the summation symbol Σ only the index (indices) of summation; the absence of such an index, marked by an *, signifies the average (or mean) with respect to the corresponding factor.

For instance, the average of all observations, denoted by \bar{x} , is expressed by

$$\bar{x} \equiv x_{*****} = \frac{\sum_{i,j,k,l,m} x_{ijklm}}{I \cdot J \cdot K \cdot L \cdot M}, \quad (2)$$

the average of the i^{th} level of the A factor ($1 \leq i \leq I$) has the form

$$x_{i*****} = \frac{\sum_{j,k,l,m} x_{ijklm}}{J \cdot K \cdot L \cdot M}, \quad (3)$$

the average of the k^{th} level of the C factor ($1 \leq k \leq K$) is computed by

$$x_{**k**} = \frac{\sum_{i,j,l,m} x_{ijklm}}{I \cdot J \cdot L \cdot M}, \quad (3')$$

while the average of the combination between the i^{th} level of the A factor ($1 \leq i \leq I$) and the j^{th} level of the B factor ($1 \leq j \leq J$) is expressed by

$$x_{ij***} = \frac{\sum_{k,l,m} x_{ijklm}}{K \cdot L \cdot M}. \quad (4)$$

The averages of the combinations of three and four factors can be defined in a similar way.

We call *main effect* of the i^{th} level of the A factor ($1 \leq i \leq I$) the quantity [7]

$$\alpha_i^A = x_{i*****} - \bar{x}. \quad (5)$$

The main effects of levels j, k , and l for the factors B, C, and D, respectively, are defined analogously

$$\alpha_j^B = x_{*j***} - \bar{x}, \quad (6)$$

$$\alpha_k^C = x_{**k**} - \bar{x}, \quad (7)$$

$$\alpha_l^D = x_{***l*} - \bar{x}. \quad (8)$$

In order to point out the cooperative effects of the factors, their interactions are also considered; one defines the 2nd, 3rd and 4th order *interactions*. For instance

$$\alpha_{ij}^{AB} = x_{ij***} - \alpha_i^A - \alpha_j^B - \bar{x} \quad (9)$$

represents the interaction between the i^{th} level of the A factor and the j^{th} level of the B factor,

$$\alpha_{ijk}^{ABC} = x_{ijk**} - \alpha_{ij}^{AB} - \alpha_{ik}^{AC} - \alpha_{jk}^{BC} - \alpha_i^A - \alpha_j^B - \alpha_k^C - \bar{x} \quad (10)$$

represents the interaction between the i^{th} level of the A factor, the j^{th} level of the B factor, and the k^{th} level of the C factor, while

$$\begin{aligned} \alpha_{ijkl}^{ABCD} = & x_{ijkl*} - \alpha_{ijk}^{ABC} - \alpha_{ijl}^{ABD} - \alpha_{ikl}^{ACD} - \alpha_{jkl}^{BCD} - \alpha_{ij}^{AB} - \alpha_{ik}^{AC} - \alpha_{il}^{AD} \\ & - \alpha_{jk}^{BC} - \alpha_{jl}^{BD} - \alpha_{kl}^{CD} - \alpha_i^A - \alpha_j^B - \alpha_k^C - \alpha_l^D - \bar{x} \end{aligned} \quad (11)$$

represents the interaction between the i^{th} level of the A factor, the j^{th} level of the B factor, the k^{th} level of the C factor, and the l^{th} level of the D factor.

The equation (11), rewritten in the form

$$x_{ijkl*} = \bar{x} + \alpha_i^A + \alpha_j^B + \alpha_k^C + \alpha_l^D + \alpha_{ij}^{AB} + \alpha_{ik}^{AC} + \alpha_{il}^{AD} + \alpha_{jk}^{BC} + \alpha_{jl}^{BD} + \alpha_{kl}^{CD} + \alpha_{ijk}^{ABC} + \alpha_{ijl}^{ABD} + \alpha_{ikl}^{ACD} + \alpha_{jkl}^{BCD} + \alpha_{ijkl}^{ABCD}, \quad (12)$$

represents the decomposition of the average of the cell (i, j, k, l) into: average of the whole experiment, main effects of the factor levels and their interactions.

2.2. Variances

In order to point out the effects and interactions, the sums of squares of the corresponding deviations are computed. By means of the fictitious factor R, one can introduce, among the parameters α , defined in section 2.1, other parameters which measure the experimental errors.

The variance σ^2 is defined as

$$\sigma^2 = \frac{SS}{v}, \quad (13)$$

where SS represents the sum of squares, while v denotes the number of degrees of freedom. Table 1 contains the equations for those quantities, where the subscripts A, B, C, D, R have the same meaning as the superscripts for main effects and interactions.

Table 1. Equations for SS and v		
SS	v	Analogously for
$SS_A = \sum_i (\alpha_i^A)^2$	$I - 1$	B, C, D, R
$SS_{AB} = \sum_{i,j} (\alpha_{ij}^{AB})^2$	$(I - 1) \cdot (J - 1)$	AC, AD, AR, BC, BD, BR, CD, CR, DR
$SS_{ABC} = \sum_{i,j,k} (\alpha_{ijk}^{ABC})^2$	$(I - 1) \cdot (J - 1) \cdot (K - 1)$	ABD, ABR, ACD, ACR, ADR, BCD, BCR, BDR, CDR
$SS_{ABCD} = \sum_{i,j,k,l} (\alpha_{ijkl}^{ABCD})^2$	$(I - 1) \cdot (J - 1) \cdot (K - 1) \cdot (L - 1)$	ABCR, ABDR, ACDR, BCDR
$SS_{ABCDR} = \sum_{i,j,k,l,m} (\alpha_{ijklm}^{ABCDR})^2$	$(I - 1) \cdot (J - 1) \cdot (K - 1) \cdot (L - 1) \cdot (M - 1)$	

The sum of squares of the experimental errors SS_e is computed by summing up all SS sums which contain the R factor:

$$SS_e = SS_R + SS_{AR} + SS_{BR} + SS_{CR} + SS_{DR} + SS_{ABR} + SS_{ACR} + SS_{ADR} + SS_{BCR} + SS_{BDR} + SS_{CDR} + SS_{ABCR} + SS_{ABDR} + SS_{ACDR} + SS_{BCDR} + SS_{ABCDR}, \quad (14)$$

while the number of degrees of freedom v_e is the sum of the degrees of freedom corresponding to the sums SS in (14).

The sum (14) can be rewritten as

$$SS_e = SS_{eA} + SS_{eB} + SS_{eC} + SS_{eD} + SS_{eR}, \quad (15)$$

where the first four terms in the right-hand side represent the sums of the squares of the errors referred to the four factors. Table 2 lists the equations for these sums and the corresponding degrees of freedom.

Table 2. Sum of squares of the errors and the corresponding degrees of freedom	
SS	v
$SS_{eA} = SS_{AR}$	$v_{eA} = v_{AR}$
$SS_{eB} = SS_{BR} + SS_{ABR}$	$v_{eB} = v_{BR} + v_{ABR}$
$SS_{eC} = SS_{CR} + SS_{ACR} + SS_{BCR} + SS_{ABCR}$	$v_{eC} = v_{CR} + v_{ACR} + v_{BCR} + v_{ABCR}$
$SS_{eD} = SS_{DR} + SS_{ADR} + SS_{BDR} + SS_{CDR} + SS_{ABDR} + SS_{ACDR} + SS_{BCDR} + SS_{ABCDR}$	$v_{eD} = v_{DR} + v_{ADR} + v_{BDR} + v_{CDR} + v_{ABDR} + v_{ACDR} + v_{BCDR} + v_{ABCDR}$

2.3. Checking the statistical hypotheses

As the other chapters of experimental statistics, AV also needs the settlement of the extent to which the differences in the results of an experiment are (or not) placed between the limits of the experimental error. In order to point out the statistical significance of the results, we use three tests: the t test, the F test and the Duncan test.

Theoretical values of the Student's test t are used for computing the theoretical *limit differences* DL

$$DL = t_e \cdot s_d, \quad (16)$$

where s_d is the standard error of differences, and t_e is taken from statistical tables, as function of the degrees of freedom of the error e . The equations for s_d are given in Table 4.

In order to compare the different variants of a polyfactorial experiment, one usually determine the theoretical DL s corresponding to the confidence limits P 5%, P 1% and P 0,1%. Denoting by Δ the difference between the averages of comparing variants (also called *deviation*), and by DL P 5%, DL P 1%, and DL P 0,1% the corresponding limit differences, the interpretation of results is given in Table 3.

Case	Statistical significance
$ \Delta < DL$ P 5%	The deviation does not exceed the limits of experimental error.
$ \Delta \in [DL$ P 5%, DL P 1%)	The deviation is <i>statistically</i> significant
$ \Delta \in [DL$ P 1%, DL P 0,1%)	The deviation is <i>distinct</i> significant.
$ \Delta \geq DL$ P 0,1%	The deviation is <i>very</i> significant.

The F (Fisher-Snedecor) test, computed by:

$$F_{\text{computed}} = \frac{\sigma^2}{\sigma_e^2}, \quad (17)$$

where σ^2 denotes the variance of some variant (factor or factor combination), while σ_e^2 is the corresponding variance of the error, is used to determine the homogeneity of the compared variances. The computed F value, F_{computed} is compared with the theoretical F value, $F_{\text{theoretical}}$ taken from statistical tables, as a function of the corresponding degrees of freedom and a given confidence limit. The variances are homogeneous if $F_{\text{computed}} < F_{\text{theoretical}}$, i.e. the casual and systematic factors have a comparable contribution to the result. If $F_{\text{computed}} \geq F_{\text{theoretical}}$, the deviation is statistically significant into the considered confidence interval.

The F test indicates only the presence or absence of certain effects, being therefore a quantitative indicator. In the presence of effects, the AV processing continues normally with the comparison of variances based on limit differences DL .

The general formula (16) which computes DL is made explicit by Table 4 and equations (18) - (23). Table 4 contains formulae (taken from [1]) for the main effects and the second- and third-order interactions, where the order of considered factors matters.

The factors/factor combination	t_e	s_d	Analogously for
A – compares the means of the levels of A factor	t_{eA}	$s_{d_A} = \sqrt{\frac{2 \cdot SS_{eA}}{J \cdot K \cdot L \cdot M}}$	B, C, D
BA – compares the means of the levels of B to the same level of A	t_{eB}	$s_{d_{BA}} = \sqrt{\frac{2 \cdot SS_{eB}}{K \cdot L \cdot M}}$	CA, DA, CB, DB, DC
AB – compares the means of the levels of A to the same level of B	eq. (18)	eq. (19)	AC, AD, BC, BD, CD
CAB – compares the means of the levels of C to the same levels of A and B	t_{eC}	$s_{d_{CAB}} = \sqrt{\frac{2 \cdot SS_{eC}}{L \cdot M}}$	DAB, DAC, DBC
BAC – compares the means of the levels of B to	eq. (20)	eq. (21)	BAD, CAD, CBD

the same levels of A and C			
ABC – compares the means of the levels of A to the same levels of B and C	eq. (22)	eq. (23)	ABD, ACD, BCD

In the following equations, the values t_{eA} , t_{eB} și t_{eC} are taken from statistical tables (t values) for the degrees of freedom v_{eA} , v_{eB} and v_{eC} and the desired significance threshold

$$t_{eAB} = \frac{(J-1) \cdot SS_{eB} \cdot t_{eB} + SS_{eA} \cdot t_{eA}}{(J-1) \cdot SS_{eB} + SS_{eA}}, \quad (18)$$

$$s_{d_{AB}} = \sqrt{\frac{2 \cdot (J-1) \cdot SS_{eB} + SS_{eA}}{J \cdot K \cdot L \cdot M}}, \quad (19)$$

$$t_{eBAC} = \frac{(K-1) \cdot SS_{eC} \cdot t_{eC} + SS_{eB} \cdot t_{eB}}{(K-1) \cdot SS_{eC} + SS_{eB}}, \quad (20)$$

$$s_{d_{BAC}} = \sqrt{\frac{2 \cdot (K-1) \cdot SS_{eC} + SS_{eB}}{K \cdot L \cdot M}}, \quad (21)$$

$$t_{eABC} = \frac{J \cdot (K-1) \cdot SS_{eC} \cdot t_{eC} + (J-1) \cdot SS_{eB} \cdot t_{eB} + SS_{eA} \cdot t_{eA}}{J \cdot (K-1) \cdot SS_{eC} + (J-1) \cdot SS_{eB} + SS_{eA}}, \quad (22)$$

$$s_{d_{ABC}} = \sqrt{\frac{2 \cdot J \cdot (K-1) \cdot SS_{eC} + (J-1) \cdot SS_{eB} + SS_{eA}}{J \cdot K \cdot L \cdot M}}. \quad (23)$$

The comparison of variants by using limit differences DL is valid only inside the groups of factors/factor combination given in Table 4 which differ by the levels of the first factor considered, called *neighbours with respect to the classification*. In order to compare the means of other variants, multiple comparison methods need to be used. Among these methods, the Duncan test is based on the notion of significant difference DS , computed by

$$DS\ 5\% = s_x^- \cdot q_{5\%} \quad (24)$$

where s_x^- is the *mean error*, generically given by $s_x^- = \frac{s_d}{\sqrt{2}}$, s_d being the error of differences, given in Table 4.

In this case the place of the variant in the classification does not matter, as Table 5 shows.

The factors/factor combination	Degrees of freedom	Mean error s_x^-	Analogously for
A – compares the means of the levels of A factor	v_{eA} (Table 2)	$s_{x_A}^- = \sqrt{\frac{SS_{eA}}{J \cdot K \cdot L \cdot M}}$	B, C, D
BA – compares the means for all combinations of levels of B and A	v_{eB} (Table 2)	$s_{x_{BA}}^- = \sqrt{\frac{SS_{eB}}{K \cdot L \cdot M}}$	CA, DA, CB, DB, DC
CAB – compares the means for all combinations of levels of C, A and B	v_{eC} (Table 2)	$s_{x_{CAB}}^- = \sqrt{\frac{SS_{eC}}{L \cdot M}}$	DAB, DAC, DBC

The *Duncan test* uses the notion of *studentized range*, explained below. The considered range contains values (means) sorted in ascending order. The theoretical value is computed by $q = W/s$, where W is the size of the range (the difference between the largest and the smallest value), and s is the standard deviation of the error, with v degrees of freedom. The value of q also depends on the number of the values in the considered range, d , and on the significance threshold (usually 5%).

When comparing the means of variants VM, the variants are identified by their position within the considered range. Multiple comparisons are performed: the variant at the i^{th} position is compared with all variants at the j^{th} position ($j > i$), and the number of comparisons is given by $d = j - i + 1$. The theoretical values $q(v, d)$ are taken from two-entry tables and are used to compute the significant differences $DS(d)$ with the aid of equation (24).

If $|VM(j) - VM(i)| \geq DS(j-i+1)$ then the difference of means for the variants i and j is considered significant. The next step is to group all variants which have no significant differences, producing as a final result a classification of variants into non-disjoint groups.

3. SPECIAL CASES OF AV

The general algorithm described in the previous section refers to the AV processing for a polyfactorial, complete and balanced experimental plan. This section introduces two other special cases.

3.1. One-factor experiments in a latin square design

The latin square experimental design has the following features [6]:

- the general plan is a square, where both rows and columns denote a complete repetition;
- the number of blocks (b), columns (l), variants (v), and repetitions (n) are equal;
- a row (column) cannot contain two identical variants;
- the AV processing allows to eliminate both the differences between blocks and the differences between columns.

Table 6 illustrates this design, in which a cell is identified by the pair (ib, ic) , where ib represents the block index, while ic is the column index, both varying from 1 to $b = l = v$. Every cell contains two distinct informations: the index of the variant $iv_{ib, ic}$ and the observed value $val_{ib, ic}$. The variant indices attached to the cells in the row (block) i denote a complete repetition; this means that $(iv_{i,1}, iv_{i,2}, \dots, iv_{i,l})$ is a permutation of $(1, 2, \dots, v)$. Analogously, the variant indices attached to the cells in the column j also contain a complete repetition.

The AV processing in this special case is close to the general algorithm; for details, see [6].

3.2. One-factor experiments in a latin rectangle design

The latin rectangle experimental design has the following features [6]:

- the general plan is a rectangle, where both rows and columns denote a complete repetition;
- the columns are divided into sub-columns; the number of sub-columns (s) is the same for all columns;
- the number of blocks (b), columns (l), and repetitions (n) are equal;
- the total number of sub-columns ($l \cdot s$) is equal to the number of variants;
- a row (column) cannot contain two identical variants;
- the AV processing allows to eliminate both the differences between blocks and the differences between columns.

The latin square is a particular case of the latin rectangle, where the number s of sub-columns is equals to 1. The AV computations in these two special experimental designs are almost identical.

4. THE COMPUTER PACKAGE

Three programs constitute the current version of the computer package for AV: POLIFACT, DREPTLAT, and PATRLAT. The user interacts with them via a common user interface, discussed below.

4.1. The user interface

The user interface of the programs is using Romanian language. All programs in the package have some common features with respect to the user interface, as follows:

- input data from keyboard can be stored in a file;
- the produced results are displayed in a separate window and can be either saved in a text file or printed.

The main window of each program exposes the full functionality of it, being primarily dedicated to entering and visualization of the input data, while the output is displayed in a separate editor-like window.

The input data of every AV program are corresponding to a specific experiment, and can be classified into three categories:

- the general data identifying the experiment;
- the data describing the factors;
- the observational data.

A detailed description of the input data is given below; there are slight variations from program to program. The main window has two areas:

- the data area, containing the input data; the factor and observational data are displayed in a spreadsheet-like way;
- the command area, in the right part of the window, with five command buttons: **Încarcă (Load)** for loading data from a file, **Salvează (Save)** for saving the current data in a file; **Prelucrează (Process)** for performing the computations and displaying the results; **Afișează (Display)** which shows **Results Window** (see below); **Închide (Close)** which ends the execution of the program.

The buttons **Încarcă** and **Salvează** display standard Microsoft Windows dialog boxes **Open** and **Save As**. The button **Afișează** shows **Results Window** (see Figures 1 and 2).



Fig. 1. The Results Window with its menu visible.

Indicatorul urmarit: Productia de grau
 Unitatea de masura: q
 Numarul de factori: 3
 Numarul de repetitii: 3

Factorul	Denumirea factorului	Graduari	Martor
1 N	Ingrasaminte cu azot	2	0
2 K	Ingrasaminte cu potasiu	2	0
3 P	Ingrasaminte cu fosfor	3	0

Media observatiilor: 30,3889

Tabelul analizei variantei

Sursa variatiei	Suma patratelor	Grade de libertate	Patratul mediu	Proba F
N	765,44450	1	765,44450	207,188
K	16,00000	1	16,00000	4,840
NK	0,44444	1	0,44444	0,134
P	91,55556	2	45,77778	30,519
NP	20,22222	2	10,11111	6,741
KP	2,66667	2	1,33333	0,889
MKP	21,55556	2	10,77778	7,185
R	6,05556	2	3,02778	
NR	7,38889	2	3,69444	
KR	3,50000	2	1,75000	
MKR	9,72222	2	4,86111	
PR	3,44444	4	0,86111	
NPR	4,44445	4	1,11111	
KPR	8,33333	4	2,08333	
MKPR	7,77778	4	1,94444	
Eroarea N	7,38889	2	3,69444	
Eroarea K	13,22222	4	3,30556	
Eroarea P	24,00000	16	1,50000	
T o t a l	968,55560	35		

Fig. 2. The Results Window showing the computed results.

The **Results Window** contains a text box which fills all its client area and a menu **Fișiere (File)**. The text box shows all the results produced by the program; the user can scroll in both horizontal and vertical directions. The **Fișiere** menu has the following menu options:

- **Încarcă (Load)** – loads in the text box the content of a text file, previously saved with **Salvează**;
- **Salvează (Save)** – saves the content of the text box in a text file;
- **Listează (Print)** – print the content of the text box;
- **Închide (Close)** – close **Results Window** and redisplay main window.

4.2. What results are displayed

Most of the results computed by the general AV algorithm described in Section 2 are presented in a tabular form, and are displayed in the client area of the **Results Window**, shown in fig. 2:

- the general data of the problem, the factor data, and the observational matrix (optional, the check box **Datele experientei**);
- the arithmetic mean of the observed data;
- the AV table, containing columns for the source of the variation, sum of squares, degrees of freedom, mean square and F test; there are rows for main effects, and interactions of two and three factors, as well as the errors introduced by each factor;
- (optional, the check box **Diferentele limita**) a table with t values and the corresponding limit differences DL for the significance thresholds P 5%, P 1%, and P 0,1%, and all variants given in Table 4;
- (optional, the check box **Interactiuni factori**) comparison tables, one table for each variant in Table 4. There are columns for the variant identification, the averages being compared, the deviation from the witness variant (absolute value and percentual value) and the significance of the deviation;
- (optional, the check box **Testul Duncan**) Duncan test results, for each variant in Table 5. The results are included in three tables. The first table contains the differences in means; its rows and columns contain the means of variants considered, sorted in ascending order, while a cell on row i and column j ($j > i$) contains the difference between the mean of the variant i and variant j . The second table refers to the significant differences DS, with rows for the mean errors, theoretical q values, and theoretical DS values. Finally, the third table contains the final results, with rows corresponding to the compared variants and columns for variant identification, its mean, and classification. The variant classification is a single letter or a group of letters. All variants which have a common letter in the classification form a cluster in the sense that their differences in means are not statistically significant.

The frame **Parametri de afisare (Display parameters)**, disposed in the right side of the main window, contains the already referred check boxes which control the display of the above-described tables.

4.3. The POLIFACT program

The POLIFACT program performs the AV computations for randomised complete polyfactorial designs.

The *general data* identifying the experiment are disposed in the upper part of the main window (fig. 3), in the frame **Datele problemei (Problem data)**. They are:

- the name of the experiment (the text box labeled **Denumirea experientei**);
- the place of the experiment (the text box labeled **Locul experientei**);
- the date of the experiment (the text box labeled **Perioada experientei**);
- the author(s) of the experiment (the text box labeled **Autorul experientei**);
- the significance of observed data (the text box labeled **Indicatorul urmarit**);
- the unit of measure (the text box labeled **UM indicator (per ha)**);

- the number of factors (the text box labeled **Numărul de factori**);

- the number of repetitions (the text box labeled **Numărul de repetiții**).

Fig. 3. The main window of the POLIFACT program – initial view.

Factorul	Cod	Denumire	Graduări	Martor
1				
2				
3				

Fig. 4. The main window of the POLIFACT program after entering the general data and pressing the button **Setează tabelul de factori**

The *data describing the factors* are contained in a table disposed in the middle part of the **Datele problemei** frame. This table is configured after the general data are already set, by pressing the command button **Setează tabelul de factori** (**Set factor table**); every line of the table describes a separate factor, having the following columns (fig. 4):

- the factor identifier (code), a letter used to unique by identify the factor in the displayed results; every upper-case letter is allowed, the letter **R** identifies the repetitions;
- the factor name;
- its number of levels – at least two;
- the index of the witness level, with valid values between 0 (the mean with respect to the current factor) and the number of levels.

Factor data can be entered rowwise (**pe linii** option button, default) or columnwise (**pe coloane**), as the frame **Introducerea factorilor (Entering factor data)** specifies.

Observational data are contained in a separate table, in the lower part of the **Datele problemei** frame. This table is configured after the factor data are already in place, by pressing the button **Setează tabelul de observații (Set observation table)**. Its number of rows equals the number of factor combinations, whereas the number of columns equals the number of repetitions (see Figure 5). Table header contains (from left to right) columns with factor identifiers, followed by columns for repetitions. The cells in the factor columns contain the factor levels, being automatically generated, while those in the repetition columns contain the observed data.

The screenshot shows the main window of the POLIFACT program. The title bar reads "PolifAct - analiza varianței pentru experiențe polifactoriale randomizate complet". The window is divided into several sections:

- Datele problemei:** C:\Mimi\vb\PoliFact\sau3.dat
- Denumirea experienței: Test trifactoriale Saulescu p. 262
- Locul experienței: Lovrin
- Perioada experienței: 1967
- Autorul experienței: Saulescu
- Indicatorul urmărit: Productia de grau
- UM indicator (per ha): q
- Numărul de factori: 3
- Numărul de repetiții: 3

Buttons: Salvează, Încarcă, Setează tabelul de factori, Introducerea factorilor (pe linii, pe coloane), Setează tabelul de observații, Introducerea observațiilor (pe linii, pe coloane), Parametri de afișare (Datele experienței, Diferențele limita, Interacțiuni factori, Testul Duncan), Prelucreează, Afișează, Închide.

Factorul	Cod	Denumire	Graduări	Martor
1	N	Ingrasaminte cu azot	2	0
2	K	Ingrasaminte cu potasiu	2	0
3	P	Ingrasaminte cu fosfor	3	0

N	K	P	R1	R2	R3
1	1	1	22	22	21
1	1	2	29	27	30
1	1	3	30	30	26
1	2	1	23	23	24
1	2	2	24	27	25
1	2	3	26	27	28
2	1	1	36	37	33
2	1	2	36	36	35
2	1	3	37	36	36
2	2	1	35	31	31
2	2	2	36	34	35
2	2	3	37	34	35

Fig. 5. The main window of the POLIFACT program after entering the factor data and pressing the button **Setează tabelul de observații**.

Observational data can be entered rowwise (**pe linii**, default) or columnwise (**pe coloane**), as the frame **Introducerea observatiilor (Entering observational data)** specifies. After all the data are in place, the main window looks as Figure 6 shows. The name of the current data file is shown in the caption of data area frame.

Fig. 6. The main window of the POLIFACT program – input data loaded from a file.

4.4. The PATRLAT program

The PATRLAT program performs the AV computations for latin square experimental designs. Its main window is shown in fig. 7; the command area resembles well those in POLIFACT main window, whereas the data areas are different.

The new controls contained in the *general data area* are as follows:

- the number of variants (the text box labeled **Numărul de variante**);
- the number of blocks (the text box labeled **Numărul de blocuri**);
- the significance of the variants (the text box labeled **Semnificația variantelor**);
- the index of the witness variant, with valid values between 0 (the mean with respect to the mean of the experiment) and the number of variants (the text box labeled **Varianta martor**).

As fig. 7 shows, there is no factor table, because the experiments are considered monofactorial. There are data for a sole factor - called here variant – (the significance, the number of levels, and the witness variant) included in the general data area.

Observational data are contained in a separate table, in the lower part of the **Datele problemei** frame. This table is configured after the factor data are already in place, by pressing the button **Setează tabelul de observații (Set observation table)**. Its number of rows and columns are the same, i.e. the number of blocks, respectively columns. The first column denotes the block, while the next ones are denoting the columns of the experimental plan (Roman numerals). The cells in the columns part contain the observed data. Every cell contains two values, separated by a semicolon: the index of the variant set in the current block and column, and the observed value, as discussed in Section 3.1.

Validations include both formal (numerical values) and logical (the variant indices lying between 0 and the number of variants) tests.

Datele problemei: C:\Mim\vb\PatrLat\PL Saulescu p 217.dat

Denumirea experienței: Test patrat latin Saulescu p. 217

Locul experienței: Lovrin Perioada experienței: 1967 toamna

Autorul experienței: Saulescu Indicatorul urmărit: Productia de porumb

UM indicator: q Numărul de variante: 7 Numărul de blocuri: 7

Semnificatia variantelor: Soiuri Varianta mator: 0

Suprafata parcelei recoltabile (mp): 10000

Setează tabel observații **Culegere date observații**
 pe linii pe coloane

Bloc	I	II	III	IV	V	VI
1	1;38.8	2;49.8	3;45.9	4;45.3	5;49.8	6;49.4
2	6;35.7	7;68.8	4;40.2	1;60.4	3;59	5;60.6
3	4;42.1	1;42	6;47.1	3;55.7	7;42.9	2;72.6
4	5;45.1	4;39.6	7;44.1	2;68.8	6;44.5	3;60.8
5	3;45.9	6;40.4	1;66.7	5;62	2;75.1	7;66.9
6	7;65.3	5;53.9	2;77.3	6;49	1;68.2	4;53.5
7	2;72.6	3;49.6	5;45.3	7;67.8	4;58.8	1;61.2

Parametri de afișare

Datele experienței

Diferențele limita

Interacțiuni factori

Testul Duncan

Salvează **Încarcă**

Prelucrează

Afișează

Închide

Fig. 7. The main window of the PATRLAT program.

4.5. The DREPTLAT program

The DREPTLAT program performs the AV computations for latin rectangle experimental designs. Its main window is shown in fig. 8. The only differences with respect to the PATRLAT main window are:

- an additional text box placed in the data area, which contains the number of columns (the text box labeled **Numarul de coloane**);
- the columns in the table of observed data are splitted into sub-columns. A typical sub-column header is of the form $C-s$, where C is the column index (Roman numeral), and s is the sub-column index (Arab numeral).

The number of columns must divide the number of variants, as discussed in 3.2. Every cell in the sub-column area contains two values, separated by a semicolon: the variant index and the observed value.

DreptLat - analiza varianței pentru experiențe monofactoriale asezate după metoda dreptunghiului latin

Datele problemei: C:\Mimi\vb\DreptLat\Saulescu p 221.dat

Denumirea experienței: Test dreptunghi latin Saulescu p. 221

Locul experienței: Lovrin Perioada experienței: 1967 toamna

Autorul experienței: Saulescu Indicatorul urmărit: Productia de grau

UM indicator (per ha): q Numărul de variante: 10 Numărul de blocuri: 5

Semnificatia variantelor: Linia Varianta martor: 10

Suprafata parcelei recoltabile (mp): 10000

Numărul de coloane: 5

Setează tabelul de observații **Culegere date observații**
 pe linii pe coloane

Bloc	I-1	I-2	II-1	II-2	III-1	III-2	IV-1	IV-2	V-1
1	1,37	2,40	3,39	4,43	5,35	6,41	7,39	8,38	9,42
2	7,35	10,40	8,38	1,39	3,37	9,38	5,37	2,39	4,40
3	6,39	3,35	5,28	2,36	10,33	8,30	9,40	4,36	7,36
4	5,34	9,37	6,39	7,35	2,33	4,37	10,31	1,35	3,35
5	4,39	8,32	10,42	9,34	7,33	1,38	6,33	3,44	5,34

Parametri de afișare:
 Datele experienței
 Diferențele limita
 Interacțiuni factori
 Testul Duncan

Salvează Încarcă

Prelucrează

Afișează

Închide

Fig. 8. The main window of the DREPTLAT program.

5. Conclusions and further work

The programs described in this paper are used currently in our University. The users of the old versions [4, 5] appreciate both the user-friendly graphical interface and their new features. Future versions of the package will include new programs dedicated to other special cases of AV.

REFERENCES

- [1] Cochran, W.G, C. Cox, *Experimental designs*, 2nd ed., John Wiley and Sons, 1957.
- [2] Hartley. H.C., *Varianzanalyse*, in *Mathematische Methoden für Digitalrechner* (eds. Ralston, A., Wilf, H.S.), R. Oldenbourg Verlag, München, 1972. (German)
- [3] IBM Scientific Subroutine Package (SSP), IBM Corp., 1969.
- [4] Pârv, B., M. Pârv, *ANVAR – A Software package for analysis of variance*, in *Lucrările Conferinței Naționale de Matematică Aplicată și Mecanică*, 20-23 oct. 1988, Institutul Politehnic Cluj-Napoca, vol. II, 513-522.
- [5] Pârv, B., D. Stegăroiu, G. Morar, M. Pârv – *Ghid de utilizare: AVEPRC Analiza varianței privind experiențele polifactoriale randomizate complet*, Sistem-model-calculator. Aplicații, Tipografia Agronomia, Cluj-Napoca, 1988.
- [6] Săulescu, N.A., N. N. Săulescu, *Câmpul de experiență*, ed. a II-a, Ed. Agro-Silvică, București, 1967.
- [7] Văduva, I., *Analiza dispersională*, Ed. Tehnică, București, 1970.

**EDUCATION SECTION
(SECTIUNEA DIDACTICA)**

Inegalități algebrice folosite în demonstrarea unor inegalități geometrice

Liliana Antonescu,
Școala "Liviu Rebreanu" Mioveni-Argeș

Inegalitățile sunt importante în toate domeniile matematicii, uneori chiar mai mult decât egalitățile.

Inegalitățile geometrice sunt atractive în special, deoarece pot fi ușor înțelese și în același timp ele reprezintă o introducere excelentă în gândirea matematică creativă și în spiritul matematicii moderne. De multe ori inegalități geometrice sau trigonometrice se demonstrează cu ajutorul unor inegalități algebrice.

Inegalitatea mediilor ne ajută să demonstrăm multe inegalități legate de elementele unui triunghi sau ale altor figuri sau corpuri geometrice.

Teorema 1. Dacă x_1, x_2, \dots, x_n sunt numere pozitive, atunci

$$\min_{1 \leq i \leq n} \{x_i\} \leq m_h \leq m_g \leq m_a \leq m_p \leq \max_{1 \leq i \leq n} \{x_i\} \quad (1)$$

unde $m_h = \frac{n}{\frac{1}{x_1} + \frac{1}{x_2} + \dots + \frac{1}{x_n}}$ este media lor armonică, $m_g = \sqrt[n]{x_1 \cdot x_2 \cdot \dots \cdot x_n}$ este media lor geometrică,

$m_a = \frac{x_1 + x_2 + \dots + x_n}{n}$ este media lor aritmetică și $m_p = \sqrt{\frac{x_1^2 + x_2^2 + \dots + x_n^2}{n}}$ este media lor pătratică.

Să dăm câteva aplicații ale inegalității (1).

Problema 1. Dintre toate paralelipedele dreptunghice cu arie totală dată, cubul are volumul cel mai mare.

Demonstrație. Să notăm cu a, b și c lungimea, lățimea și înălțimea unui paraleliped dreptunghic cu aria totală S și volumul V . Evident, $V=abc$ and $S=2(ab+bc+ca)$. Din ipoteza că S este constantă rezultă că suma celor trei numere ab, bc și ac este constantă. Putem aplica Teorema 1 pentru aceste numere și avem că

$$(ab \cdot bc \cdot ca)^{1/3} \leq \frac{ab + bc + ca}{3} \quad \text{sau } (V^2)^{1/3} \leq S/6. \quad \text{Astfel, } V \leq (S/6)^{3/2}. \quad \text{Egalitatea are loc dacă și numai dacă}$$

$ab=bc=ac$. De aici rezultă că volumul este maxim dacă $a=b=c$, i.e. atunci când paralelipedul dreptunghic este cub.

Problema 2. Un cilindru circular drept cu volumul V are arie minimă dacă diametrul său este egal cu înălțimea.

Demonstrație. Notând cu S, r și h , respectiv aria, raza și înălțimea unui cilindru circular drept cu volumul V , avem că $S=2\pi(r^2 + rh)$ și $V = \pi r^2 h$. Deci, $S = 2\pi \left(r^2 + \frac{V}{\pi r} \right) = 2\pi \left(r^2 + \frac{V}{2\pi r} + \frac{V}{2\pi r} \right)$. Putem considera, astfel, că $S/(6\pi)$ este media aritmetică a numerelor $r^2, V/(2\pi r)$ și $V/(2\pi r)$, deci cu Teorema 1,

obținem $\frac{S}{6\pi} \geq \left(\frac{V^2}{4\pi^2} \right)^{1/3}$. Membrul drept al inegalității este constant. Deci S este minimă când are loc

egalitate, și anume $r^2 = V/(2\pi r)$ sau $V = 2\pi r^3$. Astfel, S este minimă când $2r = h$.

Să ne întoarcem acum la figurile geometrice.

Problema 3. Să se demonstreze că în orice triunghi ABC avem $6Rr \leq \sqrt[3]{a^2 b^2 c^2}$, unde notațiile sunt cele obișnuite.

Demonstrație. Să observăm că putem scrie $abc=4RS=4Rpr=2Rr(a+b+c)$. Deci $[Rr(a+b+c)]/(abc)=1/2$ sau $(Rr)/(bc)+(Rr)/(ca)+(Rr)/(ab)=1/2$. Inegalitatea dintre media geometrică și media aritmetică a numerelor $(Rr)/(bc), (Rr)/(ca), (Rr)/(ab)$ și egalitatea precedentă conduc la $(R^3 r^3)/(a^2 b^2 c^2) \leq 1/6^3$, de unde $6Rr \leq \sqrt[3]{a^2 b^2 c^2}$.

Problema 4. Să se arate că într-un triunghi oarecare ABC are loc inegalitatea $\sin A + \sin B + \sin C \geq 3[S/(2R^2)]^{1/3}$, unde notațiile sunt cele obișnuite.

Demonstrație. Să folosim inegalitatea $m_g \leq m_a$, pentru $n=3$, i.e. $\sqrt[3]{x_1 x_2 x_3} \leq \frac{x_1 + x_2 + x_3}{3}$. Să

alegem $x_1 = \sin A$, $x_2 = \sin B$, $x_3 = \sin C$ și să înlocuim $\sin A + \sin B + \sin C \geq 3 \sqrt[3]{\sin A \sin B \sin C}$, dar (folosind Teorema sinusurilor) $\sin A \sin B \sin C = S/(2R^2)$, de unde rezultă inegalitatea cerută.

Problema 5. Se dă romb $ABCD$. Sa se demonstreze că $2\sqrt{2} AB \geq AC + BD$.

Demonstrație. Fie $AC \cap BD = \{O\}$. $AO = AC/2$, $BO = BD/2$. În triunghiul AOB avem $AB^2 = AO^2 + BO^2$, deci $AB = \sqrt{\frac{AC^2 + BD^2}{4}}$, $AB = \frac{1}{\sqrt{2}} \sqrt{\frac{AC^2 + BD^2}{2}}$. Aplicând Teorema 1 avem $AB = \frac{1}{\sqrt{2}} \sqrt{\frac{AC^2 + BD^2}{2}} \geq \frac{1}{\sqrt{2}} \cdot \frac{AC + BD}{2}$. Deci $2\sqrt{2} AB \geq AC + BD$.

Problema 6. Să se demonstreze că în orice triunghi ABC este satisfăcută inegalitatea $a \cdot \operatorname{tg}(A/2) + b \cdot \operatorname{tg}(B/2) + c \cdot \operatorname{tg}(C/2) \geq 6p \operatorname{tg}(A/2) \operatorname{tg}(B/2) \operatorname{tg}(C/2)$.

Demonstrație. Numerele $a \cdot \operatorname{tg}(A/2)$, $b \cdot \operatorname{tg}(B/2)$, $c \cdot \operatorname{tg}(C/2)$ sunt pozitive. De aceea inegalitatea mediilor ne permite să scriem $a \cdot \operatorname{tg} \frac{A}{2} + b \cdot \operatorname{tg} \frac{B}{2} + c \cdot \operatorname{tg} \frac{C}{2} \geq 3 \sqrt{abc \cdot \operatorname{tg} \frac{A}{2} \operatorname{tg} \frac{B}{2} \operatorname{tg} \frac{C}{2}}$. Calculând produsul P al tangențelor jumătăților de unghiuri, găsim $P = \operatorname{tg}(A/2) \operatorname{tg}(B/2) \operatorname{tg}(C/2) = S/(p^2) = r/p$. Cantitatea de sub radical (cu $abc = 4RS$, $S = p^2 P$, $1/p = P/r$) se scrie $4RSP = 4R \cdot p^2 P \cdot P = 4Rp^3 \cdot (1/p) \cdot P^2 = p^3 \cdot 4R \cdot (P/r) \cdot P^2 = p^3 \cdot P^3 \cdot 4(R/r)$. Relația lui Euler, care dă distanța d dintre centrele cercurilor circumscris și înscris într-un triunghi și anume $d^2 = R^2 - 2Rr (\geq 0)$, ne arată că $(R/r) \geq 2$.

Teorema 2. Dacă x_1, x_2, \dots, x_n sunt numere pozitive, atunci $(x_1 + x_2 + \dots + x_n)[(1/x_1) + (1/x_2) + \dots + (1/x_n)] \geq n^2$. Această inegalitate este de fapt $m_h \leq m_a$, scrisă puțin altfel. Scriind această inegalitate pentru $n = 3$, avem $(x_1 + x_2 + x_3)[(1/x_1) + (1/x_2) + (1/x_3)] \geq 9$.

Această ultimă inegalitate are foarte multe aplicații, cu ajutorul ei putându-se demonstra multe inegalități relative la elementele unui triunghi.

Problema 7. Sa se arate că într-un triunghi ABC există relația $h_a + h_b + h_c \geq 9r$, notațiile fiind cele obișnuite.

Demonstrație. Să alegem $x_1 = h_a$, $x_2 = h_b$, $x_3 = h_c$ și să ținem cont că aria unui triunghi poate fi scrisă $S = (1/2)ah_a = (1/2)bh_b = (1/2)ch_c = pr$, și deci $1/h_a = a/(2S)$, $1/h_b = b/(2S)$, $1/h_c = c/(2S)$, $1/r = (2p)/(2S)$. Obținem că $(1/h_a) + (1/h_b) + (1/h_c) = 1/r$. Folosind inegalitatea din Teorema 2 avem $(h_a + h_b + h_c)[(1/h_a) + (1/h_b) + (1/h_c)] \geq 9$, deci $(h_a + h_b + h_c)/r \geq 9$.

Problema 8. Să se arate că într-un triunghi are loc inegalitatea $r_a + r_b + r_c \geq 9r$, unde r_a, r_b, r_c sunt razele cercurilor exânscrie triunghiului.

Demonstrație. Este ușor de demonstrat că aria triunghiului, S , se mai poate scrie și astfel

$S = r_a(p-a) = r_b(p-b) = r_c(p-c) = pr$ și, deci, $(1/r_a) + (1/r_b) + (1/r_c) = [(p-a)/S] + [(p-b)/S] + [(p-c)/S] = (3p-2p)/S = 1/r$. Inegalitatea cerută este acum evidentă.

Să demonstrăm acum, cu ajutorul inegalităților algebrice o inegalitate cunoscută ca inegalitatea lui Euler $R \geq 2r$. Folosim tot Teorema 2, unde drept x_1, x_2, x_3 luăm tot razele cercurilor exânscrie triunghiului.

Atunci $(r_a + r_b + r_c)(r_a r_b + r_b r_c + r_c r_a) \geq 9r_a r_b r_c$. Pe de altă parte, $r_a + r_b + r_c = 4R + r$, $r_a r_b + r_b r_c + r_c r_a = p^2$, $r_a r_b r_c = p^2 r$. Deci $(4R + r)p^2 \geq 9p^2 r$, de unde rezultă imediat ca $R \geq 2r$.

O altă inegalitate algebrică, folositoare în multe situații este:

Inegalitatea lui Jensen. Fie E un interval $f: E \rightarrow \mathbf{R}$. Dacă f este convexă pe E , atunci oricare ar fi $x_1, x_2, \dots, x_n \in E$ și oricare ar fi numerele pozitive p_1, p_2, \dots, p_n , are loc inegalitatea

$$f\left(\frac{p_1 x_1 + p_2 x_2 + \dots + p_n x_n}{p_1 + p_2 + \dots + p_n}\right) \leq \frac{p_1 f(x_1) + p_2 f(x_2) + \dots + p_n f(x_n)}{p_1 + p_2 + \dots + p_n}.$$

Să aplicăm inegalitatea lui Jensen în:

Problema 9. Să se arate că într-un triunghi oarecare are loc inegalitatea $\sin A + \sin B + \sin C \leq \frac{3\sqrt{3}}{2}$,

egalitatea având loc când triunghiul este echilateral.

Demonstrație. Se ia funcția $f(x) = \sin x$ pe intervalul $[0, \pi]$. Funcția este concavă pe acest segment. Atunci, luând două puncte $x_1, x_2 \in [0, \pi]$, relația $[f(x_1)+f(x_2)]/2 = (\sin x_1 + \sin x_2)/2$, devine $[f(x_1)+f(x_2)]/2 = \sin[(x_1+x_2)/2] \cos[(x_1-x_2)/2] \leq \sin[(x_1+x_2)/2]$, deoarece $0 \leq \cos[(x_1-x_2)/2] \leq 1$, pentru orice $(x_1-x_2)/2$. Deci $(\sin x_1 + \sin x_2)/2 \leq \sin[(x_1+x_2)/2]$. Inegalitatea lui Jensen, aplicată pentru $\sin x$, $x \in [0, \pi]$ și $x_i \in [0, \pi]$, $i = 1, 2, \dots, n$, ne dă $\frac{1}{n} \sum_{i=1}^n \sin x_i \leq \sin\left(\frac{1}{n} \sum_{i=1}^n x_i\right)$. Atunci, luând $x_1 = A$, $x_2 = B$, $x_3 = C$ cu $A + B + C = \pi$, $A, B,$

$C > 0$, avem imediat $(\sin A + \sin B + \sin C)/3 \leq \sin(\pi/3) = \frac{\sqrt{3}}{2}$ și deci inegalitatea cerută a rezultat imediat.

Acest rezultat se poate aplica imediat și în:

Problema 10. În orice triunghi se verifică inegalitatea $(r_a/a)(r_b/b)(r_c/c) \leq \left(\frac{\sqrt{3}}{2}\right)^3$.

Demonstrație. Inegalitatea din enunț se transformă succesiv

$$(pS^2)/S^3 \leq \frac{3\sqrt{3}}{8} abc \Leftrightarrow \frac{p}{R} \leq \frac{3\sqrt{3}}{2} \Leftrightarrow \sin A + \sin B + \sin C \leq \frac{3\sqrt{3}}{2}. \text{ Ultima inegalitate este cea}$$

demonstrată anterior cu ajutorul inegalității lui Jensen.

O altă inegalitate algebrică, inegalitatea *Cauchy-Buniakovski-Schwartz*, poate fi de asemenea folosită în demonstrarea unor inegalități geometrice. Folosind această inegalitate obținem

$$\sin \alpha \cos \beta + \sin^2 \beta \leq \sqrt{\sin^2 \alpha + \sin^2 \beta} \cdot \sqrt{\cos^2 \beta + \sin^2 \beta}, \forall \alpha, \beta \in R$$

$$\sin \beta \cos \alpha + \sin^2 \alpha \leq \sqrt{\sin^2 \alpha + \sin^2 \beta} \cdot \sqrt{\cos^2 \alpha + \sin^2 \alpha}, \forall \alpha, \beta \in R$$

Adunând cele două inegalități membru cu membru obținem inegalitatea

$$\forall \alpha, \beta \in R, \sin^2 \alpha + \sin^2 \beta + \sin(\alpha + \beta) \leq 2\sqrt{\sin^2 \alpha + \sin^2 \beta}, \quad (2)$$

egalitatea având loc dacă $\alpha = k_1\pi$, $\beta = k_2\pi$, $k_1, k_2 \in Z$ sau $\alpha + \beta = 2k\pi + (\pi/2)$, $k \in Z$.

Să indicăm acum două aplicații ale acestei inegalități și anume:

Problema 11. În orice triunghi ABC are loc inegalitatea

$$\sin^2 A + \sin^2 B + \sin C \leq 2\sqrt{\sin^2 A + \sin^2 B}, \quad (3)$$

egalitatea având loc dacă $m(C) = 90^\circ$.

Demonstrație. Se alege $\alpha = A$, $\beta = B$ și se aplică inegalitatea (2) ținând cont că $\sin(A+B) = \sin C$.

Problema 12. Să se arate că în orice triunghi ABC avem $a^2 + b^2 + 2Rc \leq 4R\sqrt{a^2 + b^2}$.

Demonstrație. Această inegalitate se obține ușor folosind (3) și Teorema sinusurilor.

Inegalitățile algebrice sunt o sursă importantă de rezultate în domeniul geometriei și în special în cel al geometriei triunghiului.

BIBLIOGRAFIE

- [1] Georgescu-Buzău, E., Onofraș, E., Metode de rezolvare a problemelor de matematică în liceu, Editura Didactică și Pedagogică, București, 1983.
- [2] Kazarinoff, N., Geometric inequalities, New Mathematical Library, 1961.
- [3] Voda, V., Triunghiul – ringul cu trei colțuri, Editura Albatros, București, 1979.
- [4] Tudor, I., Probleme de geometrie pentru gimnaziu, Editura Paco, 1992.
- [5] Gazeta Matematica, Nr.7-8/1993, Societatea de Științe Matematice din România.
- [6] Gazeta Matematica, Nr.4/1981, Societatea de Științe Matematice din R.S.R.

Mulțimea lui Mandelbrot

Ștefăniță Hăbuc
Școala „Liviu Rebreanu” Mioveni-Argeș

Studiul sistemelor dinamice în planul complex a luat amploare odată cu apariția calculatorului și reprezentări grafice. Surprinzătoarele imagini obținute au incitat și mai mult studiul atât din punct de vedere matematic, cât și din cel informatic furnizând o teorie în acest sens. Bazele matematice au fost pornite încă de la începutul secolului XX de către matematicianul francez Gaston Julia, dezvoltată mai apoi de Benoit Mandelbrot al cărui nume îl poartă mulțimea la care vom face referire.

Plecând de la spațiul complex C organizat ca un spațiu topologic și alegând o funcție polinomială $f: C \rightarrow C$, $f(z) = a_0 + a_1z + a_2z^2 + \dots + a_nz^n$, cu $n \geq 2$ și $a_k \in C$ pentru $k = 0, n$, în spațiul fazelor C funcția f generează un sistem dinamic sau semidinamic discret.

Fie w un punct periodic de perioadă p , cu $(f^p)'(w) = \lambda$. Punctul w este numit: *superatractiv* dacă $\lambda = 0$, *atractiv* dacă $0 < |\lambda| < 1$, *indiferent* dacă $|\lambda| = 1$ și *repulsiv* dacă $|\lambda| > 1$. Închiderea mulțimii punctelor repulsive a sistemului discret și deci a funcției f se numește *mulțime Julia* și va fi notată cu $J(f)$.

Vom considera cazul când funcția f este $f(z) = z^2 + c = f_c(z)$. Numim **mulțime Mandelbrot** (notată cu \mathbf{M}), mulțimea parametrilor c pentru care mulțimea Julia a lui f_c este conexă. Adică avem $\mathbf{M} = \{c \in C \mid J(f_c) \text{ este conexă}\}$. La prima vedere \mathbf{M} apare ca o mulțime rezultată dintr-o restricție impusă mulțimii $J(f)$, de fapt, după cum vom vedea, \mathbf{M} conține o mulțime de informații referitoare la structura mulțimilor Julia.

Definiția dată este destul de incomodă pentru scopurile noastre de a face reprezentarea pe calculator. De aceea vom da o definiție echivalentă cu aceasta, care este mult mai comodă pentru a investiga forma cu totul excepțională a mulțimii Mandelbrot și de a determina efectele reprezentării acesteia în planul complex al parametrului c .

Se poate demonstra că mulțimea Mandelbrot poate fi caracterizată în funcție de șirul de iterate al funcției f_c în punctul zero. Astfel $\mathbf{M} = \{c \in C \mid \{f_c^k(0)\}_{k \geq 1} \text{ este mărginit}\}$ sau $\mathbf{M} = \{c \in C \mid f_c^k(0) \rightarrow \infty \text{ când } k \rightarrow \infty\}$.

Dinamica punctelor este studiată cu ajutorul mulțimilor de nivel. Alegem o rețea de puncte $c \in C$ și testăm pentru fiecare punct dacă după N iterații termenul corespunzător șirului $0 \rightarrow c \rightarrow c^2 + c \rightarrow \dots$, depășește sau nu un disc centrat în origine de rază suficient de mare. Această rețea se va găsi în interiorul unui dreptunghi ale cărui dimensiuni $x_{\min}, y_{\min}, x_{\max}, y_{\max}$ vor fi citite inițial de la terminal. Așadar, pentru un punct (x, y) din rețea se va parcurge următoarea procedură:

Procedura Mandelbrot

$k \leftarrow 0$

$x_n \leftarrow x$

$y_n \leftarrow y$

cat timp $(x_n^2 + y_n^2 < M)$ și $(k < N)$ repetă

$k \leftarrow k + 1$

$aux \leftarrow x_n^2 - y_n^2 + x$

$y_n \leftarrow 2 * x_n * y_n + y$

$x_n \leftarrow aux$

stop procedura

Parametrii M și N vor fi aleși astfel încât convergența să fie destul de rapidă dar să se obțină și o imagine cât mai bună. De exemplu în experiențele noastre (realizate în limbajele Pascal și C) am folosit $M=100$ și $N=64$. Vom observa că dacă mărim pe N imaginea devine din ce în ce mai bună, iar convergența nu este mai rapidă și, în consecință, timpul de redare al imaginii este mai mare.

În funcție de valoarea lui k se va face și colorarea punctului din interiorul porțiunii dreptunghice aleasă pentru reprezentare. Este indicat să alegem o schemă de culori astfel încât pentru valori foarte mari ale lui k se va calcula $k \bmod nrcul$, unde $nrcul$ reprezintă numărul maxim de culori alese.

Analizând reprezentarea din fig. 1 a mulțimii Mandelbrot observăm mai întâi o regiune delimitată prin o cardioidă, simetrică față de o axă centrală cu adâncitura de $+0.25$ și cuprinsă la stânga până la -0.75 . Urmează apoi un disc cu centrul în -1 și raza 0.25 , tangent la cardioidă. Mai observăm o mulțime infinită de discuri tangente la M , iar pentru ele există de asemenea o infinitate de discuri tangente.

Mergând mereu în stânga, pornind din cardioidă, se ajunge la punctul numit Myrbert-Feigenbaum, situat la -1.401 . În acest punct se poate observa apariția primei noi cardioide. Segmentul conținut între acest punct și -2 este conținut în mulțimea lui Mandelbrot. Pe acest segment se găsesc o infinitate de componente cardioide, foarte mici.

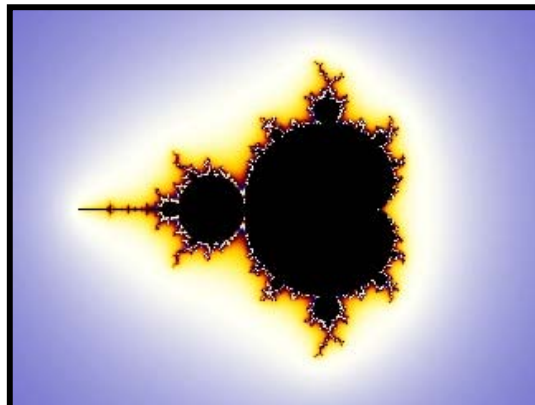


Fig. 1. Mulțimea Mandelbrot

Ceea ce este interesant, este apariția mulțimilor

Julia în structura mulțimii Mandelbrot. Folosind poziționarea parametrului c se pot observa următoarele cazuri:

- dacă c este în interiorul corpului principal al lui M , care are forma unei cardioide, mulțimea Julia se prezintă ca un cerc deformat fractal în jurul unui punct fix atractiv;
- dacă c este în interiorul unui „mugure”, mulțimea Julia constă dintr-o mulțime infinită de cercuri deformate fractal în jurul unui atractor periodic și a preimaginilor sale;
- dacă c este în punctul de germinație al unui „mugure”, avem așa numitul caz parabolic: frontiera se prezintă asemenea unor „cârcei” care se îmbogățesc spre margine în raport cu un atractor stabil;
- dacă c este pe frontiera cardioidei sau a unui mugur, avem cazul numit *discurile lui Siegel*, în interiorul regiunii limitate de mulțimea Julia găsim cercuri în jurul unor puncte fixe.

Descompunerea binară

O reprezentare interesantă a mulțimii lui Mandelbrot se poate obține folosind o tehnică denumită **descompunerea binară**. În acest caz colorarea nu se va mai face în funcție de parametru k , această tehnică constând în colorarea anumitor zone în negru și a altora în alb în funcție de condiția $y \leq 0$ ce va pusă la ieșirea din ciclu. Astfel se realizează împărțirea fiecărui nivel de atracție în benzi, ce vor crește de la nivelul exterior către cel interior. Fiecare nivel va prezenta 2^k benzi, unde k reprezintă numărul nivelului de atracție calculat de la exterior către interior, de fapt numărul de iterații ce va calculat prin procedura Mandelbrot.

Această tehnică va fi realizată folosind următoarea procedură de calcul:

Procedura Mandelbrot

```

k ← 0
xn ← x
yn ← y
cat timp (xn*xn+yn*yn<M) si (k<N) repetă
    k ← k+1
    aux ← xn*xn - yn*yn + x
    yn ← 2*xn*yn + y
    xn ← aux
daca yn >= 0 atunci culoare ← alb
    altfel culoare ← negru
stop procedura

```

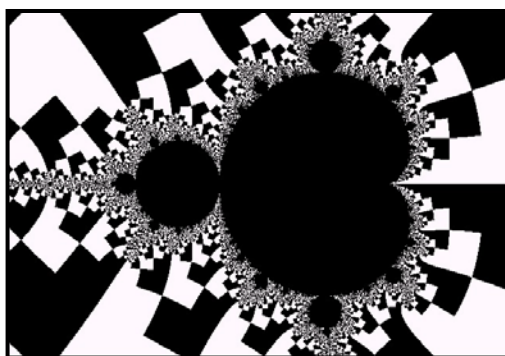


Fig. 2. Mulțimea lui Mandelbrot – descompunere binară.

Colorarea se face numai pentru punctele din exteriorul mulțimii, nu și pentru cele din interior.

În ceea ce privește limbajul de programare, este mai indicat să folosim C-ul deoarece acesta are inclusă o bibliotecă de funcții pentru numere complexe, față de limbajul Pascal ce nu prezintă acest lucru. Programul pentru realizarea unei imagini pentru mulțimea Mandelbrot în descompunere binară este următorul:

```
#include<iostream.h>
#include<conio.h>
#include<graphics.h>
#include<math.h>
#include<complex.h>

const M=100;
const N=64;
const xmin=-2.5;
const xmax=1.5;
const ymin=-1.5;
const ymax=1.5;

complex f(complex z)
{
    return z*z;
}

int mandelbrot(complex z)
{
    int k=0;
    complex zn=z;
    while ((norm(zn)<M) && (k<N))
    {
        k=k+1;
        zn=f(zn)+z;
    }
    if (norm(zn)>=M)
        if (imag(zn)<=0) k=0;
        else k=15;
    return k;
}

void main()
{
    clrscr();
    int gm;
    int gd=DETECT;
    initgraph(&gd,&gm,"C:\\BORLANDC\\BGI");
    int w=640; double sx=(xmax-xmin)/double(w);
    int h=480; double sy=(ymax-ymin)/double(h);
    double s=sx; if (sy>s) s=sy;
    h=int((ymax-ymin)/s);
    w=int((xmax-xmin)/s);
    double x=xmin,y;
    int i,j;
    complex z;
    for(i=0;i<=w;i++)
    {
        y=ymin;
        for(j=0;j<=h;j++)
```

```
{  
    z=complex(x, y);  
    putpixel(i, h-j, mandelbrot(z));  
    y=y+s;  
}  
x=x+s;  
}  
getch();  
}
```

BIBLIOGRAFIE

- [1] Holmgren, R. A., A First course in discrete dynamical systems, Springer, New York, 1994.
- [2] Peitgen, H.O., Richter, P.H., The beauty of fractals. Images of complex dynamical systems, Springer, New York, 1996.
- [3] Niculescu, C. P., Sisteme dinamice haotice, Univ. Craiova, 1995.
- [4] Vlada, M., Nistor, I., Postea, A., Constantinescu, C. Grafică pe calculator în limbajele Pascal și C, Ed. Tehnică, București, 1992.
- [5] Retrisor, E., Sisteme dinamice haotice, Editura Universitatii din Timisoara, 1992.
- [6] Constantinescu, C.-P., Haos, fractali si aplicatii, Ed. The Flower Power, Pitesti, 2003

Probleme distractive de matematică

Gruia Monica
Sc. „L. Rebreanu”, Mioveni, Arges

Problemele rezolvate cu ajutorul matematicii sunt foarte variate și, pe lângă probleme tradiționale de algebră sau geometrie, apar probleme cu tablouri cu numere sau culori aranjate în diferite feluri, cu lăcuste sau purici care sar dintr-un loc în altul, cu piese de șah și deplasări de obiecte. În spatele acestei matematici făcute cu zâmbetul pe buze găsim autori celebri și teoreme „serioase”.

În această lucrare propunem câteva probleme de determinare a numărului de metode pentru atingerea unui scop.

Probleme cu sărituri

I) În câte moduri un săritor, care stă în fața unor celule desenate pe pământ, poate să ajungă la celula n , dacă el sare de la celula 1 la celula n , atingând pământul numai în interiorul celulelor, iar lungimea salturilor este arbitrară?

Fie u_s numărul de moduri prin care se poate ajunge în celula s . Pentru rezolvarea problemei trebuie să ținem seama că săritorul are posibilitatea (una singură) de a sări în celula a n -a, fără să atingă pământul în vreoa celulă intermediară. Dacă el sare întâi prin k celule intermediare, el o poate face în C_{n-1}^k moduri. Prin urmare, avem

$$u_n = 1 + C_{n-1}^1 + C_{n-1}^2 + \dots + C_{n-1}^{n-1} = 2^{n-1}. \quad (1)$$

Observația 1. Evident, cu acest prilej am găsit numărul diverselor reprezentări ale numărului n sub forma unei sume de termeni pozitivi întregi (inclusiv cazul „ sumei” formate dintr-un singur termen), două reprezentări fiind considerate distincte dacă diferă fie prin natura termenilor, fie prin ordinea lor.

Observația 2. Dacă săritorul are voie să atingă pământul numai de un număr par de ori, atunci el va dispune de $1 + C_{n-1}^2 + C_{n-1}^4 + \dots$ adică de 2^{n-2} moduri.

II) În câte moduri un săritor poate să ajungă în celula a n -a dacă el are voie să efectueze numai salturi simple (până în celula vecină) și duble (sărind din două în două) ?

Să notăm cu v_s numărul de moduri prin care se poate ajunge în celula cu nr. s . Deoarece săritorul poate ajunge în celula s numai din celulele cu numerele $s-1$ și $s-2$, iar pentru a ajunge aici el dispune respectiv de v_{s-1} și de v_{s-2} moduri, rezultă că, pentru $s > 2$, are loc egalitatea

$$v_s = v_{s-1} + v_{s-2}. \quad (2)$$

Ne convingem ușor că

$$v_1 = 1 \text{ și } v_2 = 2. \quad (3)$$

Plecând de la (3), cu ajutorul relației (2), putem determina consecutiv valorile lui v_3, v_4, v_5, \dots , adică să scriem soluția problemei sub forma tabelului

s	1	2	3	4	5	6	7	8	9	10	11	12	13	14
v_s	1	2	3	5	8	13	21	34	55	89	144	233	377	610

Ecuția (2) este un caz particular al ecuațiilor denumite *cu diferențe finite*

$$v_{x+m} = F(v_x, v_{x+1}, \dots, v_{x+m-1}), \quad (5)$$

care formează obiectul de studiu al calculului cu diferențe finite.

Fiind cunoscute valorile lui $v_0, v_1, v_2, \dots, v_{x+m-1}$, este ușor să găsim consecutiv, cu ajutorul egalității (5), care este și ea o relație de recurență pentru funcția v_x , valorile lui $v_m, v_{m+1}, v_{m+2}, \dots$, adică să obținem soluția ecuației (5) cu ajutorul unui tabel. Ar fi, însă, incomod să căutăm pe această cale, de exemplu, pe v_{1000} , astfel că, de obicei, căutăm să reprezentăm soluția sub forma $v_x = f(x)$.

Ne convingem ușor că ecuațiile (2) și (3) sunt satisfăcute de funcția

$$v_n = (1/5^{1/2}) \{ [(1+5^{1/2})/2]^{n+1} - [(1-5^{1/2})/2]^{n+1} \} \quad (6)$$

Calculând cu ajutorul ei valorile lui v_3, v_4, v_5, \dots , vom obține aceleași numere ca în tabelul (4).

Observația 3. În prima problemă cu săritorul am fi putut stabili relația

$$u_s = 1 + u_1 + u_2 + \dots + u_{s-1}, \quad (7)$$

de unde rezultă ușor (pentru $u_1 = 1$) egalitatea $u_n = 2^{n-1}$.

Observația 4. În problema cu săritorul putem varia condițiile după cum vrem: de exemplu, putem să admitem în general salturile simple, duble și triple, făcând însă restricția că din celulele ale căror numere de ordine sunt multipli de cinci (inclusiv din poziția inițială) se pot face numai salturi simple.

În aceste condiții, să notăm cu w_s numărul de moduri în care se poate ajunge la celula nr. s ; atunci, în locul unei singure ecuații, vom avea sistemul

$$\begin{aligned} w_s &= w_{s-1} + w_{s-2} + w_{s-3} && \text{pentru } s = 5k \text{ și pentru } s = 5k + 1 \text{ (sau } s = 5k - 1 \text{)}, \\ w_s &= w_{s-1} + w_{s-3} && \text{pentru } s = 5k + 2, \\ w_s &= w_{s-1} + w_{s-2} && \text{pentru } s = 5k + 3; \text{ unde } w_1 = 1, w_2 = 1, w_3 = 2. \end{aligned}$$

Să trecem acum la probleme în care funcția căutată depinde de două sau mai multe argumente care sunt numere întregi.

Problema turnului

În câte moduri (cu cel mai mic număr de mutări) putem trece un turn de pe câmpul $(0, 0)$ în câmpul (m, n) dacă îl deplasăm prin mutări simple, adică mutări în care turnul trece în câmpul vecin pe orizontală sau pe verticală ?

Numerele dintre paranteze marchează respectiv numărul coloanei și numărul liniei la intersecția cărora se află câmpul; coloanei din stânga și liniei de jos li se atribuie numărul 0 (m și n sunt întregi pozitivi).

Să notăm cu $u_{x,y}$ numărul de moduri prin care se poate trece din câmpul $(0, 0)$ în câmpul (x, y) .

Evident, oricare ar fi numerele pozitive x și y , avem

$$u_{x,0} = 1 \text{ și } u_{0,y} = 1. \quad (8)$$

Deoarece, pentru $x > 0$ și $y > 0$, turnul poate ajunge în câmpul (x, y) , fie din câmpul $(x-1, y)$, fie din câmpul $(x, y-1)$, în care el poate ajunge, respectiv, în $u_{x-1,y}$ și $u_{x,y-1}$ moduri, rezultă că

$$u_{x,y} = u_{x-1,y} + u_{x,y-1}. \quad (9)$$

Am obținut o relație de recurență pentru funcția $u_{x,y}$, care depinde de două argumente întregi. Scriind în fiecare câmp (x, y) al tablei de șah valoarea lui $u_{x,y}$ corespunzătoare, atunci, în virtutea relațiilor (8) și (9), toate câmpurile din coloana din stânga și din linia de jos vor putea fi completate imediat cu cifra 1. Apoi vom putea completa treptat celelalte câmpuri, scriind în ele numere egale cu suma a două numere vecine (de jos și din stânga). Tabelul reprezintă soluția ecuației (9) în condițiile (8).

1						
	5					
1	4	10				
1	3	6	10			
1	2	3	4	5		
	1	1	1	1	1	1

Problema turnului poate fi rezolvată mai simplu, rezolvare care, totodată, va reprezenta soluția ecuației (9) printr-o formulă comodă.

Să remarcăm că pentru a trece turnul din câmpul $(0, 0)$ în câmpul (m, n) sunt necesare în total $m + n$ mutări; m pe orizontală și n pe verticală. Fiecare mod de a deplasa turnul poate fi caracterizat printr-o schemă formată din literele o și v , care ne arată în ce ordine trebuie să efectuăm mutările orizontale și pe cele

verticale. Pentru a alege cele m locuri ocupate de litera o (dintre cele $m + n$ locuri existente în schemă), avem în total $C_{m+n}^m = \frac{(m+n)!}{m!n!}$ posibilități distincte, ceea ce ne dă soluția problemei turnului de șah.

Soluția ecuației (9) în condițiile (8) este funcția

$$u_{x,y} = \frac{(x+y)!}{x!y!} \quad (10)$$

Problema păianjenului

În câte moduri un păianjen aflat în originea coordonatelor poate să treacă (pe drumul cel mai scurt!) în nodul k, l, m al unei rețele spațiale ? (Într-o rețea spațială orice nod, adică orice punct cu coordonate întregi, este legat cu mici bare paralele cu axele de coordonate de cele șase noduri vecine). Această problemă este o generalizare naturală a problemei turnului.

Notând cu $u_{x,y,z}$ numărul de moduri prin care se poate ajunge în nodul (x, y, z) , atunci pentru valori naturale ale numerelor x, y, z , avem

$$u_{x,y,z} = u_{x-l,y,z} + u_{x,y-l,z} + u_{x,y,z-l} \quad (11)$$

Pe lângă această ecuație cu diferențe, care conține o funcție necunoscută depinzând de trei argumente întregi, avem și condițiile

$$u_{x,y,0} = \frac{(x+y)!}{x!y!}; u_{x,0,z} = \frac{(x+z)!}{x!z!}; u_{0,y,y} = \frac{(y+z)!}{y!z!}$$

care rezultă din soluția problemei precedente.

Orice mod concret prin care păianjenul poate trece din nodul $(0, 0, 0)$ în nodul (k, l, m) poate fi caracterizat prin șirul de litere x, y, z , care arată în ce ordine se succed deplasările păianjenului după direcțiile axelor Ox, Oy, Oz ; de aceea, $u_{k,l,m}$ va fi egal cu numărul de moduri în care putem completa $k+l+m$ locuri, cu k litere „ x ”, cu l litere „ y ” și cu n litere „ z ”.

Dintre cele $k+l+m$ locuri de care dispunem, putem alege k locuri pentru a le umple cu litera „ x ” în $\frac{(k+l+m)!}{k!(l+m)!}$ noduri; fiecăruia din aceste noduri îi corespund $\frac{(l+m)!}{l!m!}$ posibilități de a alege l locuri (dintre cele $l+m$ rămase libere) pentru a le completa cu litera „ y ”. Prin urmare, numărul total de posibilități de a alege k locuri în schemă (pentru a le umple cu litera „ x ”) și l locuri (pentru a le umple cu litera „ y ”) este

$$\frac{(k+l+m)!}{k!(l+m)!} \cdot \frac{(l+m)!}{l!m!} = \frac{(k+l+m)!}{k!l!m!}$$

Așadar, soluția ecuației (11) în condițiile limită (12) va fi funcția $u_{x,y,z} = \frac{(x+y+z)!}{x!y!z!}$.

Probleme multidimensionale

Problema păianjenului poate fi generalizată la o rețea cvadridimensională, ale cărei noduri se obțin din nodurile rețelei tridimensionale, deplasând-o cu una, cu două, cu trei etc. unități de lungime în direcția celei de-a „ i -a axe” Ou . Nu este necesar să orientăm axa Ou perpendicular pe axele Ox, Oy, Oz , căci pentru aceasta ar trebui „ să ieșim în spațiul cvadridimensional”, ci este suficient să ne imaginăm o serie de rețele tridimensionale, ale căror noduri sunt unite prin mici bare „ unitare”, adică egale cu unitatea, și „ paralele” cu axa Ou .

Pentru rețele tridimensionale cu număr finit de noduri aceasta se poate realiza sub forma unui model, unind efectiv prin „ sârme unitare” nodurile corespunzătoare ale rețelelor tridimensionale de ordinul „zero”, de ordinul întâi, al doilea etc.

Într-o astfel de rețea cvadridimensională, fiecare nod va fi caracterizat prin patru numere întregi (coordonatele sale).

Păstrând terminologia geometrică, un ansamblu de m numere reale a_1, a_2, \dots, a_m reprezintă coordonatele unui punct în „ spațiul m dimensional” \mathbf{R}^m .

Dacă numerele a_1, a_2, \dots, a_m sunt întregi și pozitive și considerăm drept vecine nodurile în care numai una dintre coordonate diferă cu o unitate, pe când toate celelalte sunt egale, atunci pentru $a_1 + a_2 + \dots + a_m = n$, trecerea de la nodul $O(0, 0, \dots, 0)$ la nodul $A(a_1, a_2, \dots, a_m)$ se poate efectua în n mutări (trecând de fiecare dată în câte un nod vecin).

Dacă mutările în cadrul cărora crește numai prima coordonată sau numai a doua etc. sunt notate respectiv cu literele x_1, x_2 etc., atunci numărul de moduri pentru a trece prin n mutări din nodul $O(0, 0, \dots, 0)$ în nodul $A(a_1, a_2, \dots, a_m)$ este egal cu numărul de „permutări cu repetiție” care se pot face cu n elemente:

$$\underbrace{x_1, x_1, \dots, x_1}_{a_1 \text{ elemente}}; \quad \underbrace{x_2, x_2, \dots, x_2}_{a_2 \text{ elemente}}; \quad \dots; \quad \underbrace{x_m, x_m, \dots, x_m}_{a_m \text{ elemente}}$$

Ca exemplu de problemă multidimensională este problema umplerii cu apă a unor butoaie: butoaiile cu numerele de ordine $1, 2, \dots, m$ au respectiv capacitățile a_1, a_2, \dots, a_m căldări; în câte moduri se pot umple toate butoaiile dacă trebuie respectată condiția că apa dintr-o căldare plină se toarnă în întregime într-unul din butoaie ?

Acest număr este egal cu:
$$\frac{(a_1 + a_2 + \dots + a_m)!}{a_1! a_2! \dots a_m!}.$$

Studiul matematic și fizic al centrului de greutate al unui corp: teorie, experiment, metodică

Consuela Luiza Voica și Elena Popescu
Școala cu clasele I-VIII nr.12, Sector 1, București.

În prezenta lucrare încercăm să răspundem la întrebările: Există o legătură între noțiunile matematice și cele fizice cu denumiri asemănătoare? Putem utiliza proprietăți fizice în demonstrarea unor rezultate de geometrie? Pot fi realizate orele de curs prin colaborarea unor profesori de specialități diferite? În lucrare sunt prezentate modalitățile practice, rezultate dintr-un experiment cu elevii, de realizare a unor ore de curs, în care se leagă cunoștințele de matematică cu cele de fizică. Sunt arătate atât implicațiile studiului matematic asupra modelului fizic, și invers, cât și modul în care pot colabora profesorii de matematică cu cei de fizică. Prin aceasta se urmărește din punct de vedere metodic modul de pregătire al elevilor pentru abordarea în viitor a unor probleme interdisciplinare și modalități de conlucrare a unor specialiști din discipline diferite.

1. INTRODUCERE

Lucrarea realizează o legătură între premodelarea matematică și cea fizică într-o problemă implicând calculul centrului de greutate al unui tetraedru și al unui paralelipiped dreptunghic. Acest centru joacă un rol important în alunecarea corpului pe un plan înclinat. De aceea lucrarea de față arată modalitatea concretă și metodică prin care se induce aceasta elevilor.

În accepțiunea didacticii contemporane, o componentă majoră a procesului didactic o constituie aplicarea cunoștințelor și competențelor dobândite într-un context dat, prin transferarea acestora în contexte noi, variate. În acest sens, exersarea se poate realiza prin “interconectarea diferitelor tipuri de reprezentări, construirea relațiilor dintre reprezentare și obiect, privirea conceptelor în lumina noilor experiențe” [1, p.26], ceea ce aratăm în lucrarea de față. Experimentul nostru concordă cu programele școlare care solicită printre altele “identificarea utilizării unor concepte și metode matematice studiate, în diferite domenii” și “aplicarea cunoștințelor dobândite prin studiul fizicii în domenii conexe acestora” [2].

2. SCOPUL ȘI OBIECTIVELE EXPERIMENTULUI

Primul constă în realizarea transferului de cunoștințe între diferite discipline din aria curriculară matematică și științe, evidențiindu-se legăturile dintre noțiunile matematice și fizice cu aceeași denumire dar cu conținut diferit, specific științei respective. În plus, arătăm posibilitatea utilizării unor proprietăți fizice în demonstrarea unor rezultate de geometrie, abordare rar întâlnită în literatura matematică. Dintre puținele lucrări în acest domeniu menționăm [3].

Experimentul reprezintă o premieră, realizând ore de curs cu colaborarea unor profesori de specialități diferite. Uneori, această colaborare se folosește și la cursurile de matematică aplicată din universități [4].

Obiectivele experimentului sunt: identificarea unor posibilități de colaborare între profesori de discipline diferite; organizarea și desfășurarea unor ore de curs, în care doi profesori de discipline diferite au rol de conducere a învățării; identificarea legăturilor dintre proprietăți învățate în contexte diverse; aplicarea unor metode matematice în rezolvarea problemelor practice, precum și utilizarea proprietăților fizice în demonstrarea unor rezultate matematice; utilizarea lucrului în grup, ca metodă de dinamizare a clasei și de eficientizare a activității; crearea unui cadru ambiental atractiv și agreabil de desfășurare a orelor (prin audiții muzicale în timpul realizării de către elevi a modelului decupat din hârtie); determinarea desfășurării unor experimente de același tip, prin popularizarea modului de organizare și a concluziilor obținute.

Considerăm că aceste obiective vor crea premiza grupurilor mixte de cercetare formate din matematicieni aplicați, fizicieni, chimiști, ingineri etc.

3. PROIECTAREA EXPERIMENTULUI ȘI PREMODELAREA

În proiectarea experimentului, am plecat de la următoarea întrebare, adresată elevilor: “care este legătura dintre centrul de greutate învățat la matematică, și aceeași noțiune din fizică?” Am constatat că elevii nu pot realiza legături între aceste noțiuni; pentru ei, centrul de greutate înseamnă sau “punctul de intersecție a medianelor” (la matematică) sau “punctul de aplicație a forței de greutate” (fizică).

Situații-problemă au apărut și la matematică, și la fizică, în cazul în care am încercat să lucrăm cu configurații plane sau spațiale mai complicate. De exemplu, am întrebat: “Ce punct ar putea să fie denumit centru de greutate al unui tetraedru? Dar al unui paralelipiped oblic?” Deoarece elevii nu au putut formula răspunsuri coerente, ne-am propus să organizăm ore în comun, în care să evidențiem legăturile dintre concepte cu denumire comună, dar cu definiții diferite, din matematică și din fizică. În acest fel, am putut explica coerent modul în care apar unele definiții în matematică și în fizică, ceea ce depășește, de fapt, simpla legătură dintre matematică și fizică. Într-adevăr astfel se abordează și problema mult mai dificilă a derivării unor rezultate matematice din cele fizice (e.g. în mecanică [3]).

Experimentul s-a derulat pe parcursul ultimilor 2 ani școlari, cu diferite clase a VIII-a. De la un an la altul, precum și de la o clasă la alta, am modificat atât modul de organizare a clasei, cât și sarcinile de lucru. **Modelul fizic:** centrul de greutate a două puncte materiale de mase egale este punctul de aplicație al rezultantei forțelor de greutate. Poziția centrului de greutate se determină din relația fizică de egalitate a momentelor forțelor de greutate față de centrul de greutate, i.e. $G \cdot b_1 = G \cdot b_2 \Rightarrow b_1 = b_2$. Rezultă că acest centru de greutate G se află la mijlocul segmentului imaginar ce unește cele două puncte (fig.1).

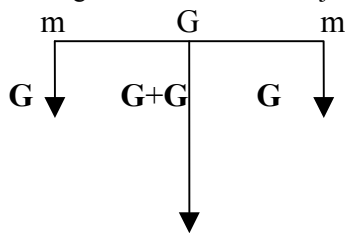


Fig.1.

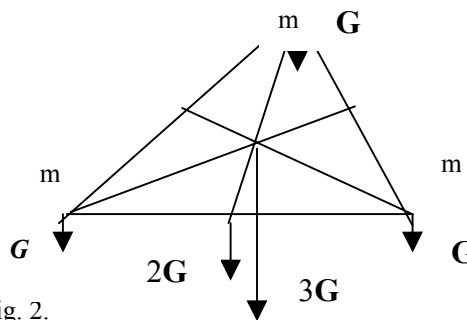


Fig. 2.

Pentru trei puncte materiale necoliniare, de mase egale, centrul de greutate al triunghiului determinat de cele trei puncte este și punctul de aplicație al forței de greutate a sistemului format de cele trei puncte (fig. 2). Acesta reprezintă punctul de intersecție al medianelor triunghiului și, scriind relația de egalitate a celor două momente, obținem

$$2G \cdot b_1 = G \cdot b_2 \Rightarrow 2 b_1 = b_2.$$

Modelul matematic: Ultima relație arată că acest centru se află pe fiecare mediană la o treime de bază și două treimi de vârf [5]. Prin urmare, modelul matematic al centrului de greutate este același cu cel din fizică.

Cazul tetraedrului. Firește, ne întrebăm dacă putem aplica aceeași metodă pentru determinarea poziției centrului de greutate al unui tetraedru. Mai întâi elevii au fost solicitați să demonstreze matematic afirmația: “medianele” tetraedrului se intersectează într-un punct situat pe fiecare din ele la $\frac{1}{4}$ de bază și $\frac{3}{4}$ de vârf” (fig.3).

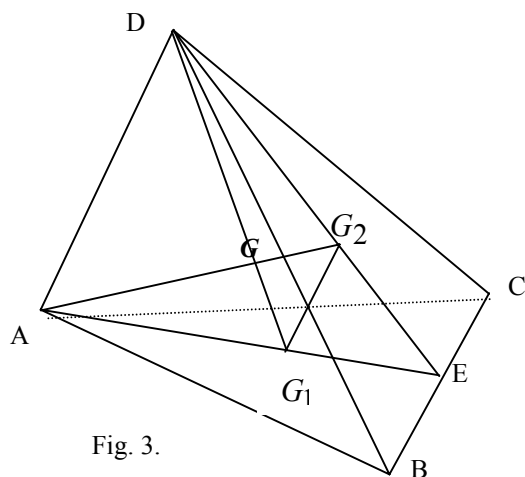


Fig. 3.

Demonstrația matematică pentru medianele AG_2 și DG_1 s-a făcut astfel: în triunghiul ADE , avem $\frac{EG_1}{AG_1} = \frac{EG_2}{DG_2} = \frac{1}{3}$, G_1 și G_2 fiind centrele de greutate ale fețelor ABC și DBC . Conform reciprocei teoremei lui Thales rezultă că $G_1G_2 \parallel AD$ și $\frac{G_1G_2}{AD} = \frac{1}{3}$. Aplicând apoi teorema fundamentală a asemănării pentru

triunghiurile AGD și G_2GG_1 obținem $\frac{GG_1}{DG} = \frac{GG_2}{AG} = \frac{1}{3}$. Deci $AG = 3GG_2$.

Aplicând același raționament pentru medianele CG_3 și BG_4 , unde G_3 și G_4 sunt centrele de greutate respective ale fețelor ABD și DAC, am demonstrat afirmația de mai sus.

Apoi, demonstrația acestui rezultat s-a făcut utilizând metode fizice și s-a dovedit a fi mult mai ușoară. Ea se bazează pe ideea prezentată anterior pentru un triunghi.

Astfel, considerând că în vârfurile tetraedrului se află puncte materiale de aceeași masă (fig.4) și aplicând raționamentul fizic prezentat anterior pentru fiecare față a tetraedrului, obținem poziția centrului de greutate al fiecărei fețe, în care acționează o greutate de modul $3G$ și este chiar cea găsită matematic.

Pentru fiecare vârf și centrul de greutate al feței opuse aplicăm relația fizică de egalitate a momentelor față de centrul de greutate, i.e. $3G \cdot b_1 = G \cdot b_2 \Rightarrow 3b_1 = b_2$, fapt ce demonstrează afirmația făcută.

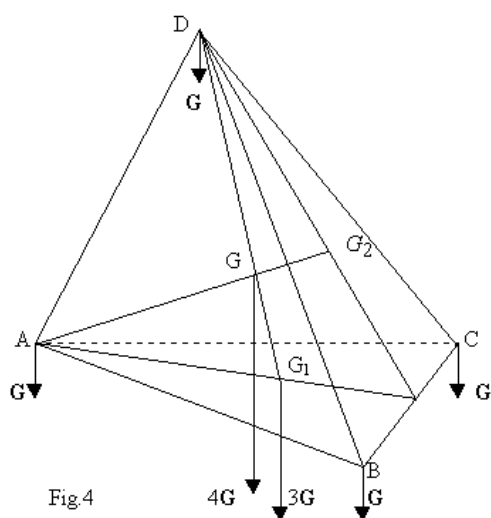


Fig.4

Centrul de greutate al unui sistem de puncte materiale se poate determina, prin gruparea arbitrară a punctelor date. În rezolvarea anterioară, am grupat trei puncte, lăsând unul dintre puncte separat.

Elevii au fost întrebați “ Ce proprietate geometrică obțineți dacă grupați punctele două câte două și utilizați rezultatul cunoscut din fizică?”

Punctul M, mijlocul segmentului AD, este centrul de greutate al sistemului format din punctele A și D iar punctul N, mijlocul segmentului BC, este centrul de greutate al sistemului format din punctele B și C iar în fiecare din ele acționează o forță de modul $2G$. Din egalitatea momentelor forțelor de greutate față de centrul de greutate, obținem că G se află la mijlocul lui MN. Analog se demonstrează că G este mijlocul segmentelor PQ și RS, unde P, Q, R, S sunt mijloacele segmentelor: AB, CD, AC și BD.

Astfel, utilizând raționamente fizice am demonstrat următoarea proprietate matematică:” Într-un tetraedru, segmentele ce unesc mijloacele laturilor opuse

sunt concurente.”

Demonstrația matematică se bazează pe proprietățile paralelogramului și este mai lungă iar experimentul nostru este o pledoarie pentru realizarea obiectivului: “aplicarea unor metode matematice în rezolvarea problemelor practice, precum și utilizarea proprietăților fizice în demonstrarea unor rezultate matematice.”

În particular, pentru un tetraedru regulat, centrul de greutate și centrul de simetrie coincid, ceea ce ne induce ideea de generalizare a afirmației pentru corpurile cu centru de simetrie. Deci pentru un paralelipiped centrul de greutate se va afla la intersecția diagonalelor lui.

4. PREMODELAREA FIZICĂ A ALUNECĂRII UNUI CORP PE PLANUL ÎNCLINAT

Să considerăm următoarea problemă: un paralelipiped dreptunghic confecționat din lemn cu volumul de 54 cm^3 și densitatea de 600 kg/m^3 alunecă pe un plan înclinat de unghi $\alpha = 30^\circ$. Să se calculeze valoarea forței de frecare știind că $\mu = 0,2$ și $g = 10 \text{ N/kg}$. Pentru rezolvarea problemei confecționăm un paralelipiped dreptunghic din carton având dimensiunile de 3 cm , 2 cm , 9 cm [6, p.21] și îl lăsăm să alunece pe planul înclinat, având $\alpha = 30^\circ, 45^\circ, 60^\circ$. Apoi procedăm la fel cu un paralelipiped din lemn și cu unul din aluminiu. Observăm că forța de frecare depinde de: natura suprafețelor de contact; valoarea unghiului planului înclinat și greutatea corpului.

Experimentul a pus în evidență pe de o parte o realitate fizică: existența unei legături între forța de alunecare și câțiva factori de care ea depinde, iar pe de altă parte, modalitatea de definire empirică,

experimentală, a așa-numitelor legi de material, care în cazul corpurilor rigide sunt tocmai legile de frecare. Desigur, în cazul simplu de mai sus, această lege este liniară, constanta de material fiind μ . Prin aceasta am pus în evidență și o modalitate ușoară de înțelegere ulterioară a unor legități fizice profunde.

Revenind la problema dată, acum avem o bază fizică pentru calcularea forței de frecare. Pentru rezolvarea problemei reprezentăm forțele care acționează asupra corpului lăsat să alunece pe planul înclinat (fig.5) și deducem formula forței de frecare

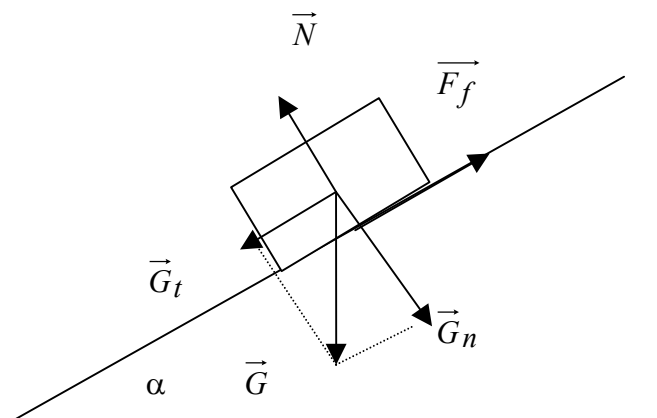


Fig.5.

$$F_f = \mu \cdot N = \mu \cdot G_n = \mu \cdot G \cdot \cos\alpha = \mu \cdot m \cdot g \cdot \cos\alpha = \mu \cdot \rho \cdot V \cdot g \cdot \cos\alpha .$$

Înlocuind datele de mai sus ale problemei în această formulă rezultă valoarea forței de

frecare: $F_f = 0,2 \cdot 600 \text{ kg} / \text{m}^3 \cdot 54 \cdot 10^{-6} \text{ m}^3 \cdot 10 \text{ N} / \text{kg} \cdot \frac{\sqrt{3}}{2} = 560,52 \cdot 10^{-4} \text{ N}$, direcția ei fiind tangență la suprafața planului înclinat (suprafața de contact).

5. EVALUAREA EXPERIMENTULUI

Am evaluat experimentul prin intermediul unor fișe cu întrebări, adresate elevilor și profesorilor asistenți. În urma prelucrării răspunsurilor, am constatat următoarele: elevii au apreciat ca bună colaborarea cu colegul de bancă, orele le-au plăcut mult, iar întrebările au avut grad de dificultate mediu. Din acest experiment ei au aflat lucruri interesante: cum se pot îmbina cunoștințele dobândite la fizică cu cele de la matematică, au înțeles mai bine noțiunea de centru de greutate și au recapitulat în acest fel mai ușor lucruri studiate separat la cele două materii. Au aflat despre paralelipipedul oblic și cel drept, au vizualizat un model al lor și o realizare practică a unei probleme de fizică, rezolvarea devenind astfel mai ușoară. Profesorii au găsit experimentul foarte interesant de urmat, iar colaborarea dintre elevi și respectiv dintre profesori a fost bună și foarte bună.

Toate aceste premize ne-au determinat să prezentăm acest experiment și să ne formulăm ca obiectiv popularizarea lui. În plus, încercarea de a prefigura modelarea matematică de la facultăți și conlucrarea unor grupuri eterogene dar unitare de specialiști din cercetarea de mai târziu, a fost și ea un real succes.

BIBLIOGRAFIE

[1] ***, *Ghiduri metodologice pentru aplicarea programelor de matematică, fizică, chimie, biologie*, editate de CNC și MEC, Ed.Aramis, București, 2001.

- [2] ***, *Curriculum național. Programe școlare pentru clasele a V-a -a VIII-a. Aria curriculară matematică și științe ale naturii*, eEditate de MEC și CNC, București, 1999.
- [3] *Iacob C., Matematică aplicată și mecanică*, Ed. Acad. R. S. R. , București 1989.
- [4] *Comunicare particulară cu doamna Prof. dr. Adelina Georgescu.*
- [5] *Pufu, E. – C., Elemente de geometrie a maselor – centre de greutate*, în această carte.
- [6] *Singer, M., Voica, C., Voica, C.-L., Decupează, construiește și verifică!* Ed. Sigma, București, 2000.

Nota asupra unor relații trigonometrice

Valerian Opreșor,
„Liviu Rebreanu” Mioveni-Argeș

În unele aplicații industriale (e.g. în aviație), cunoașterea exactă a valorii unor funcții se poate dovedi deosebit de importantă.

Pe de altă parte, stimularea gândirii geometrice a elevilor mici este un obstacol în calea pericolului ca aceștia, folosind calculatorul, să nu mai fie convinși de utilitatea acestei gândiri. Cum nu elevul este cel care știe ce-i va folosi în profesiunea sa, profesorul are datoria să-i furnizeze elevului mic astfel de motive, una din surse dovedindu-se CAIM-ul.

În același timp la CAIM, Secția de Învățământ trebuie să dezbată printre altele, probleme metodologice de formare a gândirii matematice a elevului, posibil matematician, fizician sau inginer de mâine. De aceea, lucrarea noastră prezintă două metode de calcul, geometric și trigonometric, al valorilor funcțiilor trigonometrice ale unghiurilor de 18° și de 15° , reliefindu-se avantajul uneia față de cealaltă, ca și complementaritatea lor.

Problema 1. a) Arătați că există punctele O, A, B, C astfel încât $A \in (OC)$, $[OA] = [AB] = [BC]$ și $[OB] = [OC]$. b)

Demonstrați că în situația de mai sus are loc relația: $2 \cdot |AC| = (\sqrt{5} - 1) \cdot |OA|$.

c) Deduceți relațiile: $4 \cdot \sin 18^\circ = \sqrt{5} - 1$ și $4 \cdot \cos 36^\circ = \sqrt{5} + 1$.

Soluție. a) Pentru un unghi $\angle AOB$ de măsură $u < 45^\circ$ rezultă $m(\angle OBA) = u$, $m(\angle BAC) = 2 \cdot u$, $m(\angle OCB) = 2 \cdot u$, $m(\angle ABC) = 180^\circ - 4 \cdot u$ (fig.1). Are loc $[OB] = [OC]$ dacă (și numai dacă) $\angle OBC = \angle OCB$, i.e. $u = 180^\circ / 5 = 36^\circ$.

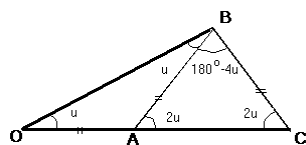


Fig. 1

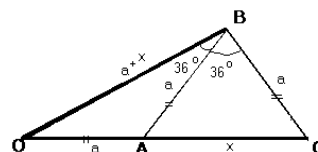


Fig. 2

b) Notând $|OA| = a$ și

$|AC| = x$ rezultă $|OB| = a + x$ și în $\triangle OBC$, unde BA

este bisectoarea $\angle OBC$, aplicând teorema bisectoarei obținem $(a+x)/a = a/x \Leftrightarrow x^2 + a \cdot x - a^2 = 0 \Leftrightarrow (2 \cdot x + a)^2 = 5 \cdot a^2$ de unde $2 \cdot x = a \cdot \sqrt{5} - a$, de unde rezultă relația cerută.

c) Fie D mijlocul segmentului (AC) (fig.3). Cunoscând lungimile tuturor segmentelor din fig. 3 se calculează $\sin 18^\circ$ din $\triangle DBC$ și $\cos 36^\circ$ din $\triangle DBO$ și se obțin relațiile cerute.

Observație. Putem calcula valori ale funcțiilor trigonometrice și pentru unghiurile cu măsurile de 54° și 72° .

Folosirea mai multor procedee pentru rezolvarea unei aceleiași probleme permite să se constate mai ușor eficiența unei anumite metode, să se decidă în care din situații se realizează o mai mare economie de timp în rezolvare sau se obține o soluție mai simplă și elegantă. Astfel, la trigonometrie la clasa a IX-a sunt frecvente aplicațiile de tipul:

Problema 2. Să se calculeze fără tabele trigonometrice $\sin 18^\circ$ sau $\cos 18^\circ$.

Soluție. Li se poate sugera elevilor, ca aplicații la formulele învățate, abordabile de altfel și la nivelul clasei a VII-a, să găsească o ecuație care să aibă drept necunoscută pe $\sin 18^\circ$. În acest sens observăm că $2 \cdot 18^\circ = 36^\circ$, $3 \cdot 18^\circ = 54^\circ$ iar $36^\circ + 54^\circ = 90^\circ$. Prin urmare arcele $2 \cdot 18^\circ$ și $3 \cdot 18^\circ$ sunt complementare, deci $\sin 2 \cdot 18^\circ = \cos 3 \cdot 18^\circ$. Utilizând formulele trigonometrice care dau funcțiile arcului dublu și triplu se găsește relația: $2 \cdot \sin 18^\circ \cdot \cos 18^\circ = 4 \cdot \cos^3 18^\circ - 3 \cdot \cos 18^\circ$, din care prin împărțire cu $\cos 18^\circ \neq 0$ și ținând cont că avem $\cos^2 18^\circ = 1 - \sin^2 18^\circ$, se obține $4 \cdot \sin^2 18^\circ + 2 \cdot \sin 18^\circ - 1 = 0$, de unde rezultă $\sin 18^\circ = (\sqrt{5} - 1) / 4$. De aici, prin utilizarea formulelor trigonometrice învățate, se pot deduce valorile și ale altor funcții

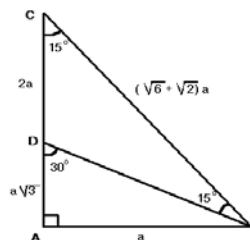
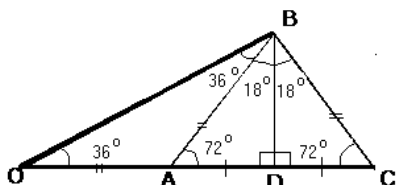


Fig. 3

Fig. 4

Problema 3. *Aplicând definițiile funcțiilor trigonometrice în triunghiul dreptunghic, calculați $\sin 15^\circ$.*

Soluție. Fie ρABC cu $m(\angle A) = 90^\circ$, $m(\angle C) = 15^\circ$ și $D \in (AC)$ astfel încât $m(\angle DBC) = 15^\circ$ (fig.4).

Notând $|AB| = a$ și având $m(\angle ADB) = 30^\circ$, rezultă că $|BD| = |DC| = 2 \sqrt{3} a$, $|AD| = a \sqrt{3}$ și utilizând teorema lui Pitagora avem că: $|BC| = (\sqrt{6} + \sqrt{2}) a$. În ρABC cu $m(\angle A) = 90^\circ$ avem: $\sin(\angle ACB) = |AB| / |BC|$, i.e. $\sin 15^\circ = a / (\sqrt{6} + \sqrt{2}) a \Leftrightarrow \sin 15^\circ = (\sqrt{6} - \sqrt{2}) / 4$.

Problema 4. *Calculați fără tabele trigonometrice $\sin 15^\circ$.*

Soluție. Utilizând formula $\sin 2x = 2 \sin x \cos x$ și punând $x = 15^\circ$ obținem: $4 \sin 15^\circ \cos 15^\circ = 1$. Ținând cont că $\sin^2 15^\circ + \cos^2 15^\circ = 1$, obținem: $\sin^2 15^\circ - 4 \sin 15^\circ \cos 15^\circ + \cos^2 15^\circ = 0$ de unde prin împărțire cu $\cos^2 15^\circ \neq 0$ și notând $\sin x / \cos x = t$ obținem $t^2 - 4t + 3 = 0 \Leftrightarrow t = 2 - \sqrt{3}$.

De aici rezultă $\sin 15^\circ / \cos 15^\circ = 2 - \sqrt{3} \Leftrightarrow \cos 15^\circ = (2 + \sqrt{3}) \sin 15^\circ$. Înlocuind în $4 \sin 15^\circ \cos 15^\circ = 1$ obținem $4 \sin^2 15^\circ = 2 - \sqrt{3} \Leftrightarrow 16 \sin^2 15^\circ = (\sqrt{6} - \sqrt{2})^2$, de unde găsim $\sin 15^\circ = (\sqrt{6} - \sqrt{2}) / 4$.

Știind $\sin 15^\circ$, putem deduce $\cos 15^\circ$, respectiv $\cos 75^\circ$ și $\sin 75^\circ$.

Spiritul creator ca și originalitatea gândirii pot fi evidențiate în cadrul multor altor probleme.

BIBLIOGRAFIE

- [1] Barbu R., Barbu N., Valențe educative ale lecțiilor de matematica, Revista de pedagogie, 1987.
- [2] Brânzei D., Negrilă A., Negrilă M., Algebră și geometrie, Ed. Paralela 45, Pitesti, 2001.
- [3] Cuculescu I., Ottescu C., Geometrie, manual pentru clasa a VII-a, E.D.P., Bucuresti, 1985.
- [4] Cota A., Marta R., Mariana R., Vornicescu F., Geometrie și trigonometrie, manual pentru clasa a IX-a, E.D.P., București, 1992.

Geometria maselor și centre de greutate

Elena – Camelia Pufu
Școala „Liviu Rebreanu” – Mioveni

Ana Moșteanu
Liceul „Dacia” – Pitești

1. INTRODUCERE

Studiul fizicii începe obligatoriu cu cel al mecanicii, ale cărei noțiuni fundamentale: traiectorie, viteză, accelerație, forță, masă, lucru mecanic, energie, impuls, moment cinetic etc. și ale cărei legi fundamentale (princiipiile mecanicii): ale impulsului, momentului cinetic, energiei cinetice, conservării energiei mecanice, sunt apoi folosite în toate capitolele fizicii.

În mecanică s-a făcut cea dintâi aplicație a matematicii la studiul cantitativ și calitativ al fenomenelor naturii. Mecanica este cea mai matematizată știință, mult timp ea fiind considerată drept o ramură a matematicii.

În această lucrare evidențiem necesitatea cunoașterii noțiunilor matematice pentru rezolvarea problemelor fizicii, legate de determinarea centrelor de greutate ale corpurilor.

Se poate arăta, atât teoretic, cât și experimental că poziția centrului de greutate este bine determinată pentru fiecare corp și nu depinde de orientarea lui în spațiu. Existența sa ne arată că forța gravitațională acționează asupra unui corp ca și cum ar fi aplicată într-un anumit punct cu toate că forța este un vector alunecător.

Centrele de greutate ale corpurilor omogene, cu forme geometrice regulate, care posedă axe sau plane de simetrie, sunt întotdeauna situate pe axa, respectiv pe planul de simetrie.

Pentru corpuri de formă mai complicată pentru determinarea centrului de greutate este necesară divizarea aceluși sistem de puncte materiale în mai multe subsisteme distincte și compunerea forțelor paralele de greutate ale punctelor materiale ce alcătuiesc corpul și care sunt supuse acțiunii greutății.

Considerațiile privind centrul de greutate al unui sistem discret de puncte materiale pot fi extinse și la cazul sistemelor materiale continue care ocupă un segment de dreaptă, un arc de curbă sau un domeniu plan sau tridimensional. Ele stau la baza însăși a calculului integral.

2. CENTRUL DE GREUTATE AL UNUI SISTEM RIGID DE PUNCTE MATERIALE

Să considerăm un sistem alcătuit din n puncte materiale A_1, A_2, \dots, A_n de mase m_1, m_2, \dots, m_n , ce au pozițiile fixe unele față de altele deci sistemul este nedeformabil și plasat în câmp gravitațional uniform. Fiecare punct al sistemului este supus acțiunii unei forțe de greutate $G_i = m_i g$, unde g este un vector constant oricare ar fi poziția punctului material.

Așa cum am menționat, acest sistem de forțe paralele ce au direcția și sensul verticalei descendente este întotdeauna echivalent cu o forță unică aplicată în centrul acelor forțe paralele.

Centrul forțelor de greutate paralele se numește **centrul de greutate** al sistemului, iar forța unică G , echivalentă cu forțele sistemului, se numește **greutatea sistemului de puncte materiale**.

Modulul forței G este dat de relația

$$G = m_1 g + m_2 g + \dots + m_n g = (m_1 + m_2 + \dots + m_n) g$$

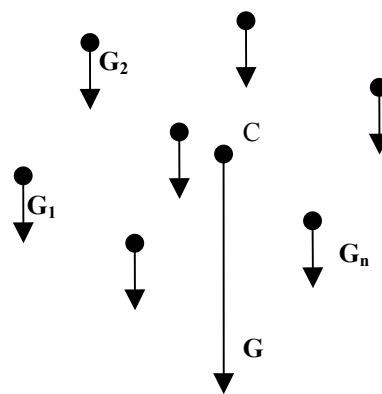


Fig. 1. Centrul de greutate al unui sistem rigid de puncte materiale.

sau

$$G = M \cdot g$$

unde $M = m_1 + m_2 + \dots + m_n$ reprezintă masa sistemului de puncte materiale.

3. DETERMINAREA POZIȚIEI CENTRULUI DE GREUTATE AL UNUI SISTEM DE DOUĂ PUNCTE MATERIALE ÎN RAPORT CU UN SISTEM DE COORDONATE

Cele două puncte materiale ale sistemului au masele m_1 și respectiv m_2 și coordonatele x_1, y_1 și respectiv x_2, y_2 în raport cu un sistem de axe perpendiculare xOy (fig. 2).

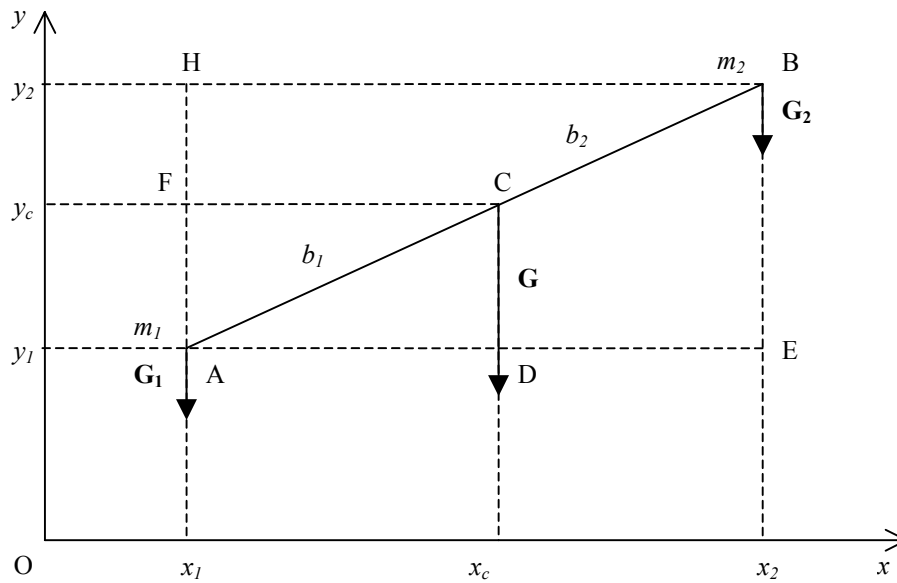


Fig. 2. Centrul de greutate al unui sistem de două puncte materiale de coordonate x_c și y_c .

$$\begin{aligned} G_1 &= m_1 g \\ G_2 &= m_2 g \\ G &= G_1 + G_2 \end{aligned} \quad (2.1)$$

Să notăm $AC=b_1$ și $CB=b_2$. Ținând seama de compunerea forțelor paralele de același sens avem

$$G_1 \cdot b_1 = G_2 \cdot b_2 \quad \text{sau} \quad \frac{b_1}{b_2} = \frac{G_2}{G_1}. \quad (2.2)$$

Din asemănarea triunghiurilor AEB și ADC, ca și a triunghiurilor AHB și AFC obținem

$$\frac{x_c - x_1}{x_2 - x_c} = \frac{b_1}{b_2} \quad \text{și} \quad \frac{y_c - y_1}{y_2 - y_c} = \frac{b_1}{b_2}. \quad (2.3)$$

Utilizând și relația (2.2) găsim

$$\frac{x_c - x_1}{x_2 - x_c} = \frac{G_2}{G_1} \quad \text{și} \quad \frac{y_c - y_1}{y_2 - y_c} = \frac{G_2}{G_1}, \quad (2.4)$$

relații ce ne permit să determinăm coordonatele centrului de greutate

$$x_c = \frac{G_1 x_1 + G_2 x_2}{G_1 + G_2}; \quad y_c = \frac{G_1 y_1 + G_2 y_2}{G_1 + G_2}. \quad (2.5)$$

Raportând sistemul de puncte materiale la trei axe de coordonate obținem și a treia coordonată a centrului de greutate

$$z_c = \frac{G_1 z_1 + G_2 z_2}{G_1 + G_2}. \quad (2.6)$$

Relațiile (2.5) și (2.6) se pot generaliza la cazul unui sistem format dintr-un număr oarecare de puncte materiale și obținem coordonatele centrului de greutate al acestui sistem

$$\begin{aligned} x_c &= \frac{G_1 x_1 + G_2 x_2 + \dots + G_n x_n}{G_1 + G_2 + \dots + G_n}, \\ y_c &= \frac{G_1 y_1 + G_2 y_2 + \dots + G_n y_n}{G_1 + G_2 + \dots + G_n}, \\ z_c &= \frac{G_1 z_1 + G_2 z_2 + \dots + G_n z_n}{G_1 + G_2 + \dots + G_n}. \end{aligned} \quad (2.7)$$

4. DETERMINAREA POZIȚIEI CENTRULUI DE GREUTATE (CENTRUL DE MASĂ) PENTRU UN SISTEM DE TREI PUNCTE MATERIALE SITUATE ÎN VÂRFURILE UNUI TRIUNGHIU ABC

Teoremă. Centrul maselor G corespunzător maselor m_1, m_2, m_3 plasate respectiv în vârfurile A, B, C ale triunghiului ABC se găsește în interiorul triunghiului.

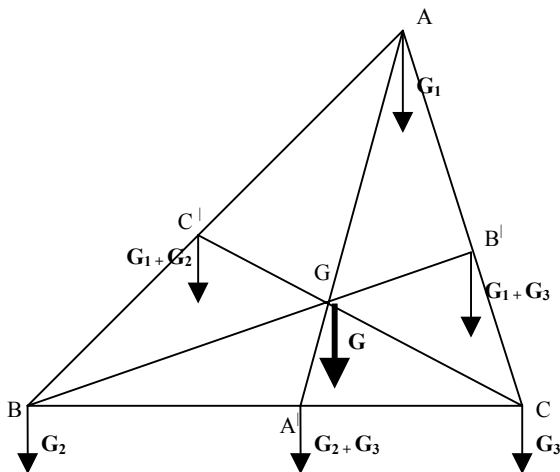


Fig. 3. Centrul de greutate al unui sistem de trei puncte materiale.

În vârfurile triunghiului ABC avem plasate masele m_i ($i = 1, 2, 3$) și greutatețile $G_i = m_i \cdot g$ ($i = 1, 2, 3$).

Compunând greutatețile G_2 și G_3 , aplicate în B și C , se găsește la greutatea $G_2 + G_3$ aplicată în punctul $A' \in BC$ astfel încât

$$m_2 BA' = m_3 CA' \quad (3.1)$$

Centrul de greutate al celor trei greutateți se va afla în punctul G de pe segmentul AA' și va fi dat de

$$m_1 AG = (m_1 + m_2) A'G \quad (3.2)$$

Dacă am fi compus întâi greutatețile G_3 și G_1 aplicate respectiv în C și A , ar fi rezultat greutatea $G_3 + G_1$ aplicată în punctul $B' \in (CA)$. Dacă apoi pe aceasta am fi compus-o cu greutatea G_2 aplicată în B am fi obținut același centru G situat pe segmentul BB' și am fi avut

$$m_3 CB' = m_1 AB'; m_2 BG = (m_3 + m_1) B'G. \quad (3.3)$$

În mod analog, compunând întâi greutatea G_1 din A și G_2 din B, am fi obținut greutatea $G_1 + G_2$ aplicată în $C' \in (AB)$. Aceasta compusă cu greutatea G_3 aplicată în C, ar fi condus la centrul de greutate $G \in (CC')$. Avem astfel

$$m_1 AC' = m_2 BC'; m_1 CG = (m_1 + m_2) C'G. \quad (3.4)$$

Dreptele AA' , BB' , CC' se intersectează evident în punctul G care se găsește în interiorul triunghiului ABC.

Putem enunța și demonstra următoarea teoremă reciprocă.

Teoremă. Fiind dat un punct oarecare G în interiorul triunghiului ABC, există întotdeauna o distribuție de mase m_1, m_2, m_3 plasate respectiv în A, B, C astfel ca G să fie centrul acestor mase.

Această teoremă constituie o sursă de nenumărate aplicații geometrice privind în special concurența de drepte (aplicații la geometria triunghiului).

Prin urmare:

- medianele unui triunghi se intersectează într-un punct G interior acestuia, care este centrul maselor egale plasate în vârfuri. Punctul G de intersecție al medianelor se numește centrul medianelor triunghiului;
- bisectoarele triunghiului ABC se intersectează în centrul de greutate al maselor plasate în vârfuri numit și ortocentrul triunghiului.

După cum s-a arătat, într-un câmp gravitațional omogen, centrul de greutate coincide cu centrul de masă. Totuși noțiunea de centru de masă este mai generală decât cea de centru de greutate, deoarece centrul de masă al unui sistem fizic se poate determina întotdeauna independent de forțele gravitaționale, ceea ce nu este adevărat pentru centrul de greutate.

În determinarea centrului maselor unui sistem material (S) putem întotdeauna să divizăm acel sistem în două sau mai multe subsisteme disjuncte (S_j), $j=1,2,3,\dots,n$, să determinăm centrele maselor G_j ale acestor sisteme, să le atribuim apoi acestora masele totale ale sistemelor (S_j). Centrul G al maselor sistemului (S) va fi atunci și centrul maselor sistemului material format din punctele G_j dotate cu masele m_j ($j=1,2,\dots,n$).

De exemplu, în spațiul cosmic un experimentator va putea determina centrul de masă al unui corp, dar centrul său de greutate nu, deoarece forța gravitațională nu are nici un efect în locul în care se face experimentul.

Cele prezentate arată că se impune o corelare a programei de fizică cu cea de matematică astfel încât elevii să aibă noțiuni elementare de geometrie și de analiză matematică atunci când studiază fenomenele mărimile fizice.

BIBLIOGRAFIE

- [1] Hristev, A., Mecanică și acustică, Editura Didactică și Pedagogică, București, 1984.
- [2] Compediu de fizică, Editura Științifică și Enciclopedică, București, 1988.
- [3] Brenneke, R., Schuster, G., Fizică – Manual pentru cursul superior al liceului, Editura Didactică și Pedagogică, București, 1973.
- [4] Iacob, C., Matematică aplicată și mecanică, Editura Academiei, București, 1989.
- [5] Manualele de fizică pentru clasa a VIII-a și a IX-a.

Contents

Preface	3
A. Bica - The remainder estimation in terms of the third derivative for the successive approximations method applied to ODE's	5
E. Codeci, D. Codeci - The bifurcation codimension for the Goodwin's model	13
M. Gâta - Authentication of the message as part of the protocol with public key	15
C. Georgescu, A. Georgescu - On the saddle-node bifurcations in the demand-supply model	19
E. Laslo, D. Noje - Amorphous Viral Operator	24
B.V. Loginov, K.M. Petrov - Bifurcation problems for the equation $\operatorname{div} (g (\nabla f ^2) \nabla f) + \lambda f = 0$	28
D. Noje, B. Bede, V. Kos - Image contrast modifiers using vectorial MV-algebras	32
G.I. Oros, A. Cătaș - On a differential inequality	37
G.I. Oros - A new differential inequality I	40
F. Todor - A probability interpretation for the zeta function of Riemann and some technical regularization for spectral representation	43
M. Trifan - Some mathematical models in cancer theory	48
A. Ionescu - On the variation of the stationary states for an excitable model	52
I. Miodrag - Determination of an extremal domain for the functions of class S	55
L. Ungureanu - Cusp bifurcation in a problem of microeconomic dynamics	61
L. Antonescu, G.I. Oros - Observatii metodice asupra compunerii functiilor	65
O.D. Makinde, M.O. Adebayo - Fluid dynamics of expanding and contracting tube: a case study of methods of series summation	68
L. Marra - Project of a hybrid-heuristic algorithm	79
A. Șerbanescu, P. Ciotirnae, D. Andrei - Applications of nonlinear dynamics to communications security	92
O. Tanasescu, H. Tanasescu - Hydrodynamics of dispersed, multiphase flows by wave breaking	99
R. Turcan, V. Turcan - Mathematics modelling of main process of pollutants transport	107
M. Văleanu, G. Moldovan - Integrity in distributed database	114
I. Dzițac, S. Dzițac, H. Oros - Some applications of the Perov's fixed point theorem	120
Aripov, M., Sadullaeva, Sh. A - Properties of the solutions of a parabolic equation of non divergent type	130
M. Vereș - Modelarea agregatelor hidroenergetice prin metoda elementelor finite	136
M. Parv - A computer package for analysis of variance	143
 Education section	
L. Antonescu - Inegalități algebrice folosite în demonstrarea unor inegalități geometrice	157
Ș. Hăbuc - Mulțimea lui Mandelbrot	160
G. Monica - Probleme distractive de matematică	164
C.L. Voica, E. Popescu - Studiul matematic și fizic al centrului de greutate al unui corp: teorie, experiment, metodică	168
V. Oprișor - Nota asupra unor relații trigonometrice	173
E.C. Pufu, A. Moșteanu - Geometria maselor și centre de greutate	175