HU ISSN 1586–2070

JOURNAL OF COMPUTATIONAL AND APPLIED MECHANICS

A Publication of the University of Miskolc

VOLUME 4, NUMBER 2 (2003)

MISKOLC UNIVERSITY PRESS

HU ISSN 1586–2070

JOURNAL OF COMPUTATIONAL AND APPLIED MECHANICS

A Publication of the University of Miskolc

VOLUME 4, NUMBER 2 (2003)

MISKOLC UNIVERSITY PRESS

EDITORIAL BOARD

István PÁCZELT, Editor in Chief, Department of Mechanics, University of Miskolc, 3515 MISKOLC, Hungary, mechpacz@gold.uni-miskolc.hu

László BARANYI, Department of Fluid and Heat Engineering, University of Miskolc, 3515 MISKOLC, Hungary, arambl@gold.uni-miskolc.hu

Edgár BERTÓTI, Department of Mechanics, University of Miskolc, 3515 MISKOLC, Hungary, mechber@gold.uni-miskolc.hu

Tibor CZIBERE, Department of Fluid and Heat Engineering, University of Miskolc, 3515 MISKOLC, Hungary, aramct@gold.uni-miskolc.hu

Wolfram FRANK, Institut für Fluid- und Thermodynamik, Universität Siegen, Paul-Bonatz-Strasse 9-11, 57076 SIEGEN, Germany, frank@ift.mb.uni-siegen.de

Ulrich GABBERT, Institut für Mechanik, Otto-von-Guericke-Universität Magdeburg, Universitätsplatz 2, 39106 MAGDEBURG, Germany, ulrich.gabbert@mb.uni-magdeburg.de

Zsolt GÁSPÁR, Department of Structural Mechanics, Budapest University of Technology and Economics, Műegyetem rkp. 3, 1111 BUDAPEST, Hungary, gaspar@ep-mech.me.bme.hu

Robert HABER, Department of Theoretical and Applied Mechanics, University of Illinois at Urbana-Champaign, 216 Talbot Lab., 104 S. Wright Str., URBANA, IL 61801, USA r-haber@uiuc.edu

Gábor HALÁSZ, Department of Hydraulic Machines, Budapest University of Technology and Economics, Műegyetem rkp. 3, 1111 BUDAPEST, Hungary, HALASZ@vizgep.bme.hu

Ji Huan HE, Department of Mathematics, College of Basic Science, Shanghai Donghua University, No. 1882 Yan'anxilu Road, 200051 Shanghai, China, jhhe@dhu.edu.cn

Károly JÁRMAI, Department of Materials Handling and Logistics, University of Miskolc, 3515 MISKOLC, Hungary, altjar@gold.uni-miskolc.hu

László KOLLÁR, Department of Strength of Materials and Structures, Budapest University of Technology and Economics, Műegyetem rkpt. 1-3. K.II.42., 1521 BUDAPEST, Hungary, lkollar@goliat.eik.bme.hu

Vladimir KOMPIŠ, Department of Mechanics, Faculty of Mechanical Engineering, University of Žilina, ŽILINA, Slovakia, kompis@fstroj.utc.sk

Imre KOZÁK, Department of Mechanics, University of Miskolc, 3515 MISKOLC, Hungary, mechkoz@gold.uni-miskolc.hu

Márta KURUTZ, Department of Structural Mechanics, Budapest University of Technology and Economics, Műegyetem rkp. 3, 1111 BUDAPEST, Hungary, kurutzm@eik.bme.hu

R. Ivan LEWIS, Room 2-16 Bruce Building, Newcastle University, NEWCASTLE UPON TYNE, NE1 7RU, UK, R.I.Lewis@NCL.AC.UK

Gennadij LVOV, Department of Mechanics, Kharkov Polytechnical Institute, 2 Frunze Str., 310002 KHARKOV, Ukraine, lvovgi@kpi.kharkov.ua

Herbert MANG, Institute for Strength of Materials, University of Technology, Karlsplatz 13, 1040 VIENNA, Austria, Herbert.Mang@tuwien.ac.at

Zenon MROZ, Polish Academy of Sciences, Institute of Fundamental Technological Research, Swietokrzyska 21, WARSAW, Poland zmroz@ippt.gov.pl

Tibor NAGY, Department of Physics, University of Miskolc, 3515 MISKOLC, Hungary, fiznagyt@uni-miskolc.hu

Gyula PATKÓ, Department of Machine Tools, University of Miskolc, 3515 MISKOLC, Hungary, mechpgy@uni-miskolc.hu

Jan SLADEK, Ústav stavbeníctva a architektúry, Slovenskej akadémie vied, Dubróvska cesta 9, 842 20 BRATISLAVA, Slovakia, usarslad@savba.sk

Gábor STÉPÁN, Department of Mechanics, Budapest University of Technology and Economics, Műegyetem rkp. 3, 1111 BUDAPEST, Hungary, stepan@mm.bme.hu

Barna SZABÓ, Center for Computational Mechanics, Washington University, Campus Box 1129, St. LOUIS, MO63130, USA, szabo@ccm.wustl.edu

Szilárd SZABÓ, Department of Fluid and Heat Engineering, University of Miskolc, 3515 MISKOLC, Hungary, aram2xsz@uni-miskolc.hu

György SZEIDL, Department of Mechanics, University of Miskolc, 3515 MISKOLC, Hungary, Gyorgy.SZEIDL@uni-miskolc.hu

LOCAL EDITORIAL COUNCIL

T. CZIBERE, I. KOZÁK, I. PÁCZELT, G. PATKÓ, G. SZEIDL

Journal of Computational and Applied Mechanics, Vol. 4., No. 2., (2003), pp. 97–104

LOCAL BOUNDARY CONDITIONS WITH A HIGH-RESOLUTION NUMERICAL SCHEME FOR NON-OSCILLATORY SHOCK ABSORPTION AND REFLECTION

Satya Baboolal
Department of Computer Science, University of Durban-Westville
Private Bag X54001, Durban 4000, South Africa
sbab@pixie.udw.ac.za

Richard Naidoo
Department of Mathematics and Physics, Durban Institute of Technology
P.O. Box 1334, Durban 4000, South Africa
naidoor@yoda.cs.udw.ac.za

[Received: July 24, 2002]

Abstract. In this work we examine some simple non-oscillatory boundary conditions that can be applied to a typical class of modern finite-difference shock capturing schemes. In particular we concentrate on a non-staggered source-term version of the Nessyahu-Tadmor scheme [3] applied to some one-dimensional model equations admitting shocks that can interact with the system boundaries.

Mathematical Subject Classification: 65M06, 35L65, 35L67
Keywords: boundary conditions, shock capturing, high resolution schemes

1. Introduction

Recently many schemes have been employed in numerical studies of hyperbolic systems. Such systems occur in a wide variety of problems, including shock propagation in fluids [1]. New methods for the numerical solution of such systems fall into several classes, of which the fully discrete schemes of Nessyahu and Tadmor [2] and their extensions constitute one important class of Riemann-solver-free schemes for shock propagation problems. Here we concentrate on a non-staggered variation [3] of [2] and pay particular attention to what transpires on a wave structure at a boundary, especially when it is to be fully absorbed and when it is to be reflected. To illustrate the application of these two types of boundary conditions, we examine their effect on shock structures modelled by such equations as the inviscid Burgers' equation and the equations for a Broadwell gas, which is a two-dimensional neutral gas model allowing for one-dimensional spatial variation. We formulate and test local boundary conditions required to smoothly absorb and to smoothly reflect shocks at the boundaries. Because of the spatial extent of the finite-difference grid stencils of modern high resolution schemes, it is required to employ "thick" boundary conditions, whereby, not only a single end point is treated as a boundary point but so too are its immediate neighbours. Tests conducted with this device in conjunction with the above [3] high resolution scheme indicate that simple local conditions can indeed result in smooth boundary interactions.

2. Model equations and the numerical scheme

The model equations studied are of the type,

$$\frac{\partial u(x,t)}{\partial t}+\frac{\partial f\left(u\right)}{\partial x}=g\left(u\right), \tag{2.1}$$

a one-dimensional hyperbolic system [1] of partial differential equations. Here $u(x,t)$ is the unknown m-dimensional vector function, $f(u)$ is the flux vector, $g(u)$ is a continuous source vector function, with x the single spatial coordinate and t the temporal coordinate. For the numerical integration of (2.1) we consider uniform spatial and temporal grids with the spacings, $\Delta x{=}x_{j+1}-x_j$; $\Delta t{=}t^{n+1}-t^n$ (with j and n being suitable integer indices) and employ the non-staggered-grid version [3] of [2]:

$$\begin{aligned}\bar{u}_j^{n+1} \;=\;& \frac{1}{4}\left(\bar{u}_{j+1}^n+2\bar{u}_j^n+\bar{u}_{j-1}^n\right)-\frac{1}{16}\left(u_{xj+1}^n-u_{xj-1}^n\right)-\frac{1}{8}\left[u_{xj+\frac{1}{2}}^{n+1}-u_{xj-\frac{1}{2}}^{n+1}\right]\\ &+\frac{\Delta t}{8}\left[g\left(u_{j+1}^n\right)+2g\left(u_j^n\right)+g\left(u_{j-1}^n\right)\right]\\ &+\frac{\Delta t}{8}\left[g\left(u_{j+1}^{n+1}\right)+2g\left(u_j^{n+1}\right)+g\left(u_{j-1}^{n+1}\right)\right]\\ &-\frac{\lambda}{4}\left[\left(f_{j+1}^n-f_{j-1}^n\right)+\left(f_{j+1}^{n+1}-f_{j-1}^{n+1}\right)\right], \end{aligned} \tag{2.2}$$

where $\lambda=\Delta t/\Delta x$ and the subscript x denotes differentiation with respect to x. This scheme has been obtained by extending the scheme [2] based on a staggered spatial grid to one based on a non-staggered grid, following the procedure outlined in [4]. In addition, we have also incorporated a non-zero source term $g(u)$ so that (2.1) above can be applied to non-homogeneous systems. Full details are available from the authors [3]. Consequently, of course, it is much easier to apply boundary conditions on non-staggered grids. Further, in order to apply this scheme it is customary to employ suitable non-oscillatory derivative approximations. In all our calculations we used the "UNO" derivative approximation [2] for the derivative terms indicated by the subscript x in (2.1). In particular we utilize the form,

$$\begin{aligned}u_{xj} \;=\;& MM(u_j-u_{j-1}+\frac{1}{2}MM(u_j-2u_{j-1}+u_{j-2},u_{j+1}-2u_j+u_{j-1}),\\ & u_{j+1}-u_j-\frac{1}{2}MM(u_{j+1}-2u_j+u_{j-1},u_{j+2}-2u_{j+1}+u_j)), \end{aligned} \tag{2.3}$$

where the min-mod non-linear limiter MM is defined by

$$MM(s_1, s_2, ...) = \begin{cases} \min\{s_j\} \ \mathit{if}\ s_j > 0 \quad \forall j \\ \max\{s_j\} \ \mathit{if}\ s_j < 0 \quad \forall j \\ 0 \quad \mathit{otherwise}\,. \end{cases}$$

Many other derivative approximations are used in practice (see for example [5]).

3. **Boundary conditions**

3.1. **Absorption.** In this section we examine boundary conditions conditions for absorbing shocks.

First for computation over $x \in [x_L, x_R]$ with corresponding $j = L, L+1, ..., R-1, R$, we note from (2.2) and (2.3) that the spatial index range must be $L+2 \le j \le R-2$. Thus the u-values at the grid points corresponding to $j = L, L+1$ and $j = R-1, R$ need to be computed at the next time level. For complete absorption we employ successive double quadratic extensions in the forms:

$$u_{L+1} = u_{L+4} - 3u_{L+3} + 3u_{L+2}\,, \quad u_L = u_{L+3} - 3u_{L+2} + 3u_{L+1}\,, \tag{3.1}$$

$$u_{R-1} = u_{R-4} - 3u_{R-3} + 3u_{R-2}\,, \quad u_R = u_{R-3} - 3u_{R-2} + 3u_{R-1}\,, \tag{3.2}$$

which apply respectively to the left ($j = L, L+1$) and right ($j = R-1, R$) boundaries. These expressions can be shown to be second order accurate in the grid spacing Δx and hence are consistent with the second-order numerical scheme (2.2). Issues relating to the specific components on which they are to be applied are discussed in the next section.

3.2. **Reflection.** For reflection of a shock we shall find that specifically the density component should satisfy the homogeneous Neumann condition ($\partial u/\partial x = 0$), and quadratic extensions for the other two components again on two successive points at each end as above. We thus employ the successive Neumann conditions:

$$u_{L+1} = (4u_{L+2} - u_{L+3})/3;\ u_L = (4u_{L+1} - u_{L+2})/3 \tag{3.3}$$

$$u_{R-1} = (4u_{R-2} - u_{R-3})/3;\ u_R = (4u_{R-1} - u_{R-2})/3. \tag{3.4}$$

These are one-sided derivatives which can again be shown to be second-order accurate and thus consistent with (2.2). An interpretation of these conditions is given in the context of the applications below. In the following we present two test cases.

4. **Test applications**

4.1. **Burgers' equation.** First, as a simple illustration we apply the scheme (2.2) to the one-dimensional inviscid Burgers' equation:

$$\begin{aligned} &\frac{\partial u(x,t)}{\partial t} + \frac{1}{2}\frac{\partial\left(u^2\right)}{\partial x} = 0, \\ &u(x,0) = u_0(x), \quad t > 0. \end{aligned} \tag{4.1}$$

A characteristic analysis [1] of this system admits waves that propagate with a constant speed in one direction only. Hence we can only expect to study absorption of a disturbance into a boundary and no reflection from it. Further, it is well known [1] that any initial waveform will evolve into a shock or a rarefaction wave. We allow for the smooth passage of such a shock by setting the outflow conditions (3.2). These naturally allow free adjustment of the flow, thus catering for smooth absorption. Figure 1 depicts a typical initial condition, which evolves into a shock. Frame (b) depicts the situation when the shock has moved close to the boundary and illustrates smooth oscillation-free absorption. Frame (c) depicts the situation when the shock has been fully absorbed, resulting in a newer equilibrium level.

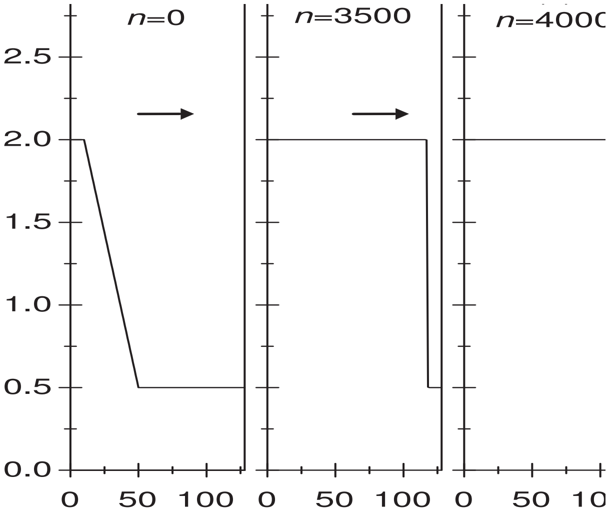


Figure 1. Absorption of a Burgers' equation shock. The plots show $u(x,t)$ as a function of the x-coordinate at fixed times $t = n \times 0.01$

4.2. **Broadwell gas model.** In this case we consider a multi-component problem modelled by the Broadwell gas equations [6,7],

$$\frac{\partial \rho}{\partial t} + \frac{\partial m}{\partial x} = 0, \quad \frac{\partial m}{\partial t} + \frac{\partial z}{\partial x} = 0, \quad \frac{\partial z}{\partial t} + \frac{\partial m}{\partial x} = \frac{1}{\varepsilon}\left(\rho^2 + m^2 - 2\rho z\right). \tag{4.2}$$

Here ε is the mean free path and $\rho(x,t), m(x,t), z(x,t)$ are the density, momentum and flux, respectively. For illustration we compute the 'stiff' case with $\varepsilon = 10^{-8}$ on a fine grid with $\Delta t = 0.1 \times \varepsilon$ and $\Delta x = 3 \times \Delta t$. An analysis [6,7] of the corresponding eigensystem shows that both left and right moving waves are admissible, with characteristic speeds of $+1$, -1, and 0. Thus we can propagate waves from the left and also obtain waves that can be reflected back into the system from the boundaries.

In order to achieve this for shocks, we choose a typical Riemann initial condition to produce a right propagating shock, such as,

$$\text{Rim1}: \begin{cases} \rho = 2, m = 1, z = 1; & x < 200 \\ \rho = 1, m = 0.13962, z = 1; & x > 200 \end{cases}$$

on $0 \leq x \leq 800$.

Then in the first instance we choose for the boundary conditions the free flow conditions (3.1)-(3.2) to study shock absorption. Figure 2 (a)-(b) depicts the situation of smooth absorption of the shock after about $t = 2300 \times \varepsilon \times 0.01$.

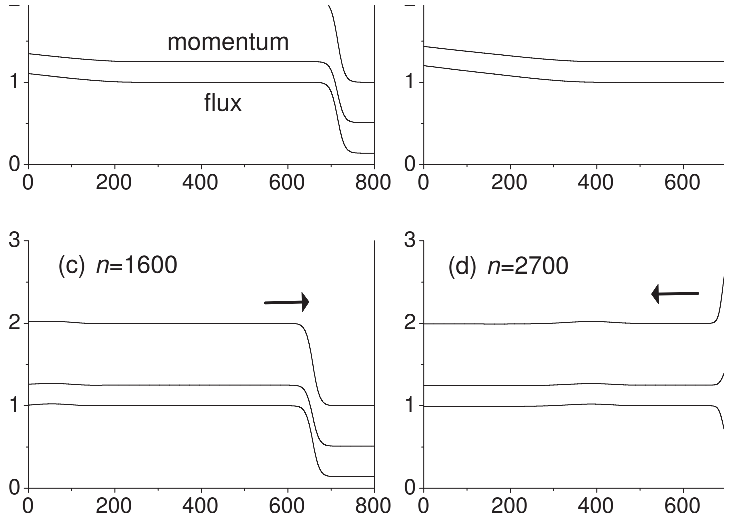


Figure 2. Absorption: (a)-(b) and reflection: (c)-(d) of Broadwell gas shocks at various time steps n. Plots of the density $\rho(x,t)$, momentum $m(x,t)$ and flux $z(x,t)$ are shown as functions of the x-coordinate

In the second instance to study shock reflection, we employ the same initial conditions but with the Neumann condition (3.4) on the density, together with the free flow conditions (3.2) on the momentum and flux. Here again, Figure 2 (c)-(d) shows that the shock is smoothly reflected with no spurious oscillations. The interpretation of the boundary conditions in this case may be seen as follows: the Neumann condition for the density forces the density profile at the boundary to level off, i.e. to prohibit any density gradient from forming. The latter would be the case when for example, free-flow density profiles are imposed. Thus since the momentum and flux fields are allowed to freely adjust for this condition, the fluid near the right boundary will be forced to dam-up due to the flow from the left, resulting in increasing density at the right. This continues until the upstream shock level has completely reached the boundary. Then the resulting compression in the vicinity of the right boundary

will tend to propagate out back into the system. This propagating front develops into a shock which travels to the left, with a higher upstream level. Use of such a homogeneous Neumann condition has also been made in other studies of shock reflection, for example in a two-dimensional shock tube problem [8]. We find such a device to be a good way to smoothly reflect the shock from the right boundary.
Whilst there are more sophisticated boundary conditions that have been reported in the literature [1,9,10,11], we are not aware of studies of their applicability to shock propagation. For example, in [9], apart from physical boundary conditions, soft or numerical ones are formulated which are treated as additional compatibility relations. These are obtained by considering the passage of inflowing and outflowing linear waves corresponding to the characteristics of the problem. Their wave amplitudes are then computed from the solution of additional local one-dimensional inviscid (LODI) relations which are compatible with the conservation laws and which have to be solved at the boundaries. Then for outgoing waves the internal grid values determine the boundary values, whilst for incoming (or reflected) waves and for other variables required by the numerical scheme the LODI relations are used. To obtain some correspondence between our simple approach for shock reflection and the corresponding LODI relations of Poinsot and Lele [9] we proceed as follows: we note that their LODI relations (which we extract for one-dimension $\sim x_1$ coordinate) and which they determined by examining the corresponding inviscid flow corresponding to ideal gas Navier-Stokes equations are

$$\begin{aligned}\frac{\partial\rho}{\partial t}+\frac{1}{c^2}\left[A_2+\frac{1}{2}(A_5+A_1)\right]&=0,\\ \frac{\partial p}{\partial t}+\frac{1}{2}\left(A_5+A_1\right)&=0,\\ \frac{\partial u_1}{\partial t}+\frac{1}{2\rho c}\left(A_5-A_1\right)&=0,\\ A_1=(u_1-c)\left(\frac{\partial p}{\partial x_1}-\rho c\frac{\partial u_1}{\partial x_1}\right)&,\\ A_5=(u_1+c)\left(\frac{\partial p}{\partial x_1}+\rho c\frac{\partial u_1}{\partial x_1}\right)&,\end{aligned}$$

where $p, \rho, u_1, c, A_1, A_5$ are the pressure, density, flow velocity, speed of sound, incoming (or reflected) wave amplitude and outgoing wave amplitude, respectively.

Now, for reflection at a solid boundary one can set the physical condition $u_1 = 0$ for all times. Then with

$$\frac{\partial u_1}{\partial t}=\frac{\partial u_1}{\partial x_1}=0 \tag{4.3}$$

at the boundary, the above implies that

$$\frac{\partial p}{\partial x_1}=0 \tag{4.4}$$

there. We observe that this condition on the density coincides with our choice (homogeneous Neumann) employed in the Broadwell gas case above.

Finally, we should mention that their particular LODI relations quoted here are valid for the case of an ideal gas, and, to our understanding, for smooth wave propagation and thus one cannot infer its validity for shock propagation. In [10] however, their approach is extended for more realistic gases, but still without consideration of shock propagation. Another point to make here is that the type of numerical scheme employed also determines the extent or number of these further boundary or compatibility relations. These are nevertheless interesting questions which are matters for further investigation. In contrast here, we are formulating and testing simple local conditions which are computationally inexpensive.

5. Conclusion

We have outlined and tested some simple boundary conditions that can be used with a class of high resolution central difference schemes which are commonly employed in studies of shock capturing and wave propagation. In particular, when we employ these boundary conditions successively on a boundary point and its immediate neighbour, a device which we designate as "thick" boundary conditions, our tests indicate that smooth shock absorption and reflection can occur. These simple local boundary conditions are thus expected to be useful for other similar difference schemes.

References

1. Godlewski, E. and Raviart, P.-A.: *Numerical Approximation of Hyperbolic Systems of Conservation Laws*, Springer-Verlag, New York, 1996.
2. Nessyahu, H. and Tadmor, E.: *Non-oscillatory central differencing for hyperbolic conservation laws*, Journal of Computational Physics, **87**, (1990), 408-463.
3. Naidoo, R. and Baboolal, S.: *Numerical Integration of the Plasma Fluid Equations with a Non-Staggered Modification of the Second-Order Nessyahu-Tadmor Central Scheme and Soliton Modelling*, University of Durban-Westville Computer Science Research Report CSCR-2003/01, (2003), 1-14.
4. Jiang, G.-S., Levy, D., Lin, C.-T., Osher, S. and Tadmor, E.: *High-resolution non-oscillatory central schemes with non-staggered grids for hyperbolic conservation laws*, SIAM Journal on Numerical Analysis, **35**, (1998), 2147-2168.
5. Kurganov, A. and Tadmor, E.: *New high-resolution central schemes for nonlinear conservation laws and convection-diffusion equations*, Journal of Computational Physics, **160**, (2000), 241-282.
6. Caflisch, R.E., Jin, Shi and Russo, G.: *Uniformly accurate schemes for hyperbolic systems with relaxation*, SIAM Journal on Numerical Analysis, **34**, (1997), 246-281.
7. Jin, Shi: *Runge-Kutta methods for hyperbolic conservation laws with stiff relaxation terms*, Journal of Computational Physics, **122**, (1995), 51-67.
8. Banda, M.K. and Seaid, M.: *A Class of the Relaxation Schemes for the Two-Dimensional Euler Systems of Gas Dynamics*, Lecture Notes in Computer Science, Springer-Verlag, 2329, (2002), 930-939.
9. Poinsot, T. J. and Lele, S. K.: Boundary conditions for direct simulations of compressible viscous flows, *J. Comput. Phys.*, **101**, (1992), 104-129.

10. Baum, M., Poinsot, T. and Thevenin, D.: *Accurate boundary conditions for multicomponent reactive flows*, Journal of Computational Physics, **116**, (1994), 247-261.

11. Givoli, D.: *High-order nonreflecting boundary conditions without high order derivatives*, Journal of Computational Physics, **170**, (2001), 849-870.

PARALLEL INCOMPLETE FACTORIZATION PRECONDITIONING OF ROTATED LINEAR FEM SYSTEMS

Gergana Bencheva, Svetozar Margenov
Central Laboratory for Parallel Processing, Bulgarian Academy of Sciences
Acad. G. Bontchev str., Bl. 25A, 1113 Sofia, Bulgaria
{gery,margenov}@parallel.bas.bg

[Received: June 19, 2002]

Abstract. The recent efforts in development of efficient solution methods for non-conforming finite element systems are inspired by their importance for various applications in scientific computations and engineering. This study is focused on the implementation of rotated bilinear elements. A locally modified approximation of the global stiffness matrix is proposed allowing for: a) a stable MIC(0) factorization; and b) a scalable parallel implementation. An optimal condition number estimate is derived for the constructed sparse matrix approximation with respect to the original global stiffness matrix. The estimates of the parallel speed-up and the parallel efficiency as well as the presented parallel numerical tests demonstrate the potential of the PCG algorithm and the MPI code developed.

Mathematical Subject Classification: 65F10, 65N30
Keywords: non-conforming FEM, preconditioning, parallel algorithms

1. Introduction

Consider the elliptic equation

$$\begin{aligned} -\nabla \cdot (a(x)\nabla u(x)) &= f(x) && \text{in} \quad \Omega, \\ u &= 0 && \text{on} \quad \Gamma_D, \\ (a(x)\nabla u(x)) \cdot n &= 0 && \text{on} \quad \Gamma_N. \end{aligned} \tag{1.1}$$

Here $\nabla u(x)$ denotes the gradient of u and $\nabla \cdot q$ denotes the divergence of the vector q. Further, we assume that Ω is a convex polygonal domain in $\mathbb{R}^2$, $f(x)$ is a given function in $L^2(\Omega)$, $a(x) = [a_{ij}(x)]_{i,j=1}^2$ is a symmetric matrix, n is the outward unit vector normal to the boundary $\Gamma = \partial\Omega$, and $\Gamma = \bar{\Gamma}_D \cup \bar{\Gamma}_N$. We assume that $a_{ij}(x)$ are piece-wise smooth functions on $\bar{\Omega}$ satisfying the uniform positive definiteness condition of the matrix $a(x)$.

Problem (1.1) can be discretized in various ways. Among the most popular and frequently used methods of approximation are the Galerkin finite element method, the finite volume method and the mixed finite element method. Each of these methods has its advantages and disadvantages when applied to particular engineering problems. For example, for petroleum reservoir problems in geometrically simple domains and heterogeneous media, the finite volume method is known to be reliable, accurate, and locally mass conservative. Many engineering problems, e.g. petroleum recovery, ground-water contamination, seismic exploration, etc. need very accurate velocity (flux) determination in the presence of heterogeneities, anisotropy and large jumps in the coefficient matrix $a(x)$. More accurate approximation of the velocity can be achieved through the use of the mixed finite element method (see, e.g. [5]). However, the technique of the mixed finite element method leads generally to an algebraic saddle point problem that is more difficult and more expensive to solve.

An alternative approach can be taken by developing hybrid methods where the continuity of the velocity vector normal to the boundary of each element is enforced by Lagrange multipliers. The important discovery of Arnold and Brezzi [2] is that the Schur system for the Lagrange multipliers can be obtained also as a discretization of (1.1) by Galerkin method using linear nonconforming elements. Namely in [2] it is shown that the lowest-order Raviart-Thomas mixed element approximations are equivalent to the usual Crouzeix-Raviart P_1-nonconforming finite element approximations when the classical P_1-nonconforming space is augmented with P_3-bubbles. Further, such a relationship has been studied for a large variety of mixed finite element spaces [1, 6].

Our study is focused on the implementation of rotated bilinear elements. These elements are an attractive discretizing tool since they possess favorable stability properties for the Stokes and the Lamé equations. An additional important feature is the regular sparsity of the stiffness matrices with no more than seven non-zero elements per row even in the case of non-regular meshes.

Two algorithms are presented, where MP and MV stand for the variants of the nodal basis functions corresponding to mid-point and integral mid-value interpolation operators.

There are two general approaches to construct parallel preconditioners, based respectively on: a) domain decomposition, or b) block incomplete/approximate factorization. The second approach does not lead to an optimal preconditioner in terms of problem size, but produces highly parallel and efficient algorithms (see, [3, 4, 7, 8, 9, 10, 11]).

Here we first locally modify the stiffness matrix, and then apply a pointwise incomplete factorization. We get as a result a well parallelizable block structure of the preconditioner, preserving the robustness with respect to the local properties of the matrix.

2. **Finite element discretization**

The domain Ω is partitioned using quadrilaterals $e \in \omega_h$. Our analysis here is concentrated on the isotropic case, and we will assume from now that $a(x)$ is a scalar function. The partitioning ω_h is aligned with the discontinuities of the coefficient $a(x)$ so that over each element $e \in \omega_h$ the function $a(x)$ is smooth. Further, we assume that the partitioning is quasi-uniform with a characteristic mesh-size h.

The Galerkin variational formulation of the above problem reads: given $f \in L^2(\Omega)$ find a function $u \in H_D^1(\Omega) = \{v \in H^1(\Omega) : v = 0 \text{ on } \Gamma_D\}$, satisfying

$$\mathcal{A}(u,v) = (f,v) \qquad \forall v \in H_D^1(\Omega), \tag{2.1}$$

where

$$\mathcal{A}(u,v) = \int_\Omega a(x)\nabla u(x) \cdot \nabla v(x) dx\cdot$$

The rotated bilinear non-conforming finite elements on quadrilaterals are implemented for the numerical solution of (2.1). The finite element space V_h corresponds to ω_h. This study is concerned with a comparison analysis of two alternative constructions of V_h, where algorithms MP and MV stand for the variants of the nodal basis functions corresponding to mid-point and integral mid-value interpolation operators (see [12]). In defining the isoparametric rotated bilinear element one uses the unit square (with sides parallel to the coordinate axes) as a reference element $\hat{e}$. For each $e \in \omega_h$, let $\psi_e : \hat{e} \to e$ be the corresponding bilinear transformation. The element nodal basis functions are determined by the relations

$$\{\phi_i\}_{i=1}^4 = \{\hat{\phi}_i \circ \psi_e^{-1}\}_{i=1}^4.$$

For algorithm MP, the reference element basis functions $\{\hat{\phi}_i\}_{i=1}^4$ are determined by the standard nodal interpolation conditions

$$\hat{\phi}_i(b_\Gamma^j) = \delta_{ij},$$

where $\{b_\Gamma^j\}_{j=1}^4$ are the mid-points of the sides $\{\Gamma_{\hat{e}}^j\}_{j=1}^4$ of $\hat{e}$, and then

$$\{\hat{\phi}_i\}_{i=1}^4 = \left\{ \left(1 \pm 2\xi_i + \xi_i^2 - \xi_{j+1}^2\right)/4,\ j = 1,2 \right\}.$$

Alternatively, for algorithm MV, an integral mid-value interpolation operator is applied in the form

$$|\Gamma_{\hat{e}}^j|^{-1} \int_{\Gamma_{\hat{e}}^j} \hat{\phi}_i = \delta_{ij},$$

and then

$$\{\hat{\phi}_i\}_{i=1}^4 = \left\{ \left(2 \pm 4\xi_j + 3\left(\xi_j^2 - \xi_{j+1}^2\right)\right)/16,\ j = 1,2 \right\}.$$

Then the finite element formulation is: find a function $u_h \in V_h$, satisfying

$$\mathcal{A}_h(u_h, v_h) = (f, v_h) \qquad \forall v_h \in V_h, \tag{2.2}$$

where

$$\mathcal{A}_h(u_h, v_h) = \sum_{e \in \omega_h} \int_e a(e)\nabla u_h \cdot \nabla v_h dx,$$

where $a(e)$ is defined as the averaged value

$$a(e) = \frac{1}{|e|}\int_e a(x)dx$$

over each $e \in \omega_h$. We note that we allow strong coefficient jumps through the interface boundaries between the elements. Now, the standard computational procedure leads to the linear system of equations

$$A\mathbf{u} = \mathbf{f}, \tag{2.3}$$

where A is the corresponding stiffness matrix. The matrix A is sparse, symmetric and positive definite. For large scale problems, the preconditioned conjugate gradient (PCG) method is known to be the best solution method of (2.3).

The goal of this study is to present a **robust and parallelizable** preconditioning algorithm for solving (2.3). The proposed method will be based on incomplete factorization of sparse matrices. The construction of the preconditioner is based on a MIC(0) factorization of a locally modified approximation of the original stiffness matrix. We have shown in our model analysis that the condition number corresponding to the applied local modification is independent of possible coefficient jumps (see Section 5). The proposed construction is aimed to improve the parallel features of the algorithm, which are discussed in Section 7. A set of numerical tests illustrating the robustness of the method, and the efficiency of the parallel implementation are shown in Sections 6, 8, respectively. Some concluding remarks are given in the last section.

3. MIC(0) preconditioning

We present here some background remarks about the modified incomplete Cholesky factorization MIC(0) preconditioner. Our presentation at this point follows those in [3], see also [9]. Let us rewrite the real $N \times N$ matrix $A = (a_{ij})$ in the form

$$A = D - L - L^t \,, \tag{3.1}$$

where D is the diagonal and $(-L)$ is the strictly lower triangular part of A. Then we consider the approximate factorization of A, which has the following form:

$$\mathcal{C}_{MIC(0)}(A) = \mathcal{C}_{MIC(0)} = (X - L)X^{-1}(X - L)^t \,, \tag{3.2}$$

where $X = diag(x_1, \cdots, x_N)$ is a diagonal matrix determined by the condition of equal row sums:

$$\mathcal{C}_{MIC(0)}\underline{e} = A\underline{e}, \qquad \underline{e} = (1, \cdots, 1)^t \in \mathcal{R}^N \,.$$

For the purpose of preconditioning, we are interested in the case when $X > 0$ and thus $\mathcal{C}_{MIC(0)}$ is positive definite. If this holds, we speak about *stable* $MIC(0)$ *factorization*. Concerning stability of $MIC(0)$ factorization, the following theorem holds.

Theorem 1. *Let $A = (a_{ij})$ be a symmetric real $N \times N$ matrix and let $A = D - L - L^t$ be the splitting (3.1) of A. Let us assume that*

$$\begin{array}{rcl} L & \geq & 0 \\ A\underline{e} & \geq & 0 \\ A\underline{e} + L^t\underline{e} & > & 0 \qquad \underline{e} = (1, \cdots, 1)^t \in \mathcal{R}^N, \end{array}$$

i.e. that A is a weakly diagonally dominant matrix with nonpositive off-diagonal entries and that $A + L^t = D - L$ is strictly diagonally dominant.

Then the relation

$$x_i = a_{ii} - \sum_{k=1}^{i-1} \frac{a_{ik}}{x_k} \sum_{j=k+1}^{N} a_{kj}$$

gives the positive values x_i and the diagonal matrix $X = diag(x_1, \cdots, x_N)$ defines stable $MIC(0)$ factorization of A.

Remark 1. *The numerical tests presented in this paper are performed using the perturbed version of $MIC(0)$ algorithm, where the incomplete factorization is applied to the matrix $\tilde{A} = A + \tilde{D}$. The diagonal perturbation $\tilde{D} = \tilde{D}(\xi) = diag(\tilde{d}_1, \ldots \tilde{d}_N)$ is defined as follows:*

$$\tilde{d}_i = \begin{cases} \xi a_{ii} & if \quad a_{ii} \geq 2w_i \\ \xi^{1/2} a_{ii} & if \quad a_{ii} < 2w_i \end{cases}$$

where

$$w_i = -\sum_{j>i} a_{ij}.$$

Here $0 < \xi < 1$ is a constant of the same order as the minimal eigenvalue of A. The computations for the model problems considered are done with $\xi = h^2$.

It is readily seen from (3.2) that the computational cost of one MIC(0) PCG iteration is proportional to the size of the matrix A. More precisely, the complexity $\mathcal{N}(\mathcal{C}^{-1}_{MIC(0)}\mathbf{v})$ is almost the same as $\mathcal{N}(A\mathbf{v})$. This will be discussed in some more details later. Unfortunately, the method is based on recursive computations, and therefore is inherently sequential. The idea of our algorithm is to apply MIC(0) factorization to a modified sparse matrix the special block structure of which allows for a scalable parallel implementation.

4. The preconditioning algorithm

The studied preconditioner $\mathcal{C}$ is constructed by a proper local modification of the stiffness matrix A. Following the standard FEM assembling procedure we write A in the form

$$A = \sum_{e \in \omega_h} L_e^T A_e L_e,$$

where A_e is the element stiffness matrix, L_e stands for the restriction mapping of the global vector of unknowns to the local one corresponding to the current quadrilateral element e. We now introduce the approximation B_e of A_e as follows

$$A_e = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}, \quad B_e = \begin{bmatrix} b_{11} & a_{12} & 0 & a_{14} \\ a_{21} & b_{22} & a_{23} & 0 \\ 0 & a_{32} & b_{33} & a_{34} \\ a_{41} & 0 & a_{43} & b_{44} \end{bmatrix}, \tag{4.1}$$

where

$$b_{11} = a_{11} + a_{13}, \quad b_{22} = a_{22} + a_{24}, \quad b_{33} = a_{33} + a_{31}, \quad b_{44} = a_{44} + a_{42},$$

that is A_e and B_e have equal row sums.

Assembling the locally defined matrices B_e we get the global one

$$B = \sum_{e \in \omega_h} L_e^T B_e L_e, \tag{4.2}$$

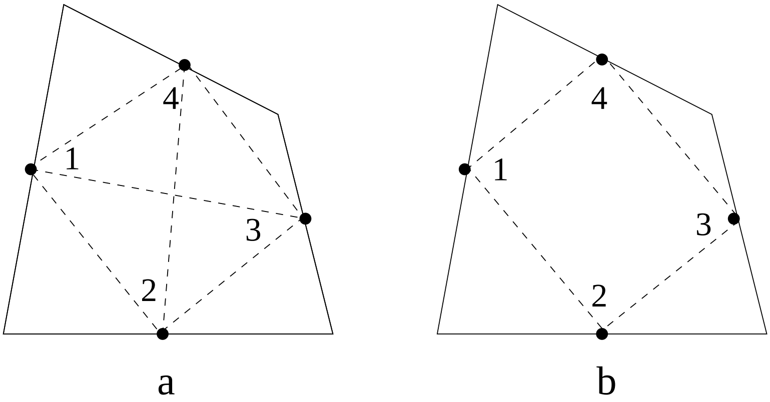


Figure 1. (a) Node numbering of a rotated bilinear quadrilateral element e; (b) Connectivity pattern of B_e.

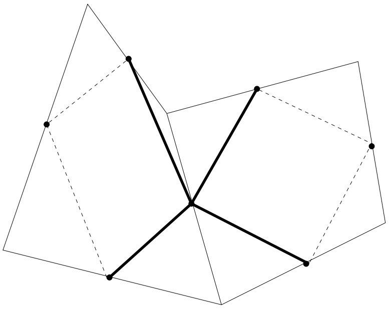


Figure 2. Skewed five point stencil

The definition of B_e corresponds to the node numbering as shown in Figure 1. Here the dash lines represent the connectivity pattern of (a) the dense element stiffness matrix A_e and (b) its locally modified sparse approximation B_e. The structure of B could be interpreted as a skewed five point stencil (see Figure 2) whereas in a very general setting A and B are spectrally equivalent.

At this point we introduce the preconditioner $\mathcal{C}$ for A which is defined as a MIC(0) factorization of B, that is,

$$\mathcal{C} = \mathcal{C}_{MIC(0)}(B).$$

This needs naturally B to allow for a stable MIC(0) factorization, which in particular will be discussed in the next section.

5. Condition number model analysis

The model problem we analyze in this section is set on a uniform square mesh. Then the element stiffness matrices corresponding to the square element $e \in \omega_h$ in the cases MP and MV have the form:

$$A_e^{MP} = \frac{a(e)}{3} \begin{bmatrix} 5 & -1 & -2 & -2 \\ -1 & 5 & -2 & -2 \\ -2 & -2 & 5 & -1 \\ -2 & -2 & -1 & 5 \end{bmatrix}, \tag{5.1}$$

$$A_e^{MV} = \frac{a(e)}{8} \begin{bmatrix} 5 & 1 & -3 & -3 \\ 1 & 5 & -3 & -3 \\ -3 & -3 & 5 & 1 \\ -3 & -3 & 1 & 5 \end{bmatrix}. \tag{5.2}$$

We consider now the local eigenvalue problem

$$A_e^{MP}\mathbf{w} = \lambda B_e^{MP}\mathbf{w}. \tag{5.3}$$

Obviously $Ker(A_e^{MP}) = Ker(B_e^{MP}) = Span\{\mathbf{e}\}$ where $\mathbf{e}^t = (1,1,1,1)$, and it is therefore enough to consider a reduced 3×3 eigenvalue problem instead of (5.3). Then the simplification using the substitution

$$\mu = 1 - \lambda$$

leads to the following characteristic equation for μ

$$det \begin{bmatrix} 1+4\mu & -1 & -2\mu \\ -1 & 1+4\mu & -2\mu \\ -2\mu & -2\mu & 1+4\mu \end{bmatrix} = 0. \tag{5.4}$$

Further computation shows that $\mu_1 = 0$ and $\mu_{2,3} = -1/2$, and therefore

$$\lambda_1 = 1, \qquad \lambda_{2,3} = 3/2.$$

The global condition number estimate directly follows from the local analysis presented. Namely, we have

$$\mathbf{v}^T A^{MP}\mathbf{v} = \sum_{e\in\omega_h} \mathbf{v}_e^T L_e^T A^{MP} L_e \mathbf{v}_e \le 3/2 \sum_{e\in\omega_h} \mathbf{v}_e^T L_e^T B_e^{MP} L_e \mathbf{v}_e = 3/2\mathbf{v}^T B^{MP}\mathbf{v}$$

and, similarly,

$$\mathbf{v}^T A^{MP}\mathbf{v} \ge \mathbf{v}^T B^{MP}\mathbf{v}.$$

The same approach is directly applied to the matrices A^{MV} and B^{MV} where $\mu_1 = 0$, $\mu_{2,3} = 1/3$, and therefore $\lambda_1 = 1$, $\lambda_{2,3} = 2/3$.

The result of our local analysis is summarized in the next theorem.

Theorem 2. *Let us consider the non-conforming FEM problem (2.2) defined on a square mesh. Then:*

(i) *the sparse approximation B of the stiffness matrix A satisfies the conditions of Theorem 1 for a stable MIC(0) factorization;*

(ii) *the matrices B and A are spectrally equivalent where the next relative condition number estimate holds uniformly with respect to any possible jumps of the diffusion coefficients.*

$$\kappa\left(B^{-1}A\right) \leq 2 \tag{5.5}$$

The above result holds for both MP and MV cases.

6. Numerical tests I

The model problem $-\Delta u = f$ in the unit square is considered, where homogeneous Dirichlet boundary conditions are assumed at the bottom side. The presented numerical tests illustrate the PCG convergence rate of the studied MIC(0) preconditioners when the size of the discrete problem is varied. A relative stopping criterion $(\mathcal{C}^{-1} r^{n_{it}}, r^{n_{it}})/(\mathcal{C}^{-1} r^0, r^0) < \varepsilon$ is used in the PCG algorithm, where r^i stands for the residual at the i-th iteration step, $(\cdot,\cdot)$ is the standard Euclidean inner product, and $\varepsilon = 10^{-6}$. A uniform mesh is used, where $h = 1/n$, and the size of the discrete problem is $N = 2n(n+1)$.

Table 1. *PCG* iterations: *MIC*(0) preconditioning in the cases MP and MV

Meshsize		n_{it}^{AA}		n_{it}^{AB}	
n	N	MP	MV	MP	MV
63	8064	51	48	34	39
127	32512	82	70	50	56
255	130560	133	101	71	81
511	523264	214	144	104	114
1023	2095104	292	208	149	167

The obtained numbers of iterations are reported in Table 1. Here again, MP and MV stand, respectively for the cases of mid-point and integral mid-value interpolation operators used to construct the nodal basis of the related non-conforming FEM basis. We denote here by n_{it}^{AA} and n_{it}^{AB} the number of iterations obtained when MIC(0) factorizations of A and B are used as preconditioners of A.

The following observations are derived from the presented numerical results:

- The number of iterations in all cases is $O(\sqrt{n}) = O(N^{1/4})$.
- The PCG convergence rate is better for the case MV. This is one more advantage of this variant of the rotated bilinear elements (see [12] for more details about the approximation properties).
- $n_{it}^{AB} < n_{it}^{AA}$. Note, that this is considerably better than what we have as a prediction from the uniform estimate from Theorem 2.

- A stable MIC(0) factorization of the matrix A^{MV} for the considered model problem has been obtained. It is important to note that the related conditions from Theorem 1 are not satisfied in this case.

7. **Parallel preconditioning algorithm**

We study in this section the parallel properties of the proposed algorithm. Our analysis is focused on the PCG solution of the linear algebraic system obtained. The preconditioner was introduced as $\mathcal{C} = \mathcal{C}_{MIC(0)}(B)$. Each PCG iteration consists of one solution of a system with the matrix $\mathcal{C}$, one matrix vector multiplication with the original matrix A, two inner products, and three linked vector triads of the form $\mathbf{v} := \alpha\mathbf{v} + \mathbf{u}$. Therefore the computational complexity of one PCG iteration is given by

$$\mathcal{N}_{PCG}^{it} \approx \mathcal{N}(\mathcal{C}^{-1}\mathbf{v}) + \mathcal{N}(A\mathbf{v}) + 10N \approx 34N.$$

In the general case, the solution of triangular systems with matrices $(X - L)$ and $(X - L)^t$ is typically recursive, see (3.2). This is the reason for considering MIC(0) an inherently sequential algorithm. We will show now how this disadvantage has been overcome by the sparse matrix B introduced.

To illustrate the basic idea, we will analyze in a more detailed form the model problem where $\Omega = (0,1)^2$, the square mesh ω_h corresponds to the mesh size $h = 1/n$. In this case the size of the discrete problem is $N = 2n(n+1)$. The structures of the matrices A and B are illustrated in Figure 3, where each of the diagonal blocks corresponds to one vertical line of the mesh if a column-wise numbering of the unknowns has been used (see also [11]). The important advantage of the matrix B is that all of its diagonal blocks are diagonal. In this case, the implementation of the PCG solution step $\mathcal{C}^{-1}\mathbf{v}$ is fully parallel within each of these blocks. One can see at this point how the construction of B has been inspired by the properties of the conforming linear FEM stiffness matrix corresponding to a skewed triangulation (see [10] for some more details). Following [13], we will assume that the computations and communications are not overlapping, and therefore, the parallel execution time is the sum of the computation and communication times. We will also assume that: a) the execution of M arithmetic operations on one processor takes time $T_a = Mt_a$, where t_a is the average unit time to perform one arithmetic operation on one processor (no vectorization); and b) the communication time to transfer M data elements from one processor to another can be approximated by $T_{com} = \ell(t_s + Mt_c)$, where t_s is the start-up time and t_c is the incremental time necessary for each of the M elements to be sent, and ℓ is the graph distance between the processors.

Let us consider a distributed memory parallel algorithm where the number of processors is p $(p > 2)$, and let $n = mp$ with some natural number m. The computational domain is split in p equally sized strips. The processor P_k is responsible for the local computations corresponding to the k-th strip. Then, we get the following expressions for the communication times related to $\mathcal{C}^{-1}\mathbf{v}$ and $A\mathbf{v}$

$$T_{com}(C^{-1}\mathbf{v}) = 8n(t_s + t_c),$$

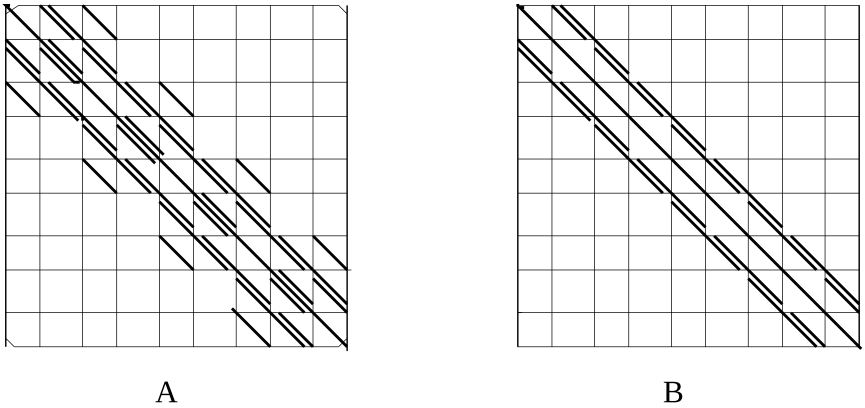


Figure 3. Sparsity pattern of the matrices A and B, $\Omega = (0,1)^2$.

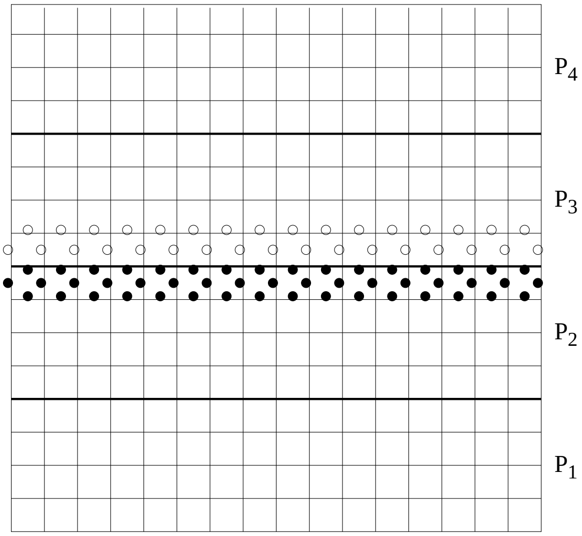


Figure 4. Stripwise data distribution between the processors in parallel implementation

$$T_{com}(A\mathbf{v}) = 4t_s + 2(3n+1)t_c.$$

Note that the above communications are completely local. The inner product needs one broadcasting and one gathering global communication but they do not contribute to the leading terms of the total parallel time and will not be considered in our analysis. This setting leads to the following expression for the parallel time per one PCG iteration

$$T_p = T_p^{it} \approx 34\frac{2n(n+1)}{p}t_a + 8nt_s + 14nt_c. \tag{7.1}$$

What we conclude from (7.1) is that the parallel algorithm is asymptotically optimal. At the same time we should emphasize that the real speed-up is strongly dependent on the relations between t_s and t_c which means that in the general case good parallel efficiency could be achieved, if and only if, the size of the problem is large enough. This is readily seen from the results reported in the next section.

8. **Numerical tests II**

The parallel implementation of our C code is developed using the MPI (Message Passing Interface) standard. A set of numerical tests have been performed on a Beowulf type cluster. It consists of four dual processor Power Macintosh computers connected with a Bay Stack 350 Switch. Each node has 512 MB RAM and two processors Power PC G4/450MHz. The same model problem is used to illustrate the properties of the proposed parallel PCG algorithm and the related code. The results obtained are given in Table 2 in terms of the speed-up S_p and the efficiency E_p where

$$S_p = \frac{T_1}{T_p}, \qquad E_p = \frac{S_p}{p},$$

and where T_p stands for the parallel time to perform the code on p processors. A well expressed asymptotic scalability of the algorithm is demonstrated by the test data presented. The parallel efficiency $E_8 \approx 0.65$ for $n = 1024$ can be evaluated as a good achievement for the problem considered.

Table 2. Parallel performance of $PCG/MIC(0)$: $S_p = \frac{T_1}{T_p}$, $E_p = \frac{S_p}{p}$

MP					MV				
$\frac{n}{n_{it}}$	p	*cpu*	S_p	E_p	$\frac{n}{n_{it}}$	p	*cpu*	S_p	E_p
$\frac{128}{49}$	1	6.02			$\frac{128}{56}$	1	6.82		
	2	3.32	1.81	0.91		2	3.70	1.84	0.92
	4	3.91	1.54	0.39		4	4.29	1.59	0.40
	8	3.67	1.64	0.21		8	4.18	1.63	0.20
$\frac{256}{71}$	1	35.44			$\frac{256}{81}$	1	40.40		
	2	19.52	1.82	0.91		2	22.21	1.82	0.91
	4	15.83	2.24	0.56		4	18.11	2.23	0.56
	8	11.96	2.96	0.37		8	13.60	2.97	0.37
$\frac{512}{104}$	1	208.95			$\frac{512}{119}$	1	238.94		
	2	114.47	1.83	0.92		2	130.97	1.82	0.91
	4	75.85	2.75	0.69		4	86.89	2.75	0.69
	8	49.48	4.22	0.53		8	56.56	4.22	0.53
$\frac{1024}{148}$	1	1198.45			$\frac{1024}{167}$	1	1335.60		
	2	654.51	1.83	0.92		2	762.31	1.75	0.88
	4	382.87	3.13	0.78		4	432.61	3.09	0.77
	8	227.52	5.27	0.66		8	255.75	5.22	0.65

Remark 2. *The cluster we have performed the reported numerical tests on has in fact a heterogeneous architecture, which is not the case for the best performance of a standard MPI code. For such a case, a combination of Open MP (within each of the shared memory two-processor node) and MPI could be recommended to improve total parallel efficiency.*

9. Concluding remarks

In this paper we have proposed a new MIC(0) preconditioner for the rotated linear non-conforming finite element systems for second order elliptic equations. We have proved that the introduced locally modified approximation of the original stiffness matrix has a relative condition number that is bounded uniformly with respect to both the problem size and the possible jumps of the coefficients. The algorithm has been analyzed in the cases of coefficient and mesh isotropy. Further, the derived estimates for the parallel time show that a good parallel scalability can be achieved for large scale problems. The presented numerical results show that the proposed scalable parallel preconditioner preserves and even improves the robustness and the computational efficiency of the standard MIC(0) factorization algorithm.

Our further plans include generalizations to 3-D case including modifications allowing for efficient treatment of coefficient and mesh anisotropy.

Acknowledgement. This work has been supported in part by MES Grant MM 801/89 and by the Center of Excellence BIS-21 Grant ICA1-2000-70016.

References

1. Arbogast, T. and Chen, Z.: *On the implementation of mixed methods as nonconforming methods for second order elliptic problems*, Math. Comp., **64**, (1995), 943–972.
2. Arnold, D.N. and Brezzi, F.: *Mixed and nonconforming finite element methods: implementation, postprocessing and error estimates*, RAIRO, Model. Math. Anal. Numer., **19**, (1985), 7–32.
3. Blaheta, R.: *Displacement decomposition - incomplete factorization preconditioning techniques for linear elasticity problems*, Numer. Lin. Alg. Appl., **1**, (1994), 107–126.
4. Bencheva, G. and Margenov, S.: *On a preconditioning strategy for rotated linear FEM elliptic systems*, Proceedings, PRISM'01, University of Nijmegen, 2001, 87-90.
5. Brezzi F. and Fortin M.: *Mixed and Hybrid Finite Element Methods*, Springer-Verlag, New York, Berlin, Heidelberg, 1991.
6. Chen Z.: *Analysis of mixed methods using conforming and nonconforming finite element methods*, RAIRO, Math. Model. Numer. Anal., **27**, (1993), 9–34.
7. Chow, E.: *Parallel implementation and practical use of sparse approximate inverses with a priori sparsity patterns*, Int'l J. High Perf. Comput. Appl., **15**, (2001), 56–74. (http://www.llnl.gov/CASC/linear_solvers/pubs.html)
8. Chow, E.: *A priori sparsity patterns for parallel sparse approximate inverse preconditioners*, SIAM Journal on Scientific Computing, **21**(5), (2000), 1804–1822. (http://www.llnl.gov/CASC/linear_solvers/pubs.html)

9. Gustafsson, I.: *Modified incomplete Cholesky (MIC) factorization*, In Preconditioning Methods; Theory and Applications, D.J. Evans, ed., Gordon and Breach, (1984), 265–293.

10. Gustafsson, I. and Lindskog, G.: *On parallel solution of linear elasticity problems: Part I Theory*, Numer. Lin. Alg. Appl., **5**, (1998), 123–139.

11. Lazarov, R. and Margenov, S.: *On a two-level parallel MIC(0) preconditioning of Crouzeix-Raviart non-conforming FEM systems*, in: I. Dimov, I. Lirkov, S. Margenov and Z. Zlatev, eds., Numerical Methods and Applications, Springer LNCS, **2542**, (2003) 191–200.

12. Rannacher, R. and Turek, S.: *Simple nonconforming quadrilateral Stokes Element*, Numerical Methods for Partial Differential Equations, **8**(2), (1992), 97-112.

13. Saad, Y. and Schultz, V.: *Data Communication in Parallel Architectures*, Parallel Comput., **11**, (1989), 131–150.

Journal of Computational and Applied Mechanics, Vol. 4., No. 2., (2003), pp. 119–127

ON SOME RELATIONSHIPS OF SPHERICAL KINEMATICS

István Ecsedi
Department of Mechanics, University of Miskolc
3515 Miskolc-Egyetemváros, Hungary
mechecs@uni-miskolc.hu

Dedicated to the memory of Professor István Sályi
(October 15, 1924 - October 17, 2001)

[Received: June 19, 2002]

Abstract. In this paper, some relations are developed for the spherical motion of a rigid body. Results are formulated in four theorems which describe a few geometrical properties of spherical motion using the geometrical data of the fixed and moving axode cones. An example illustrates the application of the formulae derived.

Mathematical Subject Classification: 70B10
Keywords: rigid body, spherical motion, curvature, axode, fixed and moving cones

1. **Introduction**

The term spherical motion is used when the rigid body moves around a fixed point. The spherical motion is equivalent to the moving axode cone C_m rolling without slipping over the fixed axode cone C_f. The instanteneous axis of rotation is the line of contact between these cones. The common apexes of the axode cones C_m and C_f is the fix point O [2, 3, 6].

The intersection of the axode cones C_m and C_f with the sphere whose center is point O and the radius is R are the moving polode c_m and the fixed polode c_f, respectively [2, 3, 6]. The common point of the curves c_f and c_m is denoted by P, and the common tangent unit vector of curves c_m and c_f at point P is indicated by $\mathbf{t}$. The angular velocity vector $\boldsymbol{\omega} = \omega\mathbf{e}$ describes the instantaneous motion of the rigid body considered. Here, $\mathbf{e} = \overrightarrow{OP}/R$ – see Figure 1.

The contact point P of the curves c_m and c_f moves along the curve c_f in the frame of the fixed polod cone C_f. This motion has the velocity

$$\mathbf{u}_f = u_p\left(s_f\right)\mathbf{t}\left(s_f\right) \qquad s_f = s_f\left(\tau\right), \tag{1.1}$$

and acceleration

$$\mathbf{w}_f = \dot{u}_p\mathbf{t} + \Gamma_f u_p^2\mathbf{n}_f. \tag{1.2}$$

The contact point P of the curves c_m and c_f moves along the curve c_m in the frame of the rigid body considered. The moving polode cone C_m is attached to the moving rigid body. The motion of point P on the curve c_m is characterized by its velocity and acceleration, which are as follows

$$\mathbf{u}_m = u_p\left(s_m\right)\mathbf{t}\left(s_m\right) \qquad s_m = s_m\left(\tau\right), \tag{1.3}$$

$$\mathbf{w}_m = \dot{u}_p\mathbf{t} + \Gamma_m u_p^2\mathbf{n}_m\,. \tag{1.4}$$

Here,

- τ denotes the time;
- s_m and s_f are arc coordinates defined on the moving and fixed polode c_m and c_f, respectively;
- Γ_m and Γ_f are the curvatures at point P of curves c_m and c_f, respectively;
- $\mathbf{n}_m$ and $\mathbf{n}_f$ are the principal normal vectors at point P to the curves c_m and c_f, respectively;
- over dot denotes derivation with respect to time.

The consequence of pure rolling is that

$$\dot{s}_f = \dot{s}_m = u_p, \qquad \ddot{s}_f = \ddot{s}_m = \dot{u}_p\,. \tag{1.5}$$

Starting from the equation

$$\boldsymbol{\omega} = \omega\mathbf{e} = \frac{\omega}{R}\overrightarrow{OP}$$

and using the definition of angular acceleration [1, 5]

$$\boldsymbol{\varepsilon} = \left(\frac{d\boldsymbol{\omega}}{d\tau}\right)_f,$$

where the symbol $\left(\frac{d}{d\tau}\right)_f$ denotes the time derivative computed in the fixed frame we get for the body angular acceleration the formula

$$\boldsymbol{\varepsilon} = \dot{\omega}\mathbf{e} + \frac{\omega u_p}{R}\mathbf{t}\,. \tag{1.6}$$

The fixed frame is attached to the base (fixed polode cone) and the moving frame is attached to the moving rigid body (moving polode cone). Point A of the moving rigid body instantaneously coincides with point P. Using the fundamental relationships of relative motion in connection with point P we can write

$$\mathbf{u}_f = \mathbf{v}_A + \mathbf{u}_m, \tag{1.7}$$

$$\mathbf{w}_f = \mathbf{a}_A + 2\boldsymbol{\omega}\times\mathbf{u}_m + \mathbf{w}_m. \tag{1.8}$$

Here, $\mathbf{v}_A$ is the velocity of point A and $\mathbf{a}_A$ is the acceleration of point A. $\mathbf{u}_f, \mathbf{w}_f, \mathbf{a}_A, \boldsymbol{\omega}$ are taken in the fixed frame and $\mathbf{u}_m, \mathbf{w}_m$ are regarded in the moving frame in equation 1.8), the vectorial product of two vectors is denoted by cross.

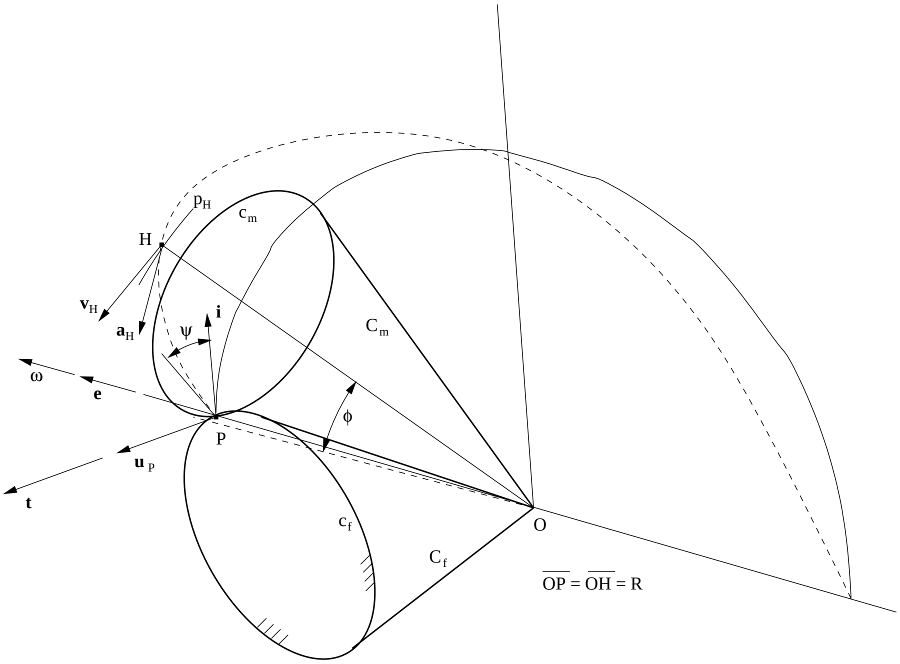


Figure 1. Moving and fixed axode cones

From pure rolling it follows that $\mathbf{v}_A = \mathbf{0}$, $\mathbf{u}_m = \mathbf{u}_f$. The latter statement was mentioned in equation (1.5). The combination of equations (1.2), (1.4), (1.5) with equation (1.8) gives the result

$$\mathbf{a}_A = 2\mathbf{u}_p \times \boldsymbol{\omega} + u_p^2(\Gamma_f \mathbf{n}_f - \Gamma_m \mathbf{n}_m), \tag{1.9}$$

where $\mathbf{u}_p = \mathbf{u}_m = \mathbf{u}_f$ is the velocity of the instantaneous contact point P.

The common tangential unit vector $\mathbf{t}$ of curves c_m and c_f is attached to point P. We know that its time derivative in a fixed frame can be expressed as [1, 3]

$$\left(\frac{d\mathbf{t}}{d\tau}\right)_f = \left(\frac{d\mathbf{t}}{d\tau}\right)_m + \boldsymbol{\omega} \times \mathbf{t}, \tag{1.10}$$

where the symbol $\left(\frac{d}{d\tau}\right)_m$ denotes the time derivative computed in the moving frame. A simple calculation shows that

$$\left(\frac{d\mathbf{t}}{d\tau}\right)_f = \frac{d\mathbf{t}}{ds_f}\dot{s}_f = u_p \Gamma_f \mathbf{n}_f, \tag{1.11}$$

$$\left(\frac{d\mathbf{t}}{d\tau}\right)_m = \frac{d\mathbf{t}}{ds_m}\dot{s}_m = u_p \Gamma_m \mathbf{n}_m. \tag{1.12}$$

Inserting these results into equation (1.10) we obtain

$$\mathbf{t} \times \boldsymbol{\omega} = u_p(\Gamma_m \mathbf{n}_m - \Gamma_f \mathbf{n}_f). \tag{1.13}$$

Using the trivial identity

$$u_p\mathbf{t}\times\boldsymbol{\omega}=\mathbf{u}_p\times\boldsymbol{\omega}=u_p^2(\Gamma_m\mathbf{n}_m-\Gamma_f\mathbf{n}_f)$$

and equations (1.9) and (1.13) we get

$$\mathbf{a}_A=\mathbf{u}_p\times\boldsymbol{\omega}\,. \tag{1.14}$$

Equation (1.14) is in harmony with the result of example 4.8 of the the book [5] by Ginsberg.

2. **Some useful relations**

Theorem 1. The angular velocity of the moving rigid body is determined by the geometry of the polode curves and the speed of the contact point according to the equation

$$\boldsymbol{\omega}=u_p\left(\Gamma_f\mathbf{b}_f-\Gamma_m\mathbf{b}_m\right), \tag{2.1}$$

where $\mathbf{b}_f$ and $\mathbf{b}_m$ are the binormal unit vectors at point P to the curves c_f and c_m.

Proof. The proof of equation (2.1) follows from the equations

$$\boldsymbol{\omega}\cdot\mathbf{t}=\frac{\omega}{R}\mathbf{R}\cdot\frac{d\mathbf{R}}{ds_f}=\frac{\omega}{R}\frac{d}{ds_f}\left(\frac{R^2}{2}\right)=0,$$

$$\mathbf{t}\times(\boldsymbol{\omega}\times\mathbf{t})=\omega\mathbf{t}^2-\mathbf{t}(\boldsymbol{\omega}\cdot\mathbf{t})=\omega,$$

and the definition of the binormal vector

$$\mathbf{b}_f=\mathbf{t}\times\mathbf{n}_f,\qquad \mathbf{b}_m=\mathbf{t}\times\mathbf{n}_m$$

and the validity of equation (1.13). Here, the dot between two vectors denotes their scalar product.

Theorem 2. The angular acceleration of the moving rigid body is determined by the geometry of the polode curves, the speed of the contact point and the rate of change of the speed of the contact point according to the equation

$$\boldsymbol{\varepsilon}=\dot{u}_p\left(\Gamma_f\mathbf{b}_f-\Gamma_m\mathbf{b}_m\right)+u_p^2\left(\frac{d\Gamma_f}{ds_f}\mathbf{b}_f-\frac{d\Gamma_m}{ds_m}\mathbf{b}_m-\Gamma_fT_f\mathbf{n}_f+\Gamma_mT_m\mathbf{n}_m\right)+\frac{\omega u_p}{R}\mathbf{t}. \tag{2.2}$$

Here, T_f and T_m are the torsions of curves c_f and c_m at point P, respectively.

Proof. Starting from the expression of $\boldsymbol{\omega}$ given by equation (2.1) we obtain

$$\boldsymbol{\varepsilon}=\left(\frac{d\boldsymbol{\omega}}{d\tau}\right)_f=\dot{u}_p\left(\Gamma_f\mathbf{b}_f-\Gamma_m\mathbf{b}_m\right)+u_p^2\left(\frac{d\Gamma_f}{ds_f}\mathbf{b}_f+\Gamma_f\frac{d\mathbf{b}_f}{ds_f}-\frac{d\Gamma_m}{ds_m}\mathbf{b}_m\right)-u_p\Gamma_m\left(\frac{d\mathbf{b}_m}{d\tau}\right)_f. \tag{2.3}$$

Making use of the fundamental equation of relative motion [1, 5]

$$\left(\frac{d\mathbf{b}_m}{d\tau}\right)_f=\left(\frac{d\mathbf{b}_m}{d\tau}\right)_m+\boldsymbol{\omega}\times\mathbf{b}_m, \tag{2.4}$$

and the equations

$$\left(\frac{d\mathbf{b}_m}{d\tau}\right)_m=\left(\frac{d\mathbf{b}_m}{ds_m}\right)\dot{s}_m=\frac{d\mathbf{b}_m}{ds_m}u_p, \tag{2.5}$$

$$\frac{d\mathbf{b}_f}{ds_f} = -T_f \mathbf{n}_f, \qquad \frac{d\mathbf{b}_m}{ds} = -T_m \mathbf{n}_m, \tag{2.6}$$

$$u_p \Gamma_m \boldsymbol{\omega} \times \mathbf{b}_m = -\frac{\omega}{R} \mathbf{u}_p \tag{2.7}$$

we get the proof of formula (2.3). The validity of equation (2.7) can be proved as

$$0 = \frac{d}{ds_m} (\mathbf{R} \cdot \mathbf{t}) = \frac{d\mathbf{R}}{ds_m} \cdot \mathbf{t} + \mathbf{R} \cdot \frac{d\mathbf{t}}{ds_m} = \tag{2.8}$$

$$= 1 + R\Gamma_m \mathbf{e} \cdot \mathbf{n}_m \qquad \text{that is} \qquad \mathbf{e} \cdot \mathbf{n}_m = -\frac{1}{R\Gamma_m}$$

and, on the other hand, we have

$$u_p \Gamma_m \boldsymbol{\omega} \times \mathbf{b}_m = u_p \Gamma_m \omega \mathbf{e} \times (\mathbf{t} \times \mathbf{n}_m) = u_p \Gamma_m \omega \left[\mathbf{t} \left(\mathbf{e} \cdot \mathbf{n}_m\right) - \mathbf{n}_m \left(\mathbf{e} \cdot \mathbf{t}\right)\right] = \tag{2.9}$$

$$= \mathbf{u}_p \Gamma_m \omega \left(\mathbf{e} \cdot \mathbf{n}_m\right).$$

The combination of equation (2.8) with equation (2.9) yields equation (2.7).

Consider point H of the moving rigid body. Let the position vector of point H be

$$\overrightarrow{OH} = R \cos \Phi \mathbf{e} + R \sin \Phi \cos \psi \mathbf{i} + R \sin \Phi \sin \psi \mathbf{t}, \qquad \mathbf{i} = \mathbf{t} \times \mathbf{e}. \tag{2.10}$$

The path of point H in the fixed reference frame lies on a sphere whose radius is R and whose center is point O – Figure 1. In (2.10), Φ is the angle between the lines OP and OH. Let $[\mathbf{t}; OP]$ be the plane whose normal vector is $\mathbf{t}$ and which contains the line OP. The plane $[OPH]$ is determined by points O, P and H. The angle formed by the planes $[\mathbf{t}; OP]$ and $[OPH]$ was denoted by ψ in equation (2.10).

The following theorems describe the relations of the geometrical properties of the path of point H with the moving and fixed polodes c_m and c_f.

Theorem 3. Let c_H be the path curve of point H. Assuming that the instantaneous axis is the line OP, then the following equations hold

$$\mathbf{t}_H = -\mathbf{i} \sin \psi + \mathbf{t} \cos \psi, \tag{2.11}$$

$$\mathbf{N}_H = -\mathbf{e} + \mathbf{i} \left(\cot \Phi \cos \psi - \frac{R}{d_p}\right) + \mathbf{t} \left(\cot \Phi \sin \psi - \frac{R}{d_p} \tan \psi\right), \tag{2.12}$$

$$\mathbf{B}_H = -\mathbf{e} \left(\cot \Phi - \frac{R}{d_p} \frac{1}{\cos \psi}\right) - \mathbf{i} \cos \psi - \mathbf{t} \sin \psi, \tag{2.13}$$

$$\left(\frac{\cos \psi}{\sin^2 \Phi} \frac{d_p}{R} - \cot \Phi\right)^2 + 1 = \left(R\Gamma_H\right)^2, \tag{2.14}$$

where $\mathbf{t}_H$, $\mathbf{N}_H$ and $\mathbf{B}_H$ are the tangential, principal and binormal vectors to the curve c_H at point H, respectively. $\mathbf{t}_H$ is a unit vector, $\mathbf{N}_H$ and $\mathbf{B}_H$ are not unit vectors. Furthermore, d_p is defined as

$$d_p = \frac{1}{\Gamma_f \cos \alpha_f - \Gamma_m \cos \alpha_m}, \tag{2.15}$$

$$\cos \alpha_f = \mathbf{b}_f \cdot \mathbf{e}, \qquad \cos \alpha_m = \mathbf{b}_m \cdot \mathbf{e} \tag{2.16}$$

and Γ_H denotes the curvature of c_H at point H.

Proof. The proof of equations (2.11), (2.12), (2.13), (2.14) is based on the definition of $\mathbf{B}_H$, which is

$$\mathbf{B}_H = \mathbf{t}_H \times \mathbf{N}_H,$$

and the following kinematical equations of a particle and a rigid body [1, 5]

$$\mathbf{t}_H = \frac{\mathbf{V}_H}{|\mathbf{V}_H|} = \frac{\boldsymbol{\omega} \times \overrightarrow{OH}}{\left|\boldsymbol{\omega} \times \overrightarrow{OH}\right|},$$

$$\mathbf{a}_H = \boldsymbol{\omega} \times \mathbf{V}_H + \epsilon \times \overrightarrow{OH},$$

$$\mathbf{a}_H = \frac{\mathbf{a}_H \cdot \mathbf{V}_H}{\mathbf{V}_H^2}\mathbf{V}_H + \Gamma_H \mathbf{V}_H^2 \mathbf{n}_H,$$

$$\mathbf{n}_H = \frac{\mathbf{N}_H}{|\mathbf{N}_H|},$$

and the equation

$$d_p = \frac{u_p}{\omega}. \tag{2.17}$$

The validity of equation (2.17) follows from equation (2.1).

Theorem 4. Let T_H be the torsion of curve c_H at point H, and let s be an arc coordinate defined on curve c_H. We have

$$|T_H| = \frac{\left|\left\{\frac{d}{ds}\ell n \frac{\Gamma}{\Gamma_H}\right\}\right|_{s=s_H}}{\left|\frac{\cos\Psi}{\sin^2\Phi}\frac{d_p}{R} - \cot\Phi\right|} \qquad (\Phi \neq 0, \pi)\,. \tag{2.18}$$

Here, $\Gamma = \Gamma(s)$ is the curvature at an arbitrary point of c_H and the position of point H on c_H is given by s_H.

Proof. Using the concept of the osculating sphere in connection with the spherical curve c_H we can write [4, 7, 8]:

$$\left(\frac{1}{\Gamma_H}\right)^2 + \left(\frac{d}{ds}\left(\frac{1}{\Gamma}\right)_{s=s_H} \frac{1}{T_H}\right)^2 = R^2. \tag{2.19}$$

The combination of formula (2.14) with equation (2.19) leads to formula (2.18).

3. Remark on the computation of d_p

This section concentrates on the computation of $\cos\alpha_f$ and $\cos\alpha_m$, which appear in formula (2.15).

Let us consider an arbitrary curve c on the sphere whose radius and center are R and O, respectively. Let $\overrightarrow{OQ} = \varrho = \varrho(s)$ be the equation of curve c, where Q is an arbitrary point of c and s is an arc coordinate defined on c. A repeated differentiation of the equation

$$\varrho^2 = R^2 = \text{constant} \tag{3.1}$$

with respect to s gives

$$\Gamma \varrho \cdot \mathbf{n} + 1 = 0, \tag{3.2}$$

where Γ is the curvature of c at point Q and $\mathbf{n}$ is the principal normal vector of c at point P. Equation (3.2) can be obtained from Meusnier's theorem as well [4, 7, 8].

The application of equation (3.2) to curve c_f at point P yields

$$\mathbf{n}_f \cdot \mathbf{e} = -\frac{1}{R\Gamma_f}. \tag{3.3}$$

The angle formed by the vectors $\mathbf{e}$ and $\mathbf{n}_f$ is denoted by β_f. It is obvious that the angle between the vectors $\mathbf{e}$ and $\mathbf{b}_f$ is

$$\alpha_f = \beta_f \pm \frac{\pi}{2}. \tag{3.4}$$

From equations (3.3) and (3.4) we get

$$\cos\alpha_f = \operatorname{sgn}(\mathbf{b}_f \cdot \mathbf{i})\sqrt{1 - \frac{1}{(R\Gamma_f)^2}}. \tag{3.5}$$

A similar formula can be derived to obtain the value of $\cos\alpha_m$:

$$\cos\alpha_m = \operatorname{sgn}(\mathbf{b}_m \cdot \mathbf{i})\sqrt{1 - \frac{1}{(R\Gamma_m)^2}}. \tag{3.6}$$

Here, we remark that the analogue pair of equation (3.6) for the curve c_m was derived in Section 2 (equation (2.8)).

4. **Example**

Figure 2 illustrates a rigid body's circular cone OPH. The point O is fixed and the cone rolls without slipping on the horizontal plane $[\mathbf{i}; OP]$ whose normal vector is $\mathbf{i}$. The fixed axode cone is the "plane $[\mathbf{i}; OP]$" (degenerate cone) and the moving axode cone is the circular cone OPH. Q is the center point of the base circle of the cone OPH. This base circle can be considered as a moving polode curve c_m. The fixed polode curve c_f is a circle in the plane $[\mathbf{i}; OP]$ whose radius is $R = \overline{OP}$ and its center is point O. Our aim is to determine the local geometrical property of the path of point H at the instant shown in Figure 2.

Using the data given in Figure 2 we can write

$$\begin{aligned}
\mathbf{n}_f &= -\mathbf{e}, \quad \mathbf{b}_f = -\mathbf{i}, \quad \Gamma_f = \frac{1}{R},\\
\mathbf{n}_m &= \mathbf{i}\cos\vartheta - \mathbf{e}\sin\vartheta,\\
\mathbf{b}_m &= -\mathbf{i}\sin\vartheta - \mathbf{e}\cos\vartheta,\\
\Gamma_m &= \frac{1}{R\sin\vartheta}, \ \Phi = 2\vartheta, \ \psi = 0.
\end{aligned}$$

$$\mathbf{b}_f \cdot \mathbf{e} = 0, \ \mathbf{b}_m \cdot \mathbf{e} = -\cos\vartheta,$$

$$d_p = R\tan\vartheta.$$

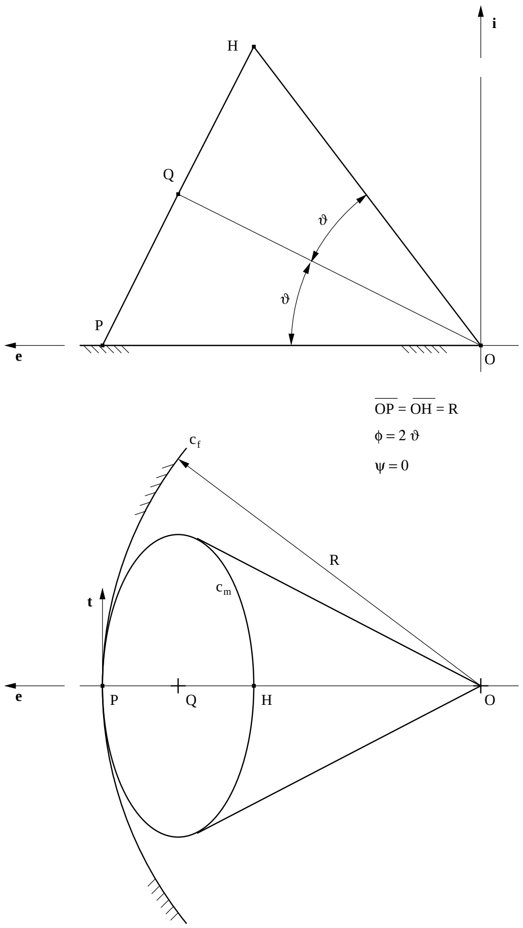


Figure 2. Rolling circular cone

The application of *Theorem 3* to this problem gives

$$
\begin{aligned}
\mathbf{t}_H &= \mathbf{t},\ \ \mathbf{n}_H = -\frac{1}{\sqrt{(\cot 2\vartheta)^2+1}}\left(\mathbf{e}+\frac{\mathbf{i}}{\sin 2\vartheta}\right),\\
\mathbf{b}_H &= \frac{1}{\sqrt{(\cot 2\vartheta)^2+1}}\left(\frac{\mathbf{e}}{\sin 2\vartheta}-\mathbf{i}\right),\\
\Gamma_H &= \frac{1}{R}\left[\left(\frac{\lambda^4+4\lambda^2-1}{4\lambda}\right)^2+1\right],\ \ \lambda=\tan\vartheta.
\end{aligned}
$$

The path of point H is a spherical cycloid and the motion analyzed can be considered a spherical cycloidal motion [2].

5. Conclusion

Some relations are derived for the spherical motion of a rigid body. The geometrical properties of spherical motion are expressed in the geometrical data of the fixed and moving axode cones. The approach applied does not use the tools of instantaneous spherical kinematics [2, 3]. The method presented is based on a vectorial approach that one can meet in [1, 5].

Curvature type relations such as (2.1), (2.14) can be considered as a form of Euler-Savary equation for spherical motion. Different forms of the Euler-Savary equation using the terminology and the concept of instantaneous invariants introduced by Bottema are given in [2, 3] for spherical and plane motions.

One example shows how we can use the derived formulas to determine the tangent, principal normal and binormal vectors together with the curvature at a point of path curve in the case of spherical motion.

References

1. Millard, F. B. Jr.: *Principles of Engineering Mechanics*, Vol. 1, Plenum Press, New York, 1986.

2. Bottema, O. and ROTH, B.: *Theoretical Kinematics*, Nort-Holland, New York, 1979.

3. Hunt, K. H.: *Kinematic Geometry of Mechanisms*, Oxford University Press, Oxford, 1978.

4. Gehér, L.-Szőkefalvi-Nagy, G. and Nagy, P.: *Differential Geometry*, Műszaki Kiadó, Budapest, 1979. (in Hungarian)

5. Ginsberg, J. H.: *Advanced Engineering Dynamics*, Cambridge University Press, Cambridge, 1995.

6. Müller, H. R.: *Sphärische kinematics*, Veb Deucher Verlag der Wissenschaften, Berlin, 1962.

7. Struik, D. J.: *Differential Geometry*, Addison-Wesley, Massachusets, 1950.

8. Willmore, T. J.: *An Introduction to Differential Geometry*, Clarendon Press, Oxford, 1959.

Journal of Computational and Applied Mechanics, Vol. 4., No. 2., (2003), pp. 129–146

APPLICATION OF CFD NUMERICAL SIMULATION FOR INTAKE PORT SHAPE DESIGN OF A DIESEL ENGINE

András Horváth
Department of Physics, Széchenyi István University
Egyetem tér 1, 9026 GYŐR, Hungary
horvatha@sze.hu

Zoltán Horváth
Department of Mathematics, Széchenyi István University
Egyetem tér 1, 9026 GYŐR, Hungary
horvathz@sze.hu

[Received: June 19, 2002]

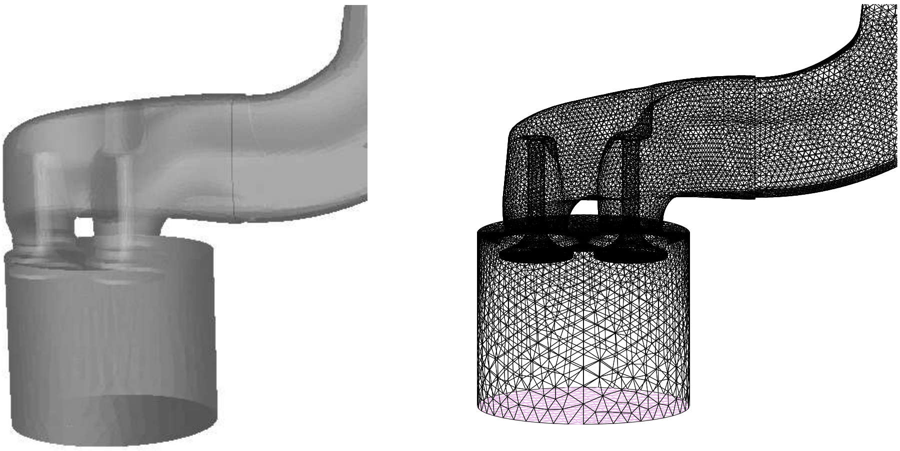
Abstract. In this paper we investigate the air flow characteristics of the intake port of a Diesel engine by numerical simulation, which is based on a self developed code. Several possibilities of the mathematical model of the engineering problem and their numerical solutions are implemented, discussed and some of them developed further and compared with actual physical measurements. As a conclusion we find that a first order finite volume method, the Vijayasundaram flux vector split method with local time-stepping is suitable for computing the flow characteristics, namely the flow and swirl coefficients, accurately. By accuracy we mean that the computed and the measured quantities differ in 0–0.5% and 0.5–10%, respectively, validating our numerical model. Applying subsequently this code and a domain deformation we are able to increase in 1% the flow coefficient under the constraint of a constant swirl number, which is significant since only small modifications were allowed.

Mathematical Subject Classification: 76M12, 76N25, 65M99
Keywords: compressible fluid flow, numerical solution of Euler equations, Diesel engine flow problems, geometric parameter optimization

1. Introduction

1.1. **The engineering problem.** The value of a Diesel engine is described by a great many variables, e.g. power, efficiency, emission of pollution. They depend on numerous parameters of the engine, e.g. geometrical structure (intake port, cylinder, combustion chamber), injection parameters etc., in a very complex way.

Hence the engineering process of developing a Diesel engine consists of several consecutive steps. At one of the first stages a suitable intake geometry is determined and then, proceeding further, the geometry of the combustion chamber and the parameters of the injection etc. are adjusted so that the resulting engine satisfies the prescribed power, air pollution, etc. values.



An opaque model

Surface grid

Figure 1. Overview of an intake geometry

In this paper we focus on the design of the intake geometry (for a typical example see Figure 1), which is of crucial importance for the engine efficiency. Namely, this determines largely the amount of fresh air intaken during an engine cycle and the rotation of the fluid in the cylinder and combustion chamber, which have a close influence on the efficiency of the combustion. The features of the intake geometry are characterized by two non-dimensional numbers computed from measurements of a quasi-stationary flow, the flux-coefficient and the swirl-coefficient, denoted by C_f and C_s respectively; for a more detailed definition see Section 3.1.

Now the engineering problem we are dealing with in this paper is formulated as follows. We have to modify a certain given intake geometry by small deformations so that the resulting intake geometry will be "optimal": the larger C_f the better while C_s belongs to a certain given interval (determined a priori from the existing model) and the volume of intake port will be smaller, if possible. We would like to emphasize that due to technical restrictions only small deformations are allowed.

An "optimal" intake geometry is sought traditionally by a sequence of consecutive measurements of these numbers and test-piece modifications; then the best model is chosen "optimal". It is clear that this process is rather expensive and time consuming.

Our task is to substitute a reliable numerical simulation for this process.

1.2. **The numerical simulation.** As part of an industrial project, our task was to simulate the step of optimal intake port design by numerical simulation. For this we had to compute C_f and C_s from a geometry given by a CAD-model and mechanical parameters, by simulation. Moreover, using these coefficients and other flow parameters such as graphs of pressure distribution we had to suggest an "optimal" intake geometry. For similar problems investigated in the literature consult e.g. [5], [7] and references therein.

For solving our complex problem we had to face – among others – the following subproblems:

- selecting an appropriate mathematical model for the gas flow;
- finding a fast numerical solution of the mathematical model;
- implementing the resulting algorithm efficiently;
- validating the numerical model with experiments;
- finding small deformations to improve the geometry

such that the resulting simulation be robust and give sufficiently accurate result compared with actual measurements.

In the literature there exist a great number of suggestions for solving the mathematical subproblems, see e.g. [2], [4], [6], [8], [11]. However, it seems there exist only few papers dealing with the engineering subproblems such as verification with measurements as well; for an example see [5] and [7].

For solving the complex problem, we found a simple and yet adequate approach, which will be presented and discussed in the paper. Its main features are the following:

- compressible Euler equations for gas flow;
- Vijayasundaram's flux vector split finite volume method on a fixed (i.e. non-adaptive) unstructured tetrahedral mesh; here we applied for time stepping a local time-stepping strategy enabling approximation even of the non-steady flow;
- ANSI C programming language for the code, which was optimized by the computer algebra program Maple;
- visualization and mesh deformation modules.

We shall see in this paper below that our computational results were verified by many actual measurements with a relative error 0–0.5% for C_f and 0.5%–10% for C_s, see Section 3. Then from several experiments we could suggest an actual new model intake port; by test-piece measurements our prediction was proven to be of 1% larger C_f, same C_s and remarkably less intake port volume than the corresponding values of an initial, a priori given intake port.

2. Components of the numerical simulation

2.1. The mathematical model. The mathematical model consists of the well-known formalization of conservation laws of mass, momenta and energy by the Euler equations and a thermodynamical formula, and the equation of states (EOS), which is specific to the material of the gas. We emphasize that we did not need to use any turbulence models because numerical simulations based on our mathematical model happened to be satisfactory, see Section 3.

To formulate the model we need some notations. Let us denote the density, the velocity, the total energy density (i.e. total energy per unit volume) and pressure of the flowing air by ρ, $\mathbf{v} = (v_1, v_2, v_3)^T$, e, p respectively, the time by $t \in [0, t_{max}]$, the points in $\mathbb{R}^3$ by $(x_1, x_2, x_3)^T$, the flow domain by $\Omega \subset \mathbb{R}^3$.

Let us introduce further the notations

$$\begin{aligned} u &= (\rho, \rho\mathbf{v}^T, e)^T : \Omega \times [0, t_{max}] \to \mathrm{I\!R}^5, \\ f &= (f_1, f_2, f_3)^T, \qquad f_i : \mathrm{I\!R}^5 \to \mathrm{I\!R}^5, \\ f_i(u) &= \left(\rho v_i, (\rho v_i \mathbf{v} + p\mathbf{e}_i)^T, v_i(e+p)\right)^T, \qquad i \in \{1,2,3\}, \\ \mathrm{div} f(u) &= \sum_{i=1}^{3} \frac{\partial f_i(u)}{\partial x_i} \end{aligned}$$

Here u is called conservative variable, $\gamma = c_p/c_v$, i.e. the ratio of heat capacity at constant pressure and volume (for air $\gamma = 1.4$).

Then our mathematical model reads (see e.g. [6], [3])

$$\frac{\partial}{\partial t} u + \mathrm{div}\, f(u) = 0 \qquad \text{on } \Omega \times [0, t_{max}] \tag{2.1}$$

$$p = \rho(\gamma - \frac{1}{\gamma})(e - \frac{\mathbf{v}^2}{2}) \tag{2.2}$$

$$u(.,0) = u_0 \tag{2.3}$$

$$+ \quad \text{boundary conditions} \tag{2.4}$$

In order to apply adequate boundary conditions in (2.4) we described the circumstances of physical measurements (c.f. Section 3). Thus, denoting by $\Gamma = \partial\Omega$ the boundary of Ω, which is divided into three disjoint parts $\Gamma = \Gamma_{in} \cup \Gamma_{out} \cup \Gamma_{wall}$ with Γ_{in} the inlet (beginning plane section of the intake tube), Γ_{out} the outlet (the cylinder bottom plane section) and Γ_{wall} the wall (the rest of Γ), we arrive at the following boundary conditions for (2.4):

- $\mathbf{v}$ is parallel to Γ_{wall}
- $p = p_{in}$, $\rho = \rho_0$ are given at Γ_{in} and $\mathbf{v} \perp \Gamma_{in}$;
- $p = p_{out}$ is given at Γ_{out}.

2.2. **The numerical algorithm.** For the numerical solution method of the mathematical model which consists of (2.1)–(2.4) we chose some flux vector splitting finite volume methods to maintain conservativity of mass, momenta and energy and keeping implementation simple. For a detailed introduction and investigation of such methods consult [6], see also [2], [3], [4] and [8]. Here we show only the most important features of our method with a more detailed description of our time-stepping scheme.

Suppose that Ω is discretized by a conform tetrahedral mesh consisting of tetrahedra T_j and the time span is divided (adaptively) by $0 = t_0 < t_1 < \ldots < t_k < \ldots < t_M = t_{max}$; we shall denote the sides of T_j by S_{jl} ($l \in \{1, \ldots, 4\}$), which induces a local labeling of tetrahedra and variables assigned to tetrahedra (for example, T_{jl} is the tetrahedron of the mesh with $T_j \cap T_{jl} = S_{jl}$).

We shall approximate the flow variables constant on each tetrahedron and each time level via $u_j^k \approx u$ on T_j at $t = t^k$. In order to derive a time-stepping scheme for the u_j^k values, we integrate (2.1) over T_j; by application of Gauss' theorem on volume integrals we get

$$\frac{d}{dt}\left(\int\limits_{T_j} udV\right) + \sum_{l=1}^{4} \int\limits_{S_{jl}} f(u) \cdot \mathbf{n}_{jl} dS = 0 \tag{2.5}$$

where $\mathbf{n}_{jl}$ is the outer unit normal vector of T_j on its faces S_{jl} and we use the notation

$$f(u) \cdot \mathbf{n}_{jl} = \sum_{i=1}^{3} f_i(u)\mathbf{n}_{jl,i} \ .$$

To approximate the terms of (2.5) observe first that $f(u) \cdot \mathbf{n}_{jl} = C_{jl}(u)u$ with a 5-by-5 matrix valued function C_{jl} the flux over S_{jl}, hence $\int\limits_{S_{jl}} f(u) \cdot \mathbf{n}_{jl} dS$ can be approximated by the numerical flux $g_{jl} = g_{jl}(u_j, u_{jl})$ of form

$$g_{jl}(u, v) := |S_{jl}|(C_{jl,1}(u, v)u + C_{jl,2}(u, v)v)$$

with

$$\begin{aligned} C_1 := C^+(u), \quad C_2 := C^-(v) \quad &: \quad \text{for the Steger-Warming method,} \\ C_1 := C^+(\frac{u+v}{2}), \quad C_2 := C^-(\frac{u+v}{2}) \quad &: \quad \text{for the Vijayasundaram method.} \end{aligned}$$

We remark that the positive and negative part of the matrices are computed after diagonalization: if $C = Q\,D\,Q^{-1}$, then $C^{\pm} = Q\,D^{\pm}\,Q^{-1}$.

Applying a simple first order approximation of the first term of (2.5) and employing the material of this section, we get

$$u_j^{k+1} = u_j^k - \frac{\tau_k}{|T_j|} \sum_{l=1}^{4} g_{jl}(u_j^k, u_{jl}^k) \quad \forall j = 1, \ldots, N, \quad k \ . \tag{2.6}$$

In the formulas above, $|S_{jl}|$ is the area of S_{jl} and $|T_j|$ is the volume of T_j, further $\nu_{jl} = |S_{jl}|\mathbf{n}_{jl}$.

2.2.1. *The time-stepping.* The usual way of time-stepping with the explicit Euler method applied to the Euler equations reads

$$\begin{aligned} u_j^0 &= \frac{1}{|T_j|} \int\limits_{T_j} u^0 dV \ , \\ u_j^{n+1} &= u_j^n - \frac{\tau}{|T_j|} \sum_{l} g_{jl}(u_j^n, u_{jl}^n) \end{aligned}$$

under the CFL-condition:

$$\max_j \frac{\tau}{|T_j|} \max_l |S_{jl}| \lambda_{jl,max}(u_j^n) \leq CFL(< 1) \ .$$

Here CFL is the so-called "Courant number", which is chosen due to stability restrictions; in our performances we equate it with 0.4.

Application of an CFL criterion implies that even if there is only one side with large $\lambda_{lj}/|T_j|$, the time-step gets a very low value and the number of steps and CPU time grow significantly, although the majority of tetrahedra would allow perhaps a much higher value for τ. In our case this phenomenon arises because the ratio of maximal and minimal volume of tetrahedra in our mesh is approximately 3000.

In order to avoid this decrease of efficiency, we used the concept of "life-time of side-fluxes", τ_{jl}. The main idea is that we reuse the calculated g_{jl} value on the side S_{jl} as long as the time-step taken in the step due to the CFL condition does not exceed τ_{jl}; naturaly, after each use we subtract the time-step taken from τ_{jl}.

The skeleton of the time-stepping with this kind of life-spans is the following: $t := 0$*;*

```
$n\_steps := 0$
$For\ all\ j\ do\ u_j := u_j^0$
$For\ all\ j, l\ do$
        $q_{jl} := g_{jl}(u_j, u_{jl})$
        $\tau_{jl} := \frac{|T_j|\,CFL}{|S_{jl}|\,\lambda_{jl,max}(u_j)}$
$While\ t < t_{max}\ do$
        $\tau := \min\{\min_{j,l} \tau_{jl},\ t_{max} - t\}$
        $t := t + \tau;\ n\_steps = n\_steps + 1$
        $For\ all\ j, l\ do$
                $u_j := u_j - \frac{\tau}{|T_j|} q_{jl}$
                $\tau_{jl} := \tau_{jl} - \tau$
                $if\ \tau_{jl} \leq \tau * 0.05\ or\ mod(n\_steps,\ 10){=}0\ then$
                        $q_{jl} := g_{jl}(u_j, u_{jl})$
                        $\tau_{jl} := \frac{|T_j|\,CFL}{|S_{jl}|\,\lambda_{jl,max}(u_j)}$
```

With this adaptive time-step control the total average CPU-cost of one iteration reduced by a factor of 1.8–2.5 in different cases, but the code used 40% more RAM, because we had to store values of g_{jl} and τ_{jl}.

We remark further that using our time stepping method we get numerical solutions approximating even the *non-steady* flow, in contrast with usual local time-stepping methods, which approximate only the steady-state solution in limit and the internal u_j^k vectors do not approximate the solution at all (c.f. [3], p. 476).

2.3. **Implementation of the numerical algorithm.** We implemented the numerical algorithms described above in ANSI C. Linux operating system was used as developing and running environment. The main reason for this choice was the stability of the system, which is an important property if the typical CPU time of one calculation is 14–18 hours.

Our software reads its input from files and writes its output to files which makes it easy to run on a remote host and/or with timing. A separate self developed program was used to visualize the results. With this software, we could examine the intermediate results during run-time.

Calculation of the 5×5 matrices above (C, their diagonalization matrices) is the most CPU consuming part of the program. We imported these formulas (see [6], [11]) into the Maple computer algebra system and generated an optimized C code with this software (see [9]). Maple found a lot of common subexpressions that were unrecognizable by humans. This C code was optimized further by hand. With these optimization steps we managed to reduce the CPU cost of C^+ calculations to 50–60%.

Besides applying the symmetry properties in both Steger-Warming's and Vijayasundaram's method (for example $C_{jl}^+ = C_{im}^-$ if the lth side of T_j is the same as the mth side of T_i (see [6] and [4]), we used an another trick in Vijayasundaram's method: we have to calculate two matrix-vector products where the matrices are positive and negative parts of the same matrix. In this way we have to calculate eigenvalues and diagonalization matrices only once.

3. Experimental verification

3.1. **Characteristics of the experiments.** In order to validate our numerical simulation software described in Section 2 we considered the following experiments which were tried by Autokut Budapest. We took a particular intake geometry (intake tube, port, valves with a fixed lift and cylinder without piston), made from a certain CAD model, see Figure 1. The air is drawn from the inlet (beginning of intake tube) to the outlet (bottom of cylinder) due to a constant pressure drop, namely constant inlet and outlet pressure, p_{in} and p_{out} were set. Then, after reaching the steady state flow, the total mass flux and the torque (flux of angular momentum) of air were measured (at the outlet). The flow and swirl coefficients, C_f and C_s are calculated from these measurements in the standard way:

$$C_f = \frac{\dot{m}\,/\rho_0}{A\,v_0}, \qquad C_s = \frac{8T}{\dot{m}\,B\,v_0}$$

where $\dot{m}=$ the measured total mass flow rate [kg/s], ρ_0 = air density at the inlet [kg/m^3], $A = 2\cdot$valve inner seat area [m^2], $v_0 = \sqrt{2(p_{in} - p_{out})/\rho_0}$ = characteristic velocity based on pressure drop [m/s], T = measured torque [Nm], B = cylinder bore [m].

In the actual physical experiments $\dot{m}$ and T are measured by standard devices.

In the mathematical model we can calculate them from the density and velocity distributions:

$$\dot{m}= \int_{S_{out}} \rho \mathbf{v} d\mathbf{S}, \qquad T = \int_{S_{out}} \rho(\mathbf{v} \times (\mathbf{r} - \mathbf{r}_0))d\mathbf{S}$$

where S_{out} is the outlet surface (the bottom of the cylinder), $\mathbf{r}_0$ is the position vector of a point in the symmetry axis of the cylinder. In the numerical model these integrals are approximated simply by sums on outlet faces of cells.

From an engineering point of view C_f and C_s are the interesting parameters, because C_f describes the total mass of fresh air flowing through the intake port into the cylinder (which feeds the combustion) and C_s is a measure of mixture formation.

The history of the intake port geometry in our work is the following. With a classical engineering development process (small deformations using rasp and putty) an initial shape was formed. We shall call it Shape 1. In the next step a CAD-model of Shape 1 was constructed based on geometrical measurements. This CAD-model was used in our calculations, namely a new shape called Shape 2 was realized using CNC-machines from this CAD-model. Shape 1 and Shape 2 are almost the same, but due to the measuring errors of CAD-model building there are small differences between them. Both Shape 1 and Shape 2 were examined experimentally. From an engineering point of view Shape 1 and Shape 2 appeared to be equivalent with acceptable differences (see [1]).

We shall present both results in comparison because the differences between the measured characteristics of Shape 1 and Shape 2 show the acceptable difference between the calculated and measured values.

Our simulation is based on the CAD-model, therefore our first goal was to get back the measured C_f and C_s values for Shape 2 at different pressures. We used experimental data for Shape 1 as a reference: if the difference between the simulated values and Shape 2 is less than the difference between Shape 1 and 2, we say that the numerical model is accurate enough for our purposes.

p_{in} was the same in all experiments: $p_{in} = 105.0\,\text{kPa}$, the temperature of steady air was 18 degree Celsius. The experiments were performed for 4 different pressures at the outflow. We shall label these cases “A”, “B”, “C” and “D”. Table 1 shows the values of pressure at the outflow part. The measured values of C_f and C_s can be found below in Table 4 together with the results of simulation.

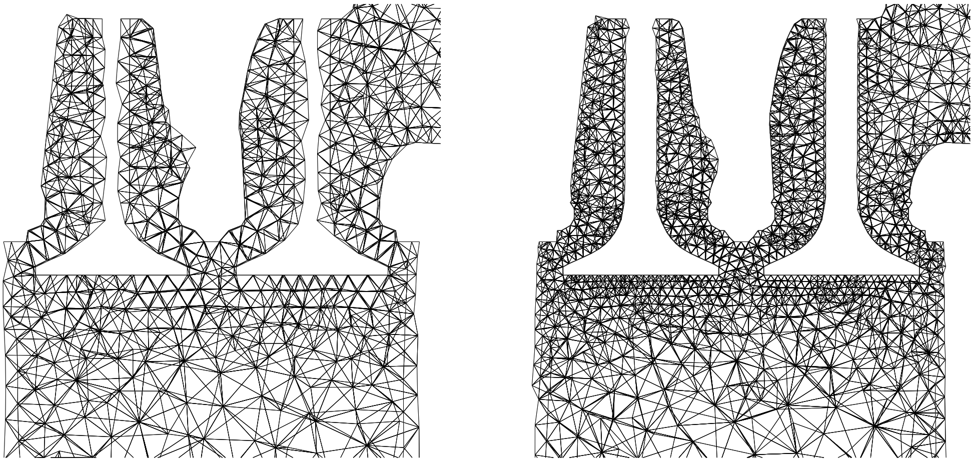
Case	A	B	C	D
p_{out} [kPa]	87.2	89.9	91.2	92.5

Table 1. Different values of p_{out}

3.2. **Discretization of the initial geometry.** Using a CAD model of the intake port, the intake tube and the top of cylinder, the flow domain was divided into tetrahedra. Two different grids were constructed, which are referred to as Grid 1 and Grid 2. The main properties are shown in Table 2. To demonstrate the difference between the resolution of Grid 1 and Grid 2 we present a cut of tetrahedral meshes in the plane of valves on Figure 2. One can observe that Grid 1 uses only 2 tetrahedra in the small gap between valve and the top of cylinder, while Grid 2 uses at least 4. This property can affect the accuracy significantly.

	Grid 1	*Grid 2*
Number of gridpoints	9 685	31 830
Number of tetrahedra	43 371	147 775

Table 2. Main properties of two tetrahedral grids



Grid 1

Grid 2

Figure 2. Comparison of Grid 1 and Grid 2 with a cut of tetrahedral mesh.

The list and coordinates of gridpoints (vertices of tetrahedra) and the list of tetrahedra were exported from CAD software and a simple Perl script was used to convert these files into the input form of our program.

As shown below in Table 4 resolution of Grid 2 is needed and is enough to get accurate results.

3.3. **Verification of the numerical simulation.**

3.3.1. *Typical characteristics of the numerical simulation.* A numerical simulation on Grid 2 needs 120 000–150 000 iterations (being equivalent to 0.02–0.05 s physical time) to reach the quasy-steady state flow. The CPU time on a 466 MHz Pentium III. based workstation was 14–18 hours and the program used 48 megabytes of RAM, when the Vijayasundaram method was applied to the Euler equations.

Figure 3 shows how the C_f and C_s values are converging to the stationary values. (The initial state was homogeneous density with zero velocities. Moreover, in fact the graph of mass flow rate and torque are displayed, which are a constant multiple of the corresponding coefficients.) It is an interesting fact that the frequency of oscillations in mass flux corresponds to the frequency of sound wave with wavelength 0.3–0.4 m which is the linear size of the whole system.

	SW	Vijaya	Measured
C_f	0.281	0.509	0.509
C_s	0.034	0.260	0.261

Table 3. Results of computations and measurements for verification

During the calculations it was possible to visualize the partial results, because some global parameters as well as the state of flow was written to hard disk periodically. The stability of the operation system was very important: Linux made no errors even if the simulation and the visualization software were running simultaneously on a machine where 2–3 users were doing their daily tasks.

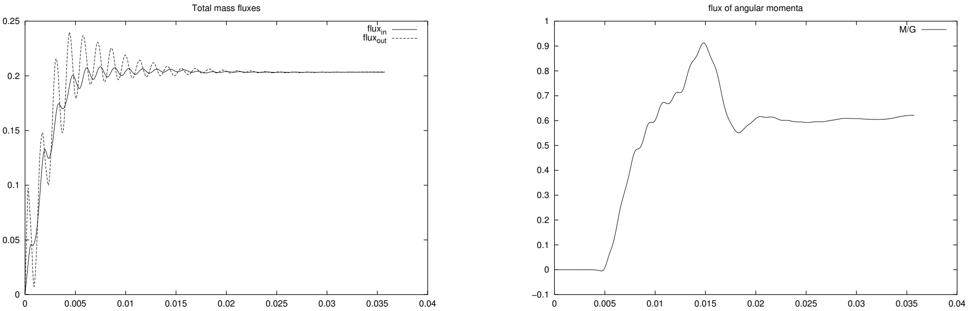


Figure 3. Fluxes as a function of time in a typical case. (Grid 2, pressure B.)

3.3.2. *Comparison with experimental data.* First we want to compare numerical simulation results with measurements; we want in this way to verify our mathematical model and the numerical methods we applied as well. Moreover, we saw in the previous subsection that there was a great difference between CPU-times when turning from the Steger-Warming method to the Vijayasundaram method (the CPU-time was doubled).

In connection with this we want to reply to the question: Is there a significant difference between the accuracy of the Steger-Warming method and the Vijayasundaram method?

In Table 3 we display both the numerical results and measurements w.r.t. the setting with parameters Shape 2, B and Grid 2. In the table SW and Vijaya refer to the Steger-Warming and Vijayasundaram methods, respectively. In the table we show the measurements (see [1]) of the corresponding experiments as well to have a base point.

The conclusion of these calculations is clear: the Steger-Warming method gives inaccurate results while the results with the Vijayasundaram method fit very well to the experimental data. Moreover, these results show also that neglecting viscosity from our mathematical model is acceptable in our case: although the flow near to the valves must be turbulent, the global C_f and C_s parameters seem not to be sensitive to this effect: the computational results with the non-viscous model and the more accurate numerical method give back the C_f and C_s values.

		Grid 1	Grid 2	Shape 2	Shape 1
A	C_f	0.478	0.517	0.517	0.513
	C_s	0.209	0.260	0.258	0.212
B	C_f	0.472	0.509	0.509	0.499
	C_s	0.197	0.259	0.261	0.211
C	C_f	0.464	0.504	0.501	0.496
	C_s	0.202	0.256	0.267	0.211
D	C_f	0.455	0.494	0.494	0.488
	C_s	0.199	0.249	0.276	0.206

Table 4. Comparing experimental data with simulations.

Both conclusions are plausible. In other calculations described in the literature the Vijayasundaram method proved to be much more accurate in a wide range of problems than the Steger-Warming method, see e.g. [10]. The viscosity of air is quite small and in our case the smallest gap was approximately 9 mm wide. In such cases simulations of other air-flow problems turned out to be well approximated neglecting the viscosity.

Although our simulation software was tested in classical test problems, we had to compare the results of simulation with experimental data. Therefore we performed test calculations for all 4 values of p_{out} on both grids with the Vijayasundaram method; Table 4 shows the result.

We can conclude that the results on the coarse Grid 1 are approximately good, while on fine Grid 2 all the results are perfect. More precisely the values in the columns Grid 2 and Shape 2 differs less than 5% in all cases but one, and this difference is significantly less than the difference between “Shape 1” and Shape 2. (See above in Section 3.1).

Note that the relative deviation of computed C_s values to measured one are higher than that for C_f, in agreement with the literature (see e.g. [7]).

As a final conclusion we may state that simulations with the Euler equations on Grid 2 with the Vijayasundaram method give reliable results.

4. Finding an optimal modification

4.1. **Modification of the original shape.** As a result of the previous section we are provided with a good numerical simulation method for reliably computing air-flow in an arbitrary intake geometry. Hence we are in the position to solve our problem posed in Section 1.1, i.e., the numerical simulation can serve as a substitute for the test-piece modification and measurements.

In order to have an idea where to deform the actual geometry we visualized the flow to find the most sensitive parts of the intake model, see Section 4.2. Using these sensitive points we examined some simple deformations and based on these results we could combine some of them to achieve our goal, see Section 4.3.

4.2. **Study of the calculated flow.** Visualization of flow gives us a great possibility to understand what happens in the intake port. Although the real strength of this tool is available only in interactive work, where the developer can move the cutting plane or rotate the model as he or she wants to, we present some figures about the stationary flow. All of our figures were made using a simulation on Grid 2 in case B of pressure. (Note that other values of pressure make the flow very similar to this one.)

In Figure 4 we present the velocity distribution in 3 cutting planes. Since the head of vectors makes the figures less usable, we neglected them. The direction of velocity can be figured out simply.

Figure 5 shows the velocities in the cylinder at different heights.

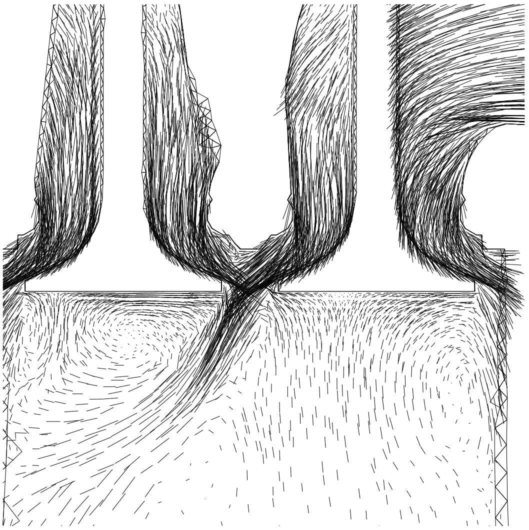
Besides getting an overall view of flow we can perform special searching tasks. For example we can visualize the distribution of pressure and search for places with high pressure gradients. This is very important information, because in such places the air has substantial acceleration, and it usually means strong resistance against the mass flux. In figure 6 we present two pictures about pressure distribution. One can easily detect the places where the gradients have high values, i.e. the places with high contrast in the Figures. This information led us to try a small deformation at the neck. (Deformation a; see in the next section.)

Similar investigations were performed e.g. to find the places with small kinetic energy density. This kind of places may be superfluous and possibly can be cut out from the volume of intake port. On the other hand large kinetic energy density indicates important parts, where the shape distorsions can affect the flow significantly. In Figure 7 we present two pictures representing the density of kinetic energy ($\rho\mathbf{v}^2/2$). We marked the low kinetic energy density regions on the left, the high kinetic energy density regions on the right picture.

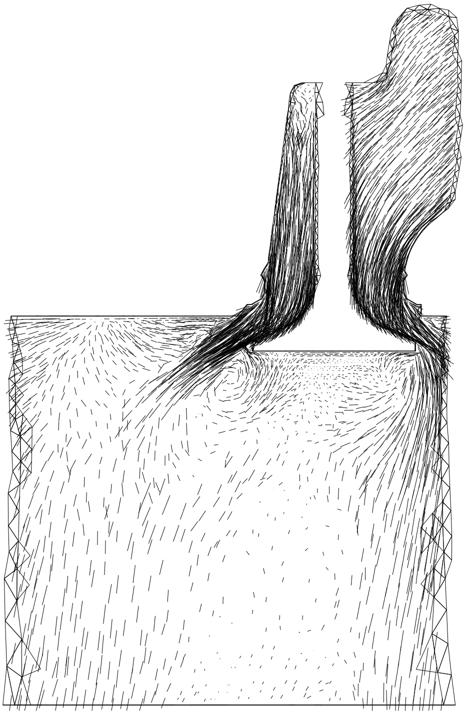
4.3. **Small deformations.** The deformations we used were local modifications on vertex coordinates. A simple algorithm was applied to perform this displacement. Here we discuss it briefly.

We found the following way of deformations very useful. For the definition of such an elementary deformation we have to give the following parameters:

- The center of deformation. Because our grid is fine enough at the surface, the center was defined by the index number of a vertex point of grid.
- The radius of deformation.
- The deepth of deformation. Let us imagine a cylinder with an axis perpendicular to the original surface and its center be the center of deformation. Let the radius of this cylinder be equal to the radius of deformation and its height be the double of the deepth. We shall displace only the vertices inside this cylinder.

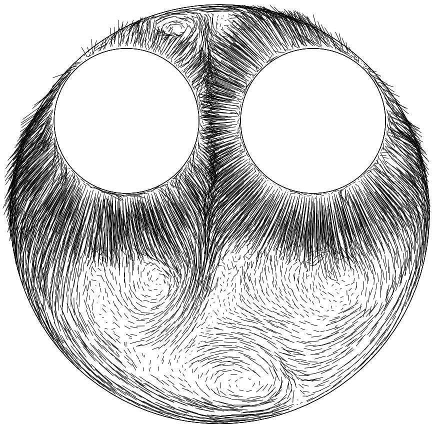


Plane with two valves

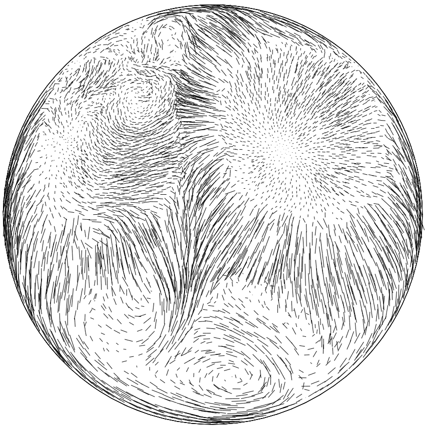


Valve 1

Figure 4. Velocity distribution in different planes



Plane A



Plane B

Figure 5. Velocity distribution in the cylinder. (Plane A contains the bottom of the valves, B is at 5 mm distance from A downwards)

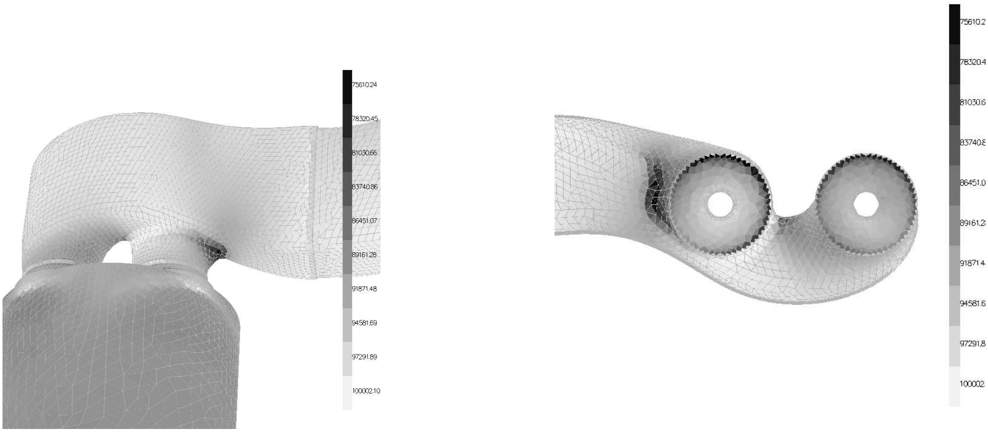


Figure 6. Pressure distribution in the intake port

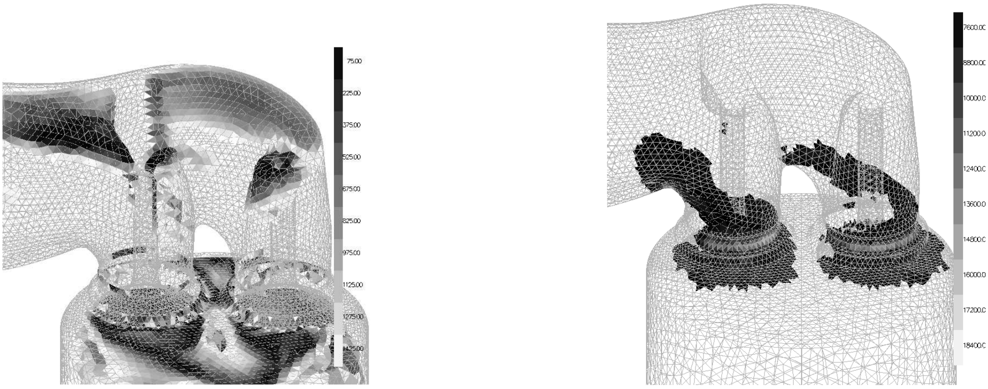
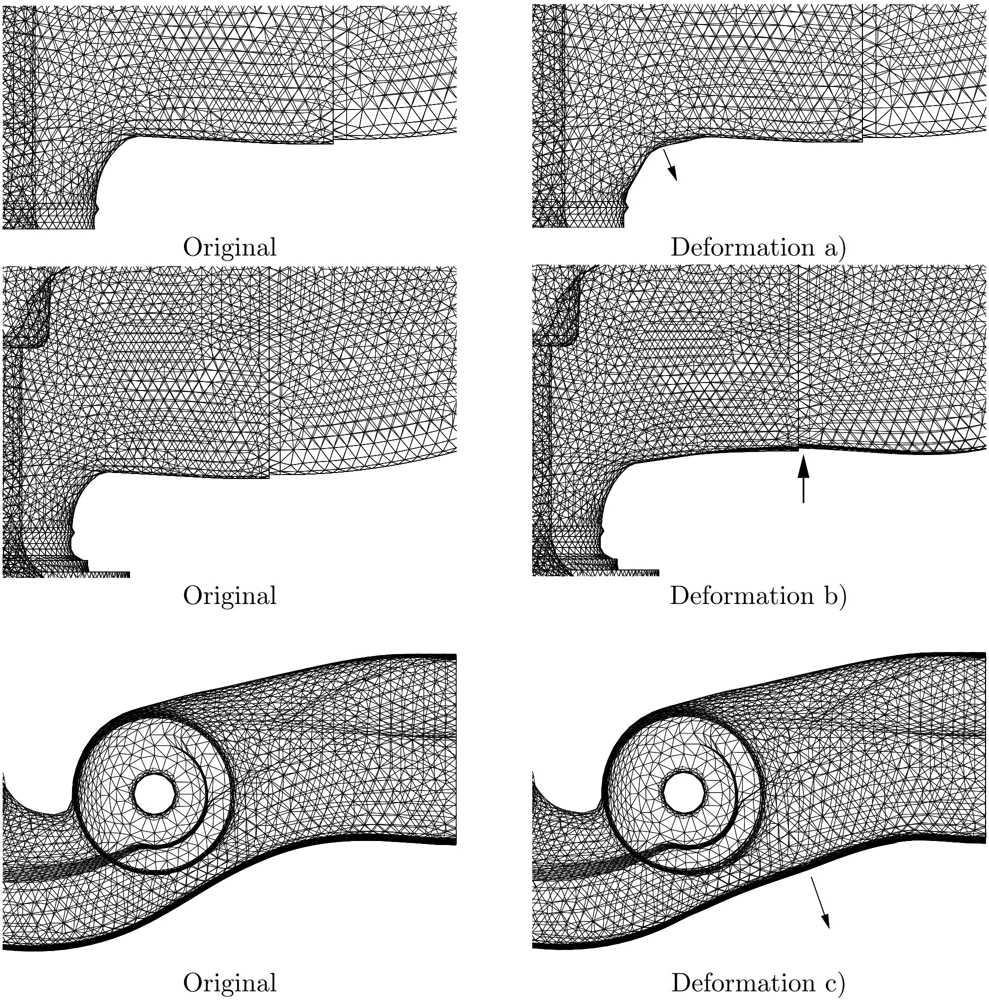


Figure 7. Low (left) and high (right) kinetic energy density regions in the intake port

- The amount of deformation. It gives the displacement at the center. The amount of displacement vanishes at the surface of the cylinder to protect us from badly deformed tetrahedra.

Study of the effect of small deformations was performed on Grid 2, in case B for 15 deformations. Here we present only 3 of them in Figure 8.

In Table 5 we present the flow and swirl coefficients of the deformed shapes and the difference between deformed and original volumes of intake port. Considering the basic deformations (a, b, c) one can observe approximately 1% differences in C_f and 5% differences in C_s.



Original

Deformation a)

Original

Deformation b)

Original

Deformation c)

Figure 8. Three selected deformations

Deformation	C_f	C_s	ΔV [cm^3]
Original	0.509	0.259	0.0
a)	0.514	0.271	1.1
b)	0.513	0.246	-18.1
c)	0.516	0.250	8.4
C1)	0.515	0.254	-15.6
C2)	0.514	0.257	-19.3

Table 5. Stationary properties of deformed ports

As mentioned above a great number of deformations were tested numerically and we used their combination to keep C_s at the original value while increasing C_f. The best combined deformations (see C1 and C2 in the Table) could gain more than 1% improvement in C_f at the same C_s while the volume of the intake port decreased significantly, see Table 5.

After numerical simulations, the most promising versions among our suggestions were tested experimentally at Autokut RT. The experiments showed that the results of our numerical simulation are reliable and suitable for improving the engine.

5. Conclusions

In our paper we presented the results of a software development for compressible fluid dynamics applied to an engineering problem.

As a general conclusion we can state that the Vijayasundaram method is suitable for a sufficiently accurate compressible flow simulation in existing engineering problems, even on a non-adaptive mesh. For the simulation the capacity of an average workstation is enough to achieve accurate results for such problems.

Our numerical results were compared to experimental results and found to be valid. Using the numerical simulation we could investigate the flow in the intake port in details, which is impossible with a usual experimental method. With numerical simulation we could examine the effects of many deformations and could find small deformations that satisfy the demands of engine designers and improve the intake port.

Finally we note that our study of numerical simulation of Diesel engines has gone further, e.g. at present our software can handle the moving piston, the motion and evaporation of fuel droplets. The core of these simulations, the results of which will be presented in the future, is the code reported in this paper.

Acknowledgement. The authors would like to express their grateful thanks to A. Csíkos (Autókut RT.) for putting his raw measurements at their disposal and to A. Csiszár for some initial tetrahedral grids.

References

1. Cser, G. and Csíkos, A.: *Measurements on the intake port with different geometries*, Technical Report of Autokut RT., Budapest, 1998. (in Hungarian)
2. Felcman, J., Dolejší, V. and Feistauer, M.: *Adaptive Finite Volume Method for the Numerical Solution of the Compressible Euler Equations*, In: Computational Fluid Dynamics 94, Proc. of the 2nd European CFD Conf., Wiley (1994) 894-901.
3. Feistauer, M.: *Mathematical Methods in Fluid Dynamics*, Pitman Monographs and Surveys in Pure and Applied Mathematics, Longman Scientific & Technical, UK, 1993.
4. Haase, G.: *Parallelization of Numerical Algorithms*, Report of Johannes Kepler University of Linz, 1999.

5. KANG, KERN Y. and REITZ, R.D.: *The effect of intake valve alignment on swirl generation in a DI diesel engine*, Experimental Thermal and Fluid Science, **20**, (1999), 94-103.

6. KRÖNER, D.: *Numerical Schemes for Conservation Laws*, Wiley and Teubner, Chichester, Stuttgart, 1997.

7. KRÜS, H.: *Numerical simulation and experimental verification of DI diesel intake port designs*, 4th Int. Conf. on Vehicle and Traffic Systems Technology Strasbourg, 16-18 June 1993. `http://www.cyclone.nl/ports/ports.htm`

8. MEISTER, A. and SONAR, TH.: *Finite Volume Schemes for Compressible Fluid*, Surv. Math. Ind., **8**, (1998), 1-36.

9. MOLNÁRKA, GY., WETTL, F., KALLÓS, G., GERGÓ, L. AND HORVÁTH, A.: *Maple V and its Applications.* Springer Budapest, 1996. (in Hungarian)

10. VIJAYASUNDARAM, G.: *Transonic flow simulations using an upstream centered scheme of Godunov in finite elements*, J. Comput. Phys. **63**, (1986), 416-433.

11. WIERSE, M.: *Higher Order Upwind Schemes on Unstructured Grids for the Compressible Euler Equations in Timedependent Geometries in 3D.* Dissertation, Preprint 393, SFB256, Bonn University, 1995.

Journal of Computational and Applied Mechanics, Vol. 4., No. 2., (2003), pp. 147–158

COMPUTATION OF TURBULENT FLOW IN AN S-SHAPED CHANNEL

Gábor Janiga
Department of Fuid and Heat Engineering, University of Miskolc
3515 Miskolc - Egyetemváros, Hungary
aramjg@uni-miskolc.hu

[Received: December 20, 2002]

Abstract. The application of a new stochastic turbulence model for curved channel flow is presented. The numerical computation was performed using the finite volume method on colocated variable arrangements and SIMPLE based pressure-correction method was used to treat the velocity-pressure coupling. The widely used approach for computation of laminar flow was extended by the discretization of the turbulent Reynolds stress tensor modelled by the new stochastic turbulence model of Czibere [1]. In this paper an application of this turbulence model is shown for curved channel flow. The computed velocity field was compared with the experimental data measured by LDV.

Mathematical Subject Classification: 76D06
Keywords: turbulent flow, curved channel, turbulence model

1. **Introduction**

Many engineering turbulence models use the Boussinesq-hypothesis to close the system of governing equations, but in some situations some of them fail to produce acceptable results [2, 3, 4]. A new stochastic turbulence model and related numerical computations are presented in this paper. The discretization methods are also shown after having introduced the governing equations.

Computations were performed in an S-shaped confuser and comparison was made with experimental data measured by LDV. Similar geometry can be found in Leoffler [5], but he made computations in an S-shaped diffuser. In this paper two-dimensional, incompressible flow is assumed.

In a statistically steady turbulent flow the Reynolds-averaged Navier-Stokes (RANS) equations read:

$$\frac{\partial\left(\rho\mathbf{v}\right)}{\partial t}+\operatorname{Div}\left(\rho\mathbf{v}\circ\mathbf{v}\right)=\mathbf{f}\rho-\nabla p+\operatorname{Div}\boldsymbol{\tau}+\operatorname{Div}\mathbf{F}_R\;, \tag{1.1}$$

where $\mathbf{F}_R=-\rho\left(\overline{\mathbf{v}'\circ\mathbf{v}'}\right)$ is the Reynolds stress tensor. This tensor introduces new unknown variables so the system of equations is not closed, and we have to use a turbulence model to close the system.

2. Application of the stochastic turbulence model

2.1. **Description of the stochastic turbulence model.** This model is the three-dimensional extension of the well-known Kármán similarity hypothesis for two-dimensional turbulent flow. The turbulent stress tensor $\mathbf{F}_R$ is defined in the so-called natural coordinate system according to the model of Czibere [1]:

$$\mathbf{F}_R = \rho\kappa^2 l^2 \mathbf{H}\,|\mathit{\Omega}|\,\mathit{\Omega}\ ,$$

where $\kappa = 0.41$ is the Kármán constant, l is the length scale and $\mathbf{H}$ is the similarity tensor:

$$\mathbf{H} = \begin{pmatrix} \alpha & 1 & \mu \\ 1 & \beta & \vartheta \\ \mu & \vartheta & \gamma \end{pmatrix} ,$$

where α, β, γ, μ and ϑ are the constants of the model [1]. The basis vectors of the natural coordinate system are defined with the time-mean velocity $\mathbf{v}$ and the vorticity vector $\nabla \times \mathbf{v}$:

$$\mathbf{e}_3' = -\frac{\nabla \times \mathbf{v}}{|\nabla \times \mathbf{v}|}, \quad \mathbf{e}_2' = \frac{\mathbf{v} \times (\nabla \times \mathbf{v})}{|\mathbf{v} \times (\nabla \times \mathbf{v})|}, \quad \mathbf{e}_1' = \mathbf{e}_2' \times \mathbf{e}_3'\ .$$

The advantage of this definition is that two components of the vorticity vector are zero in this system.

Let us introduce the following notation:

$$\mathit{\Theta}\,(q_1', q_2', q_3', t) = \rho\kappa^2 l^2\,|\mathit{\Omega}|\,\mathit{\Omega}\ , \tag{2.1}$$

where $\mathit{\Theta}$ is the dominant turbulent stress in turbulent shear flow.

The directions of the coordinate axes change from point to point, as the velocity and the curl of the velocity change. In order to perform the numerical computation it is easier to use the physical coordinate system, since in this way transformation from the natural coordinate system q_1', q_2', q_3' can be performed. The Reynolds stress tensor $\mathbf{F}_R$ in the physical coordinate system reads:

$$\mathbf{F}_R = \mathit{\Theta}\,(q_1, q_2, q_3, t)\,\mathbf{G}\ , \tag{2.2}$$

where tensor $\mathbf{G}$ is the transformation of the tensor $\mathbf{H}$ defined in the natural coordinate system:

$$\mathbf{G} = \mathbf{E}\,\mathbf{H}\,\mathbf{E}^T\ . \tag{2.3}$$

$\mathbf{E}^T$ is the transpose tensor of the tensor $\mathbf{E}$. The scalar elements of the tensor $\mathbf{E}$ are defined by the time-mean velocity $\mathbf{v}$ and the vorticity vector $\nabla \times \mathbf{v}$. For numerical computation the Cartesian coordinate system x, y, z was used instead of the

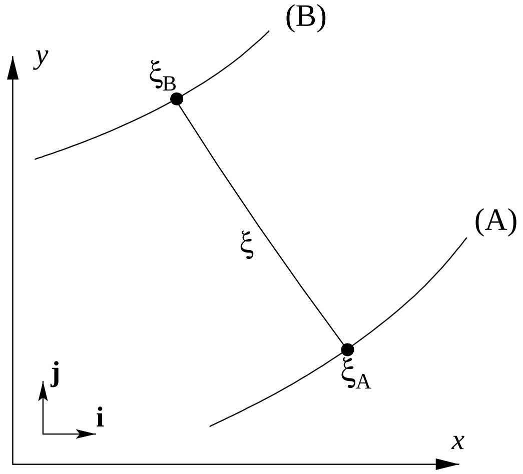


Figure 1. Definition of the scale function along a trajectory

curvilinear orthogonal coordinate system q_1, q_2, q_3:

$$E_{xx} = \frac{1}{\sqrt{1-\lambda^2}}\left(\frac{v_x}{|\mathbf{v}|} - \lambda\frac{\Omega_x}{|\mathbf{\Omega}|}\right) \quad , \quad E_{xy} = \frac{1}{\sqrt{1-\lambda^2}}\frac{v_y\Omega_z - v_z\Omega_y}{|\mathbf{v\Omega}|} \quad , \quad E_{xz} = -\frac{\Omega_x}{|\mathbf{\Omega}|} \quad ,$$

$$E_{yx} = \frac{1}{\sqrt{1-\lambda^2}}\left(\frac{v_y}{|\mathbf{v}|} - \lambda\frac{\Omega_y}{|\mathbf{\Omega}|}\right) \quad , \quad E_{yy} = \frac{1}{\sqrt{1-\lambda^2}}\frac{v_z\Omega_x - v_x\Omega_z}{|\mathbf{v\Omega}|} \quad , \quad E_{yz} = -\frac{\Omega_y}{|\mathbf{\Omega}|} \quad ,$$

$$E_{zx} = \frac{1}{\sqrt{1-\lambda^2}}\left(\frac{v_z}{|\mathbf{v}|} - \lambda\frac{\Omega_z}{|\mathbf{\Omega}|}\right) \quad , \quad E_{zy} = \frac{1}{\sqrt{1-\lambda^2}}\frac{v_x\Omega_y - v_y\Omega_x}{|\mathbf{v\Omega}|} \quad , \quad E_{zz} = -\frac{\Omega_z}{|\mathbf{\Omega}|} \quad ,$$

where $\lambda = (\mathbf{v\Omega}) \,/\, (|\mathbf{v}|\,|\mathbf{\Omega}|)$.

The (1.1) momentum equation can be reshaped according to the stochastic turbulence model:

$$\frac{\partial(\rho\mathbf{v})}{\partial t} + \mathrm{Div}\,(\rho\mathbf{v}\circ\mathbf{v}) = \mathbf{f}\rho - \nabla p + \mathrm{Div}\,\boldsymbol{\tau} + \mathrm{Div}\,(\Theta\mathbf{G}) \ . \tag{2.4}$$

The length scale l is always zero on the wall. According to the Kármán-Prandtl similarity hypothesis, the mixing length $l_{mix} = \kappa l$ is a linear function when approaching the wall. In the numerical computation the length scale l appearing in the definition of Reynolds stress tensor is approximated by a fourth-order polynomial:

$$l(\xi) = 4S\left(1 - \frac{(4S-T)}{S}\left(\frac{\xi-\xi_0}{\xi_B-\xi_A}\right)^2\right)\frac{(\xi-\xi_A)\cdot(\xi_B-\xi)}{(\xi_B-\xi_A)} \ ,$$

$$\text{where} \quad \xi_A \le \xi \le \xi_B \quad , \quad \xi_0 = \frac{\xi_A+\xi_B}{2} \quad \text{and} \quad S,\, T > 0 \ .$$

The $(\xi_B - \xi_A)$ means the distance between the two points, and ξ is the parameter of a trajectory (see Figure 1).

According to the transformation from the natural coordinate system q'_1, q'_2, q'_3 to the Cartesian coordinate system x, y, z, every (q'_1, q'_2, q'_3) triad corresponds to triad (x, y, z) according to the following equations:

$$x = x\left(q'_1, q'_2, q'_3\right) , \quad y = y\left(q'_1, q'_2, q'_3\right) , \quad z = z\left(q'_1, q'_2, q'_3\right) \ .$$

2.2. **Transformation to the computational coordinate system.** An arbitrary scalar value (such as the function Θ) is not changed during the transformation:

$$\Theta\left(x, y, z, t\right) \equiv \Theta'\left(q'_1, q'_2, q'_3, t\right) \ .$$

A vector (for example the velocity vector $\mathbf{v}$) can be transformed from the q'_1, q'_2, q'_3 natural coordinate system to the x, y, z Cartesian coordinate system with the following equation:

$$\begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} = \begin{pmatrix} E_{xx} & E_{xy} & E_{xz} \\ E_{yx} & E_{yy} & E_{yz} \\ E_{zx} & E_{zy} & E_{zz} \end{pmatrix} \cdot \begin{pmatrix} v_{1'} \\ v_{2'} \\ v_{3'} \end{pmatrix} \ . \tag{2.5}$$

The $\mathbf{G}$ tensor in the x, y, z Cartesian coordinate system can be calculated using the similarity tensor $\mathbf{H}$ defined in the q'_1, q'_2, q'_3 natural coordinate system:

$$\begin{pmatrix} G_{xx} & G_{xy} & G_{xz} \\ G_{yx} & G_{yy} & G_{yz} \\ G_{zx} & G_{zy} & G_{zz} \end{pmatrix} = \begin{pmatrix} E_{xx} & E_{xy} & E_{xz} \\ E_{yx} & E_{yy} & E_{yz} \\ E_{zx} & E_{zy} & E_{zz} \end{pmatrix} \cdot \begin{pmatrix} \alpha & 1 & \mu \\ 1 & \beta & \vartheta \\ \mu & \vartheta & \gamma \end{pmatrix} \cdot \begin{pmatrix} E_{xx} & E_{yx} & E_{zx} \\ E_{xy} & E_{yy} & E_{zy} \\ E_{xz} & E_{yz} & E_{zz} \end{pmatrix} \ .$$

This tensor equation can be written by the following six scalar equations:

$$\begin{aligned}
G_{xx} &= \alpha E_{xx}^2 + \beta E_{xy}^2 + \gamma E_{xz}^2 + 2\left(E_{xx}E_{xy} + \mu E_{xx}E_{xz} + \vartheta E_{xy}E_{xz}\right) , \\
G_{yy} &= \alpha E_{yx}^2 + \beta E_{yy}^2 + \gamma E_{yz}^2 + 2\left(E_{yx}E_{yy} + \mu E_{yx}E_{yz} + \vartheta E_{yy}E_{yz}\right) , \\
G_{zz} &= \alpha E_{zx}^2 + \beta E_{zy}^2 + \gamma E_{zz}^2 + 2\left(E_{zx}E_{zy} + \mu E_{zx}E_{zz} + \vartheta E_{zy}E_{zz}\right) , \\
G_{xy} &= G_{yx} = \alpha E_{xx}E_{yx} + \beta E_{xy}E_{yy} + \gamma E_{xz}E_{yz} + \\
&+ \left(E_{xx}E_{yy} + E_{xy}E_{yx}\right) + \mu\left(E_{xx}E_{yz} + E_{xz}E_{yx}\right) + \vartheta\left(E_{xy}E_{yz} + E_{xz}E_{yy}\right) , \\
G_{xz} &= G_{zx} = \alpha E_{xx}E_{zx} + \beta E_{xy}E_{zy} + \gamma E_{xz}E_{zz} + \\
&+ \left(E_{xx}E_{zy} + E_{xy}E_{zx}\right) + \mu\left(E_{xx}E_{zz} + E_{xz}E_{zx}\right) + \vartheta\left(E_{xy}E_{zz} + E_{xz}E_{zy}\right) , \\
G_{yz} &= G_{zy} = \alpha E_{yx}E_{zx} + \beta E_{yy}E_{zy} + \gamma E_{yz}E_{zz} + \\
&+ \left(E_{yx}E_{zy} + E_{yy}E_{zx}\right) + \mu\left(E_{yx}E_{zz} + E_{yz}E_{zx}\right) + \vartheta\left(E_{yy}E_{zz} + E_{yz}E_{zy}\right) \ .
\end{aligned}$$

3. Governing equations for two-dimensional flow

The RANS equations for steady flow can be written in the following way, as the continuity equation:

$$\frac{\partial\left(\rho u\right)}{\partial x} + \frac{\partial\left(\rho v\right)}{\partial y} = 0 \ , \tag{3.1}$$

and as the momentum equations for viscous fluid according to the model of Czibere:

$$\begin{aligned}\frac{\partial(\rho uu)}{\partial x}+\frac{\partial(\rho vu)}{\partial y} &= -\frac{\partial p}{\partial x}+\eta\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial x}\right)+\eta\frac{\partial}{\partial y}\left(\frac{\partial u}{\partial y}\right)\\ &\quad+\frac{\partial(\Theta G_{xx})}{\partial x}+\frac{\partial(\Theta G_{xy})}{\partial y}\ ,\end{aligned}\tag{3.2}$$

$$\begin{aligned}\frac{\partial(\rho uv)}{\partial x}+\frac{\partial(\rho vv)}{\partial y} &= -\frac{\partial p}{\partial y}+\eta\frac{\partial}{\partial x}\left(\frac{\partial v}{\partial x}\right)+\eta\frac{\partial}{\partial y}\left(\frac{\partial v}{\partial y}\right)\\ &\quad+\frac{\partial(\Theta G_{yx})}{\partial x}+\frac{\partial(\Theta G_{yy})}{\partial y}\ .\end{aligned}\tag{3.3}$$

Here ρ is the density, u, v and x, y are the Cartesian velocity components and coordinate directions, respectively, and η is the dynamic viscosity.

In two-dimensional problems the vorticity vector $\mathbf{\Omega}$ is perpendicular to the flow plane, so the x and y components of this vector are cancelled in the transformed tensor. For two-dimensional flow the scalar elements of the transformation tensor $\mathbf{E}$ can be simplified:

$$E_{xx}=\frac{u}{\sqrt{u^2+v^2}}\quad ,\quad E_{xy}=-\frac{v}{\sqrt{u^2+v^2}}\quad ,\quad E_{xz}=0\quad ,$$

$$E_{yx}=\frac{v}{\sqrt{u^2+v^2}}\quad ,\quad E_{yy}=\frac{u}{\sqrt{u^2+v^2}}\quad ,\quad E_{yz}=0\quad ,$$

$$E_{zx}=0\quad ,\quad E_{zy}=0\quad ,\quad E_{zz}=1\ .$$

The elements of the tensor $\mathbf{G}$ can be calculated from transformation (2.3):

$$G_{xx}=\alpha\frac{u^2}{u^2+v^2}+\beta\frac{v^2}{u^2+v^2}-2\frac{uv}{u^2+v^2}\ ,\tag{3.4}$$

$$G_{yy}=\alpha\frac{v^2}{u^2+v^2}+\beta\frac{u^2}{u^2+v^2}+2\frac{uv}{u^2+v^2}\ ,\tag{3.5}$$

$$G_{xy}=G_{yx}=(\alpha-\beta)\frac{uv}{u^2+v^2}+\left(\frac{u^2}{u^2+v^2}-\frac{v^2}{u^2+v^2}\right)\ .\tag{3.6}$$

According to the stochastic model of Czibere [1] and the measured values of Laufer [6] the constant elements of the tensor $\mathbf{H}$ are: $\alpha=3.9714$ and $\beta=1.5734$.

The turbulent dominant shear stress Θ can be written in the physical coordinate system using the relations $\Omega_x=0$, $\Omega_y=0$ and $\Omega_z=\Omega_{3'}$. The vorticity vector for two-dimensional flow (where the velocity vector contains only two components in the directions $\mathbf{i}$ and $\mathbf{j}$):

$$\mathbf{\Omega}=\nabla\times\mathbf{v}=\Omega_z\mathbf{k}=\left(\frac{\partial v}{\partial x}-\frac{\partial u}{\partial y}\right)\mathbf{k}\ ,\tag{3.7}$$

and the function Θ can be obtained from (2.1) as:

$$\Theta(x,y,t)=\rho\kappa^2l^2\left|\Omega_z\right|\Omega_z\ .\tag{3.8}$$

The turbulent terms contain the velocity components and the coefficients of the turbulence model. Using the length scale l, the system of the algebraic equations is closed. We have four equations: the continuity equation (3.1), the two momentum equations (3.2)-(3.3) and the definition of function Θ (3.8), and four unknowns: velocities u and v, the pressure and the turbulent dominant shear stress Θ.

4. **Discretization method**

4.1. **Basic equations.** The conservation equations for mass and momentum in integral form serve as the starting point for finite-volume solution methods:

$$\int_{\Delta V} \operatorname{div}(\rho \mathbf{v})\, dV = \int_{(\delta A)} \rho \mathbf{v} \cdot d\mathbf{A} = 0 \,, \tag{4.1}$$

the momentum equation using the turbulence model of Czibere:

$$\int_{\Delta V} \operatorname{Div}(\rho \mathbf{v} \circ \mathbf{v})\, dV = -\int_{\Delta V} \nabla p \, dV + \int_{\Delta V} \operatorname{Div} \boldsymbol{\tau}\, dV + \int_{\Delta V} \operatorname{Div}(\Theta \mathbf{G})\, dV \,,$$

and applying the Gauss divergence theorem the volume integrals can be transformed into surface integrals:

$$\int_{(\delta A)} \mathbf{v}\,(\rho \mathbf{v} \cdot d\mathbf{A}) = -\int_{\Delta V} \nabla p \, dV + \int_{(\delta A)} \boldsymbol{\tau} \cdot d\mathbf{A} + \int_{(\delta A)} \Theta \mathbf{G} \cdot d\mathbf{A} \,. \tag{4.2}$$

Here, ΔV is the volume and (δA) is the surface of an arbitrary rectangular control volume. Steady flow, Newtonian fluid, constant density ρ and dynamic viscosity η are assumed here; also, gravitational body force is included in the pressure p. The computation was performed using the finite volume method on a curvilinear orthogonal rectangular coordinate system (Figure 2) with collocated variable arrangement, and Cartesian velocity components were used.

The solution domain is subdivided into a finite number of control volumes (CV), and a computational node is placed at the center of each CV. The integral expressions are applied to each CV, and the integrals are numerically evaluated.

4.2. **Turbulent terms.** Only the turbulent terms are considered here; for a more detailed description of the discretization methods for laminar flow, see e. g. [7].

The function Θ contains the length scale l, which only depends on the geometry of the computational domain. In the case of non-moving boundaries this function has constant values, but the vorticity vectors must be updated in every iteration step.

The turbulent terms according to the turbulence model of Czibere can be written by surface integrals in the u-momentum equation:

$$F_u^t = \int_{(A_\mathrm{e})} (\Theta G_{xx})\, dA_\mathrm{e} - \int_{(A_\mathrm{w})} (\Theta G_{xx})\, dA_\mathrm{w} + \int_{(A_\mathrm{n})} (\Theta G_{xy})\, dA_\mathrm{n} - \int_{(A_\mathrm{s})} (\Theta G_{xy})\, dA_\mathrm{s} \,,$$

where e, w, n, s denote the east, west, north and south boundary of the control volume, respectively. In the v-momentum equation:

$$F_v^t = \int_{(A_\mathrm{e})} (\Theta G_{yx})\, dA_\mathrm{e} - \int_{(A_\mathrm{w})} (\Theta G_{yx})\, dA_\mathrm{w} + \int_{(A_\mathrm{n})} (\Theta G_{yy})\, dA_\mathrm{n} - \int_{(A_\mathrm{s})} (\Theta G_{yy})\, dA_\mathrm{s}\ .$$

The elements of the tensor $\mathbf{G}$ are defined with the relations (3.4)-(3.6). In the discretization these elements can be divided into two parts, where the coefficients of u^{m+1} and v^{m+1} in G_{xx} and G_{xy} are identical as are those of u^{m+1} and v^{m+1} in G_{yx} and G_{yy}:

$$G_{xx} = u^{m+1}\left(\alpha\frac{u}{u^2+v^2} - \frac{v}{u^2+v^2}\right)^m + v^m\left(\beta\frac{v}{u^2+v^2} - \frac{u}{u^2+v^2}\right)^m\ , \tag{4.3}$$

$$G_{xy} = u^{m+1}\left(\frac{u}{u^2+v^2} + \alpha\frac{v}{u^2+v^2}\right)^m - v^m\left(\frac{v}{u^2+v^2} + \beta\frac{u}{u^2+v^2}\right)^m\ , \tag{4.4}$$

$$G_{yx} = v^{m+1}\left(\alpha\frac{u}{u^2+v^2} - \frac{v}{u^2+v^2}\right)^m - u^m\left(\beta\frac{v}{u^2+v^2} - \frac{u}{u^2+v^2}\right)^m\ , \tag{4.5}$$

$$G_{yy} = v^{m+1}\left(\frac{u}{u^2+v^2} + \alpha\frac{v}{u^2+v^2}\right)^m + u^m\left(\frac{v}{u^2+v^2} + \beta\frac{u}{u^2+v^2}\right)^m\ . \tag{4.6}$$

Terms G_{xx}, G_{xy}, G_{yx} and G_{yy} are linearized in a way that u^{m+1} and v^{m+1} are considered unknown velocity variables, while u^m and v^m are known from the previous iteration. Due to this division of the turbulent terms, the coefficients of the two momentum equations are equal for the colocated variable arrangement.

From the previous iteration step the following can be written using the second term on the right-hand side (RHS) in equations (4.3)-(4.6):

$$\begin{aligned} Q_u^t = {} & v_\mathrm{e}\Theta_\mathrm{e}A_\mathrm{e}\left(\beta\frac{v}{u^2+v^2} - \frac{u}{u^2+v^2}\right)_\mathrm{e} - v_\mathrm{w}\Theta_\mathrm{w}A_\mathrm{w}\left(\beta\frac{v}{u^2+v^2} - \frac{u}{u^2+v^2}\right)_\mathrm{w} \\ & -v_\mathrm{n}\Theta_\mathrm{n}A_\mathrm{n}\left(\frac{v}{u^2+v^2} + \beta\frac{u}{u^2+v^2}\right)_\mathrm{n} + v_\mathrm{s}\Theta_\mathrm{s}A_\mathrm{s}\left(\frac{v}{u^2+v^2} + \beta\frac{u}{u^2+v^2}\right)_\mathrm{s}\ , \end{aligned}$$

and the implicit terms can be derived from equations (4.3)-(4.6) using the first term on the RHS:

$$\begin{aligned} F_u^t = {} & -u_\mathrm{e}\Theta_\mathrm{e}A_\mathrm{e}\left(\alpha\frac{u}{u^2+v^2} - \frac{v}{u^2+v^2}\right)_\mathrm{e} + u_\mathrm{w}\Theta_\mathrm{w}A_\mathrm{w}\left(\alpha\frac{u}{u^2+v^2} - \frac{v}{u^2+v^2}\right)_\mathrm{w} \\ & -u_\mathrm{n}\Theta_\mathrm{n}A_\mathrm{n}\left(\frac{u}{u^2+v^2} + \alpha\frac{v}{u^2+v^2}\right)_\mathrm{n} + u_\mathrm{s}\Theta_\mathrm{s}A_\mathrm{s}\left(\frac{u}{u^2+v^2} + \alpha\frac{v}{u^2+v^2}\right)_\mathrm{s}\ . \end{aligned}$$

The convective and diffusive terms were discretized with the first-order upwind difference scheme (UDS) and the second-order central difference scheme (CDS) and deferred correction was used to connect them. The discretized algebraic equations were solved with Stone's strongly-implicit procedure (SIP) [8]. The solution of the coupled set of equations for u, v and p is based on the SIMPLE algorithm [9]. The coefficients of the discretized equations are updated and solved in turn and the process is repeated until convergence.

5. **Results of computations**

In order to show how to apply the stochastic turbulence model, turbulent flow was investigated in a curved-channel. Measurements were also performed with laser doppler velocimetry (LDV) at the University of Magdeburg [10]. The present computations requires a few minutes on today's personal computers. Using the present model we do not need any additional differential equation apart from the continuity and momentum equations, in contrast with many other engineering turbulence models, so the computational time per iteration can be reduced.

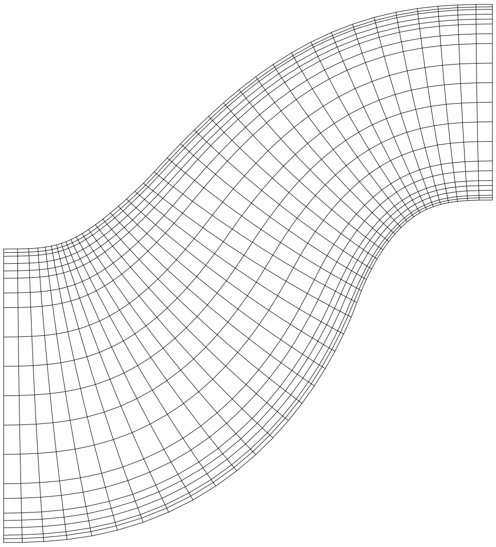


Figure 2. Computational grid

The contour plots of the computed and measured values are shown in dimension m/s in Figures 3-6. The colors and the scales of these figures mean the same values in pairs. Figures 7 and 8 represent the contour plots of the differences of the calculated and measured velocity components. As can be seen the agreement is quite good.

The extension of the present numerical methods for three-dimensional problems is planned in the near future.

6. **Conclusions**

Based on the results of computations presented above, the following conclusions can be drawn.

Using the stochastic turbulence model of Czibere the system of governing equations is closed, and the two unknown velocity components and the pressure can be determined from the two momentum equations and the continuity equation.

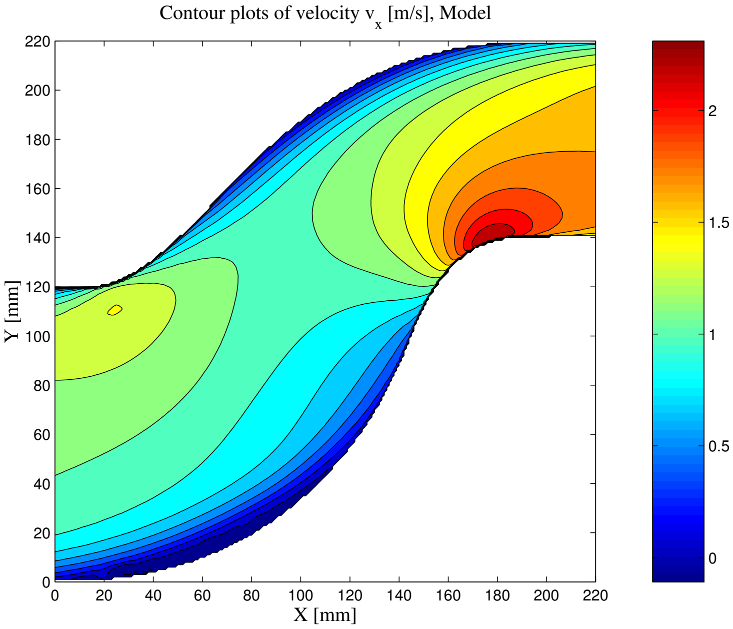


Figure 3. Calculated velocity components in x direction

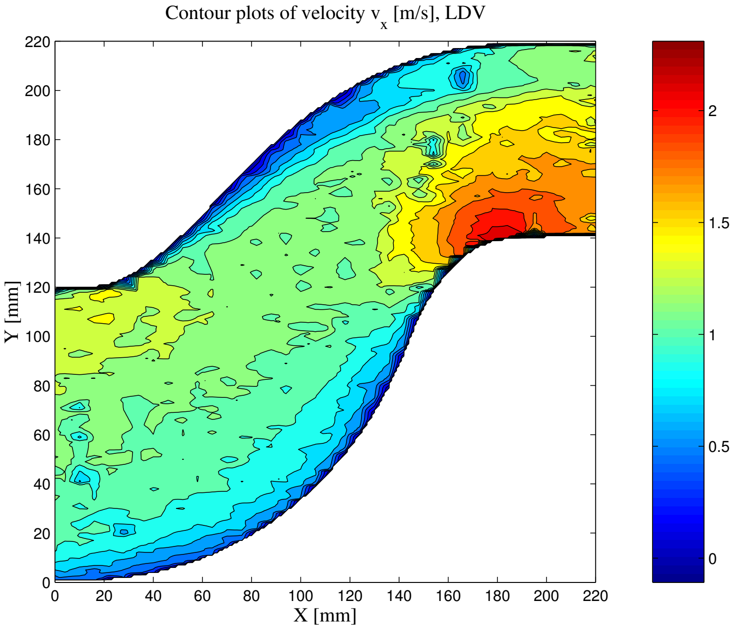


Figure 4. Measured velocity components in x direction

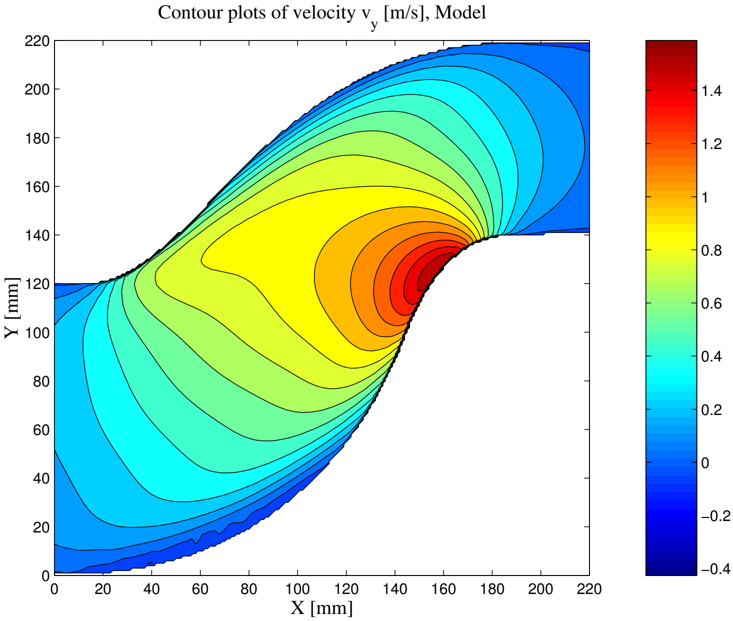


Figure 5. Calculated velocity components in y direction

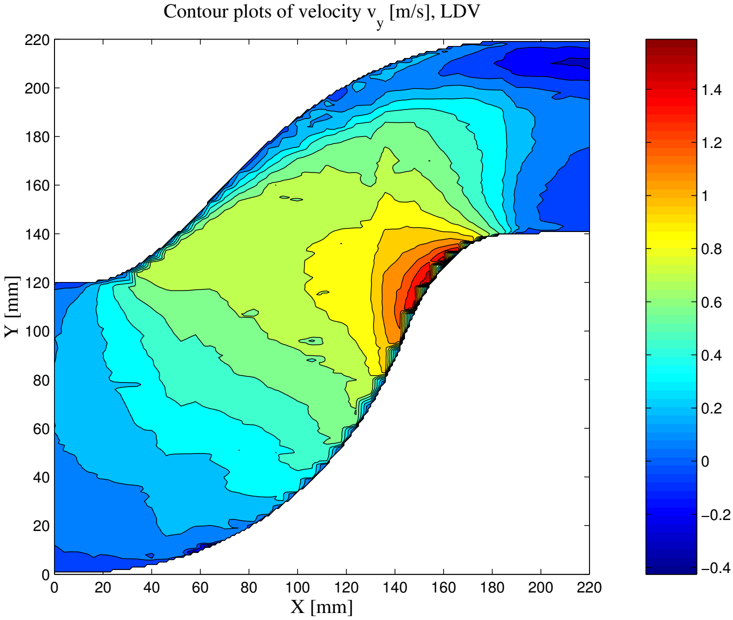


Figure 6. Measured velocity components in y direction

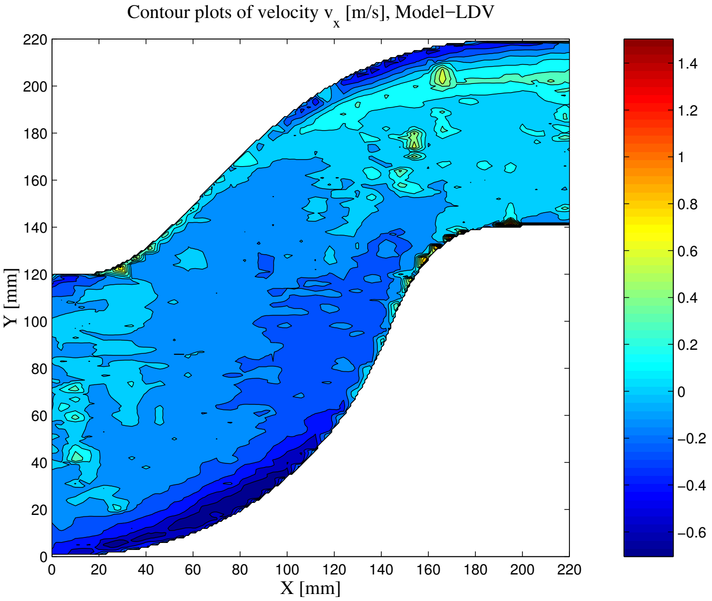


Figure 7. Differences of the calculated and measured velocity components in x direction

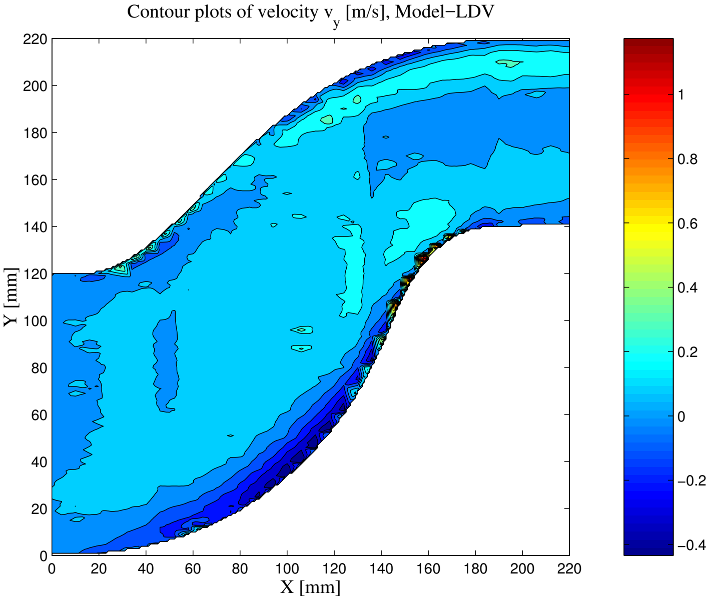


Figure 8. Differences of the calculated and measured velocity components in y direction

The application of this model does not require the solution of any additional differential equation apart from the continuity and momentum equations, in contrast with many other engineering turbulence models. Using this model the computational time per iteration step is less than that of most engineering models.

Computational results compare well with experimental results.

Acknowledgement. The author wishes to acknowledge the guidance of Professor Czibere. The support provided by the János Bolyai Research Scholarship is also greatly appreciated.

References

1. Czibere, T.: *Three dimensional stochastic model of turbulence*, Journal of Computational and Applied Mechanics, **2**, (2001), 7-20.
2. Hinze, J. O.: *Turbulence*, McGraw-Hill, New York, 1959.
3. Pope, S. B.: *Turbulent Flows*, Cambridge, University Press, 2000.
4. Wilcox, D. C.: *Turbulence Modeling for CFD*, DCW Industries, Inc., La Cañada, California, 2000.
5. Loeffler, Jr., A. L.: *A Navier-Stokes code for S-shaped diffusers – A review*, Int. J. Numer. Methods Fluids, **8**, (1988), 463-474.
6. Laufer, J.: *The Structure of Turbulence in Fully Developed Pipe Flow.* NACA Report 1174, 1954.
7. Ferziger, J. H. and Perić, M.: *Computational Methods for Fluid Dynamics*, Springer-Verlag, Berlin, 1999.
8. Stone, H. L.: *Iterative solution of implicit approximations of multidimensional partial differential equations*, SIAM J. Numer. Anal., **5**, (1968), 530-558.
9. Patankar, S. V. and Spalding, D. B.: *A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows*, Int. J. Heat Mass Transfer, **15**, (1972), 1787-1806.
10. Szabó, S., Farkas, A., Janiga, G., Kecke, H.-J., Praetor, R. and Wunderlich, B.: *Untersuchung der turbulenten ebenen Strömung in einem gekrümmten Kanal*, MicroCAD International Computer Science Conference, Miskolc, **N**, (2001), 33-40.

Journal of Computational and Applied Mechanics, Vol. 4., No. 2., (2003), pp. 159–173

INVESTIGATION OF THE DEFORMATION OF A MULTILAYERED PIEZOELECTRIC SEMIRING

Pál Zoltán Kovács
Department of Mechanics, University of Miskolc
3515 Miskolc-Egyetemváros A/4, Hungary
mechanik@uni-miskolc.hu

István Páczelt
Department of Mechanics, University of Miskolc
3515 Miskolc-Egyetemváros A/4, Hungary
mechpacz@uni-miskolc.hu

[Received: January 30, 2002]

Abstract. A semicircular planar prismatic beam is subjected to bending moments resulting from piezoelectric actuator devices bonded to the parallel surfaces of the beam. The bending and torsional deformations are investigated with the Finite Element Method (FEM) and the closed-form solution of the linear theory of 3D curved beams, respectively. The geometry of curved element segments is mapped exactly by using the blending function method and with the Legendre polynomials (isoparametric functions). The aim of the study is to demonstrate that a planar curved beam under piezoelectric load deflects and twists additionally.

Mathematical Subject Classification: 74E30,74F99,74K10
Keywords: piezoelectric actuator, blending functions, bending rigidity, FEM

1. **Introduction**

This study presents an investigation of a semicircular prismatic cantilever beam, the centerline of which is a planar curvilinear arc. Its parallel surfaces are covered by PIC151 piezoelectric patches (Figure 1).

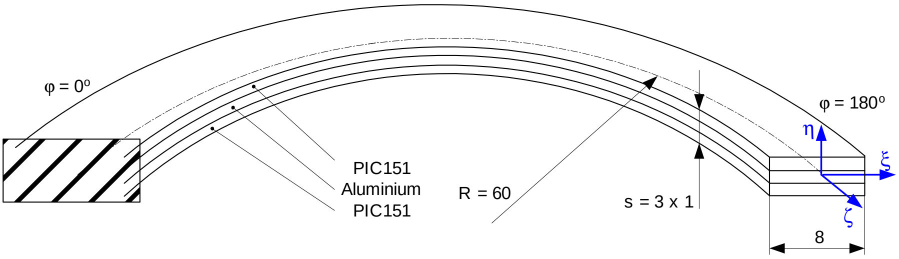


Figure 1. Composite semiring

Particularly, the linear theory of space curved beams is used and the results are compared to those obtained by the FEM. The inverse piezoelectric effect is realised by generating an electric potential difference between the piezoelectric surfaces perpendicular to the η-direction, which is the direction of its natural polarization. The longitudinal stresses in the actuators result in the contraction and the expansion of the external fibres of the basic aluminium beam, respectively. The effect generates a bending moment along the beam, which causes the deflection along the centerline and the angular rotations of the cross-sections. The aim of the study is to determine these deformations by using the closed form solution and to compare them to the approximate solution. The procedure is assumed to be quasi-static, the displacements and strains are considered to be small. The layers are bonded to each other through a glue layer with zero thickness. The aluminium beam is assumed to be isotropic, the piezoelectric material based on PIC151 is orthotropic with its polarizing axis. Both materials are homogenous.

2. Formulation of the problem by using the theory of space curved beams

The investigation follows the way as shown in [1]. Here, we simplify those equations, due to the fact that our structure has an initial planar form with constant initial radius of curvature. Furthermore, initially the cross-sections are not twisted. The geometrical arrangement is such that the so-called laminar piezoelectric effect dominates in the design, so the electric field in the η-direction causes expansion or contraction in the ζ-direction, which results in bending around the ξ-direction. Let us consider an arbitrary part of the beam with a width b and a thickness $2d + c$ as shown in Figure 2.

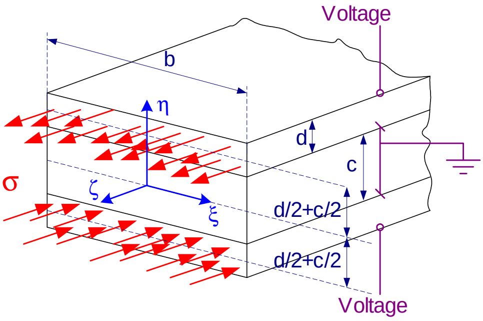


Figure 2. Tensile stresses in piezoelectric layers

The internal stress in a piezoelectric layer can be written as [2]

$$\sigma_{\varsigma pi} = E_{pi}\varepsilon_{\varsigma 0}\eta - e_{\eta\varsigma}E_{\eta}, \tag{2.1}$$

where $e_{\eta\varsigma}$ is the corresponding piezoelectric coefficient. The electric field E_{η} is known as the negativ gradient of the electric potential and is determined here as the difference ratio $-\frac{Voltage}{d}$. E_{pi} is the Young's modulus of the piezoelectric material. The strain

$\varepsilon_{\varsigma 0}$ on the centerline can be written in the form

$$\varepsilon_{\varsigma 0} = -\frac{1}{R^2}\frac{d^2v}{d\varphi^2} - \frac{1}{R}\gamma \tag{2.2}$$

which is a basic relation for the prismatic planar curvilinear beams with a constant radius of curvature R. Here, φ denotes the angle at center of the beam measured from the clamped end and is used as an independent variable. The variables v and γ represent the deflection of the beam in the η-direction and the twist of the beam, respectively.

There is a stress in the midlayer which can be calculated in the knowledge of the strain. By denoting the Young modulus of that layer by E_{Al}, we can write

$$\sigma_{\varsigma Al} = E_{Al}\varepsilon_{\varsigma 0}\eta. \tag{2.3}$$

Integrating $\sigma_{\varsigma pi}$ and $\sigma_{\varsigma Al}$ over the corresponding surfaces $A_{pi} = bd$ and $A_{Al} = bc$, respectively, we determine the global bending moment in the ξ-direction

$$M_\xi = \int\limits_{A_{pi}} \sigma_{\varsigma pi}\eta dA + \int\limits_{A_{Al}} \sigma_{\varsigma Al}\eta dA. \tag{2.4}$$

Performing the integrations in (2.4) and substituting (2.2), the moment can be expressed as

$$M_\xi = I_{\xi E}\left(-\frac{1}{R^2}\frac{d^2v}{d\varphi^2} - \frac{1}{R}\gamma\right) - 2e_{\eta\varsigma}E_\eta b\left(\frac{d}{2} + \frac{c}{2}\right), \tag{2.5}$$

where $I_{\xi E}$ is the bending rigidity of the cross-section. In a later section we wish to bring $I_{\xi E}$ into sharper focus. Similarly, the torsional mode is also taken into account and the torsional moment can be formulated as

$$M_\zeta = I_{cG}\left(\frac{1}{R}\frac{d\gamma}{d\varphi} - \frac{1}{R^2}\frac{dv}{d\varphi}\right), \tag{2.6}$$

where I_{cG} is the torsional rigidity of the-cross section.

3. Solution of the differential equation system

Since there are no external mechanical loads in our study, M_ξ and M_ζ are zero. Hence, (2.5) and (2.6) can be rearranged

$$I_{\xi E}\left(\frac{1}{R^2}\frac{d^2v}{d\varphi^2} + \frac{1}{R}\gamma\right) = -2e_{\eta\varsigma}E_\eta b\left(\frac{d}{2} + \frac{c}{2}\right), \tag{3.1a}$$

$$I_{cG}\left(\frac{1}{R}\frac{d\gamma}{d\varphi} - \frac{1}{R^2}\frac{dv}{d\varphi}\right) = 0\,. \tag{3.1b}$$

Equations (3.1a) and (3.1b) are coupled with the variables v and γ. We can eliminate the torsional rigidity in (3.1b) and by solving the differential equation

$$\frac{d\gamma}{d\varphi} - \frac{1}{R}\frac{dv}{d\varphi} = 0 \tag{3.2}$$

with the boundary conditions $v\,|_{\varphi=0}\ = 0$, $\gamma\,|_{\varphi=0}\ = 0$, we get the relation between the twist and the deflection of the beam as

$$\gamma = \frac{v}{R}. \tag{3.3}$$

Substituting (3.3) into (3.1a), an inhomogenous second order differential equation is obtained

$$\frac{d^2v}{d\varphi^2} + v = -2\frac{R^2}{I_{\xi E}} e_{\eta\varsigma} E_\eta b \left(\frac{d}{2} + \frac{c}{2}\right). \tag{3.4}$$

The solution is looked for in the form $v = A\cos\varphi + B\sin\varphi - 2\frac{R^2}{I_{\xi E}} e_{\eta\varsigma} E_\eta b \left(\frac{d}{2} + \frac{c}{2}\right)$, which is a sum of the general solution of the homogenous equation and of a particular solution. Here A and B are constants. The related boundary conditions are $v\,|_{\varphi=0}\ = 0$, $\frac{dv}{d\varphi}\,|_{\varphi=0}\ = 0$. The solution of the boundary value problem can be written in the form

$$v(\varphi) = 2\frac{R^2}{I_{\xi E}} e_{\eta\varsigma} E_\eta b \left(\frac{d}{2} + \frac{c}{2}\right)(\cos\varphi - 1)\ , \tag{3.5}$$

which is a trigonometric function of φ.

4. **Determining the bending rigidity of a composite beam**

Let us consider the symmetrical cross-section of the beam (Figure 3). The inertia of the extreme layers has the form

$$I_{\xi_{extr}} = b\left(\frac{d^3}{3} + \frac{d^2c}{2} + \frac{dc^2}{4}\right), \tag{4.1}$$

where b is the width of the cross-section, c means the thickness of the midlayer, d is the thickness of each extreme layer. The inertia of the midlayer can be written as

$$I_{\xi_{mid}} = \frac{bc^3}{12}. \tag{4.2}$$

To obtain the bending rigidity of the cross-section, (4.1) and (4.2) are coupled to each other through the Young's moduli of the midlayer E_{Al} and of the extreme layers E_{pi}, respectively

$$I_{\xi E} = 2I_{\xi_{extr}}E_{pi} + I_{\xi_{mid}}E_{Al}. \tag{4.3}$$

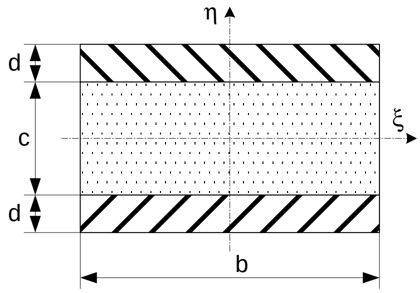


Figure 3. Cross section of a 3-layered beam

If each layer has the same size, ie., their thicknesses are the same $\left(c = d = \frac{s}{3}\right)$ (Figure 1), then the expression of rigidity is reducible and has the form

$$I_{\xi E} = \frac{bs^3}{12}\left(\frac{26}{27}E_{pi} + \frac{1}{27}E_{Al}\right). \tag{4.4}$$

With the same consideration, (3.5) needs alteration. Substituting (4.4) into (3.5) and considering that $\left(\frac{d}{2} + \frac{c}{2}\right) = \frac{s}{3}$, we obtain the deflection of the neutral elastic fibre in the form

$$v(\varphi) = \frac{72R^2 e_{\eta\varsigma} E_\eta}{s\left(26E_{pi} + E_{Al}\right)}(\cos\varphi - 1). \tag{4.5}$$

The twist of the beam obtained by substituting (4.5) into (3.3) has the form

$$\gamma(\varphi) = \frac{72Re_{\eta\varsigma} E_\eta}{s\left(26E_{pi} + E_{Al}\right)}(\cos\varphi - 1). \tag{4.6}$$

Let us notice that the width of the beam b plays no role in the solution.

5. Solution of the problem by the FEM

The general form of the functional related to the problem is presented in [3]. In what follows we simplify it with respect to the quasi-static motion, hence D'Alambert's term is missing. Further no external mechanical loads are applied to the beam. All the electromechanical devices work as actuators, hence the electric potential is not varied, and the electric charge sources are omitted. The simplified functional can be expressed as

$$\int\limits_V \left(\delta\mathbf{u}^T\partial_{\mathbf{u}}^T\right)\mathbf{C}\left(\partial_{\mathbf{u}}\mathbf{u}\right)dV + \int\limits_V \left(\delta\mathbf{u}^T\partial_{\mathbf{u}}^T\right)\mathbf{e}\left(\partial_\Phi\Phi\right)dV = 0\ , \tag{5.1}$$

where $\mathbf{C}$ is the elasticity matrix, $\mathbf{e}$ is the piezoelectric matrix, $\mathbf{u}$ is the vector of the mechanical displacements and is chosen as the primary variable, Φ is the electric potential applied on the piezoelectric surfaces. The differential operators ∂_u, ∂_Φ generate the mechanical strains and electric fields, respectively. Superscript T denotes the transpose of objects. The mechanical displacements in an arbitrary element e are written in the form

$$\mathbf{u}^e\left(x, y, z\right) = \sum_n^p N_n^e\left(\overline{\xi}, \overline{\eta}, \overline{\zeta}\right)\mathbf{q}_n^e\ , \tag{5.2}$$

where N_n^e is the approximation fuction of the n-th node and $\mathbf{q}_n^e$ denotes the nodal displacement vector. The mapping between the local system $\left\{\overline{\xi}, \overline{\eta}, \overline{\zeta}\right\}$ and the global system $\{x, y, z\}$ is performed by using the blending functions [4] or the approximation functions. No confusion should arise if we define the local system by the coordinates $\overline{\xi}, \overline{\eta}, \overline{\zeta}$, because those are not identical with ξ, η, ζ presented in the previous sections. Although we can exactly describe the geometry of the elements in a parametric form, since their curvatures are given circular segments, or given straight line segments (Figure 4), the isoparametric mapping is also applied to control this technique.

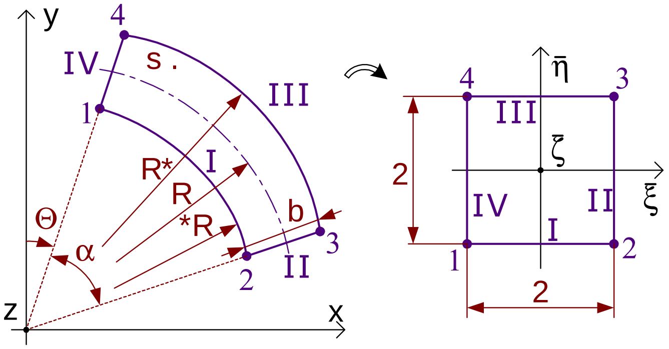


Figure 4. Mapping of an arbitrary circle-sided quadrilateral element

The coordinates x and y are obtained by

$$x = \frac{1-\bar{\eta}}{2} x_I\left(\bar{\xi}\right) + \frac{1+\bar{\eta}}{2} x_{III}\left(\bar{\xi}\right) \ , \tag{5.3a}$$

$$y = \frac{1-\bar{\eta}}{2} y_I\left(\bar{\xi}\right) + \frac{1+\bar{\eta}}{2} y_{III}\left(\bar{\xi}\right) \tag{5.3b}$$

with the functions associated with the edges I and III as

$$x_I\left(\bar{\xi}\right) =^* R \sin\left(\Theta + \frac{\alpha}{2} + \bar{\xi}\frac{\alpha}{2}\right) \ , \tag{5.4a}$$

$$x_{III}\left(\bar{\xi}\right) = R^* \sin\left(\Theta + \frac{\alpha}{2} + \bar{\xi}\frac{\alpha}{2}\right) \ , \tag{5.4b}$$

where ${}^*R$ and R^* are the radii of the inner and outer boundary curves, α is the angle at centre, Θ is the preangle, which is measured from zero to the IV^{th} edge. Taking the same geometry for each element (α) we write for the e-th element (as $e = 0, 1, 2, ..., m$), $\Theta = e\alpha$. From Figure 4 it is seen that the radius of the centerline R and the width of the element b are determined by the inner and outer radii as

$$R = \frac{{}^*R + R^*}{2} \ , \tag{5.5a}$$

$$b = \frac{R^* -^* R}{2} \ . \tag{5.5b}$$

Substituting (5.4a) and (5.4b) into (5.3a) and using (5.5a) and (5.5b) we get

$$x = \left(R + \bar{\eta}\frac{b}{2}\right) \sin\left[\frac{\alpha}{2}\left(2e + 1 + \bar{\xi}\right)\right] . \tag{5.6}$$

Similarly, the coordinate y can also be determined in the same manner, by using the corresponding blending functions belonging to the edges I and III as

$$y_I\left(\bar{\xi}\right) =^* R \cos\left(\Theta + \frac{\alpha}{2} + \bar{\xi}\frac{\alpha}{2}\right) , \tag{5.7a}$$

$$y_{III}\left(\bar{\xi}\right) = R^* \cos\left(\Theta + \frac{\alpha}{2} + \bar{\xi}\frac{\alpha}{2}\right) \tag{5.7b}$$

while y is obtained by substituting (5.7a) and (5.7b) into (5.3b) and has the form

$$y = \left(R + \overline{\eta}\frac{b}{2}\right) \cos\left[\frac{\alpha}{2}\left(2e + 1 + \overline{\xi}\right)\right] . \tag{5.8}$$

Since coordinates $\overline{\zeta}$ and z are independent of all $\overline{\xi}$, $\overline{\eta}$, x, y coordinates, the mapping between them is written as $z = \frac{s}{6}\overline{\zeta}$. The derivatives of approximation functions are said to be in the global system $\{x, y, z\}$, although $N_n^e\left(\overline{\xi}, \overline{\eta}, \overline{\zeta}\right)$ can be derived in the local $\left\{\overline{\xi}, \overline{\eta}, \overline{\zeta}\right\}$ system. Hence, the Jacobian matrix is introduced as

$$\mathbf{J}^e = \begin{bmatrix} \frac{\partial x}{\partial \overline{\xi}} & \frac{\partial y}{\partial \overline{\xi}} & \frac{\partial z}{\partial \overline{\xi}} \\ \frac{\partial x}{\partial \overline{\eta}} & \frac{\partial y}{\partial \overline{\eta}} & \frac{\partial z}{\partial \overline{\eta}} \\ \frac{\partial x}{\partial \overline{\zeta}} & \frac{\partial y}{\partial \overline{\zeta}} & \frac{\partial z}{\partial \overline{\zeta}} \end{bmatrix}^e = \begin{bmatrix} \frac{\alpha}{2}\left(R + \overline{\eta}\frac{b}{2}\right)\cos\alpha_e & -\frac{\alpha}{2}\left(R + \overline{\eta}\frac{b}{2}\right)\sin\alpha_e & 0 \\ \frac{b}{2}\sin\alpha_e & \frac{b}{2}\cos\alpha_e & 0 \\ 0 & 0 & \frac{s}{6} \end{bmatrix}, \tag{5.9}$$

where $\alpha_e = \frac{\alpha}{2}\left(2e + 1 + \overline{\xi}\right)$. The mapping is not degenerate if the inverse mapping exists. A necessary condition for this is that the determinant of the Jacobian matrix is positive. This criterium is satisfied if $R > \frac{b}{2}$.

In this paper the well known polynomial approximation of the geometry is also applied. The curved sides (I and III) of the element are written with the same Legendre polynomials which are used to approximate the mechanical displacement. This is the technique of the isoparametric mapping and detailed in the literature [5].

A p-extensional computational technique allows us to estimate the exact elasticity solution and the relative error [5]. The error $\|e\|_{(p)}$ is estimated on a nonrefined mesh as

$$\|e\|_{(p)} \leq k N_{(p)}^{-\beta}, \tag{5.10}$$

where p is the order of the approximation, $N_{(p)}$ is the number of unknowns, k and β are positive constants. By defining the estimated error difference at (p), $(p-1)$, $(p-2)$, β can be eliminated

$$\lg\|e\|_{(p)} - \lg\|e\|_{(p-1)} = -\beta(\lg N_{(p)} - \lg N_{(p-1)}), \tag{5.11a}$$

$$\lg\|e\|_{(p-1)} - \lg\|e\|_{(p-2)} = -\beta(\lg N_{(p-1)} - \lg N_{(p-2)}). \tag{5.11b}$$

By introducing the exponent

$$Q = \frac{\lg\left(\frac{N_{(p-1)}}{N_{(p)}}\right)}{\lg\left(\frac{N_{(p-2)}}{N_{(p-1)}}\right)} \tag{5.12}$$

we rewrite (5.11a) and (5.11b) in the form

$$\lg\left(\frac{\|e\|_{(p)}}{\|e\|_{(p-1)}}\right) = \lg\left(\frac{\|e\|_{(p-1)}}{\|e\|_{(p-2)}}\right)^Q . \tag{5.13}$$

By using the relationship between the energy norms of the exact elasticity solution $\|u\|$ and of the FEM-approximated solution at a given polynomial degree $\|u_{FEM}\|_{(p)}$

the error can be expressed as

$$\|e\|^2_{(p)} = \|u\|^2 - \|u_{FEM}\|^2_{(p)}. \tag{5.14}$$

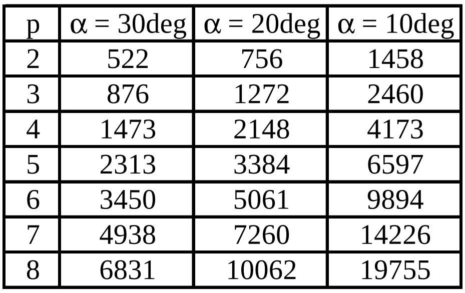
Introducing (5.14) into (5.13) we get

$$\frac{\|u\|^2 - \|u_{FEM}\|^2_{(p)}}{\|u\|^2 - \|u_{FEM}\|^2_{(p-1)}} = \left(\frac{\|u\|^2 - \|u_{FEM}\|^2_{(p-1)}}{\|u\|^2 - \|u_{FEM}\|^2_{(p-2)}} \right)^Q , \tag{5.15}$$

which makes it possible to determine the norm of the exact elasticity solution $\|u\|$ and of the real error $\|e\|_{(p)}$ by using (5.14) again.

6. **Numerical example**

Figure 1 illustrates the structure under investigation. Each layer has the thickness of $\frac{s}{3} = 1\ mm$, the radius of the curvature of the beam is $R = 60\ mm$, the width of the beam is $b = 8\ mm$. The angle at the center of the whole domain is 180°. The structure is built up by using 3D higher-order solid elements. Figure 5 shows the table with the unknown parameters as a functions of the polynomial order and the angle at the center of the element. The number of elements is 18 ($\alpha = 30°$), 27 ($\alpha = 20°$) and 54 ($\alpha = 10°$).



p	α = 30deg	α = 20deg	α = 10deg
2	522	756	1458
3	876	1272	2460
4	1473	2148	4173
5	2313	3384	6597
6	3450	5061	9894
7	4938	7260	14226
8	6831	10062	19755

Figure 5. Table of the number of the unknown parameters in each case of meshing

For approximation Legendre polynomials are used from the classical truncated space, for numerical integration 10 Gaussian points are chosen in all directions of the 3D space. Besides the blending function method the traditional isoparametric mapping is performed with the same number of approximation functions as used in the displacement approximation and the results are compared in this regard also. Both piezoelectric actuators are supplied with the constant electric potential of DC 100 Voltage and the deformation of the beam consists of two forms. First, it deflects, second, its cross-sections rotate. Figure 6 shows the displacements in a direction perpendicular to the $\{\xi, \zeta\}$ plane vs. φ. In Figure 7 the twist of the beam is plotted vs. φ. Solid lines belong to linear beam theory, dashed lines to FEM result at p=8. ($\alpha = 10°$). The results agree well. Relative discrepancies between analytical and FEM results at $\varphi = 180°$ are less than 0.1 %.

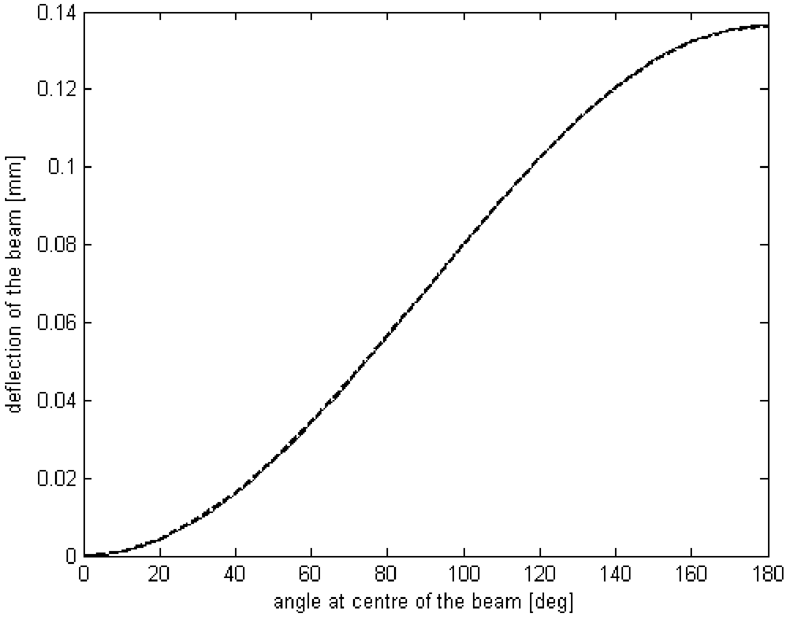


Figure 6. Bending of the beam along the centerline

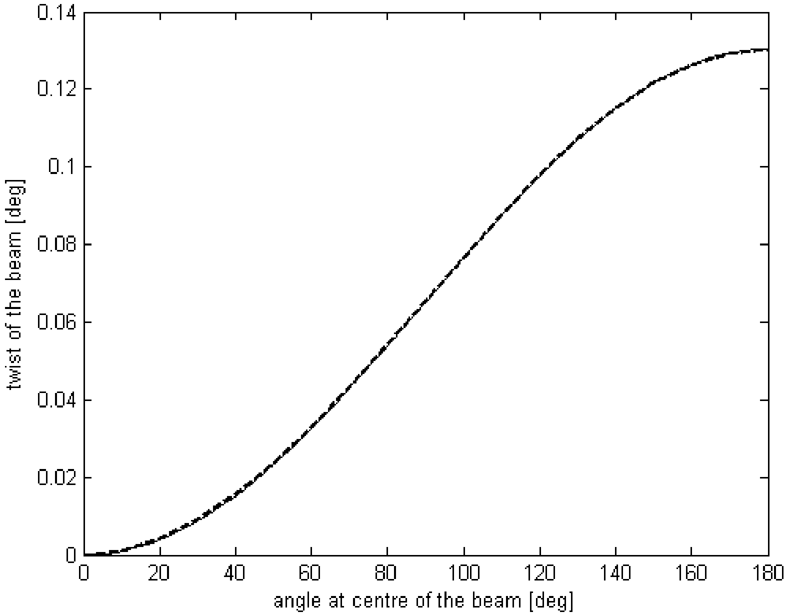


Figure 7. Rotation of the cross sections along the centerline

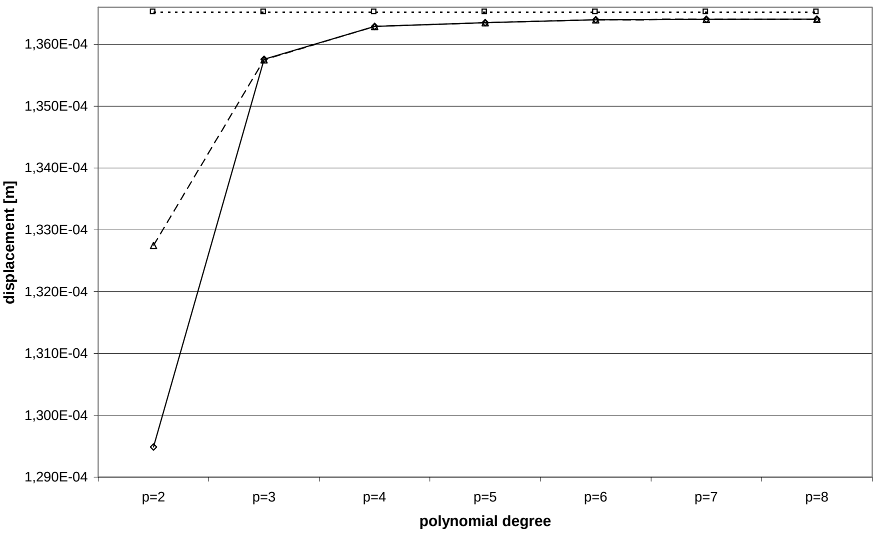


Figure 8a. Displacement of the last cross section vs. polynomial degree at $\alpha = 30°$

Displacement in the η-direction of the last cross-section is computed for different mesh and mapping. Figures 8a,b and c depict the results. Dotted lines show the beam-theory result. Dashed lines belong to the FEM isoparametric mapping, solid lines stand for the FEM blending function mapping.

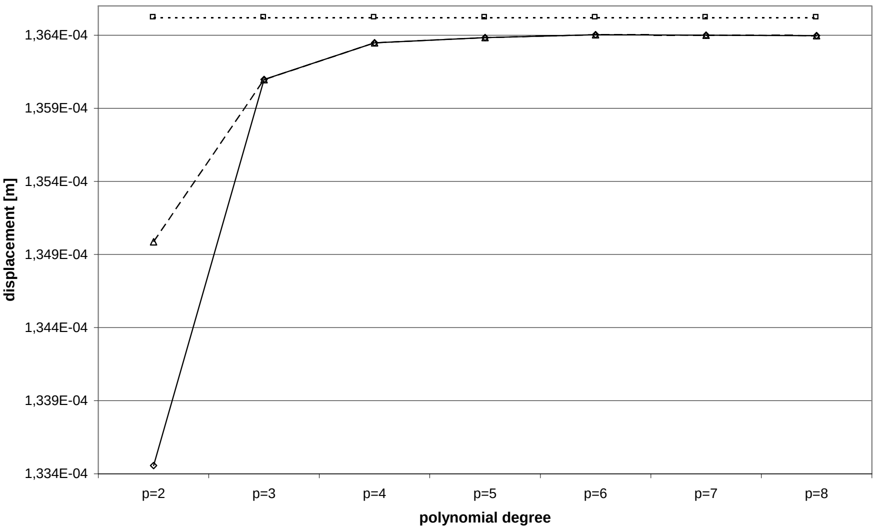


Figure 8b. Displacement of the last cross section vs. polynomial degree at $\alpha = 20°$

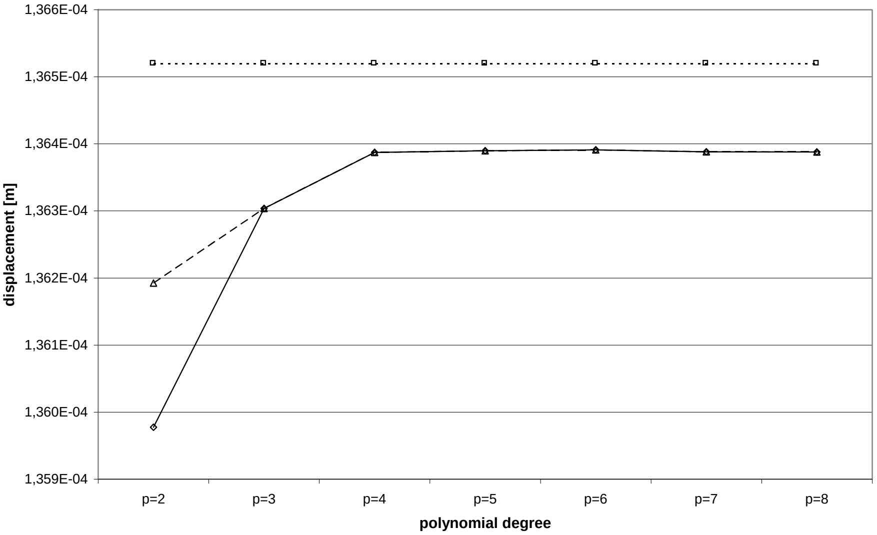


Figure 8c. Displacement of the last cross section vs. polynomial degree at $\alpha = 10°$

Figure 9 illustrates the convergence of the solution. The strain energy norm vs. order of the polynomial approximation at $\alpha = 30°$, $20°$ and $10°$ and with two different geometry mapping is drawn with dotted, dashed and solid lines, respectively. The strain energy norm of the exact elasticity solution is drawn with dashdot line.

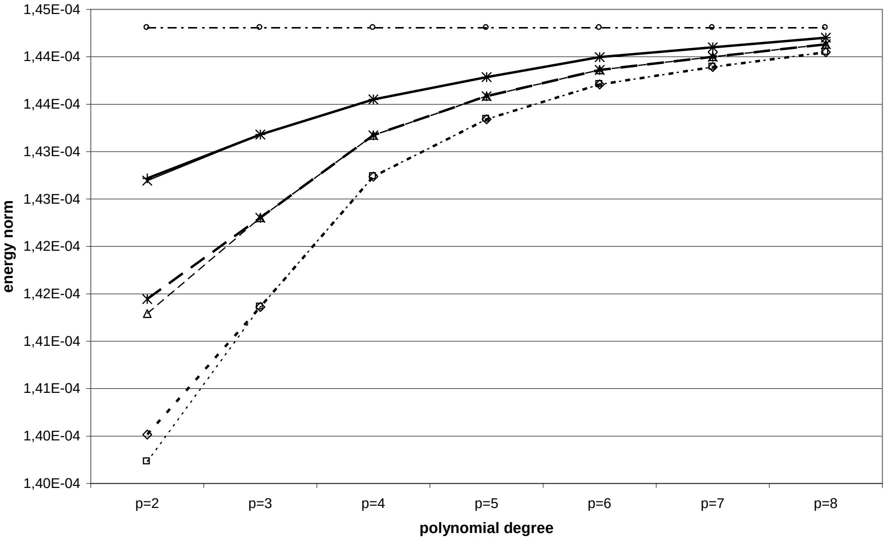


Figure 9. Strain energy norm at $\alpha = 30°$, $20°$, $10°$

It can be clearly seen that the difference between the results of the blending function method and isoparametric mapping is not significant at a higher polynomial degree. The sharp discrepancy in the displacement values arises only at $p = 2$. The relative error in energy norm is also presented for uniform $p = 2, 3, 4, 5, 6, 7, 8$ distribution. Since there is no significant difference in energy norm at $p = 6, 7, 8$ between the two types of mapping, the error estimation is performed by using the values computed with the isoparametric mapping. In Figure 10a, b, c in semilog scale the error decreases in each case ($\alpha = 30°$, $20°$, $10°$), from which it follows that the convergence is exponential.

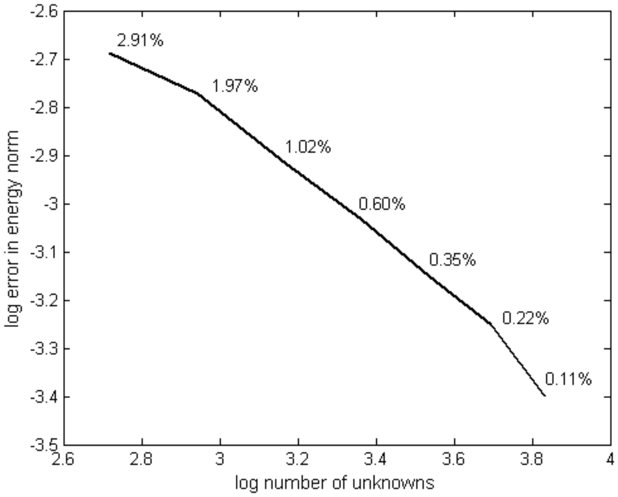


Figure 10a. Error of uniform p-distribution on nonrefined mesh ($\alpha = 30°$)

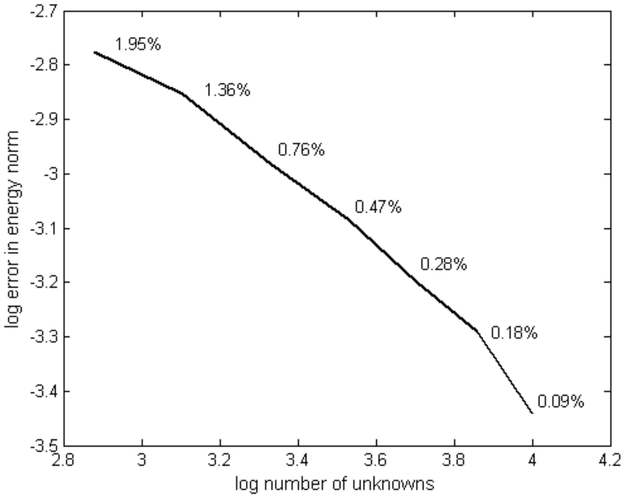


Figure 10b. Error of uniform p-distribution on nonrefined mesh ($\alpha = 20°$)

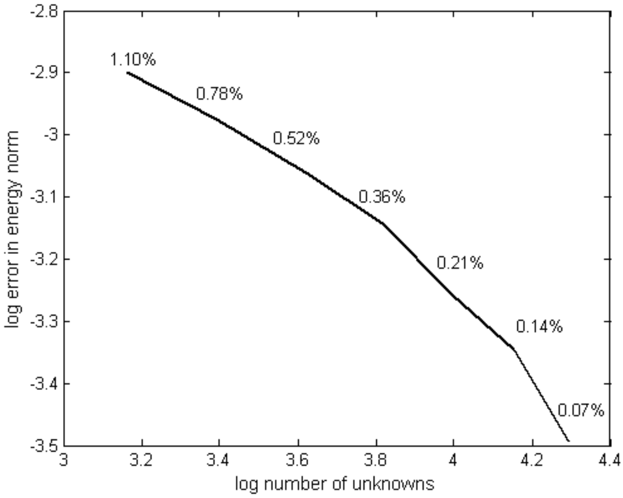


Figure 10c. Error of uniform p-distribution on nonrefined mesh ($\alpha = 10°$)

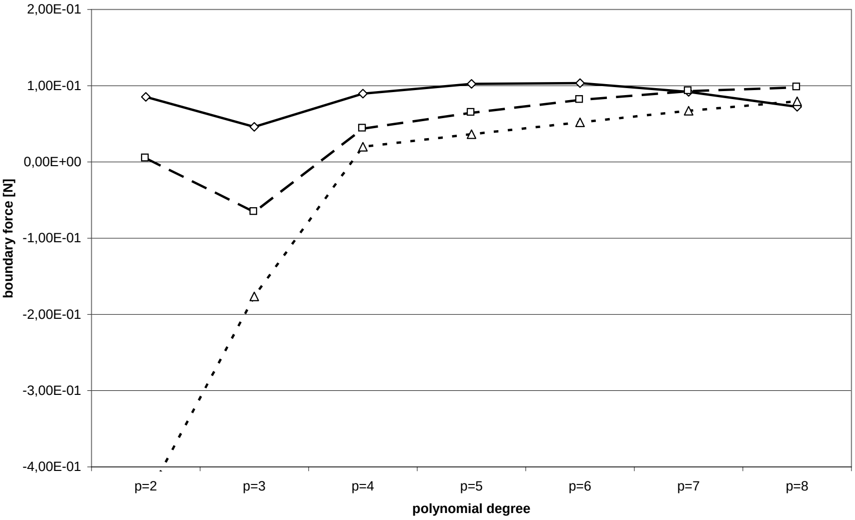


Figure 11a. Dynamical boundary condition (η-force of the last cross section) computed with the blending function method

The satisfaction of the dynamic boundary condition is also investigated. The force on the last cross-section in the η-direction is the most important variable here. This is computed for uniform $p = 2, 3, 4, 5, 6, 7, 8$ distribution in three different meshes ($\alpha = 30°$, $20°$, $10°$) and with two mapping techniques. Figure 11a illustrates the force for blending function mapping and Figure 11b for isoparametric mapping.

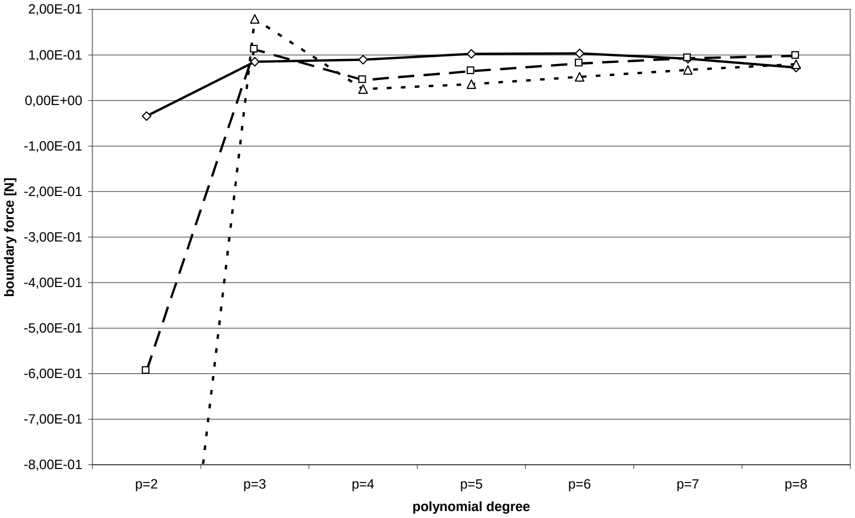


Figure 11b. Dynamical boundary condition (η-force of the last cross section) computed with isoparametric mapping

In both diagrams the dotted line belongs to $\alpha = 30°$, the dashed line to $\alpha = 20°$, and the solid line to $\alpha = 10°$. The diagram shows that at $p = 2, 3$ values the results oscillate. If $p = 4, 5, 6$ the force gets a little value, which is about $0.1N$, although at higher p values the results decrease and tend to zero. The procedure is fast if the mesh is smooth and slow with large elements. For example at $p = 8$ if $\alpha = 30°$ the force tends to its maximum, if $\alpha = 20°$, it has already reached the maximum, and if $\alpha = 10°$, the descent can be clearly seen.

7. Conclusions and remarks

In this paper the deformation of a multilayered piezoelectric semiring was presented. An electric field was applied in PZT patches bonded to the semicircular planar prismatic beam, which resulted in the deflection of the beam as it was expected and additionally in the rotation of its cross-sections. These deformations were determined analytically by using the linear theory of 3D curved beams and compared to the approximate solution obtained by the use of FEM. Solid p-extensional elements were used, due to the fact that the problem was non-symmetrical 3D. At 2, 3, 4, 5, 6, 7, 8 polynomial degrees of approximations the relative error was computed, which justifies the convergence of the numerical solution. It was presented that both the deflection and the twist of the beam were cosine functions of the angle of the arc of the centerline. It was also shown that the radius of curvature plays an important role, since it creates a proportional relationship between the deflection and the twist of the beam, but the width of the beam does not influence the result. Following this train of thought further conclusions can be drawn. Comparing the theoretical solution, which was a simple 1D case (ie. $e_{\eta\varsigma}$, E_{pi}), with the 3D FEM solution the bending around direction η is not significant, since it was neglected in the analytical solution but FEM added the energy of this mode to the other ones. It is remarkable that the change of thickness of piezoelectric layers can be also negligible in thin patches. These discrepancies cause the differences between theoretical and FEM solutions. Comparing the two mapping techniques, it is clearly seen that in lower p-distribution the blending function method gave less adequate results than the isoparametric mapping but in higher modes both techniques are equally applied, because the rigid-body rotations tend to zero [4].

Acknowledgement. This work has been supported by grant OTKA Hungarian Fund for Scientific Research No. T030096, the Hungarian Ministry of Education No. FKFP 0504/2000. The support is gratefully acknowledged.

References

1. PONOMARJOV, S. D.: *Calculus of Strength in Mechanical Engineering, Vol. 7*, Műszaki Könyvkiadó Budapest, 1966, 89-109. (in Hungarian)
2. PREUMONT, A.: *Vibration Control of Active Structures*, Kluwer Academic Publishers, 1997.
3. BERGER, H., CAO, X., KÖPPE, H. and GABBERT, U.: *Finite element based analysis of adaptive structures*, Proc. of the Euromech 373 Colloquium: Modelling and Control of Adaptive Mechanical Structures, (1998), 73-81.
4. SZABÓ, B. and BABUSKA I.: *Finite Element Analysis*, John Wiley & Sons Inc., 1991, 107-112, 97-99, 239-241.
5. PACZELT, I.: *Finite Element Method in Engineering Practice, Vol. 1*, University Press, Miskolc, 1999, 421-445. (in Hungarian)

APPENDIX

3D piezoelectric matrix (PIC151):

$$\mathbf{e}^T = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 12 \\ 0 & 0 & 0 & 0 & 12 & 0 \\ -9.6 & -9.6 & 15.1 & 0 & 0 & 0 \end{bmatrix} Cm^{-2}.$$

The 1D $e_{\eta\varsigma}$ piezoelectric coefficient is $-12.75\ Cm^{-2}$. The 3D piezoelectric elasticity matrix:

$$\mathbf{C}_{pi} = \begin{bmatrix} 10.76 & 6.312 & 6.385 & 0 & 0 & 0 \\ 6.312 & 10.76 & 6.385 & 0 & 0 & 0 \\ 6.385 & 6.385 & 10.04 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.962 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.962 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2.224 \end{bmatrix} \times 10^{10} Nm^{-2}.$$

The Young modulus in 1D is $E_{pi} = 5.943 \times 10^{10}\, Nm^{-2}$. The isotropic aluminium material has a Young's modulus $E_{Al} = 6.865 \times 10^{10}\, Nm^{-2}$ and Poisson's ratio: $\nu = 0.34$.

PENALTY-REGULARIZATION OF A DISSIPATIVE VIBRO-IMPACTING SYSTEM

Ingolf Müller and Peter Vielsack
Institute of Mechanics, University of Karlsruhe
D-76131 Karlsruhe, Germany
mechanik@ifm.uni-karlsruhe.de

[Received: June 18, 2002]

Abstract. The so-called penalty method in FE-calculation regularises the strong contact conditions by introducing contact stiffnesses and damping in order to reduce the mathematical effort. The problem, however, lies in an appropriate choice of the values of parameters for these artificially introduced springs and dampers. The principal problem of regularisation, however, can be studied for simple rigid body systems. As an example, two neighbouring physical pendulums with different natural frequencies are treated. During the motion sudden impacts and states of permanent contact interchange with states of separated motions of the two pendulums.

The first step in the consideration comprises the calculation of a semi-analytical reference to classify the properties of the motion with regard to the main features of the non-linear system's response. The results are verified by experimental investigations in the next step. Finally, the system is regularised by the penalty method and integrated by NEWMARK's method. This procedure needs three unknown numbers, two regularisation parameters and a time step. Their correct choice depends on detailed information from the experimental results for each type of motion.

Keywords: impact, penalty method, non-linear oscillation

1. **Introduction**

The presence of damage in sandwich materials, in particular delaminations between adjacent laminae, degrade severely the mechanical behaviour of a structure. A vibration-based non-destructive damage identification needs a suitable model to capture the non-linear phenomena of the oscillation [3]. Experimental investigations show that oscillations of delaminated structures are dominated by impacts [4]. They occur when separated parts of the structure come into contact during the motion. Each contact gives rise to an impact, which leads to energy dissipation. The actual available mechanical model with minimal DOF is based on an elastic beam with lumped masses and a simple law of impact [3]. The integration of this non-smooth dynamic system leads to a sequence of smooth systems, whose analytical solutions are known. They must be patched together at those times when irregularities due to contact occur [6]. This simple model captures the main oscillation phenomena and allows a discussion in principle of the influence of the internal dissipation due to the impacts on the non-linear system's response and the evolution of the impacts near resonance.

An improvement of the mechanical description can be expected by the utilisation of the finite element method. In order to reduce the numerical effort the regularisation of the strong contact conditions is required. The penalty method introduces contact stiffnesses and damping for regularisation [1]. Despite the fact that FE-calculations lead to oscillations with multi-degrees of freedom, the fundamental problem of an appropriate choice of the values for these artificially introduced springs and dampers can be discussed for simple rigid body systems.

As an example the forced vibrations of two neighbouring pendulums will be considered. The first step comprises the consideration of a semi-analytical reference to classify the properties of the oscillations with regard to the main features of the non-linear system's response. These results will be verified by experimental investigations in the next step. Finally, the validity of two different mechanical models for the contact, namely the classical theory of impact and the regularisation-technique for impacts, is compared.

2. The investigated system and its semi-analytical description

As an example, let us consider two neighbouring physical pendulums with different natural frequencies and different damping. The pendulums touch each other with a vanishing contact force in the equilibrium state. Vibrations are induced by a harmonic base excitation. This non-smooth dynamic system gives a first approximation for a delaminated sandwich beam [4]. The mechanical description is based on the model shown in Figure 1. It consists of two rigid bodies with different masses and different elastic suspensions and dampers at the top.

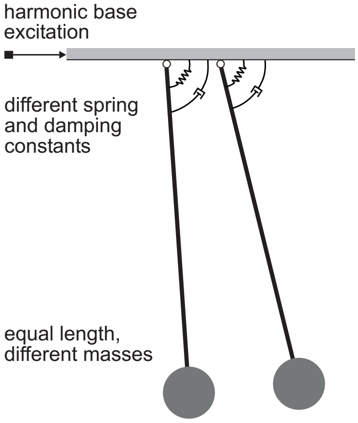


Figure 1. Mechanical model

Firstly the semi-analytical procedure for integration of the non-smooth dynamic system is considered. In this case, only the coefficient of restitution e has to be determined from experiments. Exciting the system, discontinuities of the motion due to impacts occur. This leads to sudden changes in the system's behavior at unknown separation times. The only numerical task is to find these separation times. Between two successive separation times the system is a linear one and the solutions of the equations of motion are known explicitly. Three different states must be considered (see Figure 2). In the case of a separated motion, both pendulums move independently of each other, characterised by the natural frequencies ω_1, ω_2 and the damping constants d_1, d_2, respectively. When the two pendulums come into contact, an impact occurs. In this second state the sudden impact is modelled by NEWTON's assumption with a coefficient of restitution $e = 0.5$. A third possible state is a motion in permanent contact, where the two pendulums behave as a single one with a frequency ω_3 and a damping constant d_3. All constants can be found in Table 1. They came from the real physical system under experimental investigation, considered later on. The calculation procedure is described in [4]. A detailed discussion is therefore omitted. For a better understanding only some hints are needed. All results are given in a non-dimensional representation. The non-dimensional time $\tau = \frac{t}{\omega_1}$ is refers the lowest natural frequency. The values ξ_1 and ξ_2 are non-dimensional displacements of the end masses of the pendulums (a motion in permanent contact gives $\xi_1 = \xi_2$). The corresponding velocities are ${\xi_1}'$ and ${\xi_2}'$. A transition from a separated motion to a motion in permanent contact theoretically leads to a sequence of infinite numbers of impacts with time intervals tending to zero. The beginning of a motion in permanent contact is therefore defined by a small threshold ${\xi_2}' - {\xi_1}' \leq 0.002$ to avoid numerical problems. The frequency ratio $\eta = \frac{\Omega}{\omega_1}$ indicates the frequency of excitation. In the following only stationary system's responses are considered. Depending on the frequency of excitation η the system's response shows a broad variety of bifurcated motions. The POINCARÉ-section method is used to collect samples of stationary

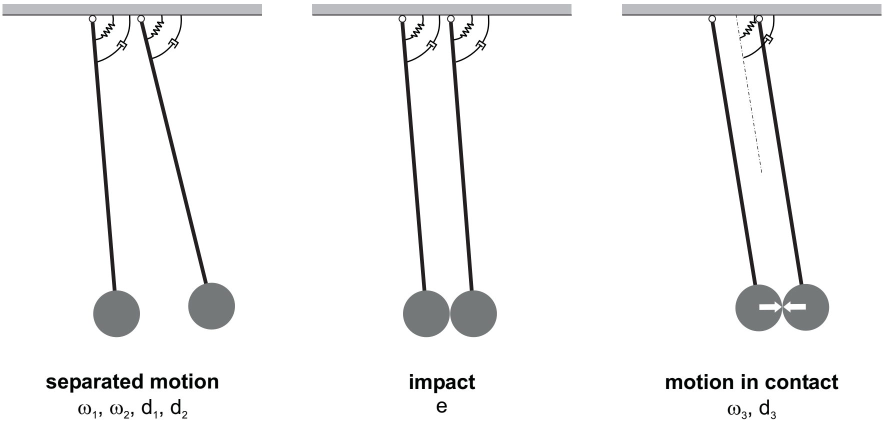


Figure 2. Possible states of motion

	Natural frequency ω_i [1/s]	**Damping d_i [-]**
Pendulum 1 (separate)	$\omega_1 = 1.00$	$d_1 = 0.0033$
Pendulum 2 (separate)	$\omega_2 = 2.41$	$d_2 = 0.0120$
Two pendulums (fixed connection)	$\omega_3 = 1.80$	$d_3 = 0.0330$

Table 1. Parameters for natural frequencies and viscous damping

responses of the displacements ξ_i, which can be assembled into a bifurcation diagram (Figure 3). The typical feature of the bifurcation diagram is an alternation of regions of irregularity and windows of periodic responses. As an example, only four typical kinds of motion will be considered (Figure 3, sections [A] - [D]) by their phase plots in Figure 4. In the vicinity of the frequency $\eta = 1.08$ (case [A]) the oscillation is non-bifurcated. As can be seen in Figure 4 [A], this type of motion contains multiple impacts in one period and a phase of permanent contact. Section [B] (Figure 3 [B]), taken at a frequency of excitation $\eta = 1.70$, shows a quasi-periodic motion. The case of quasi-periodic motions can be seen in the bifurcation diagram (Figure 3, section [B]) as widening of the lines to stripes of different widths. Increasing η to the range of $\eta = 2.76$, the system's response changes to a non-bifurcated one (Figure 3 [C]). The corresponding phase plot (Figure 4 [C]) shows one impact in one period. Finally, in the region of $\eta = 3.25$ (Figure 3, [D]) a period-doubling exists.

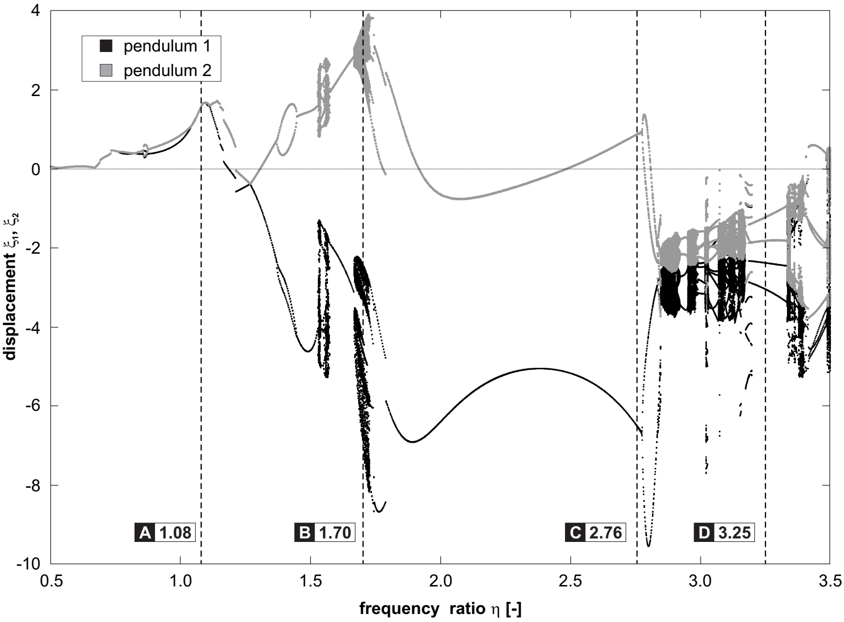


Figure 3. Bifurcation diagram

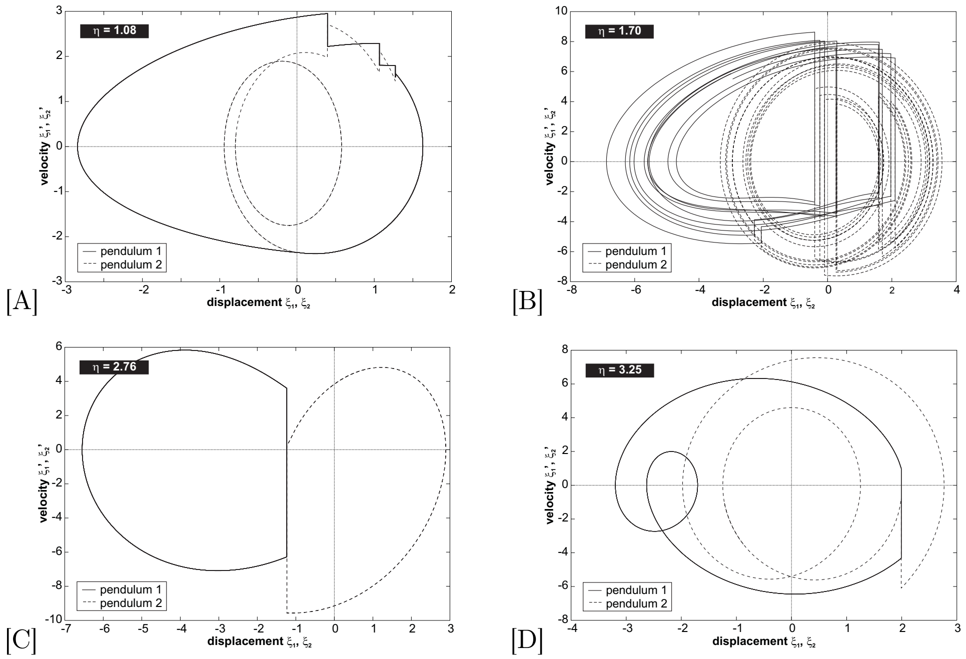


Figure 4. Phase plots of four typical kinds of oscillations

Despite this broad variety of motions, the following treatment will be restricted from now on to two typical kinds of oscillations, which are of major interest in the investigation of a delaminated beam. These are a motion caused by an excitation $\eta = 1.08$ leading to multiple impacts and permanent contact (Figure 4 [A]) and a motion caused by $\eta = 2.76$ leading to one sudden impact in one response period (Figure 4 [C]).

3. Experimental confirmation of the semi-analytical results

The experimental equipment is shown in Figure 5. It consists of two physical pendulums of length 618 mm with the vibrational parameters given in Table 1. A shaker induces vibrations as an adjustable harmonic base excitation. The amplitude of excitation is kept constant at $1.07mm$. The above mentioned excitations $\Omega = 1.08\frac{1}{s}$ and $\Omega = 2.76\frac{1}{s}$ (Figure 4, [A] and [C]) are chosen for an experimental verification of the semi-analytical results. Opto-electronical displacement sensors give the absolute positions x_i characterising the response of the system. The frequency of excitation Ω can be monitored. A contact sensor controls the opening and closing of an electric circuit and gives information about contact or no contact. Considering the stationary system's response in form of time-displacement plots of about two excitation periods, the chosen cases of the frequency of harmonic base excitation ($\eta = 1.08$ and $\eta = 2.76$) show an excellent agreement between experimental and numerical results (Figure 6, upper pictures). It must be noticed, however, that the experimental time-displacement

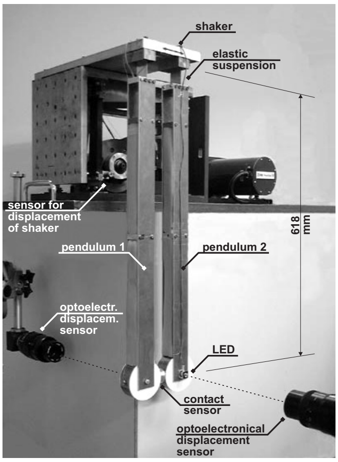

shaker
elastic suspension
sensor for displacement of shaker
pendulum 1
pendulum 2
618 mm
optoelectr. displacem. sensor
LED
contact sensor
optoelectronical displacement sensor

Figure 5. Experimental equipment

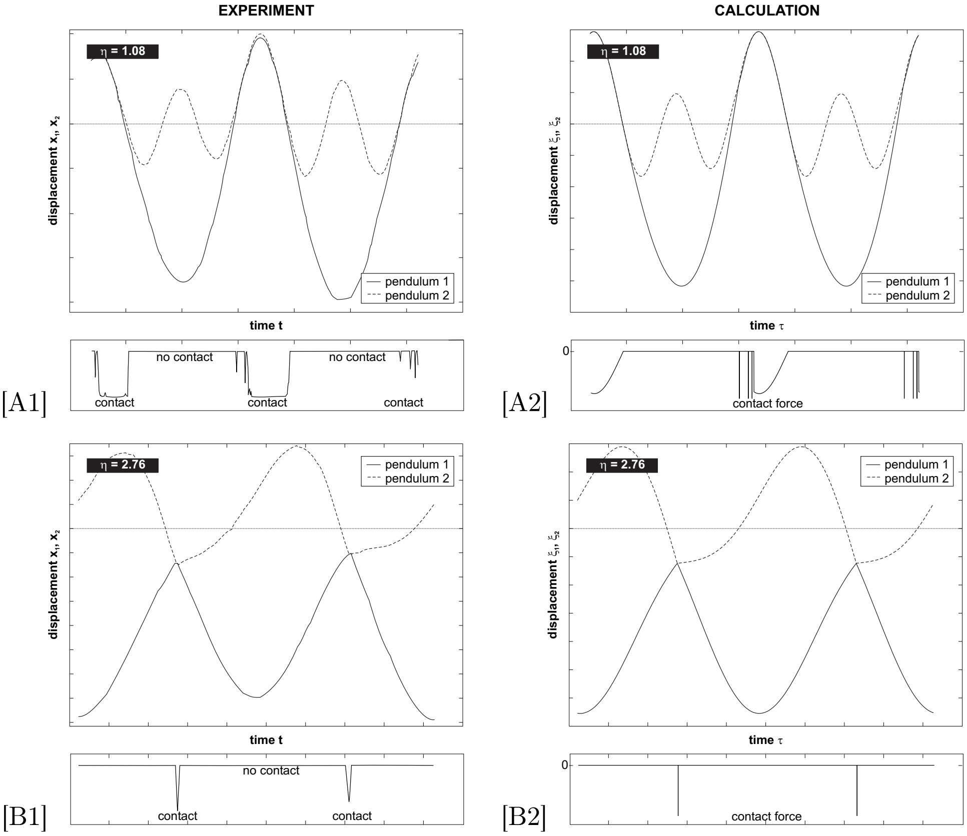


Figure 6. Comparison of stationary displacement and contact force versus time from experiment [A1], [B1] and calculation [A2], [B2]

he numerical results in a non-dimensional form. In addition, the change of the contact force in the same time domain (Figure 6, lower pictures) confirms the correctness of all calculations. Here, the experiments only give information about contact and no contact, whereas the numerical result shows the course of the contact force. The case $\eta = 1.08$ clearly shows multiple impacts with decreasing time intervals leading to a motion with permanent contact.

4. **Penalty regularisation**

Regularisation of the strong contact conditions leads to a smoothing of the points of discontinuity. In contrast to the semi-analytical procedure the number of DOF does not change in the regularized system in all partial states. This allows a fast numerical integration.

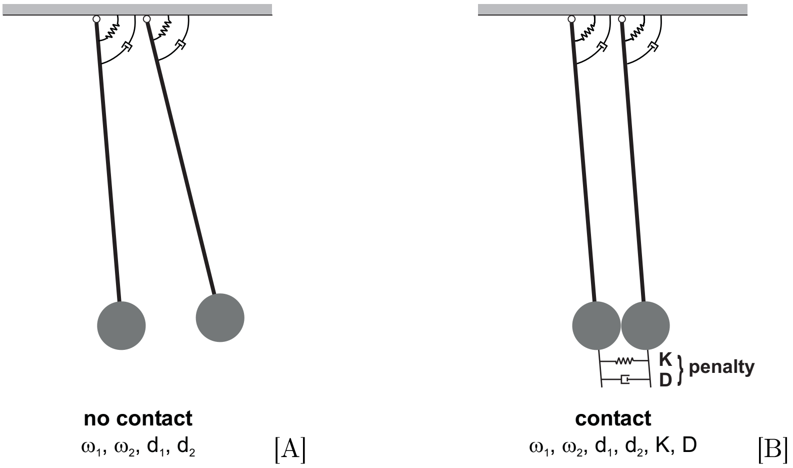


Figure 7. Mechanical systems of the different partial states
[A] no contact, [B] contact

As illustrated in Figure 7, only two states exist, namely a motion with or without contact. The state of motion without contact (Figure 7 [A]) is kept unaltered (cp. Figure 1) compared to the preceding system. In the case of contact, which means a vanishing or negative relative displacement $\xi_2 - \xi_1$, a contact spring with stiffness K is added to the basic system (Figure 7 [B]). An additionally introduced viscous damper D captures the dissipation of impact, comparable to the coefficient of restitution e. Introducing the ratio $\kappa = \frac{\omega_2}{\omega_1}$, a non-dimensional representation of the equations of motion for both states is given in Figure 8. Starting at the state without contact the mathematical description consists of two non-coupled equations. If contact occurs, the equations are linked by penalty stiffness K and damping D. It is obvious that the non-linearity of the regularized system only consists on the mutual change of the system from a free motion of both pendulums to a common motion in contact and vice versa. In addition to the fact that the number of DOF is constant in time, the second advantage is the simplification of the switching conditions. Only a control of the relative displacement decides about a transition from one state to the next. The semi-analytical procedure controls the contact force in a state of common motion.

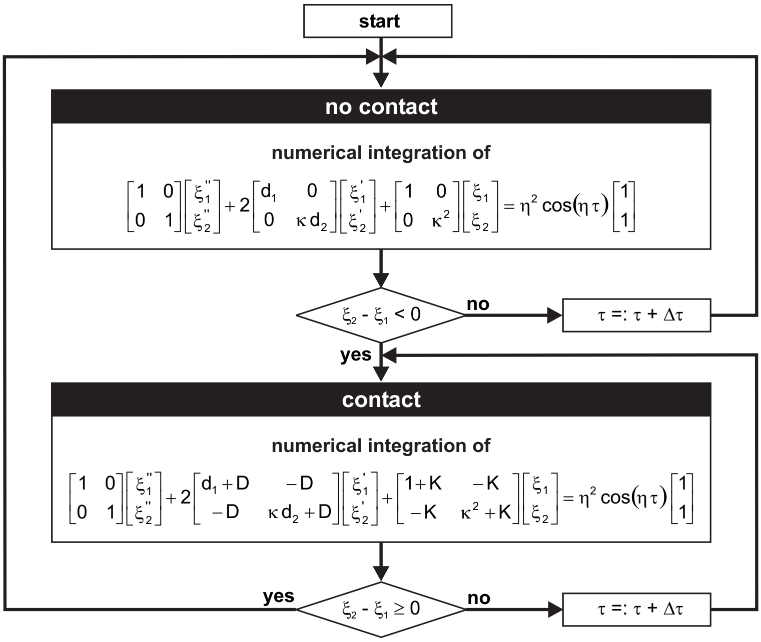

start
no contact
numerical integration of
$\begin{bmatrix}1&0\\0&1\end{bmatrix}\begin{bmatrix}\xi_1''\\\xi_2''\end{bmatrix}+2\begin{bmatrix}d_1&0\\0&\kappa d_2\end{bmatrix}\begin{bmatrix}\xi_1'\\\xi_2'\end{bmatrix}+\begin{bmatrix}1&0\\0&\kappa^2\end{bmatrix}\begin{bmatrix}\xi_1\\\xi_2\end{bmatrix}=\eta^2\cos(\eta\tau)\begin{bmatrix}1\\1\end{bmatrix}$
$\xi_2 - \xi_1 < 0$
no
yes
$\tau =: \tau + \Delta\tau$
contact
numerical integration of
$\begin{bmatrix}1&0\\0&1\end{bmatrix}\begin{bmatrix}\xi_1''\\\xi_2''\end{bmatrix}+2\begin{bmatrix}d_1+D&-D\\-D&\kappa d_2+D\end{bmatrix}\begin{bmatrix}\xi_1'\\\xi_2'\end{bmatrix}+\begin{bmatrix}1+K&-K\\-K&\kappa^2+K\end{bmatrix}\begin{bmatrix}\xi_1\\\xi_2\end{bmatrix}=\eta^2\cos(\eta\tau)\begin{bmatrix}1\\1\end{bmatrix}$
yes
$\xi_2 - \xi_1 \geq 0$
no
$\tau =: \tau + \Delta\tau$

Figure 8. Scheme for switching from one state to the other

The fundamental disadvantage, however, lies in an appropriate choice of the artificially introduced constants K and D for a certain type of motion, because these values do not represent real physical or mechanical parameters. As will be shown in the following, a correct choice of K and D needs a reference. This can be achieved by matching the input data with experimental information. In the present case, the solutions of the semi-analytical procedure can be taken. In general, low values of K give wrong results caused by the poorly satisfied contact condition, followed by a strong penetration of the subsystems. The opposite case of a large contact stiffness K gives rise to a stiff set of equations leading to problems of integration. Furthermore, the choice of the parameters K and D depends on each other and requires a correct adjustment.

The numerical integration needs a time step $\Delta\tau$. NEWMARK's method, commonly used in FE-method, is applied taking $\alpha = \frac{1}{2}$ and $\beta = \frac{1}{4}$. Therefore, the time step is constant. This fact can lead to severe errors and even totally wrong responses [5]. The reason lies in the inaccurate determination of the transition points. Therefore, the time step should be as small as possible. In the following, two kinds of motions with excitation $\eta = 1.08$ and $\eta = 2.76$ (Figure 6) will be investigated to show the problems in choosing the three numbers K, D and $\Delta\tau$.

4.1. **Motion with a state of permanent contact:** $\eta = 1.08$**.** The time step $\Delta\tau$ refers the non-dimensional periodical time $T = \frac{2\pi}{\eta}$. A time step $\Delta\tau = \frac{T}{2000}$ is taken and kept constant. Now, only contact stiffness K and damping D can be chosen freely. Figure 9 compares the phase plots, which are obtained by regularization using the sets of parameters ([A] $K = 2500$, $D = 8$ and [B] $K = 100$,$D = 50$), with the semi-analytical result [C]. As evident from Figure 9, the set of penalty parameters in case [A] captures the phenomena of oscillation given by the exact solution (case [C]). The choice of $K = 100$, $D = 50$ (case [B]) yields a completely different system behavior, which is far from reality.

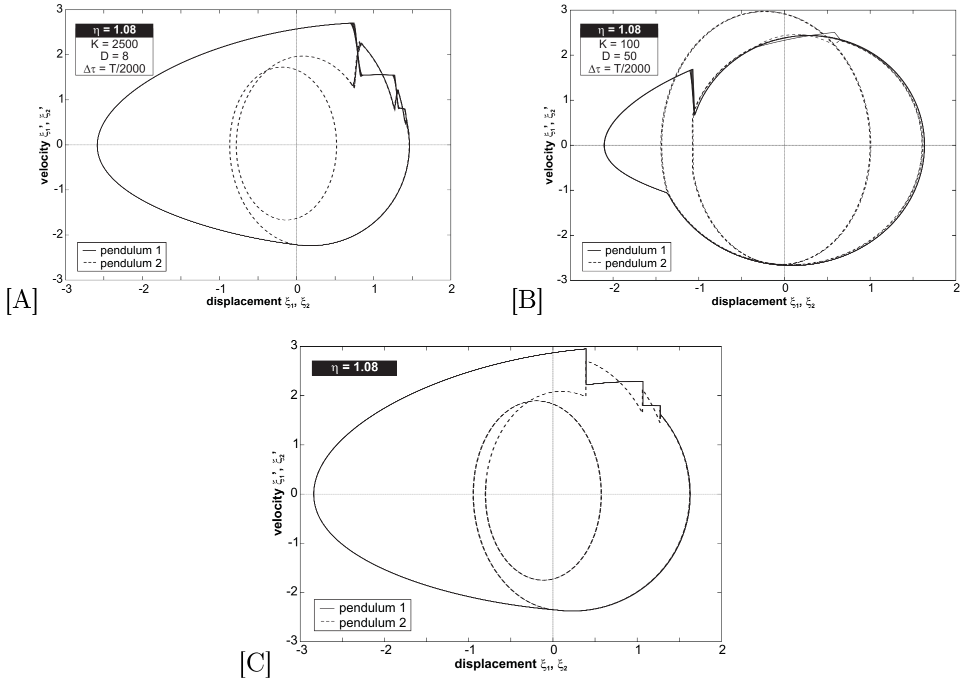


Figure 9. Comparison of numerical results gained by using different penalty parameters [A], [B] and the correct (semi-analytical) result [C]

As a conclusion it is evident that detailed information is needed with regard to the expected type of motion to determine the parameters K, D and $\Delta\tau$. The basic conditions are the properties of the response (bifurcated / non-bifurcated, periodic motion / quasiperiodic motion), number of impacts and instants of impacts in a response period. Remembering the broad variety of different kinds of oscillations shown in the bifurcation diagram (Figure 3), it must be emphasized here that the solution in Figure 9 [B] could be considered the correct one, if no information existed.

4.2. **Separated motion with one impact:** $\eta = 2.76$**.** At the beginning of the investigation the same time step $\Delta\tau = \frac{T}{2000}$ as before is taken. The simplest information which is needed to determine K and D is the non-existence of bifurcations. The

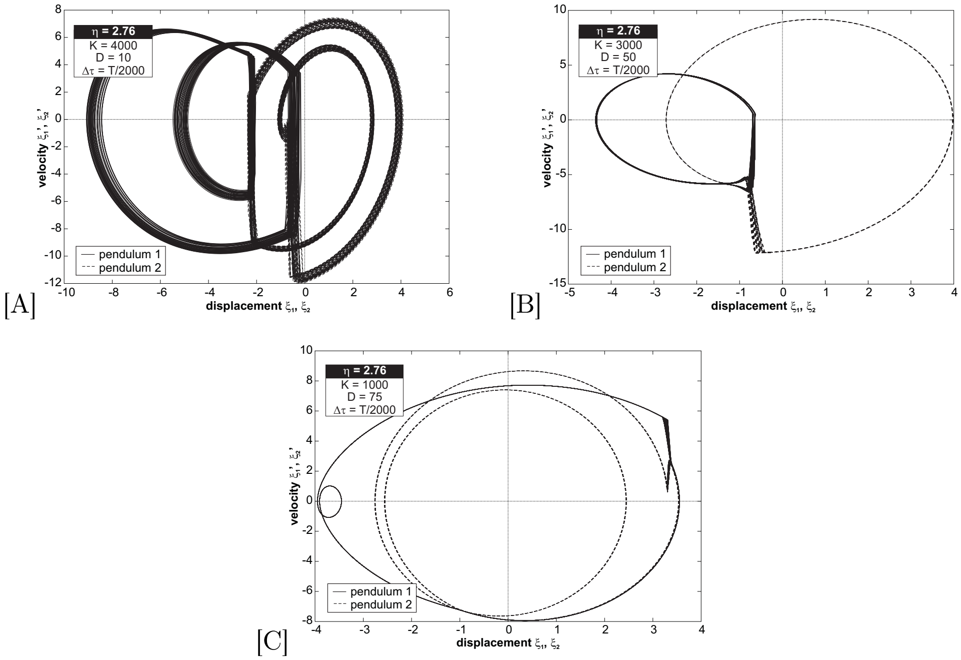


Figure 10. Examples of phase plots for bifurcated and non-bifurcated motions

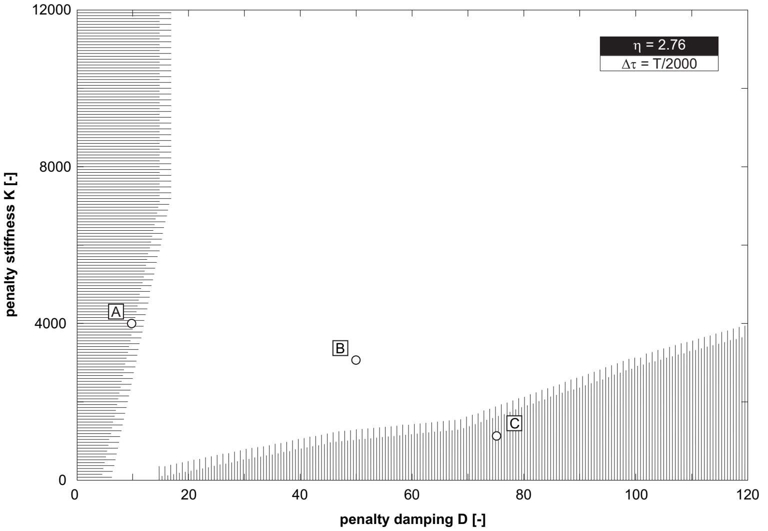


Figure 11. Parametric plane for sets K and D showing regions of bifurcated and non-bifurcated oscillations

responses for three different sets of parameters are shown in Figure 10. Two of them (Figure 10, [A] and [C]) exhibit a bifurcated motion. A systematical variation of K and D excludes the sets of K and D leading to bifurcations. This allows us to construct a parametric plane, as can be seen in Figure 11. The dashed regions in Figure 11 are out of interest. The examples of Figure 10 give three points [A], [B] and [C].

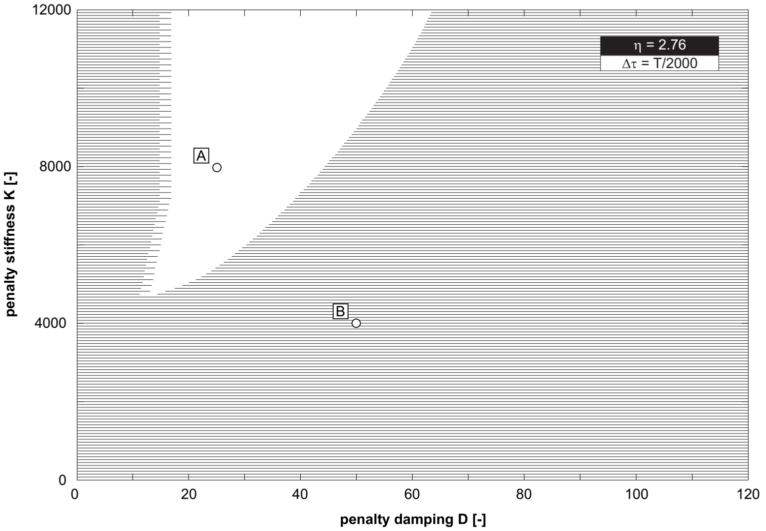


Figure 12. Updated parametric plane for sets K and D

Two examples are given in Figure 13. They correspond to the points [A] and [B] in Figure 12.

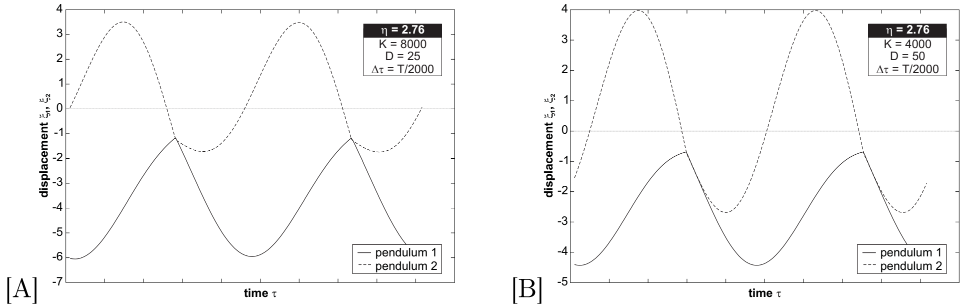


Figure 13. Time-displacement plots for responses with a [A] "sudden" impact and a [B] state of permanent contact

More detailed information is the knowledge about the number of impacts and the states of permanent contact in a period. In the present example ($\eta = 2.76$) one sudden impact occurs in one response period. The regularisation by the penalty method does not allow the reproduction of sudden impacts. That means that sudden impacts are modelled by a short interval of permanent contact. As an example, less than 30 time steps $\Delta\tau = \frac{T}{2000}$ are assumed to describe a "sudden" impact. This assumption diminishes the region of possible values K and D in an updated parametric plane (Figure 12).

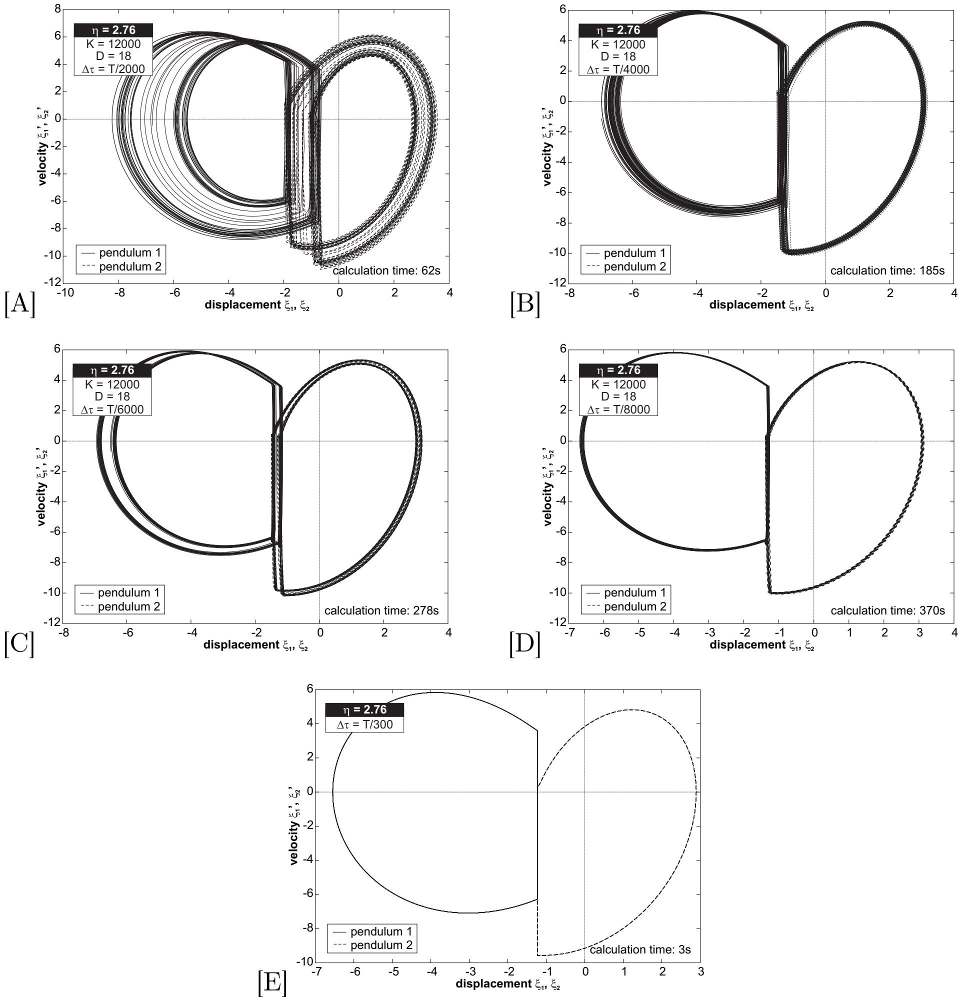


Figure 14. Stationary phase plots for various time steps $\Delta\tau$ [A]-[D] compared to the semi-analytical result [E]

A further improvement of the values for K and D can be achieved by consideration of the instant of impact during a response period. This procedure demands a comparative work and is not executed here.

Finally, the problem of the choice of a sufficiently small step $\Delta\tau$ for an orbital stable solution [6] must be considered. Assuming now $K = 12000.0$ and $D = 18.0$ according to the previous investigations, stationary phase plots are computed with different time steps $\Delta\tau$. Figure 14 contains the results for time steps $\Delta\tau = \frac{T}{2000}$, $\Delta\tau = \frac{T}{4000}$, $\Delta\tau = \frac{T}{6000}$ and $\Delta\tau = \frac{T}{8000}$. In all cases 2050 excitation periods are calculated, but only the last 50 are plotted. As predicted, the larger values of $\Delta\tau$ cannot capture the transition times with sufficient accuracy. Permanent numerical disturbances due to this systematic error give rise to quasi-periodic responses. As already mentioned, such kinds of motion are possible when remembering the bifurcation diagram (Figure 3). Only the known reference solution qualify them to be wrong.

Summarizing the facts, the required three numbers for the regularization are given by a penalty stiffness $K = 12000.0$, a penalty damping $D = 18.0$ and time steps $\Delta\tau = \frac{T}{8000}$.

5. **Conclusions**

The semi-analytical procedure for integration of non-smooth dynamic contact problems leads to a sequence of smooth systems, whose solutions must be patched together at times when irregularities due to contact occur. In order to reduce the extensive mathematical effort, the penalty method regularizes the strong contact conditions by introducing contact stiffness and contact damping.

The regularized system keeps a constant number of DOF's in all partial states and allows a fast numerical integration by the usual methods. A smoothing of the points of discontinuity is obtained. Additionally, the regularization by the penalty method leads to a simplification of switching conditions for the transition to another state of motion.

The problem, however, lies in the appropriate choice of the values for penalty parameters K and D for each type of motion. For a correct determination of K and D a reference is required, which is given in experimental investigations or semi-analytical results. Without information about the expected motion, a decision is not possible, whether the chosen numbers for K and D are right or wrong. The reference results contain the information needed for the choice of the penalty parameters, which can be obtained by consideration of the motion properties - bifurcated or non-bifurcated motion, number and instant of impacts. Treating a new type of motion, a new validation of the values K and D by the reference is required.

Recapitulating the results of the influence of time steps $\Delta\tau$ shows that the choice of K and D is not independent of steps $\Delta\tau$. A high precision of the results needs an immense numerical effort.

References

1. Engleder, T., Vielsack, P. and Schweizerhof, K.: *F.E.-regularisation of non-smooth vibrations due to friction and impacts*, Computational Mechanics, **28**, (2002), 162-168.

2. Holmes P. J.: *The dynamics of repeated impacts with a sinussoidally vibrating table*, Journal of Sound and Vibration, **84**, (1982), 173-189.

3. Müller, I., Schmieg, H. and Vielsack, P.: *Non-smooth forced oscillations of a delaminated sandwich beam*, PAMM - Proceedings in Applied Mathematics and Mechanics, **2**, (2003), 286-287.

4. Vielsack, P.: *A vibro-impacting model for the detection of delamination*, Journal of Sound and Vibration, **253**(2), (2002), 347-358.

5. Vielsack, P.: *Orbitale Stabilität nichtglatter Bewegungen bei permanenten numerischen Störungen*, ZAMM, **79**(1), (1999), 105-108.

6. Vielsack, P. and Hartung, A.: *An example for the orbital stability of permanently disturbed non-smooth motions*, ZAMM, **79**(6), (1999), 389-397.

Journal of Computational and Applied Mechanics, Vol. 4., No. 2., (2003), pp. 189–201

PARALLEL FINITE ELEMENT DOMAIN DECOMPOSITION FOR STRUCTURAL/ACOUSTIC ANALYSIS

Duc T. Nguyen, Siroj Tungkahotara
Department of Civil and Environmental Engineering, 135 KAUF Hall
Old Dominion University, Norfolk, VA 23529, USA
dnguyen@odu.edu, toohtaah@yahoo.com

Willie R. Watson
Computational Modelling & Simulation Branch, MS 128
NASA Langley Research Center, Hampton, VA 23681, USA
w.r.watson@larc.nasa.gov

Subramaniam D. Rajan
Civil Engineering Department, ECG 252
Arizona State University, Tempe, AZ 85287, USA
s.rajan@asu.edu

[Received: July 24, 2002]

Abstract. A domain decomposition (DD) formulation for solving sparse linear systems of equations resulting from finite element analysis is presented. The formulation incorporates mixed direct and iterative equation solving strategies and other novel algorithmic ideas that are optimized to take advantage of sparsity and exploit modern computer architecture, such as memory and parallel computing. The most time consuming part of the formulation is identified and the critical roles of direct sparse and iterative solvers within the framework of the formulation are discussed. Experiments on several computer platforms using real and complex test matrices are conducted using software based on the formulation. Small-scale structural examples are used to validate the steps in the formulation and large-scale (1,000,000+ unknowns) duct acoustic examples are used to evaluate the parallel performance of the formulation. Results are presented using 64 SUN 10000, 8 SGI ORIGIN 2000 processors, and a cluster of 6 PCs (running under the Windows environment). Statistics show that the formulation is efficient in both sequential and parallel computing environments and that the formulation is significantly faster and consumes less memory than that based on one of the best available commercialized parallel sparse solvers.

Mathematical Subject Classification: 74S05,15A06
Keywords: linear algebra, sparse matrix computation, parallel computation, finite element, domain decomposition, structures, acoustics

1. Domain decomposition (DD) formulation for finite element analyses

Application of finite element analysis to engineering problems leads to the discrete equation system [1, 2]

$$[K]\{Z\} = \{S\} , \tag{1.1}$$

where S, Z are vectors of length N that contains the known nodal loads and unknown nodal quantities, respectively. Here K is a complex, nonsingular, symmetric/unsymmetric, NxN sparse matrix. Although (1.1) assumes a single loading condition (i.e., right hand side), multiple loading conditions may be treated by taking S and Z as dense matrices, so that the j^{th} column of Z corresponds to the N unknown nodal quantities associated with the loadings in the j^{th} column of S.

Using the domain decomposition concept, (1.1) is written in partitioned form

$$\begin{bmatrix} K_{BB} & K_{BI} \\ K_{IB} & K_{II} \end{bmatrix} \begin{Bmatrix} Z_B \\ Z_I \end{Bmatrix} = \begin{Bmatrix} S_B \\ S_I \end{Bmatrix} \tag{1.2}$$

where submatrices K_{IB}, K_{BI}, K_{II} and K_{BB} have dimension mxn, nxm, mxm and nxn, respectively. The interior and boundary unknowns (i.e., Z_I and Z_B) have dimensions compatible with the columns in K_{II} and K_{BB}, respectively.

Eliminating the interior unknowns from (1.2) gives

$$K_B Z_B = F_B \ , \tag{1.3}$$

where

$$K_B = K_{BB} + K_{BI}\, Q \ , \tag{1.4}$$

$$Q = -K_{II}^{-1} K_{IB} \ , \tag{1.5}$$

$$F_B = S_B + \tilde{Q} \ , \tag{1.6}$$

$$\tilde{Q} = -K_{BI} \tilde{S}_I \ , \tag{1.7}$$

$$\tilde{S}_I = K_{II}^{-1} S_I \ . \tag{1.8}$$

Here K_B is the boundary stiffness matrix for the domain, F_B is the vector of boundary forces, and the superscript (i.e., -1) denotes the matrix inverse. Efficient sparse algorithm [3]-[11] may be used to decompose sparse matrix K_{II} and solve for matrix Q in (1.5) and the vector $\tilde{S}_I$ in (1.8).

In the current DD formulation [12, 13] the computational domain is decomposed into L subdomains and K_B and F_B are synthesized by considering contributions from all subdomains. For this purpose, the discrete equation system for a subdomain (which is considered an isolated free-body) is expressed in the form (1.2)

$$\begin{bmatrix} K_{BB}^{(r)} & K_{BI}^{(r)} \\ K_{IB}^{(r)} & K_{II}^{(r)} \end{bmatrix} \begin{Bmatrix} Z_B^{(r)} \\ Z_I^{(r)} \end{Bmatrix} = \begin{Bmatrix} S_B^{(r)} \\ S_I^{(r)} \end{Bmatrix}, \qquad r = 1, 2, 3 \cdots L \tag{1.9}$$

where r refers to the r^{th} subdomain. Let $n^{(r)}$ and $m^{(r)}$ represent the number of boundary and interior unknowns of the r^{th} subdomain, respectively. The solution of (1.9) is

$$K_B^{(r)} Z_B^{(r)} = F_B^{(r)} \ , \tag{1.10}$$

$$Z_I^{(r)} = (K_{II}^{(r)})^{-1} (S_I^{(r)} - K_{IB}^{(r)} Z_B^{(r)}) \ , \tag{1.11}$$

where

$$K_B^{(r)} = K_{BB}^{(r)} + K_{BI}^{(r)} Q^{(r)} \ , \tag{1.12}$$

$$Q^{(r)} = -(K_{II}^{(r)})^{-1} K_{IB}^{(r)} \ , \tag{1.13}$$

$$F_B^{(r)} = S_B^{(r)} + \tilde{Q}^{(r)} \ , \tag{1.14}$$

$$\tilde{Q}^{(r)} = -K_{BI}^{(r)} \tilde{S}_I^{(r)} \ , \tag{1.15}$$

$$\tilde{S}_I^{(r)} = (K_{II}^{(r)})^{-1} S_I^{(r)} \ . \tag{1.16}$$

Finally, K_B and F_B may be obtained explicitly from the equations

$$K_B = \sum_{r=1}^{L} (\beta^{(r)})^T K_B^{(r)} \beta^{(r)} \ , \qquad F_B = S_B + \sum_{r=1}^{L} (\beta^{(r)})^T (Q^{(r)})^T S_I^{(r)} \ , \tag{1.17}$$

where $\beta^{(r)}$ is a Boolean transformation matrix of dimension $n^{(r)}$x$n^{(r)}$.

The sequence of steps constituting the DD formulation proposed in this paper is as follows:

(1) Decompose the large-scale finite element domain into L smaller subdomains. Algorithms and software given in [12, 13] are used for this purpose.
(2) Compute $K_{IB}^{(r)}, K_{BI}^{(r)}, K_{II}^{(r)}$, $K_{BB}^{(r)}$, $S_B^{(r)}$, and $S_B^{(r)}$ using efficient sparse assembly algorithms [3, 11].
(3) Factorize the sparse matrix $K_{II}^{(r)}$ and compute $\tilde{S}_I^{(r)}$ using (1.16) and $Q^{(r)}$ using (1.13). Algorithms and software for sparse symbolic and numerical factorization, loop unrolling techniques, equation reordering, and forward-backward solution phases ([3]-[11]) are utilized at this step.
(4) Compute $K_B^{(r)}$ and $F_B^{(r)}$ for each subdomain.
(5) Compute K_B and F_B from (1.17).
(6) Solve (1.3) using a direct or iterative solver to obtain the boundary unknowns, Z_B.
   a): Efficient parallel direct dense solvers given in [14]-[16] may be utilized at this step provided that K_B is formed explicitly.
   b): However, explicit computation of K_B is an expensive operation due to the need to perform the inner produce $K_{BI}^{(r)} Q^{(r)}$ in (1.12).
   c): Iterative solvers (such as the preconditioned conjugate gradient algorithm) [17] are therefore recommended for this step in the formulation. The use of an iterative solver eliminates the need to form K_B explicitly because each stage of the iterative solution typically requires only the product a matrix $(K_{BB} + K_{BI}^{(r)}) Q^{(r)}$ with a known vector.
(7) Finally, the solution for the interior unknowns are obtained from (1.11) by using the factorized sparse matrix $K_{II}^{(r)}$ during the forward and backward substitution phases.

The solution vectors obtained from the formulation are post-processed to obtain other quantities of interest such as stresses, strains, acoustic energy, etc. The remaining sections of this paper will focus on issues related to efficient sparse assembly procedures.

2. Simple finite element model

To facilitate the discussion a simple finite element model which consists of 4 rectangular elements with 1 degree of freedom (dof) per node and its loading condition

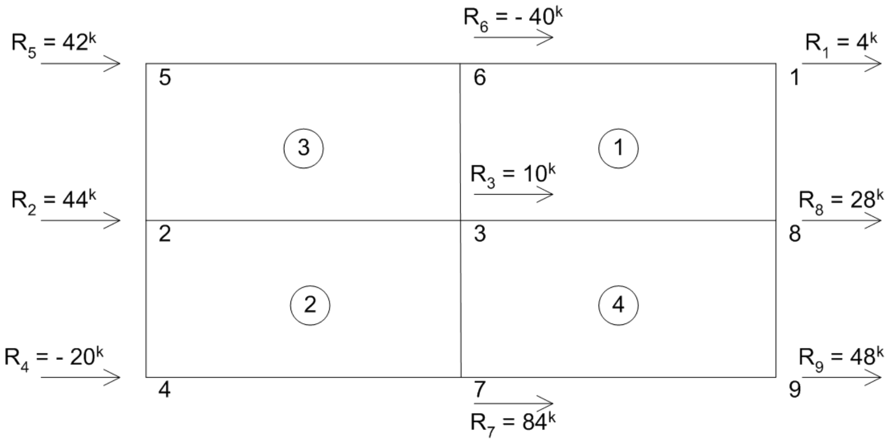


Figure 1. A simple unconstrained finite element model

(R_i) is shown in Figure 1. The global dof associated with each of the eth rectangular element in Figure 1 are given by the following "element-dof" connectivity matrix, E

$$[E] = \begin{array}{|c|c|c|c|c|c|c|c|c|l} \multicolumn{1}{c}{1} & \multicolumn{1}{c}{2} & \multicolumn{1}{c}{3} & \multicolumn{1}{c}{4} & \multicolumn{1}{c}{5} & \multicolumn{1}{c}{6} & \multicolumn{1}{c}{7} & \multicolumn{1}{c}{8} & \multicolumn{1}{c}{9} & \\ \hline x & & x & & & x & & x & & 1 \\ \hline & x & x & x & & & x & & & 2 \\ \hline & x & x & & x & x & & & & 3 \\ \hline & & x & & & & x & x & x & 4 \\ \hline \end{array} \quad . \tag{2.1}$$

The number of rows and columns in E correspond to the total number of finite elements (4 finite elements) and degrees of freedom (9 dof), in the model. The following 2 integer arrays describe the nonzero structure of E in a row oriented format

$$\{IE\}^T = \{1, 5, 9, 13, 17\} \ , \tag{2.2}$$

$$\{JE\}^T = \{3, 8, 1, 6, 7, 3, 2, 4, 5, 2, 3, 6, 7, 9, 8, 3\} \ . \tag{2.3}$$

The array IE contains the starting location of the first nonzero terms for each row of matrix E while JE contains the global dof number associated with each eth (=1,2,3,4) rectangular element. This format is called compress row format. Similarly, the transpose of the matrix E can be described by two integer arrays, IET and JET.

The "exact" numerical values for the 4x4 element stiffness matrix $K^{(e)}$ is "unimportant" at this stage of the discussions and are assumed to be given by the following formulas:

$$[K^{(1)}] = \begin{array}{|c|c|c|c|c} \multicolumn{1}{c}{3} & \multicolumn{1}{c}{8} & \multicolumn{1}{c}{1} & \multicolumn{1}{c}{6} & \\ \hline 2 & 3 & 4 & 5 & 3 \\ \hline -3 & 4 & 5 & 6 & 8 \\ \hline -4 & -5 & 6 & 7 & 1 \\ \hline -5 & -6 & -7 & 8 & 6 \\ \hline \end{array} \qquad [K^{(2)}] = \begin{array}{|c|c|c|c|c} \multicolumn{1}{c}{7} & \multicolumn{1}{c}{3} & \multicolumn{1}{c}{2} & \multicolumn{1}{c}{4} & \\ \hline 4 & 6 & 8 & 10 & 7 \\ \hline -6 & 8 & 10 & 12 & 3 \\ \hline -8 & -10 & 12 & 14 & 2 \\ \hline -10 & -12 & -14 & 16 & 4 \\ \hline \end{array} \tag{2.4}$$

$$[K^{(3)}] = \begin{array}{|c|c|c|c|c} \multicolumn{1}{c}{5} & \multicolumn{1}{c}{2} & \multicolumn{1}{c}{3} & \multicolumn{1}{c}{6} & \\ \hline 6 & 9 & 12 & 15 & 5 \\ \hline -9 & 12 & 15 & 18 & 2 \\ \hline -12 & -15 & 18 & 21 & 3 \\ \hline -15 & -18 & -21 & 24 & 6 \\ \hline \end{array} \qquad [K^{(4)}] = \begin{array}{|c|c|c|c|c} \multicolumn{1}{c}{7} & \multicolumn{1}{c}{9} & \multicolumn{1}{c}{8} & \multicolumn{1}{c}{3} & \\ \hline 8 & 12 & 16 & 20 & 7 \\ \hline -12 & 16 & 20 & 24 & 9 \\ \hline -16 & -20 & 24 & 28 & 8 \\ \hline -20 & -24 & -28 & 32 & 3 \\ \hline \end{array} \tag{2.5}$$

Note that the global row and column numbers for the e^{th} rectangular element are easily obtained from JE. For example, the global row and column numbers 7,3,2, and 4 for rectangular element 2, are contained in the 5,6,7, and 8^{th} element of JE. Further, the "simulated" element stiffness matrices, $K^{(e)}$, are unsymmetrical in value but symmetrical in locations. For example, $K^{(2)}$ has a nonzero term of 14 at location (3,4), and there is also a nonzero term of -14 at location (4,3). Following the usual finite element procedures, the system of linear unsymmetrical equations for the finite element model shown in Figure 1 can be assembled as:

$$[K]\{Z\} = \{S\}\ , \tag{2.6}$$

where

$$[K] = \sum_{e=1}^{4} [K]^{(e)}\ . \tag{2.7}$$

For example, $K_{3,3} = K^{(1)}_{3,3} + K^{(2)}_{3,3} + K^{(3)}_{3,3} + K^{(4)}_{3,3} = 2 + 8 + 18 + 32 = 60$ as indicated in (2.8)

$$[K] = \begin{array}{|c|c|c|c|c|c|c|c|c|c} \multicolumn{1}{c}{1} & \multicolumn{1}{c}{2} & \multicolumn{1}{c}{3} & \multicolumn{1}{c}{4} & \multicolumn{1}{c}{5} & \multicolumn{1}{c}{6} & \multicolumn{1}{c}{7} & \multicolumn{1}{c}{8} & \multicolumn{1}{c}{9} & \\ \hline 6 & & -4 & & & 7 & & -5 & & 1 \\ \hline & 24 & 5 & 14 & -9 & 18 & -8 & & & 2 \\ \hline 4 & -5 & 60 & 12 & -12 & 26 & -26 & -25 & -24 & 3 \\ \hline & -14 & -12 & 16 & F & F & -10 & F & F & 4 \\ \hline & 9 & 12 & & 6 & 15 & F & F & F & 5 \\ \hline -7 & -18 & -26 & & -15 & 32 & F & -6 & F & 6 \\ \hline & 8 & 26 & 10 & & & 12 & 16 & 12 & 7 \\ \hline 5 & & 25 & & & 6 & -16 & 28 & -20 & 8 \\ \hline & & 24 & & & & -12 & 20 & 16 & 9 \\ \hline \end{array} \tag{2.8}$$

$$\{S\}^T = \{4, 44, 10, -20, 42, -40, 84, 28, 48\}\ , \tag{2.9}$$

where F represents fill-in terms (shown in the upper triangular portion of matrix K only) that occur during factorization of K.

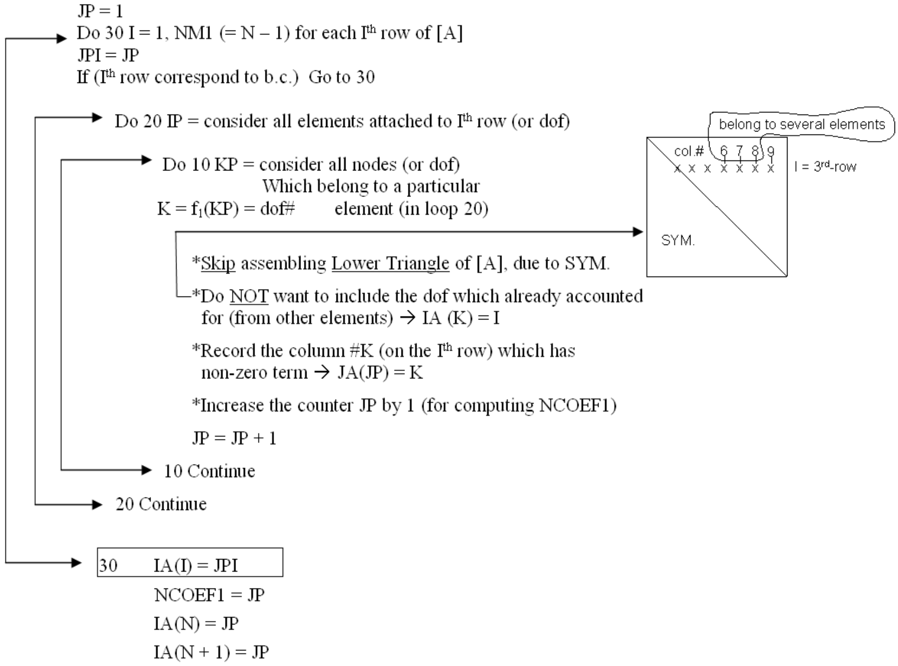
The solution to equation (2.6) is:

$$\{Z\}^T = \{1, 1, 1, 1, 1, 1, 1, 1, 1\} \tag{2.10}$$

which has been independently confirmed by the results of the computer program developed. In the upper triangular portion of K, there are 9 fill-in terms. In order to minimize the number of fill-in terms during the symbolic and numerical factorization phases, reordering algorithms [10, 13] such as multiple minimum degree (MMD), Nested-dissection (ND), and METIS are used in the DD formulation.

3. Symbolic sparse assembly for symmetrical and unsymmetrical matrices

It is useful to understand the symbolic sparse assembly for "symmetrical" matrices [3, 11] before proceeding to the unsymmetrical case. Figure 2 gives a "pseudo" Fortran coding of the symmetrical sparse assembly procedure. Only minor changes in this symmetrical assembly procedure are required to extend it to unsymmetrical matrices.



```
JP = 1
Do 30 I = 1, NM1 (= N – 1) for each I^th row of [A]
JPI = JP
If (I^th row correspond to b.c.)  Go to 30

   Do 20 IP = consider all elements attached to I^th row (or dof)

      Do 10 KP = consider all nodes (or dof)
                    Which belong to a particular
      K = f1(KP) = dof#      element (in loop 20)

         *Skip assembling Lower Triangle of [A], due to SYM.
         *Do NOT want to include the dof which already accounted
           for (from other elements) → IA (K) = I
         *Record the column #K (on the I^th row) which has
           non-zero term → JA(JP) = K
         *Increase the counter JP by 1 (for computing NCOEF1)
         JP = JP + 1

      10 Continue

   20 Continue

30      IA(I) = JPI
        NCOEF1 = JP
        IA(N) = JP
        IA(N + 1) = JP
```

Figure 2. Pseudo Fortran codes for symmetrical symbolic sparse assembly

In a symmetric matrix the "lower triangular" portion of K is identical with the "upper triangular portion." Thus, the upper triangular portion of K (neglecting fill-in terms (2.8)) can be represented in compressed row format by the following 2 integer arrays:

$$\{IA\}^T = \{1, 4, 9, 15, 16, 17, 18, 20, 21, 21\}\ , \tag{3.1}$$

$$\{JA\}^T = \{3,6,8,3,4,5,6,7,4,5,6,7,8,9,7,6,8,8,9,9\}\ , \tag{3.2}$$

where the array IA contain the starting location of the first non-zero, off-diagonal term for each row of the upper triangular portion of K. The difference between any 2 consecutive integers on the right-hand-side of (3.1) will give the number of non-zero (off-diagonal) terms in a particular row of the upper triangular portion of K. For example, IA(3)-IA(2) = 9-4 = 5. Hence, there are 5 non-zero terms (excluding the diagonal)) in row 2 of the upper triangular portion of matrix K. Additionally, JA contains the column numbers, associated with each non-zero, off-diagonal term for each row of the upper triangular portion of matrix K. Note that IA and JA arrays can also be obtained from the "pseudo" Fortran coding shown in Figure 2.

The following minor changes in the coding of Figure 2 are required to perform unsymmetrical assembly.

a): DO 30 I= 1, N (the last row will NOT be skipped)

b): Introduce a new integer array IAKEEP(N+1) which plays the role of array IA(-), for example: IAKEEP(I)= JPI .

c): Remove the IF statement in Figure 2 that skips the lower triangular portion. As a consequence of this, the original array IA (-) will contain some additional, unwanted terms.

d): The output from the "unsymmetrical" sparse assembly will be stored by IAKEEP(-) and JA(-), instead of IA(-) and JA(-) as in the symmetrical case.

4. Applications

4.1. Software. The software that is based on the parallel DD formulation presented in this paper has been developed. The parallel algorithm uses the message passing interface (MPI) for interprocess communication and is therefore highly portable. The software developed is referred to as the direct iterative parallel sparse solver (DIPSS). DIPSS (in FORTRAN) incorporates a number of lower level routines and provides options for both real and complex matrices in double precision (i.e., 64-bit arithmetic). Results are presented for symmetric matrices only. We use sparse factorization techniques presented in [3] and implement the preconditioned conjugate gradient iterative solver [17] to solve the dense system (1.3). The following three examples are used to evaluate the proposed parallel DD formulation. Performance gains are particularly evident for large problems.

4.2. Example 1– Mixed finite element types. This is a structural example for which the equation system is real and symmetric and has more than 1 finite element type. The entire finite element model is shown in Figure 3 and consists of 2-node "line" elements, 3-node "triangular" elements, and 4-node "rectangular" elements. Interior and boundary nodes are denoted by open and filled circles, respectively. This small-scale, finite element model is decomposed into 3 subdomains as indicated in Figure 4. The primary purpose of this example is to validate all intermediate and

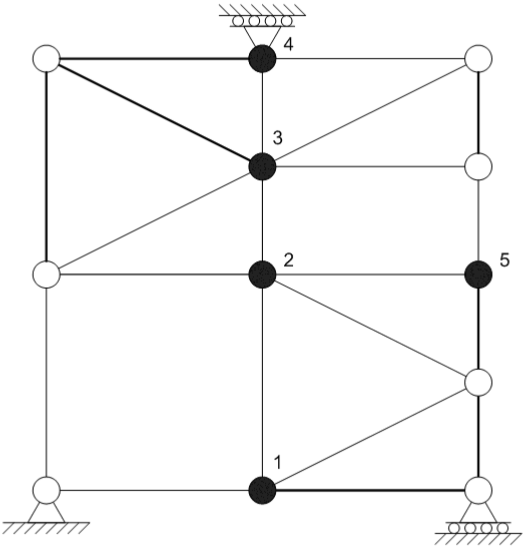


Figure 3. Finite element model with mixed elements

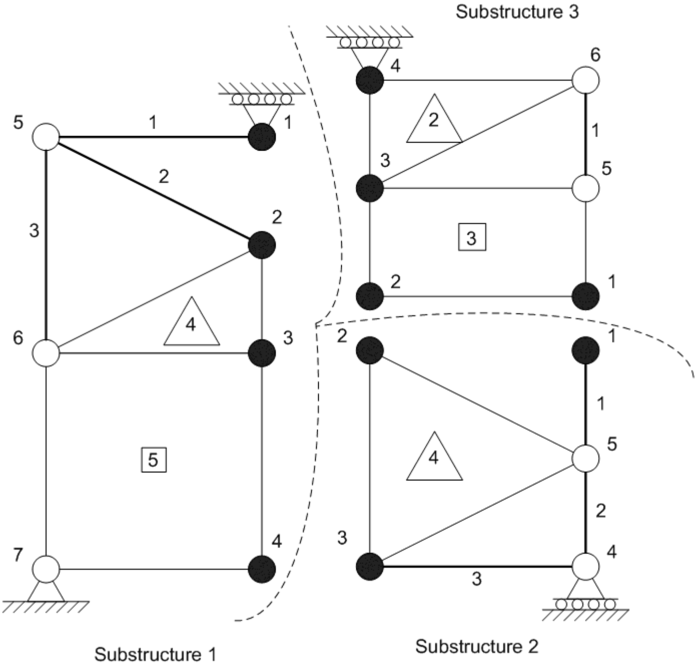


Figure 4. Decomposition of mixed model into three subdomains

final numerical results using the DIPSS software. This small-scale example was also solved in Matlab using separate software packages. Although numerical results are not presented for the sake of brevity, excellent comparison between DIPSS and the Matlab solution was obtained for this small-scale example problem.

4.3. Example 2 – Three dimensional structural bracket finite element model. The DD formulation has also been applied to solve the 3-D structural bracket problem shown in Figure 5. The finite element model contains 194,925 degrees of freedom (N=194,925) and the elements in the matrix, K, are real. Results were computed on a cluster of 1-6 personal computers (PCs) running under the Windows environment with Intel Pentium IV processors. It should be noted that the DIPSS software was not ported to the PC cluster, but the DD formulation was programmed (from scratch, in C^{++}) on the cluster.

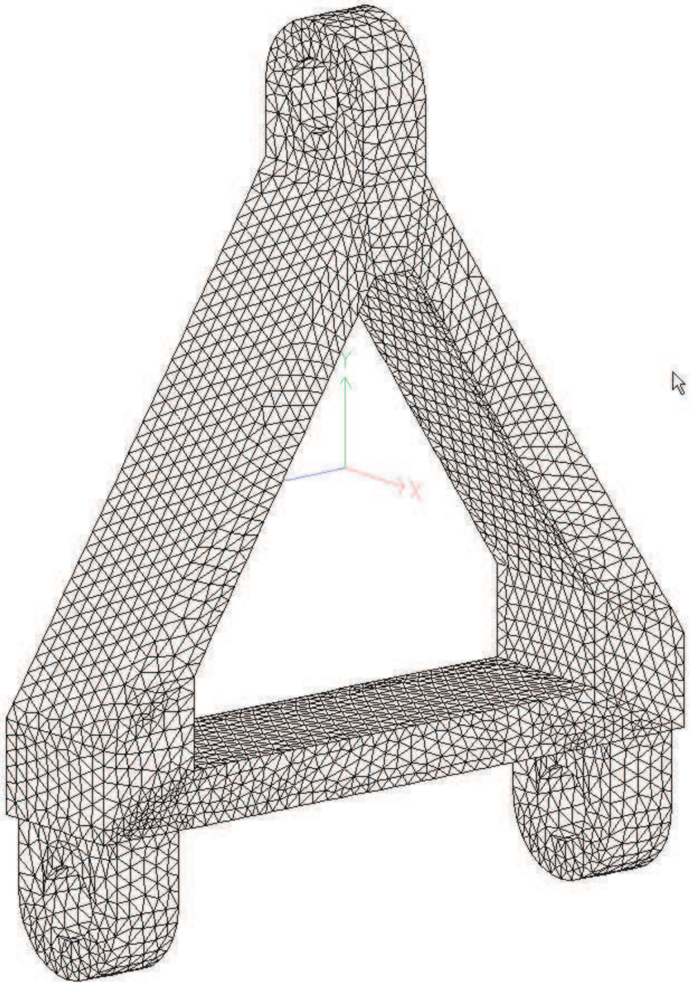


Figure 5. Finite element model for a three-dimensional structural bracket

The wallclock time (in seconds) to solve this example is documented in Table 1. A superlinear speedup factor of 10.35 has been achieved when 6 processors were used.

# of PC processors	1	2	3	4	5	6
Wallclock time (sec)	2,670	700	435	405	306	258
Speedup factor	1.00	3.81	6.14	6.59	8.73	10.35

Table 1: 3-D Structural bracket model (194,925 dofs, K real)

4.3. Example 3 – Three dimensional acoustic finite element model. In this final example, DIPSS is exercised to study the propagation of plane acoustic pressure waves in a 3-D hard wall duct without end reflection and airflow.

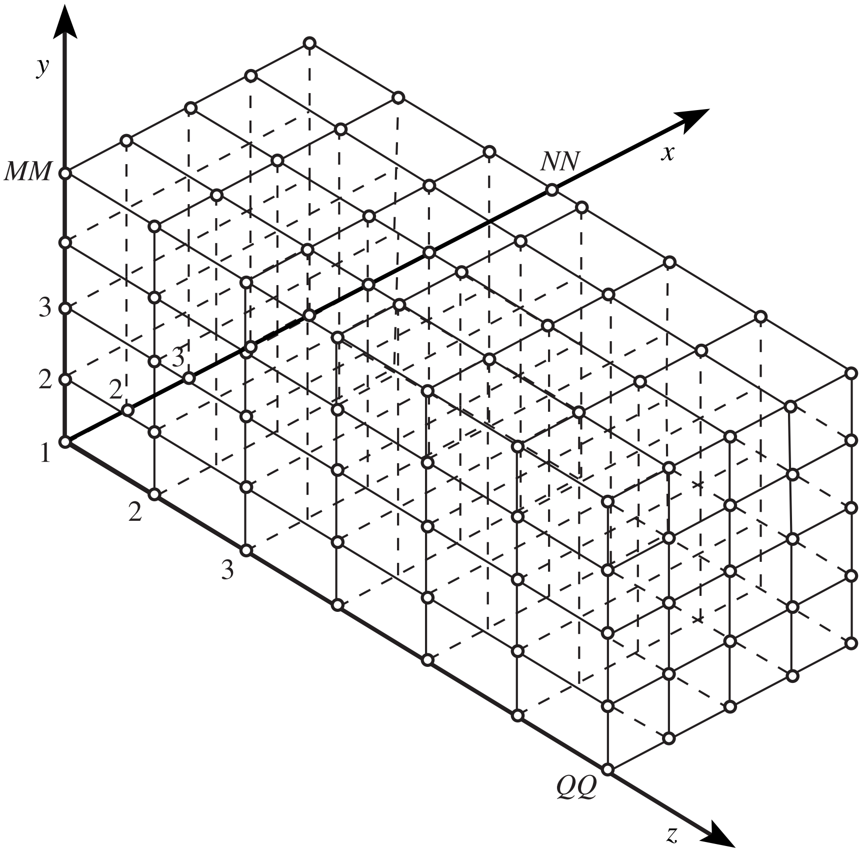


Figure 6. Finite element model for a three-dimensional hard wall duct

The duct is shown in Figure 6 and is modelled with brick elements. The source and exit planes are located at the left and right boundary, respectively. The matrix, K, contains complex coefficients and the dimension of K is determined by the product of NN, MM, and QQ (N=MMxNNxQQ). Results are presented for two grids (N=751,513 and N=1,004,400) and the finite element analysis procedure for generation of the complex stiffness matrix, K, was presented in another paper [18].

DIPSS memory and wallclock statistics were also compared to those obtained using the platform specific SGI parallel sparse solver (i.e., ZPSLDLT). These statistics were computed on an SGI ORIGIN 2000 computer platform that was located at the NASA Langley Research Center. The SGI platform contained 10 gigabytes of memory and eight ORIGIN 2000 processors were used. It should be noted that ZPSLDLT is part of the SCSL library (version 1.4 or higher) and is considered to be one of the most efficient commercialized direct sparse solvers that is capable of performing complex arithmetic. Due to the 3-D nature of hard wall duct example problem, K encounters many fill-in elements during the factorization phase. Thus, only the small grid (N=751,513) could fit within the allocated memory on the ORIGIN 2000. ZPSLDLT required 6.5 wallclock hours to obtain the solution on the small grid whereas DIPSS wallclock was only 2.44 hours. DIPSS also required nearly 1 gigabyte less memory than ZPSLDLT, and the DIPSS and ZPSLDLT solution vector (Z) were in excellent agreement.

Because DIPSS uses MPI for interprocess communications, it can be ported to other computer platforms. To illustrate this point the DIPSS software was ported to the SUN 10000 platform at Old Dominion University and used to solve the large grid duct acoustic problem (N =1,004,400). Wallclock statistics and speedup factors were obtained using as many as 64 SUN 10000 processors. Results are presented in Table 2. It should be noted that a superlinear speedup factor of 85.95 has been achieved when 64 SUN 10000 processors are used. This super-linear speedup factor is due to two primary reasons:

# of SUN processors	1	2	4	8	16	32	64
Assembly time (sec)	19.38	10.00	5.08	2.49	1.26	0.70	0.27
Factor time (sec)	131,229	58,976	26,174	10,273	3,260	909	56
Wallclock time (sec)	131,846	61,744	27,897	11,751	3,817	1,967	1,534
Speedup factor	1.00	2.14	4.73	11.22	34.54	67.03	85.95

Table 2: Statistics for 3-D Hard wall duct (N=1,004,400, K complex)

(1) The large finite element model has been divided into 64 subdomains. Since each processor is assigned to a smaller subdomain, the number of operations performed by each processor has been greatly reduced. Note that the number of operations are proportional to $(n^{(r)})^3$ for the dense matrix, or $n^{(r)}\cdot\mathrm{BW}^2$ for the banded, sparse matrix, where BW represent the half Band Width of the coefficient stiffness matrix.

(2) When the entire finite element model is analyzed by a direct conventional sparse solver, more computer "paging" is required due to a larger problem size.

5. Conclusions

A domain decomposition (DD) formulation for solving sparse linear systems of equations has been presented. The formulation incorporates lower level novel algorithmic ideas such as mixed direct/iterative sparse solvers, equation reordering, loop unrolling, efficient sparse assembly, and foward/backward solution phases that are optimized to take full advantage of sparsity and exploit modern computer architecture. Medium to large-scale examples considered in this paper show that the developed MPI parallel DD code is efficient in both sequential and parallel computing environments. Statistics show that software based on the formulation is significantly more efficient than that based on one of the best available commercialized, parallel, direct sparse solver.

Acknowledgement. The authors would like to acknowledge the many helpful comments suggested by the reviewers of this paper and the NASA Langley Research Center for providing financial support.

References

1. Desai, Chandrakant S. and Abel, John F.: *Introduction to the Finite Element Method*, Van Nostrand Reinhold Company, New York, N.Y., 1972.
2. Bathe, K. J.: *Finite Element Procedures*, Prentice-Hall, Inc., 1996.
3. Nguyen, D.T.: *Parallel-Vector Equation Solver For Finite Element Engineering Applications*, Kluwer Academic/Plenum Publishers, 2002.
4. Qin, J., Gray, Jr., C .E., Mei, C. and Nguyen, D. T.: *A parallel-vector equation solver for unsymmetric matrices on supercomputers*, Computing Systems in Engineering, **2**(2/3), (1991), 197-290.
5. Nguyen, D. T., Runesha, H., Belegundu, A. D., and Chandrupatla, T. R.: *Interior point method and indefinite sparse solver for linear programming problems*, Advances in Engineering Software, **29**(3-6), (1998) 409-414.
6. Nguyen, D.T., Hou, Gene, Runesha, H. and Bangfei, H.: *Alternative approach for solving sparse indefinite symmetrical system of equations*, Advances in Engineering Software, **31**(8-9), (2000), 581-584.
7. Ng, E. and Peyton, B.: *Block sparse Choleski algorithm on advanced uniprocessor computer*, Society for Industrial and Applied Mathematics Journal of Scientific Computing, **14**, (1993), 1034-1056.
8. Duff, I. and Reid, J.: *MA27-A set of Fortran Subroutines for Solving Sparse Symmetric Sets of Linear Equations*, AERE Technical Report, R-10533, Harwell, England, 1982.
9. Duff, I. and Reid, J.: *The multifrontal solution of indefinite sparse symmetric linear systems*, Association for Computing Machinery Transactions Mathematical Software, **9**, (1983), 302-325.
10. George, A. and Liu, J.: *Computer Solution of Large Sparse Positive Definite Systems*, Prentice-Hall, Inc., Englewood Cliffs, NJ, Chap. 5 & Chap. 10, 1981.
11. Pissanetzky, S.: *Sparse Matrix Technology*, Academic Press, Inc., London, U.K., 1984.
12. Moayyad, M. A. and Nguyen, D. T.: *An algorithm for domain decomposition in finite element analysis*, Journal of Computers and Structures, **39**(1-4), (1991), 277-290.

13. Karypis, G. and Kumar, V.: *METIS: Unstructured Graph Partitioning and Sparse Matrix Ordering*, Version 2.0, University of Minnesota, 1995.

14. Tungkahotara, S., Nguyen, D. T., Watson, W. R. and Runesha, H. B.: *Simple and efficient parallel dense equation solvers*, Ninth International Conference on Numerical Methods and Computational Mechanics, July 15-19, 2002, Univ. of Miskolc, Miskolc, Hungary.

15. Anderson, E., Bai, Z., Bischof, C., Blackford, L. S., and Demmel, J. W.: *Lapack Users' Guide (Software, Environments and Tools, 9)*, Society for Industrial & Applied Mathematics; ISBN: 0898714478; 3rd pkg edition, 2000.

16. Blackford, L. S., Choi, J., Cleary, A., D'Azevedo, E. and Demmel, J. W.: *Scalapack Users' Guide*, Society for Industrial & Applied Mathematics; ISBN: 0898713978; Bk&Cdr edition, 1997.

17. Papadrakakis, M., Bitzarakis, S. and Kotsopulos, A.: *Parallel Solution Techniques in Computational Structural Mechanics*, B. H. V. Topping (Editor), Parallel and Distributed Processing for Computational Mechanics: Systems and Tools, pp. 180-206, 1999, Saxe-Coburg Publications, Edinburgh, Scotland.

18. Watson, W. R.: *Three-dimensional rectangular duct code with application to impedance eduction*, AIAA Journal, **40**, (2002), 217-226.

Journal of Computational and Applied Mechanics, Vol. 4., No. 2., (2003), pp. 203–215

COMPARISON OF THE FARGUE-TYPE ANALYTICAL APPROXIMATION AND THE NUMERICAL SEMI-DISCRETIZATION METHOD FOR DELAYED SYSTEMS

Tamás Insperger and Gábor Stépán
Department of Applied Mechanics
Budapest University of Technology and Economics
Budapest, H-1521, Hungary
`inspi@mm.bme.hu, stepan@mm.bme.hu`

[Received: June 15, 2002]

Abstract. Stability of linear time periodic delay-differential equations is investigated. An analytical approximation, the so-called Fargue-type method, and a numerical approximation, the so-called semi-discretization method are compared for the damped and delayed Mathieu equation. It is shown that the convergence of the semi-discretization is better than that of the Fargue-type method.

Mathematical Subject Classification: 34K06
Keywords: parametric excitation, time delay, stability

1. Introduction

Time delay often occurs in different fields of science and engineering. In the modelling of cutting process dynamics, the so-called regenerative effect causes time delay [1]-[3]. In robotics applications, information delay of the control loop plays an important role [4]-[6]. Time delay also arises in neural network models, where the interactions of the neurons are delayed [7].

The delay effect is often subjected to parametric excitation. For example, in milling processes, the regenerative effect of cutting is accompanied by the tooth pass excitation [8]. While autonomous delayed systems are a well explored field of research [9], the behaviour of time periodic delayed systems is not predictable even for the simplest linear cases.

The Floquet theory of ordinary differential equations (ODEs) is extended to delay-differential equations (DDEs) [10], [11]. A so-called monodromy operator is defined that corresponds to the principal matrix of ODEs. The stability properties of time periodic DDEs are determined by the eigenvalues of the monodromy operator, via the so-called characteristic multipliers. If all these multipliers are in modulus less than 1, then the DDE is asymptotically stable. The monodromy operator, and consequently, the characteristic eigenvalues usually do not have a closed form, so no exact stability criteria can be expected. Generally, time-domain simulations and special approximation techniques are used to predict stability properties.

In this paper, two basically different approximation methods are compared: the Fargue-type approximation and the semi-discretization method. The Fargue-type method is a kind of analytical approximation method, while the semi-discretization method - as follows from its name - is a numerical method. The comparison of the two methods is made for the second order linear periodic DDE

$$\ddot{x}(t) + b_0\dot{x}(t) + c_0(t)x(t) = c_1 x(t-\tau)\,, \quad c_0(t+T) = c_0(t)\,. \tag{1.1}$$

If $c_0(t) = c_{0\,\delta} + c_{0\,\varepsilon}\cos(t)$ and $\tau = 2\pi$, then this equation gives the damped and delayed Mathieu equation. Three special cases can be defined: $c_1 = 0$, $c_{0\,\varepsilon} = 0$ and $b_0 = 0$. For these three cases, the stability charts in the parameter space $(c_{0\,\delta},\, c_1,\, b_0)$ were determined by van der Pol and Strutt [12], Hsu and Bhatt [13] and Insperger and Stépán [14], respectively. These cases serve as references for checking the accuracy of the approximation methods.

2. The Fargue-type approximation

2.1. **The type of approximation.** The Fargue-type approximation is described in two steps: first, the structure of the approximation is presented, then a time scale transformation is applied.

2.2. **The structure of the approximation.** The main step of this method is the approximation of the discrete time delay with a distributed one as follows

$$x(t-\tau) \approx \int_{-\infty}^{0} w_n(\vartheta) x(t+\vartheta)\mathrm{d}\vartheta\,, \tag{2.1}$$

where $w_n(\vartheta)$ is a special weight function series coming from the product of a polynomial and an exponential expression

$$w_n(\vartheta) = (-1)^n \frac{n^{n+1}}{\tau^{n+1} n!}\, \vartheta^n\, \mathrm{e}^{n\vartheta/\tau}\,. \tag{2.2}$$

The function $w_n(\vartheta)$ satisfies the following properties

$$\int_{-\infty}^{0} w_n(\vartheta)\mathrm{d}\vartheta = 1, \quad \lim_{n\to\infty} w_n(\vartheta) = f_\delta(\vartheta+\tau)\,, \tag{2.3}$$

where f_δ is the Dirac distribution. Figure 1 shows the weight functions for parameters $n = 2, 10, 50, 100$ and $\tau = 1$. It can be seen that the greater n is, the more correct the approximation is. Fargue [15] proved that equation (2.1) converges to $x(t-\tau)$ as n tends to infinity, i.e.

$$\lim_{n\to\infty} \int_{-\infty}^{0} w_n(\vartheta) x(t+\vartheta)\mathrm{d}\vartheta = x(t-\tau)\,. \tag{2.4}$$

Consequently, n can be called as approximation parameter.

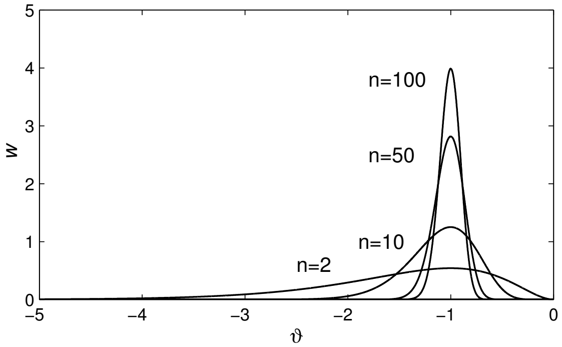


Figure 1. The Fargue-type weight function for $\tau = 1$

Application of approximation (2.1) with a fixed finite n in equation (1.1) results in the periodic DDE

$$\ddot{x}(t) + b_0\dot{x}(t) + c_0(t)x(t) = c_1 \int_{-\infty}^{0} w_n(\vartheta)x(t+\vartheta)\mathrm{d}\vartheta\,, \quad c_0(t+T) = c_0(t)\,. \tag{2.5}$$

Now, introduce the new variables $y_1, y_2, \ldots, y_{n+3}$ in several steps in the following way:

$$y_1(t) = x(t)\,, \tag{2.6}$$

$$y_2(t) = \dot{x}(t)\,, \tag{2.7}$$

$$y_3(t) = \int_{-\infty}^{0} w_n(\vartheta)x(t+\vartheta)\mathrm{d}\vartheta \approx x(t-\tau)\,. \tag{2.8}$$

Since

$$\frac{\mathrm{d}}{\mathrm{d}t}x(t+\vartheta) = \frac{\mathrm{d}}{\mathrm{d}\vartheta}x(t+\vartheta) = \dot{x}(t+\vartheta)\,, \tag{2.9}$$

the derivative of $y_3(t)$ with respect to the time t can be calculated via integration by parts as follows

$$\begin{aligned}\dot{y}_3(t) &= \int_{-\infty}^{0} (-1)^n \frac{n^{n+1}}{\tau^{n+1}n!}\vartheta^n\, \mathrm{e}^{n\vartheta/\tau}\dot{x}(t+\vartheta)\mathrm{d}\vartheta = \left[(-1)^n \frac{n^{n+1}}{\tau^{n+1}n!}\vartheta^n\, \mathrm{e}^{n\vartheta/\tau}x(t+\vartheta)\right]_{-\infty}^{0} \\ &\quad - \int_{-\infty}^{0} x(t+\vartheta)\frac{\mathrm{d}}{\mathrm{d}\vartheta}\left((-1)^n \frac{n^{n+1}}{\tau^{n+1}n!}\vartheta^n\, \mathrm{e}^{n\vartheta/\tau}\right)\mathrm{d}\vartheta = \\ &= 0 - \frac{n}{\tau}\int_{-\infty}^{0} x(t+\vartheta)(-1)^n \frac{n^{n+1}}{\tau^{n+1}n!}\vartheta^n\, \mathrm{e}^{n\vartheta/\tau}\mathrm{d}\vartheta \\ &\quad - \int_{-\infty}^{0} x(t+\vartheta)(-1)^n \frac{n^{n+1}}{\tau^{n+1}(n-1)!}\vartheta^{n-1}\, \mathrm{e}^{n\vartheta/\tau}\mathrm{d}\vartheta = \\ &= -\frac{n}{\tau}y_3(t) - \int_{-\infty}^{0} (-1)^n \frac{n^{n+1}}{\tau^{n+1}(n-1)!}\vartheta^{n-1}\, \mathrm{e}^{n\vartheta/\tau}x(t+\vartheta)\mathrm{d}\vartheta\,. \end{aligned} \tag{2.10}$$

This time, the second term in equation (2.10) is defined as the subsequent new variable

$$y_4(t) = \int_{-\infty}^{0} (-1)^n \frac{n^{n+1}}{\tau^{n+1}(n-1)!} \vartheta^{n-1} \, \mathrm{e}^{n\vartheta/\tau} x(t+\vartheta) \mathrm{d}\vartheta \, . \tag{2.11}$$

A similar calculation results in equation

$$\dot{y}_4(t) = -\frac{n}{\tau} y_4(t) - \int_{-\infty}^{0} (-1)^n \frac{n^{n+1}}{\tau^{n+1}(n-2)!} \vartheta^{n-2} \, \mathrm{e}^{n\vartheta/\tau} x(t+\vartheta) \mathrm{d}\vartheta \, , \tag{2.12}$$

where the second term defines a new variable again:

$$y_5(t) = \int_{-\infty}^{0} (-1)^n \frac{n^{n+1}}{\tau^{n+1}(n-2)!} \vartheta^{n-2} \, \mathrm{e}^{n\vartheta/\tau} x(t+\vartheta) \mathrm{d}\vartheta \, . \tag{2.13}$$

After the introduction of all the new variables in the same way, and calculating their time derivatives via integration by parts, the degree of ϑ decreases each time by 1. The $(n+2)^{\mathrm{nd}}$ new variable and its derivative read

$$y_{n+2}(t) = \int_{-\infty}^{0} (-1)^n \frac{n^{n+1}}{\tau^{n+1} 1!} \vartheta \, \mathrm{e}^{n\vartheta/\tau} x(t+\vartheta) \mathrm{d}\vartheta \, , \tag{2.14}$$

$$\dot{y}_{n+2}(t) = -\frac{n}{\tau} y_{n+2}(t) - \int_{-\infty}^{0} (-1)^n \frac{n^{n+1}}{\tau^{n+1}} \, \mathrm{e}^{n\vartheta/\tau} x(t+\vartheta) \mathrm{d}\vartheta \, , \tag{2.15}$$

where the last new variable is defined as

$$y_{n+3}(t) = \int_{-\infty}^{0} (-1)^n \frac{n^{n+1}}{\tau^{n+1}} \, \mathrm{e}^{n\vartheta/\tau} x(t+\vartheta) \mathrm{d}\vartheta \, . \tag{2.16}$$

The derivative of $y_{n+3}(t)$ reads

$$\begin{aligned} \dot{y}_{n+3}(t) &= -\frac{n}{\tau} y_{n+3}(t) - \int_{-\infty}^{0} (-1)^n \frac{n^{n+1}}{\tau^{n+1}} \, \mathrm{e}^{n\vartheta/\tau} x(t+\vartheta) \mathrm{d}\vartheta \\ &= -\frac{n}{\tau} y_{n+3}(t) - (-1)^n \frac{n^{n+1}}{\tau^{n+1}} x(t) \, . \end{aligned} \tag{2.17}$$

Equations (2.10)-(2.16) define the recursion

$$\dot{y}_j(t) = -\frac{n}{\tau} y_j(t) - y_{j+1}(t) \, , \quad j = 3, 4, \dots, n+2 \, , \tag{2.18}$$

while equation (2.17) forms a connection with $y_1(t) = x(t)$. Together with equations (2.5)-(2.8), a finite $(n+3)$ dimensional system of ODEs with a τ-periodic coefficient matrix can be defined:

$$\dot{\mathbf{y}}(t) = \mathbf{A}(t)\mathbf{y}(t) \, , \tag{2.19}$$

where $\mathbf{y} = \mathrm{col}(y_1 \; y_2 \; \dots \; y_{n+3})$ and

$$\mathbf{A}(t) = \begin{pmatrix} 0 & 1 & 0 & 0 & \cdots & 0 \\ -c_0(t) & -b_0 & c_1 & 0 & \cdots & 0 \\ 0 & 0 & -n/\tau & -1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & -n/\tau & -1 \\ (-1)^n \left(-n/\tau\right)^{n+1} & 0 & 0 & \cdots & 0 & -n/\tau \end{pmatrix} . \tag{2.20}$$

As it was shown by Fargue [15], equation (2.5) is equivalent to equation (2.19). Thus, the stability of equation (2.19) gives the stability of equation (2.5), i.e., if equation (2.5) is stable, then equation (2.19) is also stable, and vice versa.

System (2.19) is asymptotically stable, if and only if all the characteristic multipliers denoted by μ_j, $j = 1, 2, \ldots, n+3$, are in modulus less than one. There are no general methods to calculate the characteristic multipliers of system (2.19) in an algebraic form, but there are various types of approximation methods. Here, we will use the piecewise constant approximation of the coefficient matrix $\mathbf{A}(t)$.

Construct the time intervals $[t_{i-1}, t_i]$ of length Δt_i, $i = 1, 2, \ldots, k$, so that the principal period can be expressed as $T = \sum_{i=1}^{k} \Delta t_i$. The function $c_0(t)$ is approximated with constant (say average) values

$$c_{0\,i} = \frac{1}{\Delta t_i} \int_{t_{i-1}}^{t_i} c_0(t) \mathrm{d}t \approx c_0(t_i - \Delta t_i/2)\,, \quad i = 1, 2, \ldots, k\,, \tag{2.21}$$

in each time interval. Correspondingly, the piecewise constant approximation of the coefficient matrix is

$$\mathbf{A}_i = \begin{pmatrix} 0 & 1 & 0 & 0 & \cdots & 0 \\ -c_{0\,i} & -b_0 & c_1 & 0 & \cdots & 0 \\ 0 & 0 & -n/\tau & -1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & -n/\tau & -1 \\ (-1)^n \, (-n/\tau)^{n+1} & 0 & 0 & \cdots & 0 & -n/\tau \end{pmatrix}, \tag{2.22}$$

for $t \in [t_{i-1}, t_i]$, $i = 1, 2, \ldots, k$.

The Floquet transition matrix of the system can be given by coupling the solutions for each interval:

$$\mathbf{\Phi} = \exp(\mathbf{A}_k \Delta t_k) \exp(\mathbf{A}_{k-1} \Delta t_{k-1}) \cdots \exp(\mathbf{A}_1 \Delta t_1)\,. \tag{2.23}$$

This matrix is a finite dimensional approximation of the monodromy operator of system (1.1). At this point, several numerical methods can be used to determine the critical eigenvalue having the greatest modulus.

2.3. **Time scale transformation.** A numerical problem arises in the Fargue-type approximation. The bottom left element of the piecewise constant matrix (2.22) increases in modulus exponentially with the approximation parameter. It means that for large n, the matrix (2.22) is not well conditioned, and the computation of the matrix exponential in equation (2.23) needs high CPU capacity and may still result in numerical errors.

For example, if $n = 20$ and $\tau = 1$, then $(-1)^n \, (-n/\tau)^{n+1} \approx 2.0972 \times 10^{27}$, i.e. the norm of the matrix is at least 2.0972×10^{27}. Thus, the norm of the exponential matrix is about $10^{9.108 \times 10^{26}}$. This order of magnitude of numbers leads to numerical difficulties during the evaluation of the Floquet transition matrix $\mathbf{\Phi}$.

A solution for this problem is to introduce the new dimensionless time $\tilde{t} = (n/\tau)t$. The derivatives w.r.t. $\tilde{t}$ are denoted by comma, and defined by

$$\frac{\mathrm{d}}{\mathrm{d}t} = \frac{n}{\tau}\frac{\mathrm{d}}{\mathrm{d}\tilde{t}} \,. \tag{2.24}$$

For this time scale, the equation (1.1) has the form

$$\begin{gathered} x''\left(\tilde{t}\right) + \left(\frac{\tau}{n}\right) b_0\, x'\left(\tilde{t}\right) + \left(\frac{\tau}{n}\right)^2 c_0\left(\tilde{t}\right) x\left(\tilde{t}\right) = \left(\frac{\tau}{n}\right)^2 c_1\, x\left(\tilde{t} - n\right) \,, \\ c_0\left(\tilde{t} + \left(\frac{n}{\tau}\right) T\right) = c_0\left(\tilde{t}\right) \,. \end{gathered} \tag{2.25}$$

The point is that in equation (2.25) the time delay is just equal to the approximation parameter, and the approximated coefficient matrix resulting from the Fargue-type approximation is well conditioned now:

$$\mathbf{A}_i = \begin{pmatrix} 0 & 1 & 0 & 0 & \cdots & 0 \\ -\left(\frac{\tau}{n}\right)^2 c_{0\,i} & -\left(\frac{\tau}{n}\right) b_0 & \left(\frac{\tau}{n}\right)^2 c_1 & 0 & \cdots & 0 \\ 0 & 0 & -1 & -1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & -1 & -1 \\ (-1)^n & 0 & 0 & \cdots & 0 & -1 \end{pmatrix} . \tag{2.26}$$

The norm of this matrix is about 2 (it also depends on the parameters b_0, $c_{0\,i}$, c_1, n, τ, naturally), and the numerical problems mentioned before do not arise. Approximation parameter $n = 100$ can be used with a reasonable CPU capacity to determine the Floquet transition matrix

$$\mathbf{\Phi} = \exp\left(\mathbf{A}_k\, \frac{n}{\tau}\Delta t_k\right) \exp\left(\mathbf{A}_{k-1}\, \frac{n}{\tau}\Delta t_{k-1}\right) \cdots \exp\left(\mathbf{A}_1\, \frac{n}{\tau}\Delta t_1\right) \,. \tag{2.27}$$

The comparison of the CPU times for evaluating characteristic multipliers of Floquet transition matrices with and without time scale transformation can be seen in Figure 2. The figure shows the CPU time for evaluating eigenvalues of matrices (2.23) and (2.27), respectively, for various approximation parameters n and for interval number $k = 10$. It can be seen that the CPU time is higher for computations without time scale transformation. For a higher approximation parameter, the difference between the two methods increases exponentially. Furthermore, for $n > 25$, the accumulated numerical errors become too large during the calculation without time scale transformation. That is, without time scale transformation, the method can only be used for $n < 25$. With time scale transformation, the method can be used for higher approximation parameters, as well.

3. Numerical investigation by semi-discretization

In this section, the semi-discretization method [16] is used to construct the stability chart of equation (1.1).

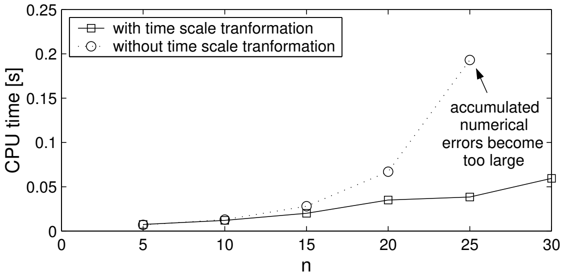


Figure 2. Comparison of Fargue-type approximations with and without time scale transformation

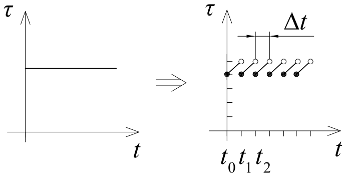


Figure 3. Approximation of the time delay for $m = 4$

The first step of semi-discretization is the construction of time interval division $[t_i, t_{i+1})$ of length Δt, $i = 0, 1, \ldots$ so that $\tau = (m + 1/2)\Delta t$, where m is called approximation parameter. In the i^{th} interval, equation (1.1) can be approximated as

$$\ddot{x}(t) + b_0\dot{x}(t) + c_{0\,i}x(t) = c_1 x_{i-m}\,, \tag{3.1}$$

where

$$c_{0\,i} = \frac{1}{\Delta t}\int_{t_i}^{t_{i+1}} c_0(t)\mathrm{d}t\,, \tag{3.2}$$

and

$$x_{i-m} = x(t_{i-m}) = x(t_i - m\Delta t)\,. \tag{3.3}$$

That is, the time periodic coefficient is approximated by a piecewise constant one, and the time delayed term is approximated by a piecewise discrete value. This corresponds to a saw-like approximation of the continuous time delay shown in Figure 3.

For the initial conditions $x(t_i) = x_i$, $\dot{x}(t_i) = \dot{x}_i$, the solution and its derivative at each time instant t_{i+1} can be determined:

$$x_{i+1} = x(t_{i+1}) = a_{0\,0}x_i + a_{0\,1}\dot{x}_i + b_{0\,m}x_{i-m}\,, \tag{3.4}$$

$$\dot{x}_{i+1} = \dot{x}(t_{i+1}) = a_{1\,0}x_i + a_{1\,1}\dot{x}_i + b_{1\,m}x_{i-m}\,, \tag{3.5}$$

where

$$\begin{aligned}
a_{0\,0} &= \kappa_{1\,0}\exp(\lambda_1\Delta t) + \kappa_{2\,0}\exp(\lambda_2\Delta t)\,,\\
a_{0\,1} &= \kappa_{1\,1}\exp(\lambda_1\Delta t) + \kappa_{2\,1}\exp(\lambda_2\Delta t)\,,\\
a_{1\,0} &= \kappa_{1\,0}\lambda_1\exp(\lambda_1\Delta t) + \kappa_{2\,0}\lambda_2\exp(\lambda_2\Delta t)\,,\\
a_{1\,1} &= \kappa_{1\,1}\lambda_1\exp(\lambda_1\Delta t) + \kappa_{2\,1}\lambda_2\exp(\lambda_2\Delta t)\,,\\
b_{0\,m} &= \sigma_1\exp(\lambda_1\Delta t) + \sigma_2\exp(\lambda_2\Delta t) + b/(\delta+\varepsilon\,c_i)\,,\\
b_{1\,m} &= \sigma_1\lambda_1\exp(\lambda_1\Delta t) + \sigma_2\lambda_2\exp(\lambda_2\Delta t)\,,
\end{aligned}$$

and

$$\lambda_{1,2} = \frac{-\kappa \pm \sqrt{\kappa^2 - 4(\delta+\varepsilon\,c_i)}}{2}\,,$$

$$\kappa_{1\,0} = \frac{\lambda_2}{\lambda_2-\lambda_1}\,,\quad \kappa_{1\,1} = \frac{-1}{\lambda_2-\lambda_1}\,,\qquad \sigma_1 = \frac{-\lambda_2}{\lambda_2-\lambda_1}\,\frac{b}{\delta+\varepsilon\,c_i}\,,$$

$$\kappa_{2\,0} = \frac{-\lambda_1}{\lambda_2-\lambda_1}\,,\quad \kappa_{2\,1} = \frac{1}{\lambda_2-\lambda_1}\,,\qquad \sigma_2 = \frac{\lambda_1}{\lambda_2-\lambda_1}\,\frac{b}{\delta+\varepsilon\,c_i}\,.$$

Equations (3.4) and (3.5) define the discrete map

$$\mathbf{y}_{i+1} = \mathbf{B}_i\mathbf{y}_i\,, \tag{3.6}$$

where the $m+2$ dimensional state vector is

$$\mathbf{y}_i = \mathrm{col}(\dot{x}_i\ \ x_i\ \ x_{i-1}\ \ \dots\ \ x_{i-m})\,, \tag{3.7}$$

and the coefficient matrix has the form

$$\mathbf{B}_i = \begin{pmatrix}
a_{1\,1} & a_{1\,0} & 0 & \dots & 0 & b_{1\,m}\\
a_{0\,1} & a_{0\,0} & 0 & \dots & 0 & b_{0\,m}\\
0 & 1 & 0 & \dots & 0 & 0\\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\
0 & 0 & 0 & \dots & 0 & 0\\
0 & 0 & 0 & \dots & 1 & 0
\end{pmatrix}. \tag{3.8}$$

So, the connection between the states at t_i and t_{i+1} is determined by the transition matrix $\mathbf{B}_i$.

Stability properties are determined by the transition matrix $\mathbf{\Phi}$ between the states at t_0 and t_0+T. First, investigate the case when the principal period is equal to the time delay: $T=\tau$. Since $t_0+\tau = t_0+(m+1/2)\Delta t$ and $t_0+2\tau = t_0+(2m+1)\Delta t$, only the transition matrix between the states at t_0 and $t_0+2\tau$ can be constructed by coupling the solutions for the discretization intervals of indeces $i=0,\,1,\,\dots,\,2m$. This gives the square of the transition matrix over the principal period:

$$\mathbf{\Phi}^2 = \mathbf{B}_{2m}\mathbf{B}_{2m-1}\dots\mathbf{B}_1\mathbf{B}_0\,. \tag{3.9}$$

The eigenvalues of $\mathbf{\Phi}^2$ give the square of the eigenvalues of $\mathbf{\Phi}$. Since $|\mu|<1$ if and only if $|\mu^2|<1$, the stability condition for $\mathbf{\Phi}^2$ is the same as for the matrix $\mathbf{\Phi}$.

If $T=2\tau$, then the matrix multiplication in equation (3.9) gives exactly the transition matrix over the principal period. If $T=\tau/2$, then (3.9) gives the 4^{th} power of the transition matrix over the principal period.

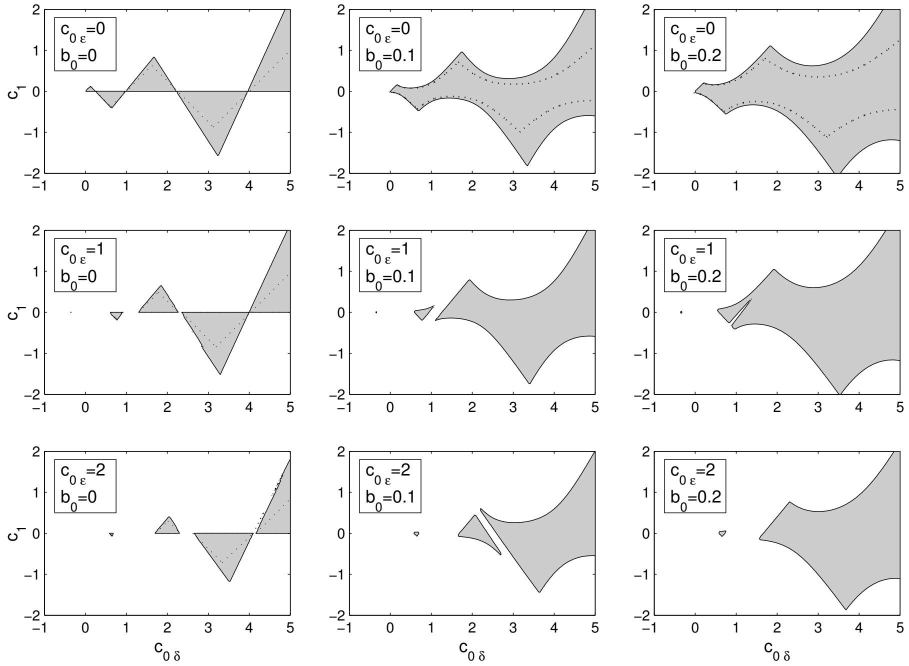


Figure 4. Stability charts for equation (4.1) with $T = 2\pi$ determined via Fargue-type method with $n = 100$

The proof of the convergence of the semi-discretization method is given in [16].

4. **Comparison of the methods**

The special case of equation (1.1)

$$\ddot{x}(t) + b_0\dot{x}(t) + c_0(t)x(t) = c_1 x(t - 2\pi)\,, \quad c_0(t) = c_{0\,\delta} + c_{0\,\varepsilon}\cos(2\pi t/T) \qquad (4.1)$$

is investigated. For $T = 2\pi$, this gives the damped and delayed Mathieu equation mentioned in the introduction.

Figure 4 shows the stability charts for equation (4.1) with $T = 2\pi$ determined by the Fargue-type method with approximation parameter $n = 100$. This means that the infinite dimensional equation (4.1) was approximated by a 103×103 sized system. For the autonomous case, when $c_{0\,\varepsilon} = 0$, the boundary curves were determined by Hsu and Bhatt [13]. For the undamped case, when $b_0 = 0$, Insperger and Stépán [14] proved that the boundaries are straight lines with slope -1, 0 and 1. For these special cases, the exact boundaries are shown by dotted lines in Figure 4. It can be seen that the stability boundaries predicted by the Fargue-type method with $n = 100$ are acceptable only in the parameter domain $-1 < c_{0\,\delta} < 1$, $-0.5 < c_1 < 0.5$.

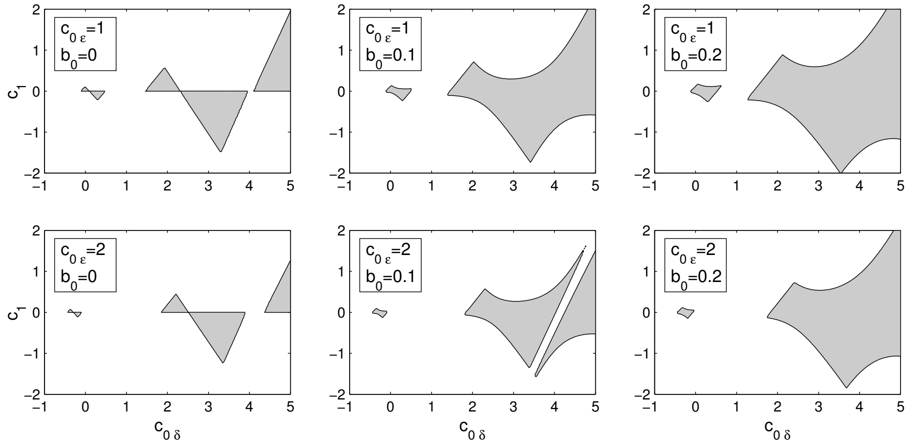


Figure 5. Stability charts for equation (4.1) with $T = \pi$ determined via Fargue-type method with $n = 100$

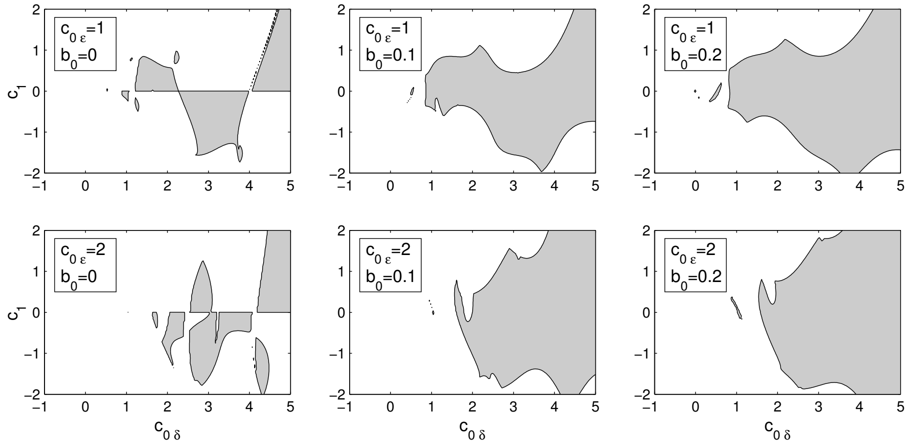


Figure 6. Stability charts for equation (4.1) with $T = 4\pi$ determined via Fargue-type method with $n = 100$

Figures 5 and 6 show the stability charts for equation (4.1) with $T = \pi$ and $T = 4\pi$, respectively, determined by the Fargue-type method with $n = 100$. In these cases, the boundary curves are not exactly known, so these results must be handled carefully. For the case $T = \pi$, the linear boundaries are still present and the stability charts have a clear structure. For the case $T = 4\pi$, the structures of the charts are not so clear, and also disjunct stable parameter domains arise.

Figure 7 shows the stability charts for equation (4.1) with $T = 2\pi$ determined by the semi-discretization method with approximation parameter $m = 20$. This means

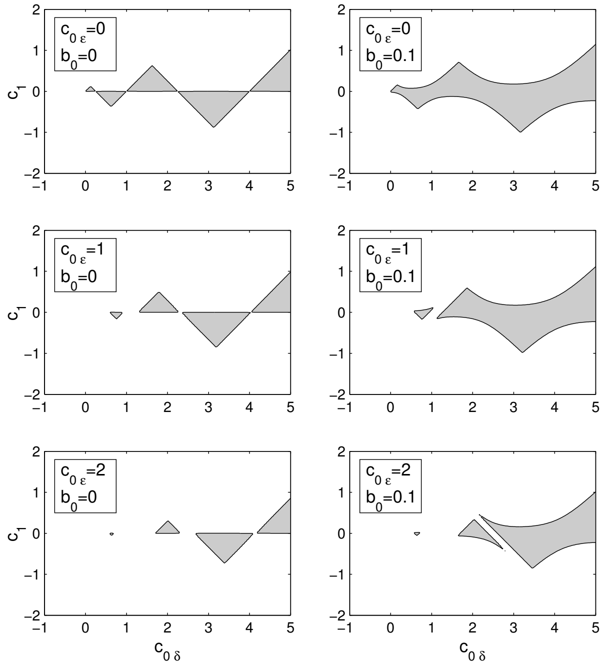


Figure 7. Stability charts for equation (4.1) with $T = 2\pi$ determined via semi-discretization method with $m = 20$

that a 22×22 sized discrete map was used to approximate the infinite dimensional equation (4.1) as opposed to the 103×103 sized Fargue-type approximation. For the special cases, when either $c_{0\,\varepsilon} = 0$ or $b_0 = 0$, the exact boundaries are also shown by dotted lines. In the presented parameter domain $-1 < c_{0\,\delta} < 5$, $-2 < c_1 < 2$, however, the difference between the exact boundaries and the boundaries determined by the semi-discretization method is not visible. These reference cases show that the semi-discretization method is more effective than the Fargue-type approximation.

Figures 8 and 9 show the stability charts for equation (4.1) with $T = \pi$ and $T = 4\pi$, respectively, determined by the semi-discretization method with approximation parameter $m = 20$. Similarly to the charts given by the Fargue-type approximation, the cases $T = \pi$ and $T = 2\pi$ show linear stability boundaries, while the case $T = 4\pi$ gives the intriguing stability charts with disjunct stable domains.

5. **Conclusions**

Two approximation methods were investigated: the Fargue-type method and the semi-discretization method. The Fargue-type method is a kind of analytical technique,

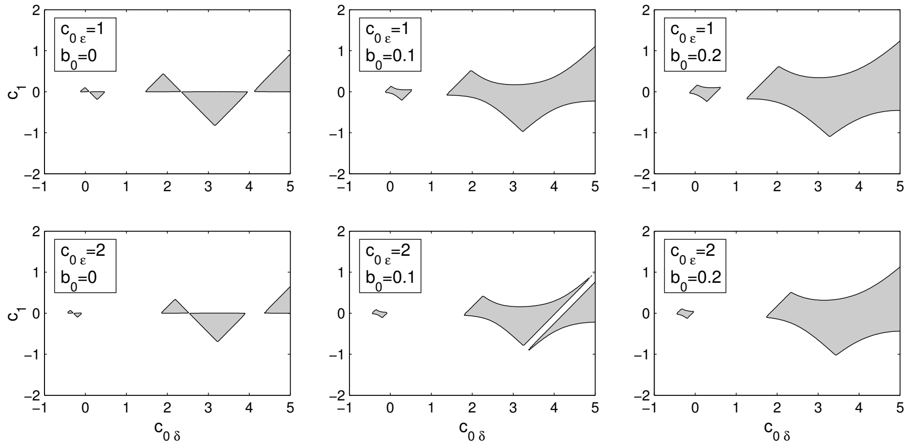


Figure 8. Stability charts for equation (4.1) with $T = \pi$ determined via semi-discretization method with $m = 20$

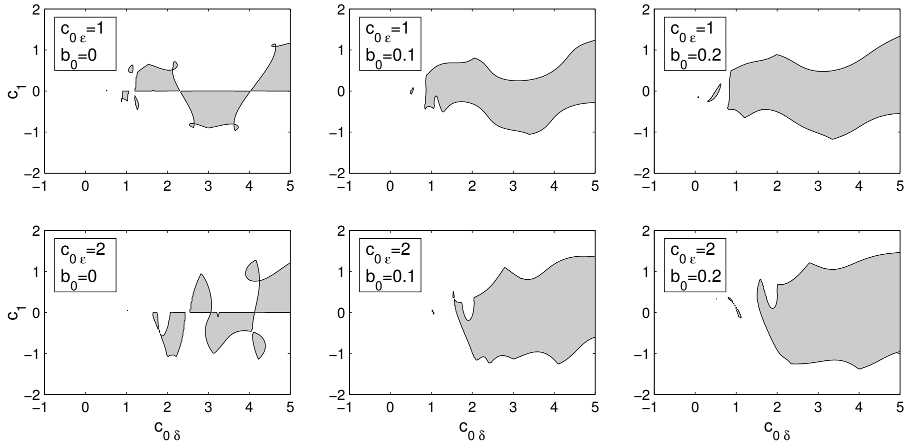


Figure 9. Stability charts for equation (4.1) with $T = 4\pi$ determined via semi-discretization method with $m = 20$

where the discrete time delay is approximated by a special distributed one. Semi-discretization is a numerical method that uses a partial time discretization. Both methods can be characterised by the dimension of the approximated system.

Both methods were applied to construct stability charts for the damped and delayed Mathieu equation (4.1). Stability charts were constructed by the Fargue-type method with 103×103 sized approximate system, and by the semi-discretization with 22×22 sized approximate system. It was shown that the semi-discretization method is much more effective than the Fargue-type approach. This result is not surprising, since the numerical methods based on discretization techniques are considered more effective than the analytical approximations.

For the cases with a time period equal to the time delay, and to half of it, the stability charts are reliable, close to each other and to the closed form results. When the time period is double the time delay, the stability charts show substantial differences, and there are still no reference results from the theory of periodic DDEs, either.

Acknowledgement. This research was supported by the Hungarian National Science Foundation under grant no. OTKA TS040792.

References

1. Tlusty, J., Polacek, A., Danek, C. and Spacek, J.: *Selbsterregte Schwingungen an Werkzeugmaschinen*, VEB Verlag Technik, Berlin, 1962.
2. Tobias, S. A.: *Machine Tool Vibration*, Blackie, London, 1965.
3. Kudinov, V. A.: *Dynamics of Tool-Lathe*, Mashinostroenie, Moscow, 1967. (in Russian)
4. Whitney, D. E.: *Force feedback control of manipulator fine motions*, Journal of Dynamics Systems, Measurement and Control, **98**, (1977), 91-97.
5. Stépán, G. and Steven, A.: *Theoretical and experimental stability analysis of a hybrid position-force controlled robot*, Proceedings of the Eighth Symposium on Theory and Practice of Robots and Manipulators, Krakow, Poland, 1990, 53-60.
6. Kim, W. S. and Bejczy, A. K.: *Special issue on space robotics*, IEEE Transactions on Robotics and Automation, **10**(5), (1993).
7. Campbell S. A., Ruan, S. and Wei, J.: *Qualitative analysis of a neural network model with multiple time delays*, International Journal of Bifurcation and Chaos, **9**(8), (1999), 1585-1595.
8. Insperger, T. and Stépán, G.: *Stability of high-speed milling*, Proceedings of Symposium on Nonlinear Dynamics and Stochastic Mechanics, Orlando, Florida, AMD-241, 2000, 119-123.
9. Stépán, G.: *Retarded Dynamical Systems*, Longman, Harlow, 1989.
10. Hale, J. K. and Lunel, S. M. V.: *Introduction to functional differential equations*, Springer-Verlag, New York, 1993.
11. Farkas, M.: *Periodic Motions*, Springer-Verlag, New York, 1994.
12. van der Pol, F. and Strutt, M. J. O.: *On the stability of the solutions of Mathieu's equation*, Philosophical Magazine, and Journal of Science, **5**, (1928), 18-38.
13. Hsu, C. S. and Bhatt, S. J.: *Stability charts for second-order dynamical systems with time lag*, Journal of Applied Mechanics, **33E**(1), (1966), 119-124.
14. Insperger, T. and Stépán, G.: *Stability chart for the delayed Mathieu equation*, Proceedings of The Royal Society, Mathematical Physical and Engineering Sciences, **458**(2024), (2002), 1989-1998.
15. Fargue, D.: *Réducibilité des systèmes héréditaires à des systèmes dynamiques*, Comptes Rendus de ľAcadémie des Sciences, Paris, **277B**, (1973), 471-473.
16. Insperger, T. and Stépán, G.: *Semi-discretization method for delayed systems*, International Journal for Numerical Methods in Engineering, **43**(1), (2003), 25-34.

Calendar of Events

2004

March 22-24, 2004: Lisbon, Portugal
Fifth International Conference on Advances in Fluid Mechanics
Conference Secretariat: Amy D'Arcy-Burt
AFM 2004, WESSEX INSTITUTE OF TECHNOLOGY
Ashurst Lodge, Ashurst
Southampton, SO40 7AA
E-mail: adarcy-burt@wessex.ac.uk
Web: http://www.wessex.ac.uk/conferences/2004/fluidmechanics04/
Phone: 44 (0) 238 029 3223; Fax: 44 (0) 238 029 2853

July 5-7, 2004: Cape Town, South Africa
Second International Conference on Structural Engineering, Mechanics and Computation
Professor A. Zingoni, University of Cape Town, South Africa
E-mail: azingon@eng.uct.ac.za
Web: http://www.ebe.uct.ac.za/~semc2004
Phone: (27) (21) 650 2601; Fax: (27) (21) 689 7471;

July 24 - 28, 2004: Jyväskylä, Finland
European Congress on Computational Methods
in Applied Sciences and Engineering
Professor Pekka Neittaanmäki, University of Jyväskylä
Department of Mathematical Information Technology,
P.O. Box 35 FIN-400, University of Jyväskylä, Finland
E-mail: pn@mit.jyu.fi
Web: http://www.mit.jyu.fi/eccomas2004
Fax: +358 14 260 2771

August 15-21, 2004: Warsaw, Poland
International Congress of Theoretical and Applied Mechanics
Professor W. Gutkowski, Polish Academy of Sciences
E-mail: ictam04@ippt.gov.pl
Web: http://www.ictam04.ippt.gov.pl
Phone: (+48 22) 826 9803; Fax: (+48 22) 826 9815

September 5-10, 2004: Bejing China
World Congress on Computational Mechanics
Professor Mingwu Yuan, Department of Mechanics & Engineering
Science Peking University, Beijing 100871, China
E-mail: general@.wccm6-apcom04.org.cn and paper@wccm6-apcom04.org.cn
Web: http://www.wccm6-apcom04.org.cn/index.htm
Phone: +86 10 6275 1826; Fax: +86 10 6275 9806

2005

March 20-25, 2005: Turin, Italy
11th International Conference on Fracture Location
Web: `http://www.ICF11.com`
Professor Alberto Carpinteri, CCI Centro Congressi Internazionale s.r.l.
Via Cervino, 60-10155 Turin
E-mail: info@congressiefiere.com
Phone: +39 011 244 69 11; Fax: +39 011 244 69 00

June 1-4, 2005: Salzburg, Austria
5th International Conference on Computation
of Shell & Spatial Structures
Web: `http://www.uni-stuttgart.de/ibs/iassiacm2005/`
Professor E. Ramm, University of Stuttgart, Institute of Structural Mechanics
Pfaffenwaldring 7, 70550 Stuttgart, Germany
E-mail: `baustatik@statik.uni-stuttgart.de`
Phone: +49 711 685-6123; Fax: +49 711 685-6130

August 7-12, 2005: Eindhoven, The Netherlands
ENOC-2005 Fifth EUROMECH
Nonlinear Oscillations Nonlinear Dynamics Conference
Web: `http://yp.wtb.tue.nl/showemp.php/822`
Professor Dick H. van Campen, Department of Mechanical Engineering
Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands
E-mail: `D.H.v.Campen@tue.nl`
Phone: +31 40 247 2710, Fax: +31 40 243 7175

Notes for Contributors

to the Journal of Computational and Applied Mechanics

Aims and scope. The aim of the journal is to publish research papers on theoretical and applied mechanics. Special emphasis is given to articles on computational mechanics, continuum mechanics (mechanics of solid bodies, fluid mechanics, heat and mass transfer) and dynamics. Review papers on a research field and materials effective for teaching can also be accepted and are published as review papers or classroom notes. Papers devoted to mathematical problems relevant to mechanics will also be considered.

Frequency of the journal. Two issues a year (approximately 80 pages per issue).

Submission of Manuscripts. Submission of a manuscript implies that the paper has not been published, nor is being considered for publication elsewhere. Papers should be written in standard grammatical English. Two copies of the manuscript should be submitted on pages of A4 size. The text is to be 130 mm wide and 190 mm long and the main text should be typeset in 10pt CMR fonts. Though the length of a paper is not prescribed, authors are encouraged to write concisely. However, short communications or discussions on papers published in the journal must not be longer than 2 pages. Each manuscript should be provided with an English Abstract of about 50–70 words, reporting concisely on the objective and results of the paper. The Abstract is followed by the Mathematical Subject Classification – in case the author (or authors) give the classification codes – then the keywords (no more than five). References should be grouped at the end of the paper in numerical order of appearance. Author's name(s) and initials, paper titles, journal name, volume, issue, year and page numbers should be given for all journals referenced.

The journal prefers the submission of manuscripts in LaTeX. Authors should prefer the $\mathcal{A}\mathcal{M}\mathcal{S}$-LaTeX article class and are not recommended to define their own LaTeX commands. Visit our home page for further details concerning the issue how to edit your paper.

For the purpose of refereeing, two copies of the manuscripts should initially be submitted in hardcopy to an editor of the journal. The eventual supply of an accepted-for-publication paper in its final camera-ready form (together with the corresponding files on an MS–DOS diskette) will ensure more rapid publication. Format requirements are provided by the home page of the journal from which sample LaTeX files can be downloaded:

`http://www.uni-miskolc.hu/home/web/pumns/mechanics`

These sample files can also be obtained directly (via e-mail) from a member of the Editorial Board, Gy. Szeidl (Gyorgy.SZEIDL@uni–miskolc.hu), upon request.

Twenty offprints of each paper will be provided free of charge and mailed to the correspondent author.

The Journal of Computational and Applied Mechanics is abstracted in Zentralblatt für Mathematik and in the Russian Referativnij Zhurnal.

Responsible for publication: Rector of the Miskolc University
Published by the Miskolc University Press under the leadership of Dr. József PÉTER
Responsible for duplication: works manager Mária KOVÁCS
Number of copies printed: 200
Put to the Press on November 1, 2003
Number of permission: TU 03-980-ME

HU ISSN 1586–2070

A Short History of the Publications of the University of Miskolc

The University of Miskolc (Hungary) is an important center of research in Central Europe. Its parent university was founded by the Empress Maria Teresia in Selmecbánya (today Banska Štiavnica, Slovakia) in 1735. After the first World War the legal predecessor of the University of Miskolc moved to Sopron (Hungary) where, in 1929, it started the series of university publications with the title *Publications of the Mining and Metallurgical Division of the Hungarian Academy of Mining and Forestry Engineering* (Volumes I.-VI.). From 1934 to 1947 the Institution had the name Faculty of Mining, Metallurgical and Forestry Engineering of the József Nádor University of Technology and Economic Sciences at Sopron. Accordingly, the publications were given the title *Publications of the Mining and Metallurgical Engineering Division* (Volumes VII.-XVI.). For the last volume before 1950 – due to a further change in the name of the Institution – *Technical University, Faculties of Mining, Metallurgical and Forestry Engineering, Publications of the Mining and Metallurgical Divisions* was the title.

For some years after 1950 the Publications were temporarily suspended.

After the foundation of the Mechanical Engineering Faculty in Miskolc in 1949 and the movement of the Sopron Mining and Metallurgical Faculties to Miskolc, the Publications restarted with the general title *Publications of the Technical University of Heavy Industry* in 1955. Four new series - Series A (Mining), Series B (Metallurgy), Series C (Machinery) and Series D (Natural Sciences) - were founded in 1976. These came out both in foreign languages (English, German and Russian) and in Hungarian.

In 1990, right after the foundation of some new faculties, the university was renamed to University of Miskolc. At the same time the structure of the Publications was reorganized so that it could follow the faculty structure. Accordingly three new series were established: Series E (Legal Sciences), Series F (Economic Sciences) and Series G (Humanities and Social Sciences). The latest series, i.e., the series H (European Integration Studies) was founded in 2001. The eight series are formed by some periodicals and such publications which come out with various frequencies.

Papers on computational and applied mechanics were published in the

Publications of the University of Miskolc, Series D, Natural Sciences.

This series was given the name Natural Sciences, Mathematics in 1995. The name change reflects the fact that most of the papers published in the journal are of mathematical nature though papers on mechanics also come out.

The series

Publications of the University of Miskolc, Series C, Fundamental Engineering Sciences

founded in 1995 also published papers on mechanical issues. The present journal, which is published with the support of the Faculty of Mechanical Engineering as a member of the Series C (Machinery), is the legal successor of the above journal.

Journal of Computational and Applied Mechanics
Volume 4, Number 2 (2003)

Contents

Contributed Papers