Programs and Algorithms of Numerical Mathematics 14

Dolní Maxov, June 1–6, 2008

Proceedings of Seminar

Edited by
J. Chleboun, P. Přikryl, K. Segeth, T. Vejchodský

Institute of Mathematics
Academy of Sciences of the Czech Republic
Prague 2008

ISBN 978-80-85823-55-4
Matematický ústav AV ČR, v.v.i.
Praha 2008

Contents

Preface

This book comprises papers that originated from invited lectures and short communications presented at the seminar Programs and Algorithms of Numerical Mathematics held in Dolní Maxov, Czech Republic, June 1–6, 2008.

The seminar was organized by the Institute of Mathematics of the Academy of Sciences of the Czech Republic and continued the previous seminars on mathematical software and numerical methods held (mostly biannually) in Alšovice, Bratříkov, Janov nad Nisou, Kořenov, Lázně Libverda, Dolní Maxov, and Prague in the period 1983–2006. The objective of this series of seminars is to provide a forum for presentation and discussion of advanced topics in numerical analysis, new approaches to mathematical modeling, and single- or multi-processor applications of computational methods.

Almost 50 participants from the field took part in the seminar, most of them form the Czech universities and from the institutes of the Academy of Sciences of the Czech Republic. The participation of a significant number of young scientists, PhD students, and also some undergraduate students is an established tradition of the PANM seminar; this year was not an exception. We believe that those who took part in the PANM seminar for the first time found the seminar worth their time and money, and that they will join the PANM community and will participate in the next PANM seminar, too.

The organizing committee consisted of Jan Chleboun, Pavel Kůs, Petr Přikryl, Karel Segeth, and Tomáš Vejchodský. Mrs Hana Bílková kindly helped in preparing manuscripts for print.

All papers have been reproduced directly from material submitted by the authors but an attempt has been made to use a unified layout for each paper.

The editors and organizers wish to thank all distinguished scientists who took a share in reviewing the submitted manuscripts.

J. Chleboun, P. Přikryl, K. Segeth, T. Vejchodský

THREE DIMENSIONAL MODELLING OF THE PEACH IN MAPLE*

Stanislav Bartoň

Abstract

Linearized Gauss-Newton iteration method is used to determine main axes of the three-dimensional ellipsoid approximating a peach. Three independent photos displaying the peach as ground, side, and front view are used as data sources. System MAPLE 11 was used as a computer environment. A practical example is presented in order to demonstrate the usage of all required commands. The quality of approximation is evaluated as a final part of the paper.

1. Introduction

The knowledge of the object's shape is usually one of the critical parameters for evaluating its mechanical properties. Concerning peaches, the proportions are significant e.g. for storage processing, deep freezing namely. Since only undamaged and healthy fruits can be processed, it is important to determine maximum levels of mechanical loading, thus to determine the modulus of elasticity [1]. The modulus of elasticity can be determined by many nondestructive methods. One of them is based on measuring of the acoustic wave propagation [2].

The most precise method of the object dimensions measuring is 3D scanning [3], but this method is very expensive, demands a lot of time and manual work, and finally produces huge amount of data. Digging the main shape parameters represents rather complicated problem. This article shows the procedure of the approximation of the peach shape by triaxial ellipsoid and using of three digital photos. These photos must be taken in mutually perpendicular orientations, see Figure 1.

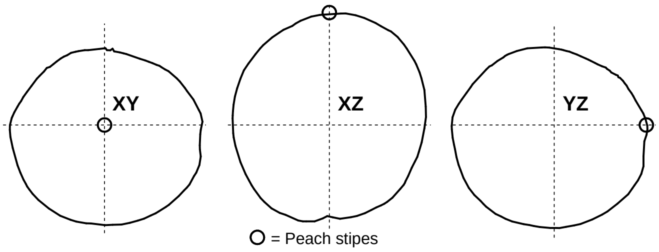


Fig. 1: *Basic peach photos.*

*This research has been supported by the Grant Agency of the Czech Academy of Sciences under Contract No. IAA201990701, *Behaviour of selected agricultural products under impact loading.*

There are many computer codes intended for the extraction of pixels coordinates from the objects perimeter. The approach described in [4] was used in this paper. Coordinates of the perimeter's pixels were saved in three independent files. Used algorithm reduces the number of pixels and orders them in such way, that resulting polygon approximates the perimeter with the accuracy better than ±1 pixel. This method reduces the number of pixels from thousands into hundreds of polygon vertices without loss of accuracy.

2. Coordinate system

Let us assume that a peach will be approximated by a triaxial ellipsoid with the axis A, B, and C in the spherical coordinate system. If the peach is properly oriented, see Figure 1, photos can be interpreted as equatorial cross section - XY, and cut along 0° meridian cross section - XZ and 90° meridian - YZ. The cross section curves may be approximated by mutually dependent ellipses. Their semiaxes will be A, B for XY, A, C for XZ and B, C for YZ. It must be assumed that these ellipses will have different centers and may be rotated by a small angle around different centers. The following description will be used: $[Ax, Ay]$ = temporary unknown center of the ellipse approximating the XY cut, Ra - rotation angle of the approximating ellipse around center. Variables $[Bx, Bz]$, Rb, $[Cy, Cz]$ and Rc have the same meaning for cross sections XZ and YZ. Parametric shape will be used to describe all three ellipses. The first ellipse will be described by the function $[A\cos(fa), B\sin(fa)]$, the second one by $[A\cos(fb), C\sin(fb)]$ and the third one by $[B\cos(fc), C\sin(fc)]$, where fa, fb, fc are mutually independent parameters, $-\pi \leq fa, fb, fc \leq \pi$. $[X, Y]$, $[Y, Z]$ and $[Y, Z]$ are coordinates of the polygon vertices, saved in three different files. Using these variables we can determine the difference between coordinates of the general vertex point and the nearest corresponding point on the ellipse. We shall use the least squares method (LSQ) to compute all unknown variables which will minimize $Qxy + Qxz + Qyz$, where:

$$\begin{aligned} Qxy = \ \Sigma_{i=1}^{Nxy} \ & (A\cos(fa_i) - (X_i - Ax)\cos(Ra) - (Y_i - Ay)\sin(Ra))^2 \\ & + (B\sin(fa_i) + (X_i - Ax)\sin(Ra) - (Y_i - Ay)\cos(Ra))^2 \ , \end{aligned} \tag{1}$$

$[X_i, Y_i]$ are coordinates of the polygon vertices of the XY cut and fa_i are parameters of the points laying on the ellipse approximating the polygon. Qxz and Qyz are defined very similarly.

3. Preparation of the iteration

Equation (1) is non-linear for all unknowns. It means that an iteration method based on the linearization must be used. One of the best approaches derived by Newton and improved by Gauss is the well-known Gauss-Newton method. Purpose of the presented article is not to explain this method, but to show how to use it on a complicated and large problem. To handle matrices, the MAPLE library "LinearAlgebra" is used. The library "plots" simplifies the visualization of results. The approach presented in this article is as generalization of [5] and [6].

```
> restart; with(LinearAlgebra): with(plots):
```

In order to limit the scope, only commands and variables describing the coordinate difference between a vertex and the corresponding point on the ellipse for XY cross section will be described. Other MAPLE commands are very similar.

```
> xa:=A*cos(fa)-((X-Ax)*cos(Ra)+(Y-Ay)*sin(Ra)):
> ya:=B*sin(fa)+((X-Ax)*sin(Ra)-(Y-Ay)*cos(Ra)):
> #  similarly xb, zb, yc, zc
```

The difference between origin of the series expansion and the actual variable value will be substituted as a new variable. This variable has D at the beginning of its name. The value of this variable will represent the correction for the corresponding variable during the iteration process.

```
> Dsu:=map(u->u-cat(u||0)=cat(D||u),Var);
> Mxa:=subs(Dsu,Mxa); # similarly Mya, Mxb, Mzb, Myc, Mzc
```

$$\begin{aligned} Mxa \;:=\; & A0\cos(fa0)-(Y-Ay0)\sin(Ra0)-(X-Ax0)\cos(Ra0)+DA\cos(fa0)\\ & -A0\sin(fa0)Dfa+DAx\cos(Ra0)+DAy\sin(Ra0)\\ & +((X-Ax0)\sin(Ra0)-(Y-Ay0)\cos(Ra0))DRa \end{aligned}$$

Complications arise with the parameters. For each vertex/point the best possible initial approximation is needed. If we have N points, we have to find N corresponding initial parameters. That is why correction for parameters from the Taylor series must be separated. Now it is possible to create a lists containing terms multiplying the individual differences.

```
> JFxa:=select(has,Mxa,Dfa)/Dfa: # similarly JFya ... JFzc
> JXa:=map(u->select(has,Mxa,u)/u,map(u->rhs(u),Dsu[4..-1]));
> # similarly JYa ... JZc
```

$$JXa := [\cos(fa0), 0, 0\cos(Ra0), 0, 0, 0, 0, (X-Ax0)\sin(Ra0)-(Y-Ay0)\cos(Ra0), 0, 0]$$

Finally, separation of free terms is necessary.

```
> RXa:=remove(has,Mxa,map(u->rhs(u),Dsu)); # similarly RYa .. RZc
```

$$RXa := A0\cos(fa0)-(Y-Ay0)\sin(Ra0)-(X-Ax0)\cos(Ra0)$$

4. Initial values iteration

At this point, the files containing coordinates of vertices can be read. The whole polygon will be moved, so its mass center will be in the center of the coordinate system. Na, Nb, Nc represents the number of vertices in the individual cross sections and Ta, Tb, Tc are corresponding centers of their masses.

```
> read "007a_3.sav": Na:=nops(XYL):
> Ta:=sum(XYL[i],i=1..Na)/Na: XY:=map(u->u-Ta,XYL):
> # similarly 007b.sav --> Nb, Tb, XZ  and  007c.sav --> Nc, Tc, YZ;
> 'Na'=Na, 'Nb'=Nb, 'Nc'=Nc, 'Ta'=Ta, 'Tb'=Tb, 'Tc'=Tc;
```

$$Na = 92,\ Nb = 80\,Nc = 96,$$
$$Ta = [407.744, 331.126],\ Tb = [446.333, 324.705],\ Tc = [400.202, 363.912]$$

The initial values for axes may be determined as average of maximal diameters of the individual cross sections and arguments of the vertices in the polar coordinate system may be used as initial values of parameters. If photos were taken carefully the initial values for displacements and rotations may be zero.

```
> A0:=0.5*(max(map(u->u[1],XY)[])+max(map(u->u[1],XZ)[])):
> Fa:=map(u->op(2,polar(u[1]+I*u[2])),XY):
> # similarly XY, YZ --> B0; XZ, YZ --> C0; XZ --> Fb; YZ --> Fc
> Ax0:=0: Ay0:=0: Bx0:=0: Bz0:=0: Cy0:=0: Cz0:=0: Ra0:=0: Rb0:=0: Rc0:=0:
```

5. Iteration

The most important variable in the iteration process is the Jacobi Matrix J. This matrix is composed from 24 submatrices, 12 of them are zero matrices, see construction of the Maple variable J. Its construction shows the Figure 2 and exactly corresponds with the structure of the command line for the variable J. Non zero elements are shown as crosses. It can be seen, that the Jacobi matrix is very sparse, partially filled up columns corespond to corrections of the variables [A, B, C, Ax, Ay, Bx, Bz, Cy, Cz, Ra, Rb, Rc].

The vector of right sides contains variable R and vector DV is vector of corrections of initial values of searched variables. The variable *eps* controls whole iteration, which runs until *eps* is less than 10^{-5}. The quadratic norm of the vector of corrections DV is assigned into *eps*. To reduce the big steps between corrections, corrections will be reduced to one half. Because at first few passes through loop corrections the parameters may be higher than 2π, it is necessary to recompute all parameters into range $< -\pi, \pi >$. Subsequently, the corrections to all other variables are added.

```
> eps:=1: while eps>1e-5 do;
>  J:=Matrix([[DiagonalMatrix(map(u->subs(fa0=u,JFxa),Fa)),ZeroMatrix(Na,Nb),
>   ZeroMatrix(Na,Nc), Matrix(zip((u,v)->subs(fa0=u,X=v[1],Y=v[2],JXa),Fa,XY))],
>   [DiagonalMatrix(map(u->subs(fa0=u,JFya),Fa)),ZeroMatrix(Na,Nb),
>   ZeroMatrix(Na,Nc),Matrix(zip((u,v)->subs(fa0=u,X=v[1],Y=v[2],JYa),Fa,XY))],
>   [ZeroMatrix(Nb,Na),DiagonalMatrix(map(u->subs(fb0=u,JFxb),Fb)),
>   ZeroMatrix(Nb,Nc), Matrix(zip((u,v)-subs(fb0=u,X=v[1],Z=v[2],JXb),Fb,XZ))],
>   [ZeroMatrix(Nb,Na),DiagonalMatrix(map(u->subs(fb0=u,JFzb),Fb)),
```

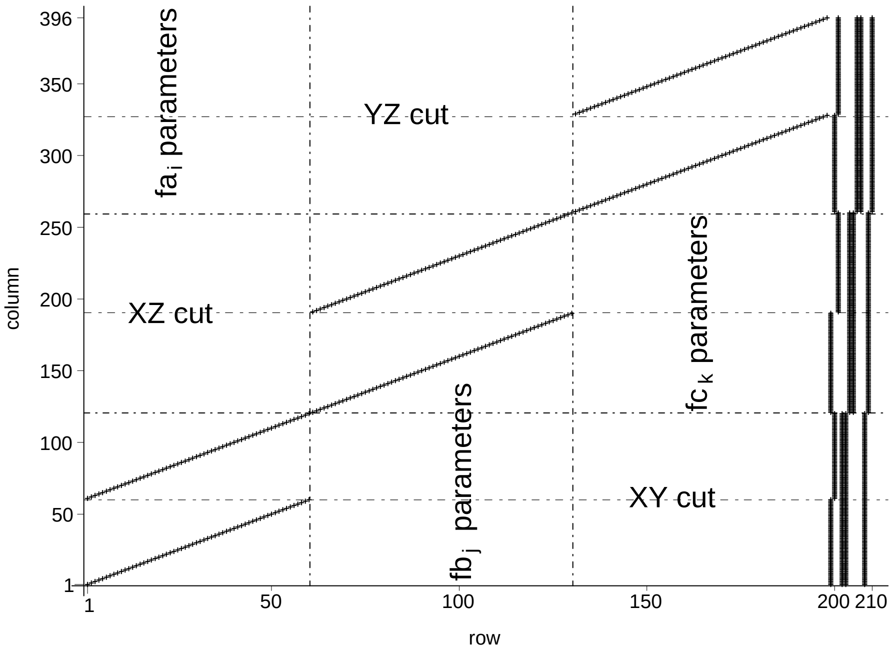


Fig. 2: *Structure of the Jacobi matrix.*

```
>   ZeroMatrix(Nb,Nc),Matrix(zip((u,v)->subs(fb0=u,X=v[1],Z=v[2],JZb),Fb,XZ))],
>   [ZeroMatrix(Nc,Na),ZeroMatrix(Nc,Nb),DiagonalMatrix(map(u->subs(fc0=u,JFyc),
>   Fc)),Matrix(zip((u,v)->subs(fc0=u,Y=v[1],Z=v[2],JYc),Fc,YZ))],
>   [ZeroMatrix(Nc,Na),ZeroMatrix(Nc,Nb),DiagonalMatrix(map(u->subs(fc0=u,JFzc),
>   Fc)),Matrix(zip((u,v)->subs(fc0=u,Y=v[1],Z=v[2],JZc),Fc,YZ))]]):
>  R:=-Vector([zip((u,v)->subs(fa0=u,X=v[1],Y=v[2],RXa),Fa,XY)[],
>   zip((u,v)->subs(fa0=u,X=v[1],Y=v[2],RYa),Fa,XY)[],zip((u,v)->subs(fb0=u,
>   X=v[1],Z=v[2],RXb),Fb,XZ)[],zip((u,v)->subs(fb0=u,X=v[1],Z=v[2],RZb),
>   Fb,XZ)[],zip((u,v)->subs(fc0=u,Y=v[1],Z=v[2],RYc),Fc,YZ)[],
>   zip((u,v)->subs(fc0=u,Y=v[1],Z=v[2],RZc),Fc,YZ)[]]):
>  DV:=LeastSquares(J,R):eps:=Norm(DV,2):DV:=0.5*DV;
>  Fa:=Fa+convert(DV[1..Na],list):Fb:=Fb+convert(DV[1+Na..Na+Nb],list):
>  Fc:=Fc+convert(DV[1+Na+Nb..Na+Nb+Nc],list):A0:=A0+DV[Na+Nb+Nc+1];
>  B0:=B0+DV[Na+Nb+Nc+2]; C0:=C0+DV[Na+Nb+Nc+3];Ax0:=Ax0+DV[Na+Nb+Nc+4];
>  Ay0:=Ay0+DV[Na+Nb+Nc+5]; Bx0:=Bx0+DV[Na+Nb+Nc+6];Bz0:=Bz0+DV[Na+Nb+Nc+7];
>  Cy0:=Cy0+DV[Na+Nb+Nc+8]; Cz0:=Cz0+DV[Na+Nb+Nc+9];Ra0:=Ra0+DV[Na+Nb+Nc+10];
>  Rb0:=Rb0+DV[Na+Nb+Nc+11]; Rc0:=Rc0+DV[Na+Nb+Nc+12];
>  Fa:=map(u->op(2,polar(cos(u)+I*sin(u))),Fa):Fb:=map(u->op(2,polar(cos(u)+
>  I*sin(u))),Fb):Fc:=map(u->op(2,polar(cos(u)+I*sin(u))),Fc):
>  Ra0:=op(2,polar(cos(Ra0)+I*sin(Ra0))):Rb0:=op(2,polar(cos(Rb0)+I*sin(Rb0))):
>  Rc0:=op(2,polar(cos(Rc0)+I*sin(Rc0))):
> end do:
```

6. Results interpretation

The accuracy of the approximation of individual cross sections can be displayed, see Figure 3.

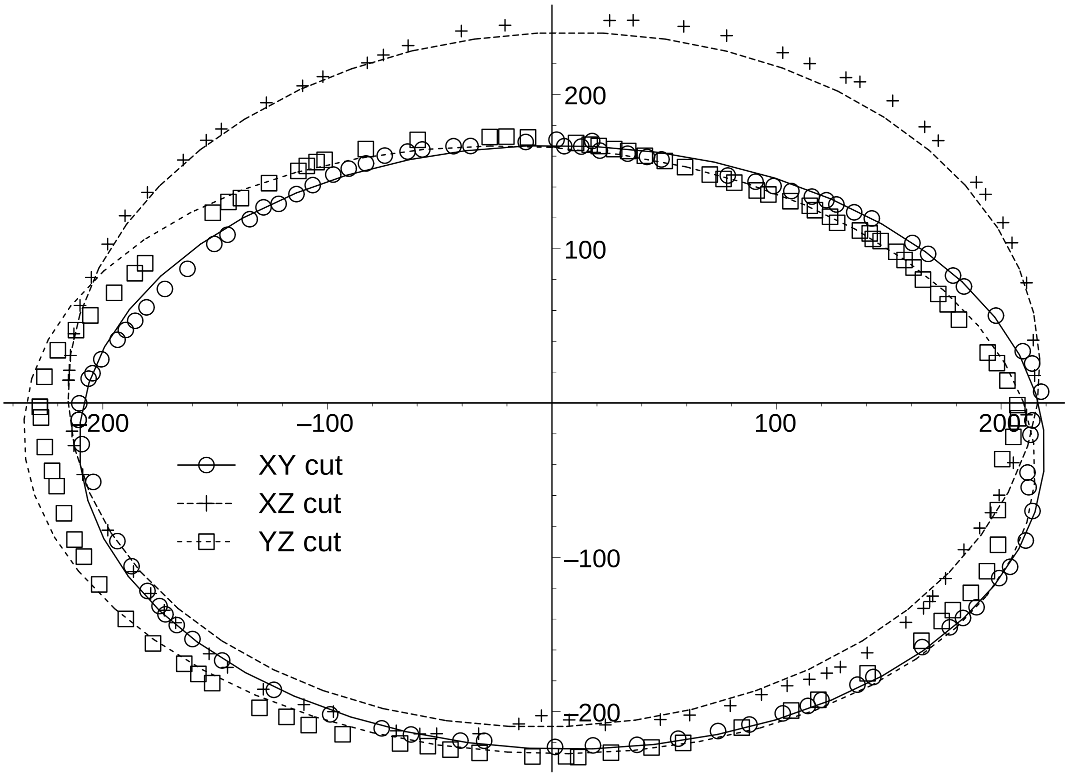


Fig. 3: *Accuracy of the individual cross sections.*

```
> Xa:=A0*cos(f); Ya:=B0*sin(f):
> Xra:=cos(Ra0)*Xa-sin(Ra0)*Ya+Ax0: Yra:=sin(Ra0)*Xa+cos(Ra0)*Ya+Ay0:
> # similarly A0,C0,f-->Xb,Zb-->Xrb,Zrb;  B0,C0,f-->Yc,Zc-->Yrc,Zrc;
> P1:=plot(XY,style=point,symbol=circle,symbolsize=20,color=black):
> P2:=plot([Xra,Yra,f=0..2*Pi],color=black,thickness=2,thickness=2,linestyle=1):
> # similarly XZ,symbol=cross --> P3, Xrb,Zrb,linestyle=3 --> P4
> # similarly YZ,symbol=box   --> P5, Yrc,Zrc,linestyle=4 --> P6
> display({P1,P2,P3,P4,P5,P6});
```

Correlation between vertex points and corresponding points on the individual ellipses and the average of the relative displacements can be computed.

```
> R1:=map(u->sqrt(u[1]^2+u[2]^2),XY):
> R2:=map(u->evalf(subs(f=u,sqrt(Xra^2+Yra^2))),Fa):
> DR:=zip((u,v)->abs(u-v),R1,R2): AR:=zip((u,v)->0.5*(u+v),R1,R2):
> RRa:=zip((u,v)->u/v,DR,AR): MRa:=sum(RRa[i],i=1..Na)/Na:
> CRa:=stats[describe,linearcorrelation](R1,R2):
> # similarly XZ-->R1, Fb-->R2, R1,R2-->Mrb,Crb
> # similarly YZ-->R1, Fc-->R2, R1,R2-->Mrc,Crc
> Correlations:=[CRa,CRb,CRc];
> Mean_relative_displacement:=[MRa,MRb,MRc];
```

$$Correlations := [0.992, 0.948, 0.084]$$
$$Mean\_relative\_displacements := [0.012, 0.021, 0.021]$$

We see that mean displacements are close to 2%, the highest individual displacement is lower than 7%, and the correlation coefficient between vertices and corresponding points on the individual ellipses is higher than 90%. These results enable to declare the approximation as successfull. Finally, 3D plot showing triaxial ellipsoid with projections of the cross sections can be displayed, see Figure 3.

```
> PCH:=plot3d([A0*cos(f)*cos(l),B0*cos(f)*sin(l),C0*sin(f)],
>  l=-Pi..Pi,f=-Pi/2..Pi/2,style=wireframe,color=grey):
> Sa:=spacecurve(map(u->[cos(Ra0)*(u[1]-Ax0)+sin(Ra0)*(u[2]-Ay0),
>   -sin(Ra0)*(u[1]-Ax0)+cos(Ra0)*(u[2]-Ay0),0],XY),
>   color=black,thickness=3,linestyle=1):
> # similarly Bx0,Bz0,Rb0,XZ,linestyle=3 --> Sb
> # similarly Cy0,Cz0,Rc0,YZ,linestyle=4 --> Sc
> display({PCH,Sa,Sb,Sc},scaling=constrained,axes=boxed);
```

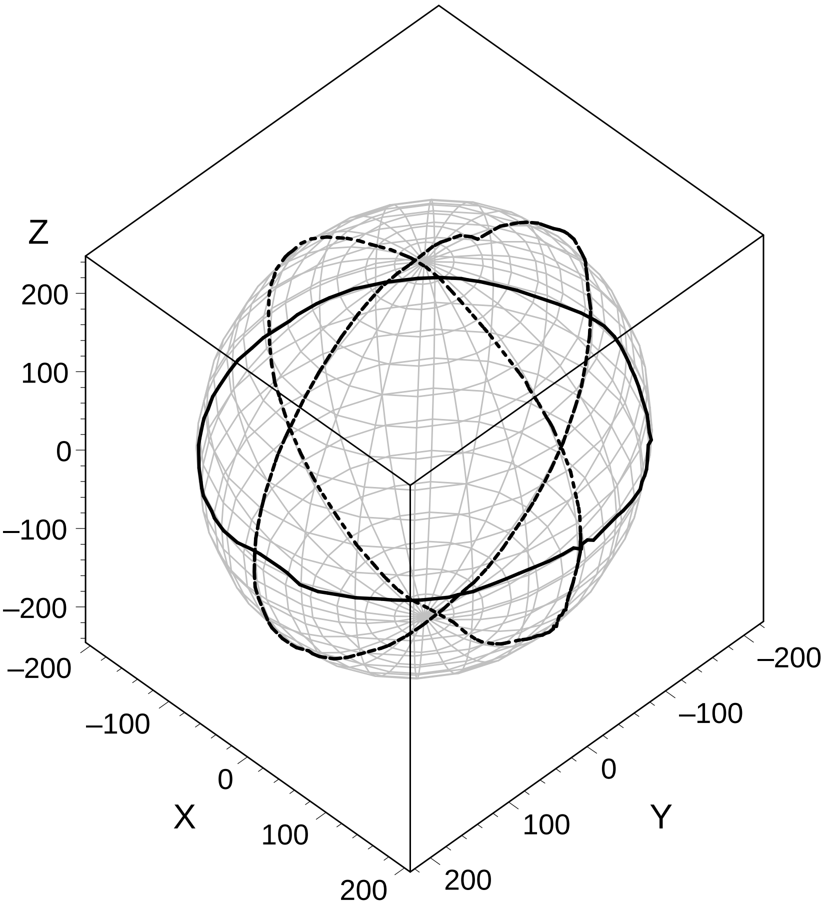


Fig. 4: *3D Peach approximation.*

7. Conclusions

Regarding above stated high correlation coefficients and relative displacements, and with the reference to the results shown in Figures 3 and 4, it can be concluded that presented approach provides excellent results. It means that it can be used as a tool for the space approximation of simple three-dimensional objects, such as peaches, apricots, apples, cucumbers, etc.

References

[1] J. Buchar, Š. Nedomová, L. Severa: *High strain rate behaviour of peaches.* Proceedings of the 2008 SEM XI International Congress and Exposition on Experimental and Applied Mechanics, USA, Society for Experimental Mechanics, Inc., 2008, pp. 163–170. ISBN 0-912053-99-2.

[2] P. Dvořáková, Š. Nedomová, L. Severa, J. Trnka, J. Buchar: *The impulse response method for measuring the overall firmness of fruit.* Book of Contributions of 46th International Scientific Conference "EXPERIMENTAL STRESS ANALYSIS 2008". Ostrava, Czech Republic: VŠB – Technical University of Ostrava, 2008, pp. 43–47. ISBN 978-80-248-1774-3.

[3] L. Severa: *Shape and strength of Red Haven peaches at the different stages of their maturity.* Acta univ. agric. et silvic. Mendel. Brun., 2008, LVI, No. 4, pp. 169–177. ISSN 1211-8516.

[4] S. Bartoň: In: J. Diblík, (Ed.), *Shape determination of an agricultural product* Proceedings of 6. Math. workshop, Brno, FAST VUT Brno, 2007, pp. 1–12. ISBN 80-214-2741-8.

[5] W. Gander and S. Bartoň: *Least Squares Fit with Piecewise Functions.* In: W. Gander, J. Hřebíček (Eds.), Solving problems in Scientific Computing using Maple and Matlab, Heidelberg, Springer, 2004, pp. 433–450. ISBN 3-540-21127-6.

[6] W. Gander and S. Bartoň: *Metod najmensich kvadratov dlja kusocno-nepreryvnych funkcij.* In: W. Gander, J. Hřebíček (Eds.), Resenie zadac v naucmych vycislenijach s primeneniem Maple i Matlab, Minsk, Vassamedia, 2005, pp. 417–442. ISBN 985-6642-06-X.

NUMERICAL MODELING OF NEUTRON FLUX IN HEXAGONAL GEOMETRY[*]

Tomáš Berka, Marek Brandner, Milan Hanuš, Roman Kužel, Aleš Matas

1. Introduction

Our concern in this paper is the *neutron flux* in the *VVER type nuclear reactors.* A nuclear reactor is composed of the fuel assemblies. An important feature of the VVER type nuclear reactors is that their fuel assemblies have *hexagonal shape.*

The *transport theory* and the *diffusion theory* are the two general ways how to model the neutron flux. In this paper we particularly study the *two-dimensional two-group neutron diffusion model.* We formulate the mathematical model in Section 2.

In the key Section 3 we present a modern variant of the *CMFD-nodal methods* suited particularly for solving the neutron diffusion equation on a hexagonal mesh. The method is build upon the *conformal mapping.* CMFD-nodal methods employ the technique called *transverse integration.* When applied to a hexagonal mesh, certain singular terms arise. Wagner's approach ([5] and [3]) is to simply ignore these terms. The approach involving conformal mapping gives more accurate results.

In Section 5 we introduce the integral part of nodal methods, the technique of *homogenization.* It determines how to transform the general equations with variable coefficients to the equations with node-wise constant coefficients. Such a procedure is based on conditions of preserving certain physical quantities.

2. Mathematical model

In the active zone of a nuclear reactor we consider the two-group neutron diffusion model. It is a *generalized eigenvalue problem* and can be written in the following way

$$\begin{aligned} \nabla\cdot\mathbf{j}^1(\mathbf{x})+\Sigma_r^1(\mathbf{x})\phi^1(\mathbf{x}) &= \frac{1}{k_{\text{eff}}}\left[\nu\Sigma_f^1(\mathbf{x})\phi^1(\mathbf{x})+\nu\Sigma_f^2(\mathbf{x})\phi^2(\mathbf{x})\right] \overset{\text{def.}}{=} s_1(\mathbf{x}),\\ \nabla\cdot\mathbf{j}^2(\mathbf{x})+\Sigma_r^2(\mathbf{x})\phi^2(\mathbf{x}) &= \Sigma_s^{1\to 2}(\mathbf{x})\phi^1(\mathbf{x}) \overset{\text{def.}}{=} s_2(\mathbf{x}). \end{aligned} \tag{1}$$

The unknown quantities in (1) are the *neutron flux* ϕ^g (eigenfunction, $g = 1, 2$) and the *reactor critical number* k_{eff} (inverse of the largest eigenvalue). The superscript g corresponds to the energy group. The term $\mathbf{j}^g$ is the *neutron current* and we link it

[*]This work has been supported by the projects MSM4977751301 and 1M0545.

to the neutron flux ϕ^g through the following constitutive relation, which is called the *Fick's law*

$$\mathbf{j}^g(\mathbf{x}) = -D^g(\mathbf{x})\nabla\phi^g(\mathbf{x}). \tag{2}$$

The other terms present in (1) are given. They characterize certain material properties of the fuel assemblies.

At the boundary of the active zone of a nuclear reactor it is usual to consider the *albedo boundary conditions* of the form

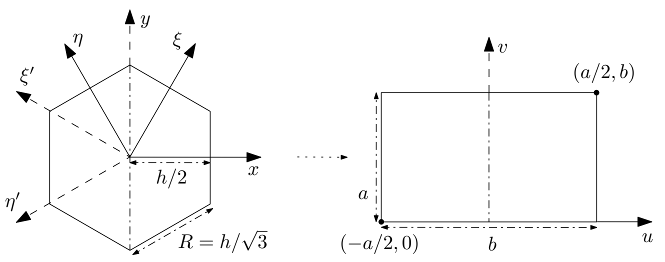
$$\gamma\phi^1(\mathbf{x}) - \mathbf{j}^1(\mathbf{x})\cdot\mathbf{n}(\mathbf{x}) = 0, \quad \gamma\phi^2(\mathbf{x}) - \mathbf{j}^2(\mathbf{x})\cdot\mathbf{n}(\mathbf{x}) = 0, \tag{3}$$

where γ is a given albedo coefficient.

3. Conformal mapping method

For numerical solution of the problem (1)–(3), we use the *conformal mapping method*. This method is a variant of the *CMFD-nodal methods* which are a combination of the CMFD (Coarse Mesh Finite Differences) and the *two-node problems* (see [5], [3]). CMFD is just a finite volume method applied to the physical node-wise discretization mesh of an active zone. It gives us the neutron fluxes averaged over particular nodes, six values of neutron currents averaged over the nodal faces and the critical number. Two-node problems are solved semianalytically and give us more accurate surface currents, which we use to correct the iteration matrix of the CMFD. These two general steps are repeated until convergence is achieved. The CMFD is a nonlinear procedure in this context and is intended to accelerate convergence of the whole process.

Conformal mapping method is designed specifically for hexagonal meshes, where it is more accurate then classical CMFD-nodal methods. The CMFD part is identical to another CMFD-nodal methods. Solution of the two-node problems is based on use of the conformal mapping, particularly the Schwarz-Christoffel transformation, which maps a complex half-plane onto the interior of a polygon. We utilize this transformation to map the interior of a hexagon in a complex plane $z = x + \mathrm{i}y$ onto the interior of a *rectangle* in a complex plane $w = u + \mathrm{i}v$ (see Fig. 1). Construction of the mentioned mapping is described in detail in [2].



a) b)

Fig. 1: *Conformally mapped hexagon onto a rectangle.*

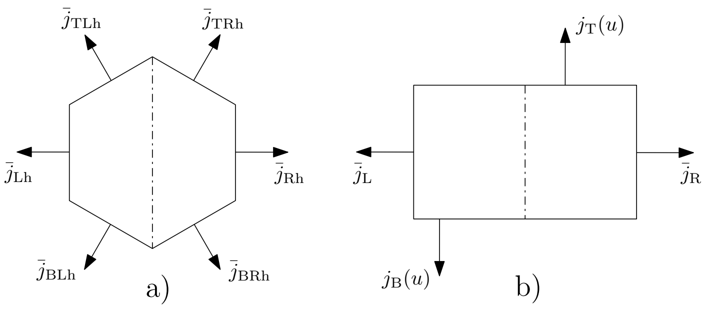


Fig. 2: *The notation of averaged currents.*

Below, we use the following notation; the quantities with the subscript "h" correspond to a hexagon, the other correspond to a rectangle. The subscripts "R", "L", "T", "B" stand for "right", "left", "top", "bottom" halves of a hexagon or a rectangle respectively; see Fig. 2 for the notation of averaged currents on the boundary of a hexagon and a rectangle. The over-bar denotes surface averaged quantities. The tilde denotes transverse integrated quantities (see Section 3.1).

3.1. Two-node problems

We solve three two-node problems corresponding to the directions x, ξ and η (see Fig. 1a) for the pairs of currents $(\bar{j}_{\mathrm{Lh}}, \bar{j}_{\mathrm{Rh}})$, $(\bar{j}_{\mathrm{TRh}}, \bar{j}_{\mathrm{BLh}})$ and $(\bar{j}_{\mathrm{TLh}}, \bar{j}_{\mathrm{BRh}})$ respectively. These steps are performed successively. We present the x-step. The other steps are analogous.

In the following, we omit the group index. If we transform the first equation in (1) valid over the interior of a hexagonal node $\mathcal{H}_i$ from the coordinate system xy to the coordinate system uv we obtain

$$-D\Delta\phi(u,v) + \Sigma_r g^2(u,v)\phi(u,v) = g^2(u,v)s(u,v) \tag{4}$$

valid over a rectangle $\mathcal{R}_i$. The second equation in (1) can be proceeded in the same way. The function $g(u,v)$ is defined on a hexagonal node as

$$g(u,v) = \left|\frac{\mathrm{d}z}{\mathrm{d}w}\right|. \tag{5}$$

The term $1/g^2(u,v)$ appears in the image of the Laplace operator as a weighting factor.

Now we apply the *transverse integration* on (4) and obtain

$$-D\frac{\mathrm{d}^2\tilde{\phi}(u)}{\mathrm{d}u^2} + \Sigma_r \tilde{g^2}(u)\tilde{\phi}(u) = \tilde{g^2}(u)\tilde{s}(u) - l_v(u), \tag{6}$$

where

$$\tilde{\phi}(u) = \frac{1}{b}\int_0^b \phi(u,v)\,\mathrm{d}v, \quad \tilde{g^2}(u) = \frac{\frac{1}{b}\int_0^b g^2(u,v)\phi(u,v)\,\mathrm{d}v}{\tilde{\phi}(u)} \sim \frac{1}{b}\int_0^b g^2(u,v)\,\mathrm{d}v. \tag{7}$$

The approximation of the term $\tilde{g}^2(u)$ is discussed in [2].

The *transverse leakage* $l_v(u)$ on a rectangle in the direction v at a point u is defined as follows

$$l_v(u) = \frac{1}{b}\big(j_{\mathrm{T}}(u) - j_{\mathrm{B}}(u)\big). \tag{8}$$

There are several ways how to approximate this term. The simplest method is to consider *constant leakage on the hexagon half-nodes*. It can be expressed on a rectangle as

$$l_{v\mathrm{L}}(u) = \frac{1}{b}g(u,0)(\bar{j}_{\mathrm{TLh}} - \bar{j}_{\mathrm{BLh}}), \quad l_{v\mathrm{R}}(u) = \frac{1}{b}g(u,0)(\bar{j}_{\mathrm{TRh}} - \bar{j}_{\mathrm{BRh}}). \tag{9}$$

The other possibility is to assume the transverse leakage *linear on the boundary half-nodes* and *constant on the inner half-nodes*, which is more complicated. We need to determine two unknowns for a linear function. We acquire them from the global boundary condition information and from the condition of preservation of the total leakage on the boundary half-nodes (see [6]).

3.2. Semi-analytic solution

We seek for a solution of (6) in the following form

$$\tilde{\phi}(u) = \underbrace{a_0p_0(u) + a_1p_1(u) + a_2p_2(u)}_{\text{particular}} + \underbrace{a_3\sinh(ku) + a_4\cosh(ku)}_{\text{homogeneous}}, \tag{10}$$

where

$$k = \sqrt{\frac{\tilde{g}^2(a/2)\Sigma_r}{D}}. \tag{11}$$

The solution (10) has two parts. The homogeneous part is the approximation of the analytical solution to the homogeneous part of the equation. The particular part is sought in the space of quadratic polynomials (p_0, p_1, p_2) in the weighted residue sense and solves approximately the complete inhomogeneous equation.

The solution has five unknown coefficients. One of them is determined from the condition of preserving the CMFD's node averaged flux. Another three are given by matching the zeroth, first, and second moment conditions of the weighted residue. The last one arises from the continuity of the flux and current at the interface of two adjacent nodes. On the boundary, we employ the boundary condition instead (see e.g. [3]).

The surface averaged currents at the points $u = \pm a/2$ can be derived from (10) and (2) as follows

$$\bar{j}_{\mathrm{L}} = \tilde{j}(u = -a/2) = -D\frac{\mathrm{d}\tilde{\phi}(u)}{\mathrm{d}u}\bigg|_{u=-a/2}, \tag{12}$$

$$\bar{j}_{\mathrm{R}} = \tilde{j}(u = +a/2) = -D\frac{\mathrm{d}\tilde{\phi}(u)}{\mathrm{d}u}\bigg|_{u=+a/2}. \tag{13}$$

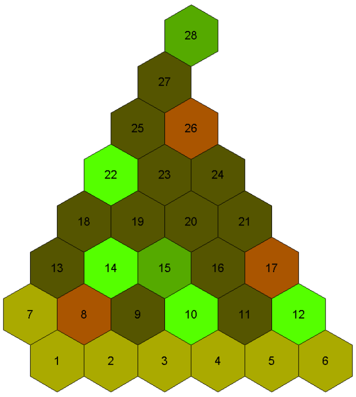
It can be proved (after proper normalization of $g(u,v)$) that the following relations between the currents on a hexagon and the currents on a rectangle hold true

$$\bar{j}_{\mathrm{Lh}} = \frac{b}{R}\bar{j}_{\mathrm{L}}, \quad \bar{j}_{\mathrm{Rh}} = \frac{b}{R}\bar{j}_{\mathrm{R}}. \tag{14}$$

The quantities $\bar{j}_{\mathrm{Lh}}$ and $\bar{j}_{\mathrm{Rh}}$ are used to correct the CMFD iteration matrix.

4. Benchmark

We undertook a numerical test on a sample configuration of a VVER-440 reactor, the experiment parameters are detailed in Benchmark no. 6 in [1]. On the picture below we see the scheme of the configuration of one sixth of the core, which is one sixth rotationally symmetric. The reactor critical number k_{eff} and the *node averaged power distributions* (PD, derived from the neutron fluxes) were computed using the approach described in previous sections with constant leakage (code of Roman Kužel). Then we compared the results with accurate result of the FVM on a fine mesh (code of Milan Hanuš).



- $k_{\mathrm{eff}} = 1.01447$
- k_{eff} error $= -7.89 \times 10^{-5}$
- PD avg. error = 1.48%
- PD max. error = 1.93%

5. Homogenization

As soon as analytical approach is to be used for solving neutron diffusion equation, it is necessary to transform the general diffusion equation with variable coefficients to an equation with node-wise constant coefficients. We call the original problem heterogeneous and the latter homogeneous. Such a transformation is possible through the technique of *homogenization*. Generally, it means that we require certain physical quantities from the heterogeneous problem to be preserved in the solution of the homogeneous problem. In this way, we acquire the constant coefficients. Let us consider only one energy group. The other groups can be proceeded in a similar way (see [4]).

We denote quantities corresponding to the homogeneous problem with a hat over a quantity. The other correspond to the heterogeneous problem. The symbol i stands for the index of a node.

5.1. Principle

Our main goal is to preserve the reactor critical number. We proceed with the one-group (for more groups, see [4]) integral formulation of the system (1) in a node $\mathcal{H}_i$

$$\sum_{k=1}^{6} \int_{\Gamma_{i,k}} \mathbf{j}(\mathbf{x}) \cdot \mathbf{n}_{i,k} \, \mathrm{d}S + \int_{\mathcal{H}_i} \Sigma_r(\mathbf{x})\phi(\mathbf{x}) \, \mathrm{d}\mathbf{x} = \frac{1}{k_{\text{eff}}} \nu \int_{\mathcal{H}_i} \Sigma_f(\mathbf{x})\phi(\mathbf{x}) \, \mathrm{d}\mathbf{x}, \tag{15}$$

where $\Gamma_{i,k}$ is a k-th surface ($k = 1, \ldots, 6$) of a hexagonal node $\mathcal{H}_i$. Preservation of the reactor critical number can be directly fulfilled by (15) if we define the constant coefficients, such that the *reaction rates* and the *surface averaged currents* are preserved

$$\int_{\mathcal{H}_i} \hat{\Sigma}^i_{(.)}\hat{\phi}(\mathbf{x}) \, \mathrm{d}\mathbf{x} = \int_{\mathcal{H}_i} \Sigma_{(.)}(\mathbf{x})\phi(\mathbf{x}) \, \mathrm{d}\mathbf{x} \quad \Rightarrow \quad \hat{\Sigma}^i_{(.)} = \frac{\int_{\mathcal{H}_i} \Sigma_{(.)}(\mathbf{x})\phi(\mathbf{x}) \, \mathrm{d}\mathbf{x}}{\int_{\mathcal{H}_i} \hat{\phi}(\mathbf{x}) \, \mathrm{d}\mathbf{x}},$$

$$\int_{\Gamma_{i,k}} \hat{\mathbf{j}}(\mathbf{x}) \cdot \mathbf{n}_{i,k} \, \mathrm{d}S = \int_{\Gamma_{i,k}} \mathbf{j}(\mathbf{x}) \cdot \mathbf{n}_{i,k} \, \mathrm{d}S \quad \Rightarrow \quad \hat{D}^i = \frac{\int_{\Gamma_{i,k}} \mathbf{j}(\mathbf{x}) \cdot \mathbf{n}_{i,k} \, \mathrm{d}S}{-\int_{\Gamma_{i,k}} \nabla\hat{\phi}(\mathbf{x}) \cdot \mathbf{n}_{i,k} \, \mathrm{d}S}.$$

5.2. Approximate theories

In the principle described above, we assume that the heterogeneous and the homogeneous solutions of the problem are known. They provide the homogenized parameters. In practice, we do not know the heterogeneous solution. At the beginning of the calculation, we also do not know the homogeneous solution. Therefore, we have to define the homogenized coefficients in a different way using *approximate homogenization theories* based on the flux-weighted constants (FWC) and on the general equivalence theory (GET), see [4], described in the following sections.

Instead of computing the heterogeneous solution of the global problem, we calculate the solution of the *one-node problems* for different types of assemblies. It means that we do not take into account the geometry of the active zone and we have to choose some suitable boundary conditions. Usually we impose zero net current at the boundary of an assembly. Then, we consider such solution as heterogeneous (exact) and use it to determine the homogenized parameters.

Another possibility is to consider the *seven-node problem.* To determine the homogenized parameters of some node $\mathcal{H}_i$ we solve the heterogeneous problem on the domain composed of $\mathcal{H}_i$ and its six neighbors. The zero net current BCs are prescribed on the surfaces of the neighbors. This approach leads to higher accuracy of the computed homogenized parameters of the considered node. It is necessary to take into account position of $\mathcal{H}_i$ in the active zone and to solve the seven-node problem for each node in the reactor.

5.3. Flux-weighted constants (FWC)

This is the simplest method for computing the homogenized coefficients. We consider two general approximations. The first is the following

$$\int_{\mathcal{H}_i} \phi(\mathbf{x})\, \mathrm{d}\mathbf{x} \approx \int_{\mathcal{H}_i} \hat{\phi}(\mathbf{x})\, \mathrm{d}\mathbf{x}. \tag{16}$$

The other approximation is the definition of the homogenized diffusion coefficient arising from the transport theory as follows

$$\frac{1}{\hat{D}^i} = \frac{\int_{\mathcal{H}_i} \phi(\mathbf{x})/D^g(\mathbf{x})\, \mathrm{d}\mathbf{x}}{\int_{\mathcal{H}_i} \phi(\mathbf{x})\, \mathrm{d}\mathbf{x}}. \tag{17}$$

The crucial problem of FWC is that we cannot determine one diffusion coefficient for one node, such that *surface averaged fluxes are continuous on each nodal surface.* For a thourough analysis of the problem, see [4].

5.4. General equivalence theory (GET)

This theory withdraws the problem of FWC by relaxing the continuity of the fluxes on the interfaces of the nodes. For this purpose, we introduce the new homogenization parameters called the *factors of discontinuity* corresponding to a node $\mathcal{H}_i$ on the surfaces $(\mathcal{H}_i, \mathcal{H}_{i+1})$ and $(\mathcal{H}_i, \mathcal{H}_{i-1})$ (considering direction x) as follows

$$f_i^{x+} = \frac{\int_{\Gamma_{i,x+}} \phi(\mathbf{x})\, \mathrm{d}S}{\int_{\Gamma_{i,x+}} \hat{\phi}(\mathbf{x})\, \mathrm{d}S}, \quad f_i^{x-} = \frac{\int_{\Gamma_{i,x-}} \phi(\mathbf{x})\, \mathrm{d}S}{\int_{\Gamma_{i,x-}} \hat{\phi}(\mathbf{x})\, \mathrm{d}S}. \tag{18}$$

To eliminate the unknown homogeneous solution from the definition of f_i^{x+} (analogically for f_i^{x-}) the following equalities must hold

$$\int_{\Gamma_{i,k}} \hat{\phi}(\mathbf{x})\, \mathrm{d}S = \int_{\mathcal{H}_i} \hat{\phi}(\mathbf{x})\, \mathrm{d}\mathbf{x} = \int_{\mathcal{H}_i} \phi(\mathbf{x})\, \mathrm{d}\mathbf{x}. \tag{19}$$

The first equality follows from the fact that the homogeneous solution of the one-node problems is constant. We can rewrite (18) for f_i^{x+} (analogically for f_i^{x-}) in the following way

$$f_i^{x+} = \frac{\int_{\Gamma_{i,x+}} \phi(\mathbf{x})\, \mathrm{d}S}{\int_{\Gamma_{i,x+}} \hat{\phi}(\mathbf{x})\, \mathrm{d}S} = \frac{\int_{\Gamma_{i,x+}} \phi(\mathbf{x})\, \mathrm{d}S}{\int_{\mathcal{H}_i} \phi(\mathbf{x})\, \mathrm{d}\mathbf{x}}. \tag{20}$$

At this point we substitute continuity of the flux on an interface with the *discontinuity condition*

$$f_i^{x+} \int_{\Gamma_{i,x+}} \hat{\phi}(\mathbf{x})\, \mathrm{d}S = f_{i+1}^{x-} \int_{\Gamma_{i+1,x-}} \hat{\phi}(\mathbf{x})\, \mathrm{d}S. \tag{21}$$

As we relaxed continuity of the flux, we can choose the diffusion coefficient entirely arbitrarily in so far it has a physical meaning. It is reasonable to use the FWC choice (17).

5.5. Algorithm

Solution procedure of the whole reactor problem together with homogenization can be summarized in the following steps:

1. Calculate the homogenized parameters from the one-node problems for different types of assemblies or from seven-node problems for the whole reactor.
2. Calculate the homogeneous solution of the whole reactor problem.
3. Calculate the homogenized parameters from the one-node or seven-node problems for the whole reactor. Use the surface currents calculated in the step 2 or 4 as a new boundary condition for these problems.
4. Calculate the homogeneous solution of the whole reactor problem.
5. Check the convergence of the homogenized parameters by inspection of two successive approximations. If the difference is too large continue from step 3.

Usually two iterations of the whole algorithm are sufficient to achieve required accuracy.

References

[1] Y.A. Chao, Y.A. Shatilla: *Conformal mapping and hexagonal nodal methods – II: Implementation in the ANC-H code.* Nuclear Science and Engineering, **121**, 1995, 210–225.

[2] Y.A. Chao, N. Tsoulfanidis: *Conformal mapping and hexagonal nodal methods – I: Mathematical foundation.* Nuclear Science and Engineering, **121**, 1995, 202–209.

[3] M. Hanuš: *Numerical modelling of neutron flux in nuclear reactors.* Bachelor's thesis, Univ. of West Bohemia, Pilsen, 2007.

[4] K.S. Smith: *Spatial homogenization methods for light water reactor analysis.* PhD Thesis, Nuclear Engineering, Massachusetts Institute of Technology, 1980.

[5] M.R. Wagner: *Three-dimensional nodal diffusion and transport theory methods for hexagonal-z geometry.* Nuclear Science and Engineering, **103**, 1989, 377–391.

[6] V.G. Zimin, D.M. Baturin: *Polynomial nodal method for solving neutron diffusion equations in hexagonal-z geometry.* Annals of Nuclear Energy, **29**, 2002, 303–335.

NUMERICAL MODELLING OF RIVER FLOW (NUMERICAL SCHEMES FOR ONE TYPE OF NONCONSERVATIVE SYSTEMS)*

Marek Brandner, Jiří Egermaier, Hana Kopincová

Abstract

In this paper we propose a new numerical scheme to simulate the river flow in the presence of a variable bottom surface. We use the finite volume method, our approach is based on the technique described by D. L. George for shallow water equations. The main goal is to construct the scheme, which is well balanced, i.e. maintains not only some special steady states but all steady states which can occur. Furthermore this should preserve nonnegativity of some quantities, which are essentially nonnegative from their physical fundamental, for example the cross section or depth. Our scheme can be extended to the second order accuracy.

1. Introduction

We are interested in solving the problem describing the fluid flow through the channel with the general cross-section area

$$\begin{aligned} a_t + q_x &= 0, \\ q_t + \left(\frac{q^2}{a} + gI_1\right)_x &= -gaB_x + gI_2, \end{aligned} \tag{1}$$

where $a = a(x,t)$ is the unknown cross-section area, $q = q(x,t)$ is the unknown discharge, $B = B(x)$ is the function of elevation of the bottom, g is the gravitational constant and

$$I_1 = \int_0^{h(x)} [h(x) - \eta]\sigma(x,\eta)d\eta, \tag{2}$$

$$I_2 = \int_0^{h(x)} (h-\eta)\left[\frac{\partial \sigma}{\partial x}\right] d\eta, \tag{3}$$

where η is the depth integration variable, h is the water depth and $\sigma(x,\eta)$ is the width of the cross-section at the depth η.

The special case are the equations reflecting the fluid flow through the varying rectangular channel

$$\begin{aligned} a_t + q_x &= 0, \\ q_t + \left(\frac{q^2}{a} + \frac{ga^2}{2l}\right)_x &= \frac{ga^2}{2l^2} l_x - gab_x, \end{aligned} \tag{4}$$

*This work was supported by the Research Plan MSM 4977751301 and by Moravian-Silesian region.

or the system with constant rectangular channel

$$\begin{aligned} h_t + (hu)_x &= 0, \\ (hu)_t + \left(hu^2 + \frac{1}{2}gh^2\right)_x &= -ghB_x, \end{aligned} \tag{5}$$

where $h(x,t)$ is the water depth and $u(x,t)$ is the horizontal velocity.

All of the presented systems can be briefly written in the matrix form

$$\mathbf{q}_t + [\mathbf{f}(\mathbf{q})]_x = \boldsymbol{\psi}(\mathbf{q}, x), \tag{6}$$

where $\mathbf{q}(x,t)$ is the vector of conserved quantities, $\mathbf{f}(\mathbf{q})$ is the flux function and $\boldsymbol{\psi}(\mathbf{q}, x)$ is the source term.

There are many numerical schemes for solving (6) with different properties and possibilities of failing. For example central, upwind or central-upwind schemes. The main requirements on the numerical schemes are the consistency (in the finite volume meaning: consistency with flux function), the conservativity (if there is possibility to rewrite the problem to the conservative form it is required to have conservative numerical scheme), positive semidefiniteness, i.e. the schemes preserve nonnegativity of some quantities, which are essentially nonnegative from their physical fundamental, and the well-balancing, i.e. the schemes maintain some or all steady states which can occur. The next properties are the order of the schemes, stability and the convergence. There are, of course, related conditions to provide mentioned requirements, for example so called CFL (Courant-Friedrichs-Levy) stability condition.

2. Augmented formulations

There are several ways how to formulate the fluid flow problems. Homogeneous, autonomous, conservative formulation, which is used for standard cases, like Euler equations or fluid flow through the channel with constant cross-section and flat bottom have the form

$$\begin{aligned} \mathbf{q}_t + [\mathbf{f}(\mathbf{q})]_x &= \mathbf{0},\ x \in \mathbf{R},\ t \in (0,T), \\ \mathbf{q}(x,0) &= \mathbf{q}_0(x),\ x \in \mathbf{R}, \end{aligned} \tag{7}$$

where $\mathbf{q} = \mathbf{q}(x,t) :\ \mathbf{R} \times \langle 0,T) \to \mathbf{R}^m$, $\mathbf{q}_0 = \mathbf{q}_0(x) :\ \mathbf{R} \to \mathbf{R}^m$, $\mathbf{f} = \mathbf{f}(\mathbf{q}) :\ \mathbf{R}^m \to \mathbf{R}^m$. This formulation corresponds to (6) with zero right hand side.

The homogeneous, nonautonomous, conservative case

$$\begin{aligned} \mathbf{q}_t + [\mathbf{f}(\mathbf{q}, \mathbf{w}(x))]_x &= \mathbf{0},\ x \in \mathbf{R},\ t \in (0,T), \\ \mathbf{q}(x,0) &= \mathbf{q}_0(x),\ x \in \mathbf{R}, \end{aligned} \tag{8}$$

where $\mathbf{w} = \mathbf{w}(x) : \mathbf{R} \to \mathbf{R}^s$ is a given function.

The system (8) can be rewritten to the homogeneous, autonomous, conservative formulation (we add the equation $\mathbf{w}_t = \mathbf{0}$)

$$\begin{aligned} \tilde{\mathbf{q}}_t + [\tilde{\mathbf{f}}(\tilde{\mathbf{q}})]_x &= \mathbf{0},\ x \in \mathbf{R},\ t \in (0,T), \\ \tilde{\mathbf{q}}(x,0) &= \tilde{\mathbf{q}}_0(x),\ x \in \mathbf{R}, \end{aligned} \tag{9}$$

where $\tilde{\mathbf{q}} = [\mathbf{q}, \tilde{\mathbf{w}}]^T$, $\tilde{\mathbf{f}}(\tilde{\mathbf{q}}) = [\mathbf{f}(\mathbf{q}, \tilde{\mathbf{w}}), \mathbf{0}]^T$ and $\tilde{\mathbf{q}}_0(x) = [\mathbf{q}_0(x), \mathbf{w}(x)]^T$.

Now we consider the system in the form (nonhomogeneous, autonomous case)

$$\begin{aligned} \mathbf{q}_t + [\mathbf{f}(\mathbf{q}, \mathbf{w}(x))]_x &= \mathbf{B}(\mathbf{q}, \mathbf{w}(x))\mathbf{w}_x,\ x \in \mathbf{R},\ t \in (0,T), \\ \mathbf{q}(x,0) &= \mathbf{q}_0(x),\ x \in \mathbf{R}, \end{aligned} \tag{10}$$

where $\mathbf{B} = \mathbf{B}(\mathbf{q}, \mathbf{w})$ is the matrix function of the type $m \times s$.
In the case of the river flow (4) this augmented formulation has following form

$$\mathbf{q} = [a, q]^T, \quad \mathbf{w}(x) = [l(x), b(x)]^T,$$
$$\mathbf{f}(\mathbf{q}, \mathbf{w}) = [q, \tfrac{q^2}{a} + \tfrac{ga^2}{2l}]^T,$$
$$\mathbf{B}(\mathbf{q}, \mathbf{w}(x)) = \begin{bmatrix} 0 & 0 \\ \frac{ga^2}{2l^2} & -ga \end{bmatrix}.$$

We can rewrite the previous system to the augmented, homogeneous, autonomous, quasilinear formulation

$$\begin{aligned} \tilde{\mathbf{q}}_t + \mathbf{C}(\tilde{\mathbf{q}})\tilde{\mathbf{q}}_x &= \mathbf{0},\ x \in \mathbf{R},\ t \in (0,T), \\ \tilde{\mathbf{q}}(x,0) &= \tilde{\mathbf{q}}_0(x),\ x \in \mathbf{R}, \end{aligned} \tag{11}$$

where

$$\mathbf{C}(\tilde{\mathbf{q}}) = \begin{bmatrix} \mathbf{f_q} & \mathbf{f_w} - \mathbf{B}(\mathbf{q}, \tilde{\mathbf{w}}) \\ \mathbf{0} & \mathbf{0} \end{bmatrix},$$

The following relation holds $\mathbf{f}_x = \mathbf{f_q}\mathbf{q}_x + \mathbf{f_w}\mathbf{w}_x$.

The next extension can be done by adding another equation in the form

$$[\mathbf{f}(\mathbf{q})]_t + \mathbf{f_q}[\mathbf{f}(\mathbf{q}, \mathbf{w}(x))]_x - \mathbf{f_q}\mathbf{B}(\mathbf{q}, \mathbf{w}(x))\mathbf{w}_x = \mathbf{0}.$$

The previous relation provides some theoretical insight into how the flux behaves.

The overdetermined system has now the form

$$\begin{aligned} \hat{\mathbf{q}}_t + \hat{\mathbf{D}}(\hat{\mathbf{q}})\hat{\mathbf{q}}_x &= \mathbf{0},\ x \in \mathbf{R},\ t \in (0,T), \\ \hat{\mathbf{q}}(x,0) &= \hat{\mathbf{q}}_0(x),\ x \in \mathbf{R}, \end{aligned} \tag{12}$$

where $\hat{\mathbf{q}} = [\mathbf{q}, \tilde{\mathbf{w}}, \hat{\mathbf{f}}]^T$,

$$\hat{\mathbf{D}}(\hat{\mathbf{q}}) = \begin{bmatrix} \mathbf{f_q} & \mathbf{f_w} - \mathbf{B}(\mathbf{q}, \tilde{\mathbf{w}}) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{f_q}\mathbf{B}(\mathbf{q}, \tilde{\mathbf{w}}) & \mathbf{f_q} \end{bmatrix},$$

where $\hat{\mathbf{q}}(x) = [\mathbf{q}_0(x), \mathbf{w}(x), \mathbf{f}(\mathbf{q}_0(x), \mathbf{w}(x))]^T$. The advantage of this formulation is in the conversion of the nonhomogeneous systems to the homogeneous one. For our model of the river flow the matrix has the form

$$\hat{\mathbf{D}}(\hat{\mathbf{q}}) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ \frac{-q^2}{a^2} + \frac{ga}{l} & \frac{2q}{a} & \frac{-ga^2}{l^2} & ga & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{-ga^2}{2l^2} & ga & 0 & 1 \\ 0 & 0 & \frac{-gqa}{l^2} & 2gq & \frac{-q^2}{a^2} + \frac{ga}{l} & \frac{2q}{a} \end{bmatrix},$$

where $\hat{\mathbf{q}} = [a, q, l, b, q, \frac{q^2}{a} + \frac{ga^2}{2l}]$. The second and fifth equations have the same unknown quantity, so the fifth equation can be rejected.

Now we can formulate new problem in the form

$$\begin{aligned} \check{\mathbf{q}}_t + \check{\mathbf{D}}(\check{\mathbf{q}})\check{\mathbf{q}}_x &= \mathbf{0}, \ x \in \mathbf{R}, \ t \in (0, T), \\ \check{\mathbf{q}}(x, 0) &= \check{\mathbf{q}}_0(x), \ x \in \mathbf{R}, \end{aligned} \tag{13}$$

where for our model of the river flow the matrix has the form

$$\check{\mathbf{D}}(\check{\mathbf{q}}) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ \frac{-q^2}{a^2} + \frac{ga}{l} & \frac{2q}{a} & \frac{-ga^2}{l^2} & ga & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{-q^2}{a^2} + \frac{ga}{l} & \frac{-gqa}{l^2} & 2gg & \frac{2q}{a} \end{bmatrix}, \tag{14}$$

and $\check{\mathbf{q}} = [a, q, l, b, \frac{q^2}{a} + \frac{ga^2}{2l}]^T$.

3. Finite volume methods

The finite volume methods are suitable for solving conservation laws, because the numerical solution is modified only by the intercell fluxes. These methods are based on the integral formulation of the problem. They use approximation of the integral averages of the unknown function instead of the approximations of the unknown functions. And the consistency of these methods is related to the flux function. See [6].

We define the following discretisation of the volume and time

$$\begin{aligned} &x_j = j\Delta x, \quad j \in \mathbf{Z}, \quad \Delta x > 0, \qquad && t_n = n\Delta t, \quad n \in \mathbf{N}_0, \quad \Delta t > 0, \\ &x_{j+1/2} = x_j + \Delta x/2, && t_{n+1/2} = t_n + \Delta t/2. \end{aligned}$$

We denote the conserved quantities at time t^n and point x_j: $\mathbf{q}_j^n = \mathbf{q}(x_j, t_n)$ and $\mathbf{q}_j(t) = \mathbf{q}(x_j, t)$ and its approximations: $\mathbf{Q}_j^n = \mathbf{Q}(x_j, t_n) \approx \mathbf{q}_j^n$, and $\mathbf{Q}_j(t) = \mathbf{Q}(x_j, t) \approx \mathbf{q}_j(t)$. The finite volumes mean the sets $(x_{j-1/2}, x_{j+1/2}) \times (t^n, t^{n+1})$.

We denote the integral averages of the conserved quantities over the finite volume

$$\bar{\mathbf{Q}}_j^n \approx \bar{\mathbf{q}}_j^n = \frac{1}{\Delta x} \int\limits_{x_{j-1/2}}^{x_{j+1/2}} \mathbf{q}(x, t_n)\ dx, \tag{15}$$

and the average flux along $x = x_{j+1/2}$

$$\bar{\mathbf{F}}_{j+1/2}^{n+1/2} \approx \bar{\mathbf{f}}_{j+1/2}^{n+1/2} = \frac{1}{\Delta t} \int\limits_{t_n}^{t_{n+1}} \mathbf{f}(\mathbf{q}(x_{j+1/2}, t))\ dt. \tag{16}$$

Fully discrete conservative method can be written as relation between approximations of the flux averages and approximations of the integral averages of the conserved quantities

$$\bar{\mathbf{Q}}_j^{n+1} = \bar{\mathbf{Q}}_j^n - \frac{\Delta t}{\Delta x}(\bar{\mathbf{F}}_{j+1/2}^{n+1/2} - \bar{\mathbf{F}}_{j-1/2}^{n+1/2}). \tag{17}$$

Sometimes it is useful to consider the discretisation in two steps. First step is discretisation only in the space (interval $(x_{j-1/2}, x_{j+1/2})$)

$$\bar{\mathbf{Q}}_j = \bar{\mathbf{Q}}_j(t) \approx \bar{\mathbf{q}}_j = \bar{\mathbf{q}}_j(t) = \frac{1}{\Delta x} \int\limits_{x_{j-1/2}}^{x_{j+1/2}} \mathbf{q}(x, t)\ dx. \tag{18}$$

This leads to the system of the ordinary differential equations in the time

$$\frac{d}{dt}\bar{\mathbf{Q}}_j = -\frac{1}{\Delta x}[\mathbf{F}_{j+1/2} - \mathbf{F}_{j-1/2}]. \tag{19}$$

4. Steady states

The steady states mean that the unknown quantities do not change in the time, i.e. $\mathbf{q_t} = \mathbf{0}$ and the flux function must balance the right hand side $[\mathbf{f}(\mathbf{q})]_x = \boldsymbol{\psi}(\mathbf{q}, x)$. For the augmented systems this means that, for example $\mathbf{D}(\hat{\mathbf{q}})\hat{\mathbf{q}}_x = \mathbf{0}$.

Some schemes are constructed to preserve some special steady states like so called rest at lake, i.e. there is no motion and the free surface height is constant:

$$q(x, t) = 0, \quad h(x, t) + b(x) = \text{const}. \tag{20}$$

This steady state has following form for our model (4)

$$q(x, t) = 0, \ \left(\frac{q^2}{a} + \frac{ga^2}{2l}\right)_x - \frac{ga^2}{2l^2} l_x + gab_x = 0. \tag{21}$$

Under the assumption $a = hl$ the mentioned relations can be rewritten into the form

$$ghl(h + b)_x = 0.$$

For general steady states the following equalities hold

$$q_x = 0, \quad \left(\frac{q^2}{a} + \frac{ga^2}{2l}\right)_x = \frac{ga^2}{2l^2} l_x - gab_x, \tag{22}$$

the left term in the second equality we can rewrite as

$$\left(\frac{q^2}{a} + \frac{ga^2}{2l}\right)_x = \left(-u^2 + \frac{ga}{l}\right) a_x - \frac{ga^2}{2l^2} l_x, \tag{23}$$

and together we have

$$\left(-u^2 + \frac{ga}{l}\right) a_x = \frac{ga^2}{l} l_x - gab_x. \tag{24}$$

From (24) we obtain the following relation for general steady states (the Bernoulli equation)

$$\left(\frac{1}{2}u^2 + gb + \frac{ga}{l}\right)_x = 0. \tag{25}$$

For numerical methods it is important to choose such approximation which conserved these steady states. The equation (25) means that the term $\frac{1}{2}u^2 + gb + \frac{ga}{l}$ is constant for differentiable steady states. Therefore following property has to be satisfied

$$\left(\frac{1}{2}u^2 + gb + \frac{ga}{l}\right)_j = \left(\frac{1}{2}u^2 + gb + \frac{ga}{l}\right)_{j+1}. \tag{26}$$

We rearrange (26) and we can express the discrete relation analogous to the smooth one $\phi_x = \left(-u^2 + \frac{ga}{l}\right) a_x - \frac{ga^2}{2l^2} l_x$

$$\Delta\Phi = \left(-|U_L U_R| + g\frac{\bar{A}\bar{L}}{L_L L_R}\right)\Delta A - \frac{g}{2}\frac{\tilde{A}^2}{L_L L_R}\Delta L, \tag{27}$$

where $X_L = \bar{X}_j$, $X_R = \bar{X}_{j+1}$, $\Delta X = X_R - X_L$, X represents U, L and A, $\bar{L} = (L_L + L_R)/2$, $\bar{A} = (A_L + A_R)/2$, $\tilde{A}^2 = (A_L^2 + A_R^2)/2$. The details can be found in [1].

5. Central methods

The central methods are universal schemes for solving hyperbolic partial differential equation, see [5]. In these schemes there is not necessary to construct the characteristic decomposition of the flux f nor to compute the approximation of the Jacobian matrix. These schemes are Riemann problem free. They are robust but they are characterized by large numerical diffusion.

One example is the first-order Lax-Friedrichs scheme

$$\bar{\mathbf{Q}}_j^{n+1} = \frac{1}{2}(\bar{\mathbf{Q}}_{j-1}^n + \bar{\mathbf{Q}}_{j+1}^n) - \frac{\Delta t}{2\Delta x}[\mathbf{f}(\bar{\mathbf{Q}}_{j+1}^n) - \mathbf{f}(\bar{\mathbf{Q}}_{j-1}^n)], \tag{28}$$

where the flux function for the conservative form can be written in the form

$$\mathbf{F}^{n+1/2}_{j+1/2} = \frac{1}{2}[\mathbf{f}(\bar{\mathbf{Q}}^n_j) + \mathbf{f}(\bar{\mathbf{Q}}^n_{j+1})] - \frac{\Delta x}{2\Delta t}(\bar{\mathbf{Q}}^n_{j+1} - \bar{\mathbf{Q}}^n_j). \tag{29}$$

For our model describing fluid flow through the constant rectangular channel

$$\begin{aligned} h_t + q_x &= 0, \\ q_t + \left(\tfrac{q^2}{h} + \tfrac{1}{2}gh^2\right)_x &= -ghb_x. \end{aligned}$$

We substitute $y = h + b$ and then we can write

$$\begin{aligned} y_t + q_x &= 0, \\ q_t + \left(\tfrac{q^2}{y-b} + \tfrac{1}{2}g(y-b)^2\right)_x &= -g(y-b)b_x. \end{aligned} \tag{30}$$

The special steady state “rest at lake” means $y(x,t) = \text{const}$ and $q(x,t) = 0$.

The flux function and discretisation of the right hand side are in the form

$$\begin{aligned} F^{n,1}_{j+1/2} &= \tfrac{1}{2}(\bar{Q}^n_j + \bar{Q}^n_{j+1}) - \tfrac{\Delta x}{2\Delta t}(\bar{Y}^n_{j+1} - \bar{Y}^n_j), \\ F^{n,2}_{j+1/2} &= \tfrac{1}{2}\left[\tfrac{(\bar{Q}^n_j)^2}{\bar{Y}^n_j - \bar{B}_j} + \tfrac{(\bar{Q}^n_{j+1})^2}{\bar{Y}^n_{j+1} - \bar{B}_{j+1}} + \tfrac{1}{2}g(\bar{Y}^n_j - \bar{B}_j) + \right. \\ &+ \left. \tfrac{1}{2}g(\bar{Y}^n_{j+1} - \bar{B}_{j+1})\right] - \tfrac{\Delta x}{2\Delta t}(\bar{Q}^n_{j+1} - \bar{Q}^n_j), \\ S^{1,n}_j &= 0, \\ S^{n,2}_j &= -\tfrac{g}{4\Delta x}(\bar{B}_{j+1} - \bar{B}_j) \\ & \quad (\bar{Y}^n_{j+1} - \bar{B}_{j+1} + \bar{Y}^n_j - \bar{B}_j + \bar{Y}^n_j - \bar{B}_j + \bar{Y}^n_{j-1} - \bar{B}_{j-1}). \end{aligned}$$

This scheme preserves only special steady state “rest at lake”. But in general these methods are not suitable for computation steady states [7]. One of their big disadvantages is the relatively large numerical dissipation.

The next type of the central method is for example Rusanovov scheme in semidiscrete form

$$\begin{aligned} \tfrac{d}{dt}\bar{\mathbf{Q}}_j = -\tfrac{1}{2\Delta x}[\mathbf{f}(\bar{\mathbf{Q}}_{j+1}) - \mathbf{f}(\bar{\mathbf{Q}}_{j-1})] + \tfrac{1}{2\Delta x}[\hat{a}_{j+1/2}(\bar{\mathbf{Q}}_{j+1} - \bar{\mathbf{Q}}_j) - \\ -\hat{a}_{j-1/2}(\bar{\mathbf{Q}}_j - \bar{\mathbf{Q}}_{j-1})], \end{aligned} \tag{31}$$

where

$$\hat{a}_{j+1/2} = \max_p\{\max\{\lambda^p_j, \lambda^p_{j+1}\}\}.$$

This scheme can be written in the conservative form (19) where the numerical fluxes have the form

$$\mathbf{F}_{j+1/2} = \frac{1}{2}[\mathbf{f}(\bar{\mathbf{Q}}_j) + \mathbf{f}(\bar{\mathbf{Q}}_{j+1})] - \frac{1}{2}|\hat{a}_{j+1/2}|(\bar{\mathbf{Q}}_{j+1} - \bar{\mathbf{Q}}_j).$$

And as will be mentioned in the next section, this scheme can be rewritten in the fluctuation form.

6. Upwind methods

6.1. Scalar case

In this subsection we consider the equation

$$\begin{aligned} q_t + aq_x &= 0,\ x \in R,\ t \in (0,T),\ a \in R, \\ q(x,0) &= q_0(x),\ x \in R. \end{aligned} \tag{32}$$

This advection equation has known solution $q(x,t) = q_0(x - at)$. Usually the REA algorithm (reconstruct-evolve-average) is used for the solution. This algorithm is based on the piecewise polynomial reconstruction of the solution from the quantities $\bar{Q}_j(t)$. This reconstruction we denote $\hat{Q}_j(x,t)$ for $x \in (x_{j-1/2}, x_{j+1/2})$. This reconstruction is considered to be the initial condition for solving sets of the Riemann problems (in this case we can use the form of the solution).

Using the forward differences for time discretisation in the semidiscrete scheme (19) the numerical flux has the form

$$F_{j+1/2} = \frac{1}{2}a(Q^-_{j+1/2} + Q^+_{j+1/2}) - \frac{1}{2}|a|(Q^+_{j+1/2} - Q^-_{j+1/2}), \tag{33}$$

where

$$Q^+_{j+1/2} = \hat{Q}_{j+1}(x_{j+1/2}+, t),\ Q^-_{j+1/2} = \hat{Q}_j(x_{j+1/2}-, t).$$

This scheme can be rewritten into so called fluctuation form

$$\frac{d\bar{Q}_j}{dt} = \frac{-1}{\Delta x}(a^-\Delta Q_{j+1/2} + a\Delta Q_j + a^+\Delta Q_{j-1/2}), \tag{34}$$

where fluctuations are defined

$$\begin{aligned} a\Delta Q_j &= a(Q^-_{j+1/2} - Q^+_{j-1/2}), \\ a^-\Delta Q_{j+1/2} &= a^-(Q^+_{j+1/2} - Q^-_{j+1/2}), \\ a^+\Delta Q_{j-1/2} &= a^+(Q^+_{j-1/2} - Q^-_{j-1/2}), \end{aligned}$$

where $a^+ = \max\{a, 0\}$, $a^- = \min\{a, 0\}$.

For simple piecewise constant reconstruction $Q^+_{j+1/2} = \bar{Q}_{j+1}$, $Q^-_{j+1/2} = \bar{Q}_j$ we obtain for $a > 0$

$$\frac{d}{dt}\bar{Q}_j = -\frac{a}{\Delta x}(\bar{Q}_j - \bar{Q}_{j-1}), \tag{35}$$

and for $a < 0$

$$\frac{d}{dt}\bar{Q}_j = -\frac{a}{\Delta x}(\bar{Q}_{j+1} - \bar{Q}_j). \tag{36}$$

6.2. Linear systems

Now we consider linear system

$$\begin{aligned} \mathbf{q}_t + \mathbf{A}\mathbf{q}_x &= \mathbf{0},\ x \in R,\ t \in (0,T), \\ \mathbf{q}(x,0) &= \mathbf{q}_0(x),\ x \in R, \end{aligned} \tag{37}$$

where $\mathbf{A}$ is real matrix $m \times m$. We suppose that the matrix $\mathbf{A}$ has distinct real eigenvalues and is diagonalisable i.e. exists regular matrix $\mathbf{R}$ such that $\mathbf{\Lambda} = \mathbf{R}^{-1}\mathbf{A}\mathbf{R}$, where $\mathbf{\Lambda}$ is diagonal matrix. So we can rewrite (37) to the form

$$\boldsymbol{\gamma}_t + \mathbf{\Lambda}\boldsymbol{\gamma}_x = \mathbf{0}, \tag{38}$$

where $\boldsymbol{\gamma}(x,t) = \mathbf{R}^{-1}\mathbf{q}(x,t)$. The system (38) represents m advection equations which can be solved analogously as in the scalar case.

After rewriting the system (37) to conservation form, where $\mathbf{f}(\mathbf{q}) = \mathbf{A}\mathbf{q}$, and solving sets of the generalized Riemann problems we get the numerical fluxes in the form

$$\mathbf{F}_{j+1/2} = \frac{1}{2}\mathbf{A}(\mathbf{Q}^-_{j+1/2} + \mathbf{Q}^+_{j+1/2}) - \frac{1}{2}|\mathbf{A}|(\mathbf{Q}^+_{j+1/2} - \mathbf{Q}^-_{j-1/2}), \tag{39}$$

and in analogy to the previous section we can write the conservative scheme in the fluctuation form

$$\frac{d\bar{\mathbf{Q}}_j}{dt} = \frac{-1}{\Delta x}(\mathbf{A}^-\Delta\mathbf{Q}_{j+1/2} + \mathbf{A}\Delta\mathbf{Q}_j + \mathbf{A}^+\Delta\mathbf{Q}_{j-1/2}), \tag{40}$$

where

$$\begin{aligned}
\mathbf{A}\Delta\mathbf{Q}_j &= \mathbf{A}(\mathbf{Q}^-_{j+1/2} - \mathbf{Q}^+_{j-1/2}),\\
\mathbf{A}^-\Delta\mathbf{Q}_{j+1/2} &= \sum_{p=1}^{m} \lambda^{-,p}\Delta\gamma^p_{j+1/2}\mathbf{r}^p,\\
\mathbf{A}^+\Delta\mathbf{Q}_{j-1/2} &= \sum_{p=1}^{m} \lambda^{+,p}\Delta\gamma^p_{j-1/2}\mathbf{r}^p,\\
\Delta\mathbf{Q}_{j+1/2} &= \sum_{p=1}^{m} \Delta\gamma^p_{j+1/2}\mathbf{r}^p,\\
\Delta\mathbf{Q}_j &= \mathbf{Q}^-_{j+1/2} - \mathbf{Q}^+_{j-1/2},
\end{aligned}$$

$\mathbf{A}^+ = \mathbf{R}\mathbf{\Lambda}^+\mathbf{R}^{-1}$, $\mathbf{A}^- = \mathbf{R}\mathbf{\Lambda}^-\mathbf{R}^{-1}$, $\mathbf{\Lambda}^+ = \mathrm{diag}(\max\{\lambda^p, 0\})$, $\mathbf{\Lambda}^- = \mathrm{diag}(\min\{\lambda^p, 0\})$, $|\mathbf{\Lambda}| = \mathrm{diag}(|\lambda^p|)$, $\Delta\boldsymbol{\gamma}_{j+1/2} = \mathbf{R}^{-1}_{j+1/2}\Delta\mathbf{Q}_{j+1/2}$.

6.3. Nonlinear systems

Now we consider nonlinear system

$$\begin{aligned}
\mathbf{q}_t + [\mathbf{f}(\mathbf{q})]_x &= \mathbf{0},\ x \in R,\ t \in (0,T),\\
\mathbf{q}(x,0) &= \mathbf{q}_0(x),\ x \in R.
\end{aligned} \tag{41}$$

The fluctuation form of the conservative scheme is as follows

$$\frac{d\bar{\mathbf{Q}}_j}{dt} = \frac{-1}{\Delta x}[\mathbf{A}^-(\Delta\mathbf{Q}_{j+1/2}) + \mathbf{A}(\Delta\mathbf{Q}_j) + \mathbf{A}^+(\Delta\mathbf{Q}_{j-1/2})], \tag{42}$$

$$\begin{aligned}
\mathbf{A}(\Delta\mathbf{Q}_j) &= \mathbf{f}(\mathbf{Q}^-_{j+1/2}) - \mathbf{f}(\mathbf{Q}^+_{j-1/2}),\\
\mathbf{A}^-(\Delta\mathbf{Q}_{j+1/2}) &= \mathbf{F}^-_{j+1/2} - \mathbf{f}(\mathbf{Q}^-_{j+1/2}),\\
\mathbf{A}^+(\Delta\mathbf{Q}_{j-1/2}) &= \mathbf{f}(\mathbf{Q}^+_{j-1/2}) - \mathbf{F}^+_{j-1/2}.
\end{aligned}$$

This scheme can be written in the form

$$\frac{d}{dt}\bar{\mathbf{Q}}_j = -\frac{1}{\Delta x}[\mathbf{F}^-_{j+1/2} - \mathbf{F}^+_{j-1/2}]. \tag{43}$$

The fluctuations have following property based on the Rankine-Hugoniot condition for the discontinuities

$$\mathbf{f}(\mathbf{Q}^+_{j+1/2}) - \mathbf{f}(\mathbf{Q}^-_{j+1/2}) = \mathbf{A}^+(\Delta\mathbf{Q}_{j+1/2}) + \mathbf{A}^-(\Delta\mathbf{Q}_{j+1/2}), \tag{44}$$

this leads to $\mathbf{F}^-_{j+1/2} = \mathbf{F}^+_{j+1/2}\ \forall j \in \mathbf{Z}$.

It is difficult to solve nonlinear Riemann problems to take exact solution. It is efficient to use some approximate Riemann solvers such as HLL or Roe's solvers. Details can be found in [3] and [4].

6.3.1. Roe's solver

This approximate Riemann solver is based on the approximation of the nonlinear system $\mathbf{q}_t + [\mathbf{f}(\mathbf{q})]_x \equiv \mathbf{q}_t + \mathbf{A}(\mathbf{q})\mathbf{q}_x = 0$, where $\mathbf{A}(\mathbf{q})$ is the Jacobian matrix, by the linear system $\mathbf{q}_t + \mathbf{A}_{j+1/2}\mathbf{q}_x = 0$, where $\mathbf{A}_{j+1/2}$ is the Roe-averaged Jacobian matrix, which is defined by suitable combination of $\mathbf{A}(\mathbf{Q}_j)$ and $\mathbf{A}(\mathbf{Q}_{j+1})$.

We define intercell numerical fluxes

$$\mathbf{F}_{j+1/2} = \frac{1}{2}[\mathbf{f}(\bar{\mathbf{Q}}_j) + \mathbf{f}(\bar{\mathbf{Q}}_{j+1})] - \frac{1}{2}|\mathbf{A}_{j+1/2}|(\bar{\mathbf{Q}}_{j+1} - \bar{\mathbf{Q}}_j), \tag{45}$$

and intercell fluctuations in the scheme (42) by

$$\begin{aligned} \mathbf{A}^-(\Delta\mathbf{Q}_{j+1/2}) &= \sum_{p=1}^{m} \lambda^{-,p}_{j+1/2}\mathbf{r}^p_{j+1/2}\Delta\gamma^p_{j+1/2}, \\ \mathbf{A}^+(\Delta\mathbf{Q}_{j+1/2}) &= \sum_{p=1}^{m} \lambda^{+,p}_{j+1/2}\mathbf{r}^p_{j+1/2}\Delta\gamma^p_{j+1/2}, \end{aligned} \tag{46}$$

where $\mathbf{r}^p_{j+1/2}$ are eigenvectors of the Roe matrix $\mathbf{A}_{j+1/2}$, $\lambda^p_{j+1/2}$ are eigenvalues called Roe's speeds and $\Delta\boldsymbol{\gamma}_{j+1/2} = \mathbf{R}^{-1}_{j+1/2}\Delta\mathbf{Q}_{j+1/2}$.

6.3.2. HLL solver

This solver does not use the explicit linearization of the Jacobian matrix, but the solution is constructed by the consideration of two discontinuities, propagating at speeds s^1 and s^2. The middle state $\bar{\mathbf{Q}}_{j+1/2}$ is determined by conservation law

$$\mathbf{f}(\bar{\mathbf{Q}}_{j+1}) - \mathbf{f}(\bar{\mathbf{Q}}_j) = s^2_{j+1/2}(\bar{\mathbf{Q}}_{j+1} - \bar{\mathbf{Q}}_{j+1/2}) + s^1_{j+1/2}(\bar{\mathbf{Q}}_{j+1/2} - \bar{\mathbf{Q}}_j), \tag{47}$$

$$\bar{\mathbf{Q}}_{j+1/2} = \frac{\mathbf{f}(\bar{\mathbf{Q}}_{j+1}) - \mathbf{f}(\bar{\mathbf{Q}}_j) - s^2_{j+1/2}\bar{\mathbf{Q}}_{j+1} + s^1_{j+1/2}\bar{\mathbf{Q}}_j}{s^1_{j+1/2} - s^2_{j+1/2}}. \tag{48}$$

When the special choice of the characteristic speeds called Einfeld speeds is used, the solver is called HLLE. The Einfeld speeds are defined by

$$s^1_{j+1/2} = \min_p\{\min\{\lambda^p_j, \lambda^p_{j+1/2}\}\}, \qquad s^2_{j+1/2} = \max_p\{\max\{\lambda^p_{j+1}, \lambda^p_{j+1/2}\}\}, \tag{49}$$

where λ^p_j are eigenvalues of the matrix $\mathbf{A}_j = \mathbf{f}'(\bar{\mathbf{Q}}_j)$.

6.4. Augmented systems

Consider the model for river flow through the varying rectangular channel (4) as was presented in Section 1 and its augmented formulation (13) and (14) presented in Section 2. The eigencomponents for the matrix $\check{\mathbf{D}}$ are

$$\lambda_1 = 0,\ \lambda_2 = 0,\ \lambda_3 = 2u,\ \lambda_4 = u + \sqrt{\frac{ga}{l}},\ \lambda_5 = u - \sqrt{\frac{ga}{l}},$$

and

$$\begin{aligned}
\mathbf{r}_1 &= [-\tfrac{ga}{\lambda_4\lambda_5}, 0, 0, -1, ga]^T, & \mathbf{r}_2 &= [-\tfrac{ga^2}{l^2\lambda_4\lambda_5}, 0, 1, 0, \tfrac{ga^2}{2l^2}]^T,\\
\mathbf{r}_3 &= [0, 0, 0, 0, 1]^T, & \mathbf{r}_4 &= [1, \lambda_4, 0, 0, \lambda_4^2]^T,\\
\mathbf{r}_5 &= [1, \lambda_5, 0, 0, \lambda_5^2]^T.
\end{aligned}$$

We realize the decomposition for the augmented quasilinear formulation i.e. for the system of five equations with Einfeld speeds

$$s_1 = 0, \quad s_2 = 0, \quad s_3 = s_4 + s_5,$$

$$s_4 = \min_p\{\min\{\lambda_L^p, \lambda_{LR}^p\}\}, \qquad s_5 = \max_p\{\max\{\lambda_R^p, \lambda_{LR}^p\}\},$$

and approximation of the eigenvectors of the matrix $\check{\mathbf{D}}$

$$\begin{aligned}
\mathbf{r}_1 &\approx [\tfrac{g\bar{A}}{\widetilde{s_4s_5}}, 0, 0, -1, \tfrac{g\bar{A}\widehat{s_4s_5}}{\widetilde{s_4s_5}}]^T,\\
\mathbf{r}_2 &\approx [\tfrac{g\bar{A}^2}{L_LL_R\widetilde{s_4s_5}}, 0, 1, 0, \tfrac{g\bar{A}^2\widehat{s_4s_5}}{L_LL_R\widetilde{s_4s_5}} - \tfrac{g\tilde{A}^2}{2L_LL_R}]^T,\\
\mathbf{r}_3 &\approx [0, 0, 0, 0, 1]^T,\\
\mathbf{r}_4 &\approx [1, s_4, 0, 0, s_4^2]^T,\\
\mathbf{r}_5 &\approx [1, s_5, 0, 0, s_5^2]^T,
\end{aligned}$$

where $\widetilde{s_4s_5} = -\bar{U}^2 + \frac{g\bar{A}\bar{L}}{L_LL_R}$, $\widehat{s_4s_5} = -|U_LU_R| + \frac{g\bar{A}\bar{L}}{L_LL_R}$, λ_L^p and λ_R^p are eigenvalues of the Jacobian matrix for the left end right values and λ_{LR}^p are eigenvalues of the Roe's matrix.

The decomposition of the augmented system has the following form

$$\begin{bmatrix} \Delta A \\ \Delta Q \\ \Delta L \\ \Delta B \\ \Delta \Phi \end{bmatrix} = \sum_{p=1}^{5} \gamma_p \mathbf{r}^p.$$

We have five linearly independent eigenvectors. The approximation is chosen to be able to prove the consistency and provide the stability of the algorithm. In some special cases this scheme is conservative and we can prove the positive semidefiniteness, but only under the additional assumptions.

The basic version of the numerical scheme is in the form

$$\frac{d\bar{\mathbf{Q}}_j}{dt} = -\frac{1}{\Delta x}[\mathbf{A}^-(\Delta\mathbf{Q}_{j+1/2}) + \mathbf{A}^+(\Delta\mathbf{Q}_{j-1/2})], \tag{50}$$

where fluctuations are defined by

$$\begin{aligned}
\mathbf{A}^-(\Delta\mathbf{Q}_{j+1/2}) &= \sum_{\substack{p=1\\ s^{p,n}_{j+1/2}<0}}^{m} \gamma^p_{j+1/2}\mathbf{r}^p_{j+1/2},\\
\mathbf{A}^+(\Delta\mathbf{Q}_{j+1/2}) &= \sum_{\substack{p=1\\ s^{p,n}_{j+1/2}>0}}^{m} \gamma^p_{j+1/2}\mathbf{r}^p_{j+1/2}.
\end{aligned}$$

7. Central-upwind method

Now we introduce so called central-upwind scheme. These schemes combine advantages of the upwind schemes i.e. lower numerical diffusion and usability for the steady states with advantages of the central schemes i.e. positive semidefiniteness. These schemes are Riemann solver free. This scheme can be found in [2]

One simple method in the conservative form (19) has the numerical flux in the form

$$\mathbf{F}_{j+1/2} = \frac{a^+_{j+1/2}\mathbf{f}(\mathbf{Q}_j) - a^-_{j+1/2}\mathbf{f}(\mathbf{Q}_{j+1})}{a^+_{j+1/2} - a^-_{j+1/2}} + \frac{a^+_{j+1/2}a^-_{j+1/2}}{a^+_{j+1/2} - a^-_{j+1/2}}\left[\mathbf{Q}_{j+1} - \mathbf{Q}_j\right], \tag{51}$$

where

$$a^+_{j+1/2} = \max\left\{\lambda_N\left(\mathbf{f}'(\mathbf{Q}_j)\right), \lambda_N\left(\mathbf{f}'(\mathbf{Q}_{j+1})\right), 0\right\},$$

$$a^-_{j+1/2} = \min\left\{\lambda_1\left(\mathbf{f}'(\mathbf{Q}_j)\right), \lambda_1\left(\mathbf{f}'(\mathbf{Q}_{j+1})\right), 0\right\},$$

represent maximal speeds of the propagation of the waves in the points $x_{j+1/2}$ and we suppose $\lambda_1 < \lambda_2, \ \ldots, \ \lambda_N$.

8. Decomposition of the flux function

Described schemes can be represented and understood by the same way. The amount of information about the structure of the solution of the Riemann problem included into schemes causes the differences between schemes. This information is employed in decomposition of the difference of the flux function.

Central schemes for example Lax-Friedrichs scheme are based on the following decomposition

$$\mathbf{f}(\mathbf{Q}^+_{j+1/2})-\mathbf{f}(\mathbf{Q}^-_{j+1/2}) = s^1(\mathbf{Q}^+_{j+1/2}-\mathbf{Q}^*_{j+1/2})+s^2(\mathbf{Q}^*_{j+1/2}-\mathbf{Q}^-_{j+1/2}) = \sum_{p=1}^{2}\mathbf{Z}^p_{j+1/2}, \quad (52)$$

where $s^1 = \frac{\Delta x}{\Delta t}$ and $s^2 = -\frac{\Delta x}{\Delta t}$. Next we define fluctuations

$$\begin{aligned} \mathbf{A}^-(\Delta\mathbf{Q}_{j+1/2}) &= \sum_{\substack{p=1\\ s^p<0}}^{2} \mathbf{Z}^p_{j+1/2}, \\ \mathbf{A}^+(\Delta\mathbf{Q}_{j+1/2}) &= \sum_{\substack{p=1\\ s^p>0}}^{2} \mathbf{Z}^p_{j+1/2}. \end{aligned} \quad (53)$$

We can use the relation (42) and we can derive the scheme in the conservative form. These schemes are not suitable for the semidiscrete formulation because of the infinite speed ($\Delta t \to 0$) of the propagating discontinuities which is typical for the parabolic type of the equations.

The semidiscrete central methods suitable for the semidiscrete formulation use estimate of upper bound of maximal speed of the propagating discontinuities. They are based on the following decomposition

$$\mathbf{f}(\mathbf{Q}^+_{j+1/2})-\mathbf{f}(\mathbf{Q}^-_{j+1/2}) = s_{j+1/2}(\mathbf{Q}^+_{j+1/2}-\mathbf{Q}^*_{j+1/2})-s_{j+1/2}(\mathbf{Q}^*_{j+1/2}-\mathbf{Q}^-_{j+1/2}) =\sum_{p=1}^{2}\mathbf{Z}^p_{j+1/2}, \quad (54)$$

where

$$s_{j+1/2} = \max_p\{\max\{|\lambda^p(\mathbf{Q}^-_{j+1/2})|, |\lambda^p(\mathbf{Q}^+_{j+1/2})|\}\}.$$

We define

$$\begin{aligned} \mathbf{A}^-(\Delta\mathbf{Q}_{j+1/2}) &= \sum_{\substack{p=1\\ s^p_{j+1/2}<0}}^{2} \mathbf{Z}^p_{j+1/2}, \\ \mathbf{A}^+(\Delta\mathbf{Q}_{j+1/2}) &= \sum_{\substack{p=1\\ s^p_{j+1/2}>0}}^{2} \mathbf{Z}^p_{j+1/2}. \end{aligned} \quad (55)$$

The central-upwind methods can be identified with HLL solver. The decomposition has the form

$$\mathbf{f}(\bar{\mathbf{Q}}_{j+1}) - \mathbf{f}(\bar{\mathbf{Q}}_j) = s^2_{j+1/2}(\bar{\mathbf{Q}}_{j+1} - \bar{\mathbf{Q}}_{j+1/2}) + s^1_{j+1/2}(\bar{\mathbf{Q}}_{j+1/2} - \bar{\mathbf{Q}}_j) = \sum_{p=1}^{2} \mathbf{Z}^p_{j+1/2}, \quad (56)$$

where $s^1_{j+1/2} = a^+_{j+1/2}$ and $s^1_{j+1/2} = a^-_{j+1/2}$. And we define

$$\begin{aligned} \mathbf{A}^-(\Delta\mathbf{Q}_{j+1/2}) &= \sum_{\substack{p=1 \\ s^p_{j+1/2}<0}}^{2} \mathbf{Z}^p_{j+1/2}, \\ \mathbf{A}^+(\Delta\mathbf{Q}_{j+1/2}) &= \sum_{\substack{p=1 \\ s^p_{j+1/2}>0}}^{2} \mathbf{Z}^p_{j+1/2}. \end{aligned} \quad (57)$$

All described schemes can be understood in the same way.

9. Conclusion

We presented various numerical schemes for solving fluid flow problems with various properties. We show that all described schemes can be understood in the same way.

References

[1] D.L. George: *Finite volume methods and adaptive refinement for tsunami propagation and innundation*, University of Washington, Ph.D. Thesis, 2006.

[2] A. Kurganov, G. Petrova: *A second-order well-balanced positivity preserving central-upwind scheme for the Saint-Venant system*, Communications in Mathematical Sciences **5** (2007), 133–160.

[3] R.J. LeVeque, M. Pelanti: *A class of approximate Riemann solvers and their relation to relaxation schemes*, Journal of Computational Physics **172** (2001), 572–591.

[4] P.L. Roe: *Approximate Riemann solvers, parameter vectors, and difference schemes*, Journal of Computational Physics **135** (1997), 250–258.

[5] A. Kurganov, G. Petrova: *Central schemes and contact discontinuities*, Mathematical Modelling and Numerical Analysis **34** (2000), 1259–1275.

[6] R.J. LeVeque: *Finite volume methods for hyperbolic problems*, Cambridge University Press, 2002

[7] A. Kurganov, E. Tadmor: New high-resolution central schemes for nonlinear conservation laws and convection-diffusion equations, Journal of Computational Physics **160** (2000), 241–282.

FINITE ELEMENT MODELLING OF SOME INCOMPRESSIBLE FLUID FLOW PROBLEMS[*]

Pavel Burda, Jaroslav Novotný, Jakub Šístek, Alexandr Damašek

Abstract

We deal with modelling of flows in channels or tubes with abrupt changes of the diameter. The goal of this work is to construct the FEM solution in the vicinity of these corners as precise as desired. We present two ways. The first approach makes use of a posteriori error estimates and the adaptive strategy. The second approach is based on the asymptotic behaviour of the exact solution in the vicinity of the corner and on the a priori error estimate of the FEM solution. Then we obtain the solution with desired precision also in the vicinity of the corner, though there is a singularity. Numerical results are demonstrated on a 2D example.

1. Introduction

One of the challenging problems in fluid dynamics is reliable modelling of flows in channels or tubes with abrupt changes of the diameter, which appear often in engineering practice.

We present two ways for getting desired precision of the FEM solution in the vicinity of corners. Both make use of qualitative properties of the mathematical model of flow that is on the Navier-Stokes equations (NSE) for incompressible fluids.

The first approach makes use of a posteriori error estimates of the FEM solution which is carefully derived to trace the quality of the solution. Especially the constant in the a posteriori error estimate is investigated with care. Then we use the adaptive strategy to improve the mesh and thus to improve the FEM solution.

The second approach stands on two legs. One is the asymptotic behaviour of the exact solution of the NSE in the vicinity of the corner. This is obtained using some symmetry of the principal part of the Stokes equation and application of the Fourier transform. Second leg is the a priori error estimate of the FEM solution, where we estimate the seminorm of the exact solution by means of the above obtained asymptotics. On the mesh we then obtain the solution with desired precision also in the vicinity of the corner, though there is a singularity.

2. Navier-Stokes equations for incompressible viscous fluids

Let Ω be an open bounded domain in $\mathbb{R}^2$ filled with a fluid and let Γ be its Lipschitz continuous boundary. The generic point of $\mathbb{R}^2$ is denoted by $\mathbf{x} = (x_1, x_2)^T$ considered in meters and t denotes time variable considered in seconds.

[*]This work was supported by grant No. 106/08/0403 of the Czech Science Foundation and by the State Research Project No. MSM 684 0770010.

2.1. Unsteady two-dimensional flow

We deal with isothermal flow of Newtonian viscous fluids with constant density. Such flow is modelled by the Navier-Stokes system (nonconservative form):

$$\begin{aligned} \rho\left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u}\right) - \mu\Delta\mathbf{u} + \nabla p_r &= \rho\,\mathbf{f} \quad \text{in } \Omega\times[0,T], \qquad (1)\\ \nabla\cdot\mathbf{u} &= 0 \qquad \text{in } \Omega\times[0,T], \qquad (2)\end{aligned}$$

where

- $\mathbf{u} = (u_1, u_2)^T$ denotes the vector of flow velocity [m/s], it is a function of $\mathbf{x}$ and t,
- p_r denotes the pressure [Pa], which is a function of $\mathbf{x}$ and t,
- ρ denotes the density of the fluid [kg/m^3],
- μ denotes the dynamic viscosity of the fluid [Pa·s], supposed to be constant,
- $\mathbf{f} = \mathbf{f}(\mathbf{x}, t)$ is the density of volume forces per mass unit [N/m^3].

Dividing both sides of the momentum equation (1) by ρ and leaving the continuity equation (2) unchanged we obtain

$$\begin{aligned} \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u} - \nu\Delta\mathbf{u} + \nabla p &= \mathbf{f} \quad \text{in } \Omega\times[0,T], \qquad (3)\\ \nabla\cdot\mathbf{u} &= 0 \quad \text{in } \Omega\times[0,T], \qquad (4)\end{aligned}$$

where

- $p = p_r/\rho$ denotes the pressure divided by the density [Pa· m^3 /kg],
- $\nu = \mu/\rho$ denotes the kinematic viscosity of the fluid [m^2/s].

The system is supplied with the initial condition

$$\mathbf{u} = \mathbf{u}_0 \ \text{ in } \Omega,\ t = 0, \qquad (5)$$

where $\nabla\cdot\mathbf{u}_0 = 0$ and with the boundary conditions

$$\begin{aligned} \mathbf{u} &= \mathbf{g} \ \text{ on } \Gamma_g\times[0,T], \qquad (6)\\ -\nu(\nabla\mathbf{u})\mathbf{n} + p\mathbf{n} &= \mathbf{0} \ \text{ on } \Gamma_h\times[0,T], \qquad (7)\end{aligned}$$

where

- Γ_g and Γ_h are two subsets of Γ satisfying $\overline{\Gamma} = \overline{\Gamma}_g \cup \overline{\Gamma}_h,\ \mu_{\mathbb{R}^1}(\Gamma_g\cap\Gamma_h) = 0$,
- $\mathbf{n}$ denotes the unit outer normal vector to the boundary Γ.

Introduced $\mathbf{g}$ is a given function of $\mathbf{x}$ and t satisfying in the case of $\Gamma = \Gamma_g$ for all $t \in [0, T]$

$$\int_\Gamma \mathbf{g}\cdot\mathbf{n}\,\mathrm{d}\Gamma = 0.$$

2.2. Steady 2D Navier-Stokes problem

In the case of steady two-dimensional flow, the Navier-Stokes equations are reduced to

$$\begin{aligned} (\mathbf{u}\cdot\nabla)\mathbf{u} - \nu\Delta\mathbf{u} + \nabla p &= \mathbf{f} \quad \text{in } \Omega, && (8)\\ \nabla\cdot\mathbf{u} &= 0 \quad \text{in } \Omega && (9) \end{aligned}$$

and boundary conditions to

$$\begin{aligned} \mathbf{u} &= \mathbf{g} \;\; \text{on } \Gamma_g, && (10)\\ -\nu(\nabla\mathbf{u})\mathbf{n} + p\mathbf{n} &= \mathbf{0} \;\; \text{on } \Gamma_h. && (11) \end{aligned}$$

2.3. Steady 2D Stokes problem

In the case of the Stokes flow the first (nonlinear) term in (8) is omitted:

$$\begin{aligned} -\nu\Delta\mathbf{u} + \nabla p &= \mathbf{f} \quad \text{in } \Omega\ , && (12)\\ \nabla\cdot\mathbf{u} &= 0 \quad \text{in } \Omega && (13) \end{aligned}$$

and boundary conditions are the same as in (10), (11).

2.4. Variational formulation of Navier-Stokes equations

Let $L_2(\Omega)$ be the space of square integrable functions on Ω and let $L_2(\Omega)/\mathbb{R}$ be the space of functions in $L_2(\Omega)$ ignoring an additive constant. Let $H^1(\Omega)$ and $H_0^1(\Omega)$ be the Sobolev spaces defined as

$$\begin{aligned} H^1(\Omega) &\equiv \left\{ v \mid v \in L^2(\Omega), \frac{\partial v}{\partial x_i} \in L^2(\Omega), i = 1,2 \right\}\ ,\\ H_0^1(\Omega) &\equiv \left\{ v \mid v \in H^1(\Omega), \mathbf{Tr}\ v = 0 \right\}\ , \end{aligned}$$

where $\mathbf{Tr}$ is the trace operator $\mathbf{Tr}:\ H^1(\Omega) \longrightarrow L_2(\Gamma)$ and derivatives are considered in the weak sense.

The inner product and norm in the space $L_2(\Omega)$ are defined as

$$(u,v)_{L_2(\Omega)} \equiv \int_\Omega uv\ \mathrm{d}\Omega\ , \qquad \|v\|^2_{L_2(\Omega)} \equiv \int_\Omega v^2 \mathrm{d}\Omega$$

and the norm of function v in the Sobolev space $H^1(\Omega)$ is considered as

$$\|v\|^2_{H^1(\Omega)} \equiv \int_\Omega \left(v^2 + \sum_{k=1}^{2} \left(\frac{\partial v}{\partial x_k} \right)^2 \right) \mathrm{d}\Omega\ .$$

Sometimes, the notation $\|\cdot\|_{L_2(\Omega)}$ is shortened to $\|\cdot\|_0$ and $\|\cdot\|_{H^1(\Omega)}$ to $\|\cdot\|_1$. Similarly, the notation $(u,v)_{L_2(\Omega)}$ is shortened to $(u,v)_0$.

Let us define vector function spaces V_g and V by

$$\begin{aligned} V_g &\equiv \left\{ \mathbf{v} = (v_1, v_2)^T \mid \mathbf{v} \in [H^1(\Omega)]^2; \mathbf{Tr}\ v_i = g_i, i = 1, 2 \right\} , \\ V &\equiv \left\{ \mathbf{v} = (v_1, v_2)^T \mid \mathbf{v} \in [H^1_0(\Omega)]^2 \right\} . \end{aligned}$$

Let us note, that the norm of vector function $\mathbf{v}$ in the spaces V_g and V is then

$$\|\mathbf{v}\|^2_{[H^1(\Omega)]^2} \equiv \sum_{i=1}^{2} \int_\Omega \left(v_i^2 + \sum_{k=1}^{2} \left(\frac{\partial v_i}{\partial x_k} \right)^2 \right) \mathrm{d}\Omega$$

and the norm of vector function $\mathbf{v}$ in the space $[L_2(\Omega)]^2$ is

$$\|\mathbf{v}\|^2_{[L_2(\Omega)]^2} \equiv \sum_{i=1}^{2} \int_\Omega v_i^2 \mathrm{d}\Omega \ .$$

The *weak unsteady Navier-Stokes problem* means seeking of $\mathbf{u}(t) \in V_g$, $\mathbf{u}(t) = (u_1(t), u_2(t))^T$, and $p(t) \in L_2(\Omega)/\mathbb{R}$ satisfying for any $t \in [0, T]$ and $\forall\ \mathbf{v} \in V$ and $\forall\ \psi \in L^2(\Omega)$:

$$\int_\Omega \frac{\partial \mathbf{u}}{\partial t} \cdot \mathbf{v} \mathrm{d}\Omega + \int_\Omega (\mathbf{u} \cdot \nabla) \mathbf{u} \cdot \mathbf{v} \mathrm{d}\Omega + \nu \int_\Omega \nabla \mathbf{u} : \nabla \mathbf{v} \mathrm{d}\Omega - \int_\Omega p \nabla \cdot \mathbf{v} \mathrm{d}\Omega = \int_\Omega \mathbf{f} \cdot \mathbf{v} \mathrm{d}\Omega \ , \tag{14}$$

$$\int_\Omega \psi \nabla \cdot \mathbf{u} \mathrm{d}\Omega = 0 \ , \tag{15}$$

$$\mathbf{u} - \mathbf{u}_g \in V. \tag{16}$$

The operation $\nabla \mathbf{u} : \nabla \mathbf{v}$ is defined as

$$\nabla \mathbf{u} : \nabla \mathbf{v} \equiv \frac{\partial u_x}{\partial x} \frac{\partial v_x}{\partial x} + \frac{\partial u_x}{\partial y} \frac{\partial v_x}{\partial y} + \frac{\partial u_y}{\partial x} \frac{\partial v_y}{\partial x} + \frac{\partial u_y}{\partial y} \frac{\partial v_y}{\partial y} \ .$$

Similarly, the *weak steady Navier-Stokes problem* reads:
Seek $\mathbf{u} = (u_1, u_2)^T \in V_g$ and $p \in L_2(\Omega)/\mathbb{R}$ satisfying $\forall\ \mathbf{v} \in V$ and $\forall\ \psi \in L^2(\Omega)$:

$$\int_\Omega (\mathbf{u} \cdot \nabla) \mathbf{u} \cdot \mathbf{v} \mathrm{d}\Omega + \nu \int_\Omega \nabla \mathbf{u} : \nabla \mathbf{v} \mathrm{d}\Omega - \int_\Omega p \nabla \cdot \mathbf{v} \mathrm{d}\Omega = \int_\Omega \mathbf{f} \cdot \mathbf{v} \mathrm{d}\Omega \ , \tag{17}$$

$$\int_\Omega \psi \nabla \cdot \mathbf{u} \mathrm{d}\Omega = 0 \ , \tag{18}$$

$$\mathbf{u} - \mathbf{u}_g \in V. \tag{19}$$

In case of the *weak steady Stokes problem* instead of (17) we require

$$\nu \int_\Omega \nabla \mathbf{u} : \nabla \mathbf{v} \mathrm{d}\Omega - \int_\Omega p \nabla \cdot \mathbf{v} \mathrm{d}\Omega = \int_\Omega \mathbf{f} \cdot \mathbf{v} \mathrm{d}\Omega \ . \tag{20}$$

3. Finite element method for Navier-Stokes equations

Let us divide the domain Ω (supposed to be polygonal from now) into N elements T_K, $K = 1, 2, \ldots, N$, of a triangulation $\mathcal{T}$ such that

$$\begin{aligned} \bigcup_{K=1}^{N} \overline{T}_K &= \overline{\Omega} \, , \\ \mu_{\mathbb{R}^2} \left(T_K \cap T_L \right) &= 0, \quad K \neq L \, . \end{aligned}$$

Let h_K mean the diameter of the element T_K.

3.1. Function spaces for velocity and pressure approximation

To solve the Navier-Stokes equations, different polynomial approximation for velocities and for pressure are usually chosen. Equal order approximation is easy to implement, but pressure exhibits instability. Approximation with different order is more suitable for practical computing, cf. [3]. I. Babuška and F. Brezzi introduced a condition (also called inf-sup condition) limitting the choice of combinations of approximation

$$\exists C_B > 0, const. \; \forall q_h \in Q_h \; \exists \mathbf{v}_h \in V_{gh} \; (q_h, \nabla \cdot \mathbf{v}_h)_0 \geq C_B \|q_h\|_0 \|\mathbf{v}_h\|_1 \; , \tag{21}$$

where Q_h and V_{gh} are the function spaces for approximation of pressure and velocity. Condition (21) is important for stability. It is satisfied, e.g., for Taylor-Hood elements we use.

3.2. Taylor-Hood finite elements

In this paper we apply Taylor-Hood finite elements on triangles and quadrilaterals. Values of velocity are approximated at corner nodes and midsides and values of pressure at corner nodes (Figure 1). It corresponds to the following function spaces on element T_K:

- *triangle*
  $v_i \in P_2(T_K)$, $i = 1, 2$, i.e. polynomials of the second order,
  $p \in P_1(T_K)$, i.e. linear polynomials
- *quadrilateral*
  $v_i \in Q_2(T_K)$, $i = 1, 2$, i.e. polynomials of the second order for each coordinate,
  $p \in Q_1(T_K)$, i.e. bilinear polynomials.

Let us employ the notation

$$R_m(\overline{T_K}) = \begin{cases} P_m(\overline{T_K}), & \text{if } T_K \text{ is a triangle} \\ Q_m(\overline{T_K}), & \text{if } T_K \text{ is a quadrilateral} \end{cases} \tag{22}$$

and let $\mathcal{C}(\overline{\Omega})$ denote the space of continuous functions on $\overline{\Omega}$.

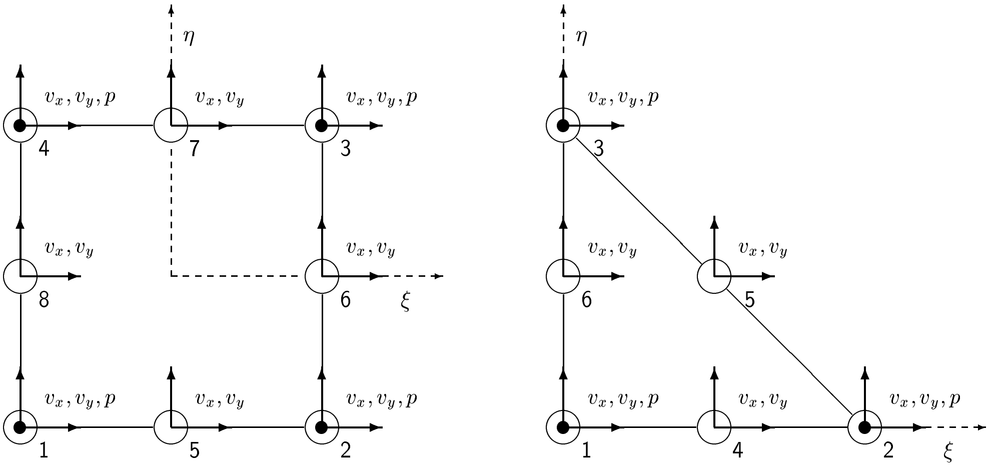


Fig. 1: *Taylor-Hood reference elements.*

Application of Taylor-Hood finite elements leads to the final approximation on the domain Ω satisfying $\mathbf{u}_h \in V_{gh}$ and $p_h \in Q_h$ where

$$V_{gh} = \Big\{ \mathbf{v}_h = (v_{h_1}, v_{h_2})^T \in [\mathcal{C}(\overline{\Omega})]^2;\ v_{h_i}\,|_{T_K} \in R_2(\overline{T_K}),\ K = 1, \dots, N,\ i = 1, 2, \quad (23)$$
$$\mathbf{v}_h = \mathbf{g} \text{ at nodes on } \Gamma \Big\} \ ,$$

$$Q_h = \Big\{ \psi_h \in \mathcal{C}(\overline{\Omega});\ \psi_h\,|_{T_K} \in R_1(\overline{T_K}),\ K = 1, \dots, N \Big\} \ . \quad (24)$$

We also need the space

$$V_h = \Big\{ \mathbf{v}_h = (v_{h_1}, v_{h_2})^T \in [\mathcal{C}(\overline{\Omega})]^2;\ v_{h_i}\,|_{T_K} \in R_2(\overline{T_K}),\ K = 1, \dots, N,\ i = 1, 2, \quad (25)$$
$$\mathbf{v}_h = \mathbf{0} \text{ at nodes on } \Gamma \Big\} \ .$$

Since these spaces satisfy $V_{gh} \subset V_g$, $V_h \subset V$, and $Q_h \subset L_2(\Omega)/\mathbb{R}$ for prescribed arbitrary value of pressure (e.g. $p_h = 0$) in one node, we can introduce *approximate steady Navier-Stokes problem*:
Seek $\mathbf{u}_h \in V_{gh}$ and $p_h \in Q_h$ satisfying

$$\int_\Omega (\mathbf{u}_h \cdot \nabla)\mathbf{u}_h \cdot \mathbf{v}_h \mathrm{d}\Omega + \nu \int_\Omega \nabla \mathbf{u}_h : \nabla \mathbf{v}_h \mathrm{d}\Omega - \int_\Omega p_h \nabla \cdot \mathbf{v}_h \mathrm{d}\Omega = \int_\Omega \mathbf{f} \cdot \mathbf{v}_h \mathrm{d}\Omega,\ \forall \mathbf{v}_h \in V_h, \quad (26)$$

$$\int_\Omega \psi_h \nabla \cdot \mathbf{u}_h \mathrm{d}\Omega = 0,\ \forall \psi_h \in Q_h \ , \quad (27)$$

$$\mathbf{u}_h - \mathbf{u}_{gh} \in V_h \ , \quad (28)$$

where $\mathbf{u}_{gh} \in V_{gh}$ is the projection of $\mathbf{u}_g$ onto the space V_{gh}.

Similarly we define *approximate steady Stokes problem*, just omitting the first term in (26).

Using the shape-regular triangulation and refining the mesh such that $h_{\max} \to 0$, where

$$h_{\max} = \max_K h_K,$$

the solution of the approximated problem converges to the solution of the continuous problem (for more detail see e.g. [3]).

4. Asymptotic behaviour of the solution near corners

We are concerned with flow in tubes and channels with abrupt changes of diameter. The results are based on the paper [4], where we investigated *the pipe flow* (axisymmetric). The asymptotic behaviour of *plane flow* with corner singularities was studied e.g. by Kondratiev [17]. The asymptotics of the biharmonic equation for the stream function ψ are basic. Let us e.g. take the case of the nonconvex domain with the internal angle $\omega = \frac{3}{2}\pi$, cf. Figure 2. Then, in polar coordinates we get the expansion

$$\psi(\rho, \vartheta) = \rho^{1.54448374}\ \phi(\vartheta)\ +\ \ldots\ (\textit{lower order terms}), \tag{29}$$

where ρ is the distance from the corner, see [4]. This result is the same as that obtained in desk geometry by Kondratiev [17], where

$$\psi^{desk}(\rho, \vartheta) = \rho^{1.5445}\ \phi^d(\vartheta)\ +\ \ldots\ (\textit{l.o.t.}). \tag{30}$$

So we get the expansion for the velocities:

$$u_l(\rho, \vartheta) = \rho^{0.54448374}\varphi_l(\vartheta) + \ldots (\textit{l.o.t.})\ ,\ l = 1, 2, \tag{31}$$

where the functions φ_l do not depend on ρ, the distance from the corner. This expansion exhibits the infinite gradient of velocity near the corner.

In Section 6 our aim is to make use of the information on the local behaviour of the solution near the corner point, in order to design local meshing subordinate to the asymptotics.

5. A posteriori error estimates for the Navier-Stokes equation

At present various a posteriori error estimates for the Stokes and Navier-Stokes problems are available. We mention e.g. Babuška, Rheinboldt [2], Ainsworth, Oden [1], Verfürth [19]. Other references can be found in [5].

5.1. A posteriori estimates for 2D steady Navier-Stokes equations

Let us consider the steady Navier-Stokes problem (8), (9), with boundary conditions (10), (11).

For the discretization by finite elements we use again Taylor-Hood elements P2/P1.

Suppose that exact solution of the problem is denoted by (u_1, u_2, p) and the approximate finite element solution by (u_1^h, u_2^h, p_h). The exact solution differs from the approximate solution in the error

$$(e_{u_1}, e_{u_2}, e_p) \equiv (u_1 - u_1^h, u_2 - u_2^h, p - p_h). \tag{32}$$

For the solution (u_1, u_2, p) we denote

$$\begin{aligned} &\mathcal{U}^2(u_1, u_2, p, \Omega) \equiv \|(u_1, u_2, p)\|_V^2 \equiv \|(u_1, u_2)\|_{1,\Omega}^2 + \|p\|_{0,\Omega}^2 \qquad (33) \\ &\equiv \int_\Omega \left(u_1^2 + u_2^2 + \left(\frac{\partial u_1}{\partial x}\right)^2 + \left(\frac{\partial u_1}{\partial y}\right)^2 + \left(\frac{\partial u_2}{\partial x}\right)^2 + \left(\frac{\partial u_2}{\partial y}\right)^2 \right) \mathrm{d}\Omega + \int_\Omega p^2 \mathrm{d}\Omega. \end{aligned}$$

The estimate proved in [5], [6], [7] for the Stokes problem can be generalized to the Navier-Stokes equations:

$$\|(e_{u_1}, e_{u_2})\|_{1,\Omega}^2 + \|e_p\|_{0,\Omega}^2 \le \mathcal{E}^2(u_1^h, u_2^h, p^h), \tag{34}$$

where (cf. [19])

$$\mathcal{E}^2(u_1^h, u_2^h, p^h, \Omega) \equiv C \left[\sum_{K \in \mathcal{T}^h} h_K^2 \int_{T_K} (r_1^2 + r_2^2) + \sum_{K \in \mathcal{T}^h} \int_{T_K} r_3^2 \mathrm{d}\Omega \right], \tag{35}$$

where h_K denotes the diameter of the element T_K and $r_i,\ i = 1, 2, 3$, are the residuals

$$\begin{aligned} r_1 &= f_x - \left(u_1^h \frac{\partial u_1^h}{\partial x} + u_2 \frac{\partial u_1^h}{\partial y} \right) + \nu \left(\frac{\partial^2 u_1^h}{\partial x^2} + \frac{\partial^2 u_1^h}{\partial y^2} \right) - \frac{\partial p^h}{\partial x}, && (36) \\ r_2 &= f_y - \left(u_1^h \frac{\partial u_2^h}{\partial x} + u_2^h \frac{\partial u_2^h}{\partial y} \right) + \nu \left(\frac{\partial^2 u_2^h}{\partial x^2} + \frac{\partial^2 u_2^h}{\partial y^2} \right) - \frac{\partial p^h}{\partial y}, && (37) \\ r_3 &= \frac{\partial u_1^h}{\partial x} + \frac{\partial u_2^h}{\partial y}. && (38) \end{aligned}$$

Let us note that due to our practical experience we use only the element residuals.

Denote also

$$\mathcal{E}^2(u_1^h, u_2^h, p^h, T_K) \equiv C \left[h_K^2 \int_{T_K} (r_1^2 + r_2^2) + \int_{T_K} r_3^2 \mathrm{d}\Omega \right]. \tag{39}$$

It is important, that C does not depend on the mesh size and so can be determined experimentally for general situation.

By computing of the estimates (11) we obtain absolute numbers, that will depend on given quantities in different problems. We are mainly interested in the error related to the computed solution, i.e. relative error. This is given by the ratio of the absolute norm of the solution error related to the unit area of the element T_K, $\frac{1}{|T_K|}\ \mathcal{E}^2(u_1^h, u_2^h, p^h, T_K)$, and the solution norm on the whole domain Ω, related to unit area $\frac{1}{|\Omega|}\ \|(u_1^h, u_2^h, p^h)\|_{V,\Omega}^2$, i.e.

$$\mathcal{R}^2(u_1^h, u_2^h, p^h, T_K) = \frac{|\Omega|\ \mathcal{E}^2(u_1^h, u_2^h, p^h, T_K)}{|T_K|\ \|(u_1^h, u_2^h, p^h)\|_{V,\Omega}^2}. \tag{40}$$

5.2. Determination of the constant C

In papers [7], [8] we investigated the problem of the constant C in the a posteriori estimates. Comparing analytical and finite element solution of some model problems we found the appropriate value of the constant. For details we refer to [7] and [8].

5.3. Numerical results and application of estimates to the construction of adaptive meshes

Consider two-dimensional flow of viscous, incompressible fluid described by Navier-Stokes equations in domain with corner singularity, cf. Figure 2.

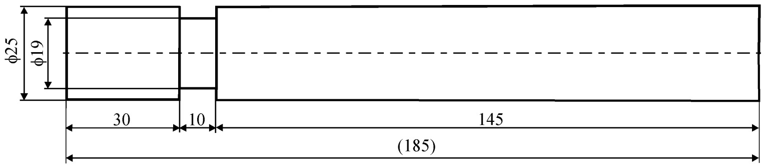


Fig. 2: *Geometry of the channel.*

Due to the symmetry, we solve the problem only on half of the channel, cf. Figure 3. On the inflow we consider parabolic velocity profile, on the outflow 'do nothing' boundary condition. On the upper wall no-slip condition and on the lower wall condition of symmetry (i.e. only y-component of velocity equals zero). We consider the following parameters: $\nu = 0.0001$ m^2/s, $u_{in} = 1$ m/s. The initial mesh is in Figure 3.

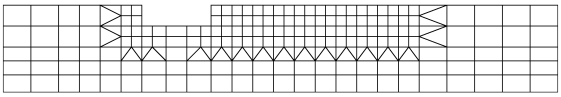


Fig. 3: *Initial finite element mesh.*

Elements, where the relative error by (40) exceeds 3 %, are refined and new solution together with new error estimates is computed. The third refinement is seen in Figure 4.

The relative errors in the vicinity of the left corner are shown in Figure 5.

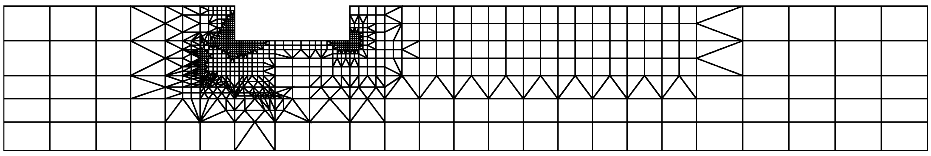


Fig. 4: *Finite element mesh after third refinement.*

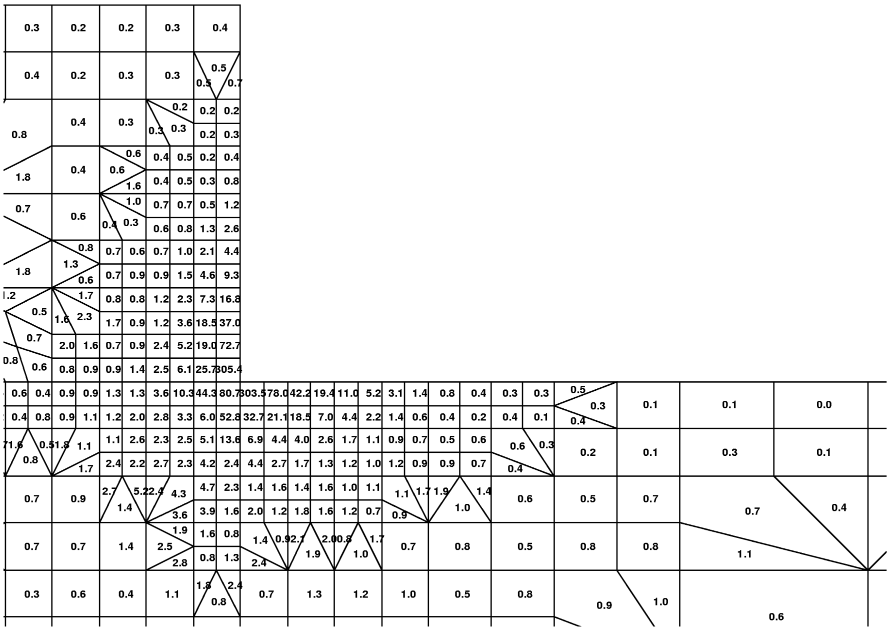


Fig. 5: *Relative errors on elements of the third refinement.*

Numerical results of velocity components and pressure are in Figures 6 and 7. The corner singularities caused by nonconvex corners are approximated with the accuracy indicated in Figure 5.

6. Application of a priori estimates for Navier-Stokes equations

The goal of this section is to summarize authors' experience with the application of a priori error estimates of the finite element method in computational fluid dynamics. This approach is applied to generate the computational mesh in the purpose of uniform distribution of error on elements and is used in precise solution on domains with corner-like singularities. Incompressible viscous flow modelled by the steady Navier-Stokes equations (17)–(19) is considered.

One possible way to improve accuracy of solution by the FEM is to refine the mesh near places, where singularity can appear, by means of adaptive refinement based on a posteriori error estimates or error estimators, as presented in Section 5. This method could be quite time demanding, since it needs several runs of solu-

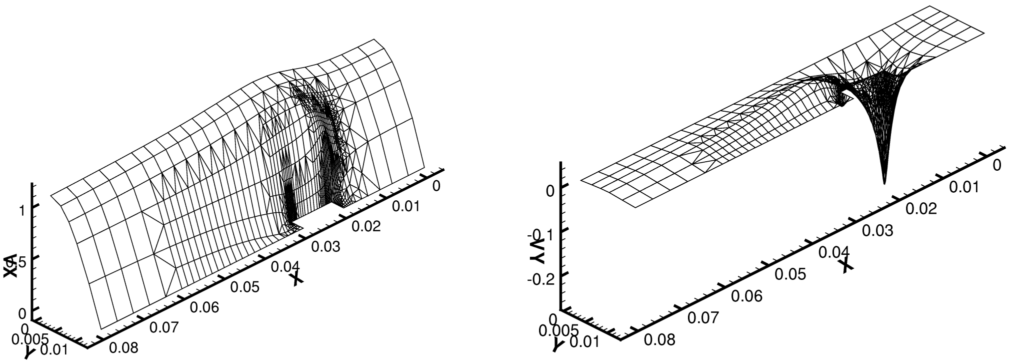


Fig. 6: *After third refinement: velocity u_x (left) and velocity u_y (right).*

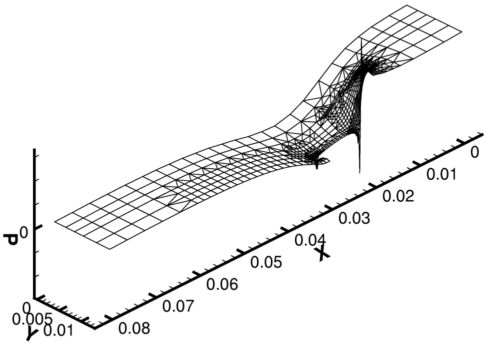


Fig. 7: *Pressure p after third refinement.*

tion. Completely different method is applied in this section. Computational mesh is prepared before the first run of the solution.

Numerical results are presented for flows in a channel with sharp obstacle and in a channel with sharp extension. Let us note that some other results were published in [9].

6.1. Algorithm for generation of computational mesh

In the derivation of the algorithm, two main 'tools' are used. The first is a priori estimate of the finite element error for the Navier-Stokes equations (17)–(19) (cf. [13]),

$$\|\nabla(\mathbf{u}-\mathbf{u_h})\|_{L_2(\Omega)} \leq C\Big[\Big(\sum_K h_K^{2k} \mid \mathbf{u} \mid^2_{H^{k+1}(T_K)}\Big)^{1/2} + \Big(\sum_K h_K^{2k} \mid p \mid^2_{H^k(T_K)}\Big)^{1/2}\Big]\,, \quad (41)$$

$$\|p-p_h\|_{L_2(\Omega)} \leq C\Big[\Big(\sum_K h_K^{2k} \mid \mathbf{u} \mid^2_{H^{k+1}(T_K)}\Big)^{1/2} + \Big(\sum_K h_K^{2k} \mid p \mid^2_{H^k(T_K)}\Big)^{1/2}\Big]\,, \quad (42)$$

where h_K is the diameter of triangle T_K of a triangulation $\mathcal{T}$ and $k = 2$ for Taylor-Hood elements, which are applied in presented numerical experiments.

The second tool is the asymptotic behaviour of the solution near the singularity. In Section 4.2 (see also [4]), it was proved for the Stokes flow in axisymmetric tubes, that for internal angle $\alpha = \frac{3}{2}\pi$, the leading term of expansion of the solution for each velocity component is

$$u_i(\rho, \vartheta) = \rho^{0.5445}\varphi_i(\vartheta) + \dots \ (l.o.t.), \ i = 1, 2 \ , \tag{43}$$

where ρ is the distance from the corner, ϑ is the angle and φ_i is a smooth function. The same expansion is known to apply to the plane flow (cf. [16]) and similar results were also proved for the Navier-Stokes equations. Differentiating by ρ, we observe $\frac{\partial u_i(\rho,\vartheta)}{\partial \rho} \ \rightarrow \ \infty$ for $\rho \ \rightarrow \ 0.$

Taking into account the expansion (43), we can estimate

$$\mid \mathbf{u} \mid^2_{H^{k+1}(T_K)} \approx C \int\limits_{r_K - h_K}^{r_K} \rho^{2(\gamma-k-1)} \ \rho \ d\rho = C \ \left[-r_K^{2(\gamma-k)} + (r_K - h_K)^{2(\gamma-k)} \right] \ , \tag{44}$$

where r_K is the distance of element T_K from the corner, cf. Figure 8.

Putting estimate (44) into the a priori error estimate (41) or (42), we derive that we should guarantee

$$h_K^{2k} \left[-r_K^{2(\gamma-k)} + (r_K - h_K)^{2(\gamma-k)} \right] \approx h_{ref}^{2k} \ , \tag{45}$$

in order to get the error estimate of order $O(h_{ref}^k)$ uniformly distributed on elements. From this expression, we compute element diameters using the Newton method in accordance to chosen h_{ref}. Similar idea was presented by C. Johnson for an elliptic problem in [15].

6.2. Geometry and design of the mesh

The algorithm was applied to the channel with sudden intake of diameter (see Figure 2). Due to symmetry, again the problem was solved only on the upper half of the channel.

The diameters of elements were computed for values $h_{ref} = 0.1732$ mm, $k = 2$, $\gamma = 0.5444837$. We started in the distance $r_1 = 0.25$ mm from the corner. This corresponds to cca 3 % of relative error on elements. Fourteen diameters of elements were obtained. The detail of the mesh refinement is in Figure 8. More in [8].

The refined detail is connected to the rest of the coarse mesh. In Figure 9 final mesh after the refinement is shown.

6.3. Measuring of error

To review the efficiency of the algorithm, we use a posteriori error estimates as derived in Chapter 5, to evaluate the obtained error on elements. Suppose that the exact solution of the problem is denoted as (u_1, u_2, p) and the approximate solution

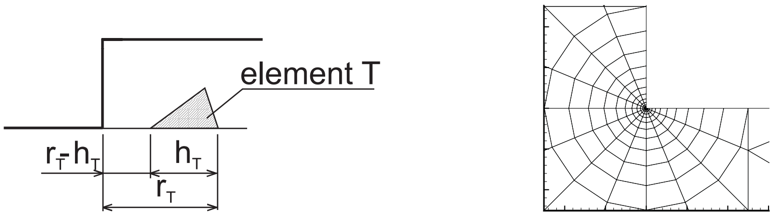


Fig. 8: *Description of element variables (left), details of refined mesh (right).*

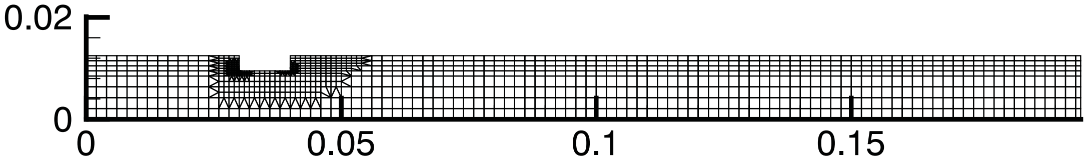


Fig. 9: *Final computational mesh for the channel.*

obtained by the FEM as (u_1^h, u_2^h, p^h). The exact solution differs from the approximate solution in the error $(e_{u_1}, e_{u_2}, e_p) = (u_1 - u_1^h, u_2 - u_2^h, p - p^h)$.

In adaptive mesh refinement in Sections 5.3–5.5 we used the error estimator (40). In this chapter, for the similarity with a priori error estimate, we use the modified absolute error defined as

$$\mathcal{A}_m^2(u_{1h}, u_{2h}, p_h, T_K, \Omega, n) = \frac{|\Omega| \mathcal{E}^2(u_{1h}, u_{2h}, p_h, T_K)}{|\overline{T_K}| \; \mathcal{U}^2(u_{1h}, u_{2h}, p_h, \Omega)} \,, \tag{46}$$

where $|\overline{T_K}|$ is the mean area of elements obtained as $|\overline{T_K}| = \frac{|\Omega|}{n}$, where n denotes the number of all elements in the domain and the symbols $\mathcal{E}^2(u_{1h}, u_{2h}, p_h, T_K)$, $\mathcal{U}^2(u_{1h}, u_{2h}, p_h, \Omega)$ are defined in (33), (39).

6.4. Numerical results

Channel with sudden intake of diameter (results for Re = 1000)

In Figures 10-11, plots of entities that characterize the flow in the channel are presented. In Figure 10, there are streamlines and a plot of the velocity component u_x. Plots of velocity component u_y and pressure are in Figure 11. Note, that the fluid flows from the right to the left on plots of u_x, u_y, and p for better view. Let us note that the relative error on the elements never exceeded 3 %. So the corner singularities caused by nonconvex corners are approximated here with very high accuracy.

7. Conclusion

Presented work is mainly focused on flow problems with singularities caused by corners in the solution domain and on the construction of the FEM solution in the vicinity of these corners as precisely as desired.

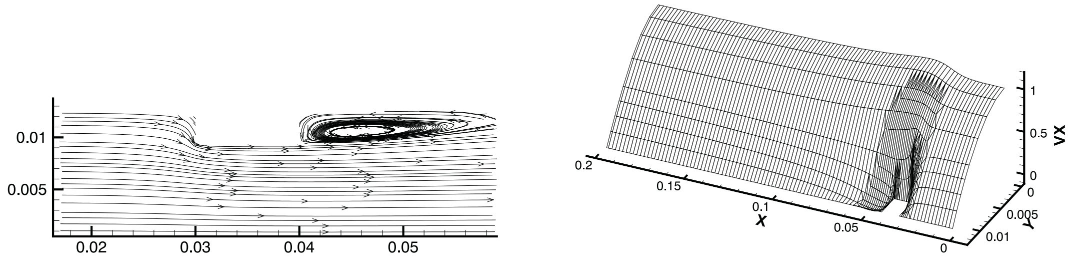


Fig. 10: *Detail of streamlines (left) and velocity component u_x (right).*

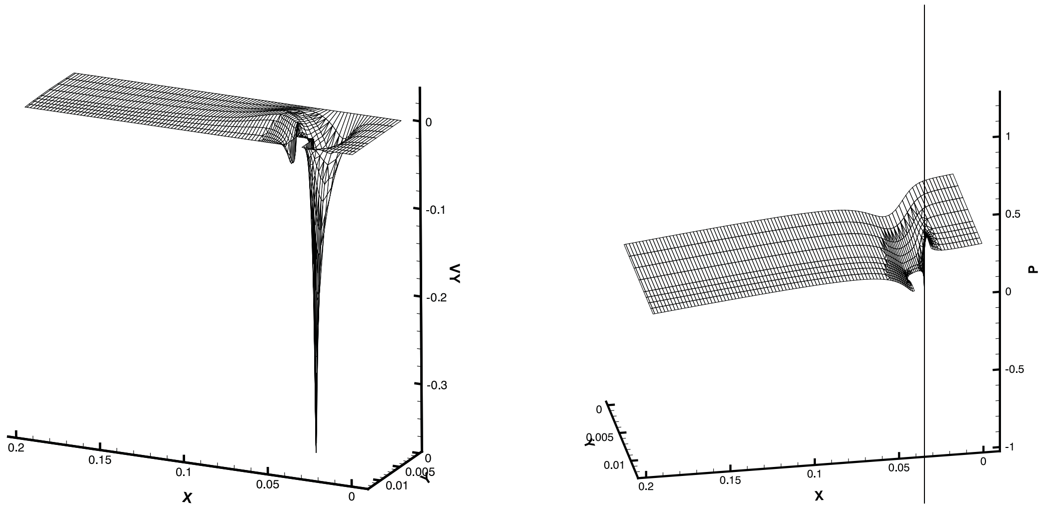


Fig. 11: *Velocity component u_y (left) and pressure (right).*

We presented two ways for getting desired precision of the FEM solution in the vicinity of corners. Both make use of qualitative properties of the mathematical model of flow. As a mathematical model we accept the Navier-Stokes equations (NSE) for incompressible fluids.

The first approach described in Section 5 makes use of a posteriori error estimates of the FEM solution which is carefully derived to trace the quality of the solution. Especially the constant in the a posteriori estimate is investigated with care. Then we use the adaptive strategy to improve the mesh and thus to improve the FEM solution. Numerical results demonstrate the robustness of this approach.

The second approach stands on two columns. In Section 4 we gave the asymptotic behaviour of the exact solution of the NSE in the vicinity of the corner. This is obtained using some symmetry of the principal part of the Stokes equation, then applying the Fourier transform and investigating the resolvent of the corresponding operator. Second column is the a priori error estimate of the FEM solution, where we estimate the seminorm of the exact solution by means of the above obtained asymptotics. In Section 6, according to these ideas, we derive an algorithm for designing the FEM mesh in advance (a priori). On the mesh we then obtain the solution with desired precision, namely in the vicinity of the corner, though there is a singularity.

The applications in Section 6 confirm the achievement of the goal – to obtain solution tinged with errors on elements satisfactorily small and uniformly distributed. Using this approach, we can save a lot of computational time using mesh 'prepared' for expected solution.

Recently we dealt also with the stabilized version of FEM to enable the calculation of flows with higher Reynolds numbers [10], [11]. We also combined stabilization with presented achievements on a posteriori error estimates. Our achievements with precise solution of problems with singularities may serve as an important tool for verification, see [12].

References

[1] M. Ainsworth, J.T. Oden: *A posteriori error estimators for the Stokes and Oseen equations.* SIAM J. Numer. Anal., **34** (1997) 228–245.

[2] I. Babuška, W.C. Rheinboldt: *A posteriori error estimates for the finite element method.* Internat. J. Numer. Meth. Engrg., **12** (1978) 1597–1615.

[3] F. Brezzi, M. Fortin: *Mixed and hybrid finite element methods*, Springer, Berlin, 1991.

[4] P. Burda: *On the FEM for the Navier-Stokes equations in domains with corner singularities.* In: M. Křížek et al. (Eds), *Finite Element Methods, Supeconvergence, Post-Processing and A Posteriori Estimates*, Marcel Dekker, New York, 1998, pp. 41–52.

[5] P. Burda: *An a posteriori error estimate for the Stokes problem in a polygonal domain using Hood-Taylor elements.* In: P. Neittaanmäki et al. (Eds) *ENUMATH 99, Proc. of the 3-rd European Conference on Numerical Mathematics and Advanced Applications.* World Scientific, Singapore, 2000, pp. 448–455.

[6] P. Burda: *A posteriori error estimates for the Stokes flow in 2D and 3D domains.* In: P. Neittaanmäki, M. Křížek (eds), *Finite Element Methods, 3D.* (GAKUTO Internat. Series, Math. Sci. and Appl., Vol. 15), Gakkotosho, Tokyo, 2001, pp. 34–44.

[7] P. Burda, J. Novotný, B. Sousedík: *A posteriori error estimates applied to flow in a channel with corners*, Math. Comput. Simulation **61** (2003), 375–383.

[8] P. Burda, J. Novotný, B. Sousedík, J. Šístek: *Finite element mesh adjusted to singularities applied to axisymmetric and plane flow.* In: M. Feistauer et al. (Eds), *Numerical Mathematics and Advanced Applications, ENUMATH 2003*, Springer, Berlin, 2004, pp. 186–195.

[9] P. Burda, J. Novotný, J. Šístek: *Precise FEM solution of corner singularity using adjusted mesh applied to axisymmetric and plane flow*, ICFD Conf. on Num. Meth. for Fluid Dynamics, Univ. Oxford, March 2004, Internat. J. Numer. Methods Fluids, **47** (2005), 1285–1292.

[10] P. Burda, J. Novotný, J. Šístek: *On a modification of GLS stabilized FEM for solving incompressible viscous flows*, Fef05, Swansea, March 2005, Internat. J. Numer. Methods Fluids, **51** (2006), 1001–1016.

[11] P. Burda, J. Novotný, J. Šístek: *Numerical solution of flow problems by stabilized finite element method and verification of its accuracy using a posteriori error estimates*, Math. Comput. Simulation **76** (2007), 28–33.

[12] P. Burda, J. Novotný, J. Šístek: *Accuracy of semiGLS stabilization of FEM for solving Navier-Stokes equations and a posteriori error estimates*, Internat. J. Numer. Methods Fluids, **56** (2008), 1167–1173.

[13] V. Girault, P.G. Raviart: *Finite element method for Navier-Stokes equations*, Springer, Berlin, 1986.

[14] R. Glowinski: *Finite element methods for incompressible viscous flow*, Handbook of Numerical Analysis, Vol. IX, Elsevier, 2003.

[15] C. Johnson: *Numerical solution of partial differential equations by the finite element method*, Cambridge University Press, 1994.

[16] V.A. Kondratiev: *Asimptotika rešenija uravnienija Nav'je-Stoksa v okrestnosti uglovoj točki granicy*, Prikl. Mat. i Mech., **1** (1967), 119–123

[17] V.A. Kondratiev: *Krajevyje zadači dlja elliptičeskich uravnĕnij v oblasťach s koniĕskimi i uglovymi točkami*, Trudy Moskov. Mat. obshch. 16 (1967), 209–292.

[18] J. Šístek: *Stabilization of finite element method for solving incompressible viscous flows*, Diploma thesis, ČVUT, Praha, 2004.

[19] R. Verfürth: *A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques*, Wiley and Teubner, Chichester, 1996.

LINEAR STABILITY OF EULER EQUATIONS IN CYLINDRICAL DOMAIN*

Libor Čermák

Abstract

The linear stability problem of inviscid incompressible steady flow between two concentric cylinders is investigated. Linearizing the transient behavior around a steady state solution leads to an eigenvalue problem for linearized Euler equations. The discrete eigenvalue problem is obtained by the spectral element method. The algorithm is implemented in MATLAB. The developed program serves as a simple tool for numerical experimenting. It enables to state rough dependency of the stability on various input velocity profiles.

1. Flow equations in the rotating cylindrical coordinate system

The inviscid incompressible flow in the cylindrical coordinate system (r, φ, z) rotating about the z-axis with the angular velocity Ω_0 is described by Euler equations

$$\begin{aligned}
&\frac{\mathrm{d}w_r}{\mathrm{d}t} - \frac{w_\varphi^2}{r} - 2\Omega_0 w_\varphi - \Omega_0^2 r + \frac{1}{\varrho}\frac{\partial p}{\partial r} = 0\,, \\
&\frac{\mathrm{d}w_\varphi}{\mathrm{d}t} + \frac{w_r w_\varphi}{r} + 2\Omega_0 w_r + \frac{1}{\varrho r}\frac{\partial p}{\partial \varphi} = 0\,, \\
&\frac{\mathrm{d}w_z}{\mathrm{d}t} + \frac{1}{\varrho}\frac{\partial p}{\partial z} = 0\,.
\end{aligned} \tag{1}$$

Here, w_r, w_φ, and w_z are radial, circumferential and axial velocities, p is the pressure,

$$\frac{\mathrm{d}}{\mathrm{d}t} = \frac{\partial}{\partial t} + w_r\frac{\partial}{\partial r} + \frac{w_\varphi}{r}\frac{\partial}{\partial \varphi} + w_z\frac{\partial}{\partial z} \tag{2}$$

is the material derivative and ϱ is the density. The continuity equation

$$\frac{1}{r}\frac{\partial}{\partial r}(r w_r) + \frac{1}{r}\frac{\partial w_\varphi}{\partial \varphi} + \frac{\partial w_z}{\partial z} = 0 \tag{3}$$

must be fulfilled as well. Equations (1)–(3) are presented, for example, in [1]. The problem is solved in the domain Q between two coaxial cylinders,

$$Q = \{(r, \varphi, z)\,|\, 0 < R_1 \le r \le R_2, 0 \le \varphi < 2\pi, 0 \le z \le L\}\,, \tag{4}$$

where $S_1 = Q \cap \{z = 0\}$ and $S_2 = Q \cap \{z = L\}$ are the pipe inlet and outlet, respectively, $\Gamma_1 = Q \cap \{r = R_1\}$ is the liquid-gas interface and $\Gamma_2 = Q \cap \{r = R_2\}$ is the pipe wall.

*This work was supported by the Council of Czech Government MSM 0021630518.

2. Linear stability

Let us suppose that the steady base flow is axially symmetric, described by functions $w_{0r}(r,z)$, $w_{0\varphi}(r,z)$, $w_{0z}(r,z)$, and $p_0(r,z)$ and by corresponding boundary conditions. To investigate the stability of the base flow to disturbances, equations that govern the evolution of these perturbations are required. To this end, the base flow is perturbed by disturbance velocities and pressure, i.e.

$$(w_r, w_\varphi, w_z, p) = (w_{0r}, w_{0\varphi}, w_{0z}, p_0) + \varepsilon(v_r, v_\varphi, v_z, \sigma)\,, \tag{5}$$

and we examine, whether $(w_r, w_\varphi, w_z, p) \to (w_{0r}, w_{0\varphi}, w_{0z}, p_0)$ for $t \to \infty$. If we substitute from (5) into (1), (3), use the fact that the stationary flow functions w_{0r}, $w_{0\varphi}$, w_{0z}, p_0 satisfy those equations, and if we neglect terms containing ε^2, we obtain linearized Euler equations

$$\begin{aligned}
&\frac{\partial v_r}{\partial t} + w_{0r}\frac{\partial v_r}{\partial r} + \frac{w_{0\varphi}}{r}\frac{\partial v_r}{\partial \varphi} + w_{0z}\frac{\partial v_r}{\partial z} + \frac{\partial w_{0r}}{\partial r}v_r + \frac{\partial w_{0r}}{\partial z}v_z - \\
&\qquad - \frac{2}{r}w_{0\varphi}v_\varphi - 2\Omega_0 v_\varphi + \frac{1}{\varrho}\frac{\partial \sigma}{\partial r} = 0\,, \\
&\frac{\partial v_\varphi}{\partial t} + w_{0r}\frac{\partial v_\varphi}{\partial r} + \frac{w_{0\varphi}}{r}\frac{\partial v_\varphi}{\partial \varphi} + w_{0z}\frac{\partial v_\varphi}{\partial z} + \frac{\partial w_{0\varphi}}{\partial r}v_r + \frac{\partial w_{0\varphi}}{\partial z}v_z + \\
&\qquad + \frac{w_{0r}}{r}v_\varphi + \frac{w_{0\varphi}}{r}v_r + 2\Omega_0 v_r + \frac{1}{\varrho r}\frac{\partial \sigma}{\partial \varphi} = 0\,, \\
&\frac{\partial v_z}{\partial t} + w_{0r}\frac{\partial v_z}{\partial r} + \frac{w_{0\varphi}}{r}\frac{\partial v_z}{\partial \varphi} + w_{0z}\frac{\partial v_z}{\partial z} + \frac{\partial w_{0z}}{\partial r}v_r + \frac{\partial w_{0z}}{\partial z}v_z + \frac{1}{\varrho}\frac{\partial \sigma}{\partial z} = 0,
\end{aligned} \tag{6}$$

and the continuity equation

$$\frac{1}{r}\frac{\partial}{\partial r}(r v_r) + \frac{1}{r}\frac{\partial v_\varphi}{\partial \varphi} + \frac{\partial v_z}{\partial z} = 0\,. \tag{7}$$

Boundary conditions are

$$v_r = v_\varphi = v_z = 0 \quad \text{on } S_1, \qquad \sigma = 0 \quad \text{on } S_2, \qquad v_r = 0 \quad \text{on } \Gamma_2. \tag{8}$$

We consider two types of boundary conditions on Γ_1:

(a) if the surface tension effect is not taken into account, zero pressure is described,

$$\sigma = 0 \qquad \text{on } \Gamma_1\,; \tag{9}$$

(b) if the surface tension effect is taken into account, we proceed in accordance with the ideas derived in [7] and [8]: we introduce the radial displacement $\Delta(\varphi, z, t)$ of the boundary Γ_1 and demand fulfillment of the impermeability equation

$$v_r = \frac{\partial \Delta}{\partial t} + w_{0z}\frac{\partial \Delta}{\partial z} \qquad \text{on } \Gamma_1\,, \tag{10}$$

the Young-Laplace equation

$$\sigma = \sigma_p \left(\frac{\partial^2 \Delta}{\partial z^2} + \frac{1}{R_1^2} \frac{\partial^2 \Delta}{\partial \varphi^2} \right) \qquad \text{on } \Gamma_1 \,, \tag{11}$$

where σ_p is the surface tension coefficient, and the initial condition

$$\Delta = 0 \qquad \text{on} \quad \Gamma_1 \cap S_1 \,. \tag{12}$$

The stability problem consists in verifying whether

$$(v_r, v_\varphi, v_z, \sigma, \Delta) \to (0,0,0,0,0) \qquad \text{for } t \to \infty. \tag{13}$$

If the condition (9) is prescribed on the boundary Γ_1, we set $\Delta = 0$ in (13).

3. The eigenvalue problem

Let $I = \sqrt{-1}$ be the imaginary unit and n be a positive whole number (so-called azimuthal wave number). Using the transformation

$$(v_r, v_\varphi, v_z, \sigma, \Delta) = e^{\lambda t + In\varphi}(u_r, u_\varphi, u_z, h, \delta), \tag{14}$$

we exclude the time t as well as the coordinate φ and obtain the eigenvalue problem

$$\begin{aligned} &\lambda u_r + w_{0r} \frac{\partial u_r}{\partial r} + w_{0\varphi} \frac{nI}{r} u_r + w_{0z} \frac{\partial u_r}{\partial z} + \frac{\partial w_{0r}}{\partial r} u_r + \frac{\partial w_{0r}}{\partial z} u_z - \\ &\qquad - \frac{2}{r} w_{0\varphi} u_\varphi - 2\Omega_0 u_\varphi + \frac{1}{\varrho} \frac{\partial h}{\partial r} = 0 \,, \end{aligned} \tag{15}$$

$$\begin{aligned} &\lambda u_\varphi + w_{0r} \frac{\partial u_\varphi}{\partial r} + \frac{nI}{r} w_{0\varphi} u_\varphi + w_{0z} \frac{\partial u_\varphi}{\partial z} + \frac{\partial w_{0\varphi}}{\partial r} u_r + \frac{\partial w_{0\varphi}}{\partial z} u_z + \\ &\qquad + \frac{w_{0r}}{r} u_\varphi + \frac{w_{0\varphi}}{r} u_r + 2\Omega_0 u_r + \frac{nI}{\varrho r} h = 0 \,, \end{aligned} \tag{16}$$

$$\lambda u_z + w_{0r} \frac{\partial u_z}{\partial r} + \frac{nI}{r} w_{0\varphi} u_z + w_{0z} \frac{\partial u_z}{\partial z} + \frac{\partial w_{0z}}{\partial r} u_r + \frac{\partial w_{0z}}{\partial z} u_z + \frac{1}{\varrho} \frac{\partial h}{\partial z} = 0 \,, \tag{17}$$

$$\frac{1}{r} \frac{\partial}{\partial r}(r u_r) + \frac{nI}{r} u_\varphi + \frac{\partial u_z}{\partial z} = 0 \,. \tag{18}$$

Here, λ is an eigenvalue and u_r, u_φ, u_z, and h are eigenfunctions (so-called normal modes) in variables r and z defined on the rectangle

$$D = \{(r,z) \,|\, 0 < R_1 \le r \le R_2, 0 \le z \le L\} \,. \tag{19}$$

The case $n = 0$ corresponds to the rotationally symmetric flow. If we take into consideration the influence of the surface tension, we add moreover equations

$$\lambda \delta(z) + w_{0z}(R_1, z)\delta'(z) - u_r(R_1, z) = 0 \,, \tag{20}$$

$$\sigma_p \left(-\delta''(z) + \frac{n^2}{R_1^2} \delta(z) \right) + h(R_1, z) = 0 \,, \tag{21}$$

where $\delta(z)$, $z \in \langle 0, L\rangle$, is a function defined on the edge $r = R_1$ of the domain D. Boundary conditions are

$$\begin{aligned} &u_r = u_\varphi = u_z = 0 & & & S_1 \cap D\,, \\ &u_r = 0 & &\text{on} & \Gamma_2 \cap D\,, \\ &h = 0 & & & S_2 \cap D\,. \end{aligned} \tag{22}$$

If we do not consider surface tension influences, then the additional boundary condition is

$$h = 0 \qquad \text{on} \quad \Gamma_1 \cap D\,, \tag{23}$$

whereas in case, when the surface tension is considered, we have, besides equations (20), (21), the additional condition

$$\delta = 0 \qquad \text{for} \quad z = 0\,. \tag{24}$$

The stability, expressed by the relation (5), occurs if and only if all eigenvalues of the problem (15)–(24) have negative real parts.

4. The discretization by the spectral element method

The approximate finite-dimensional eigenvalue problem is obtained by the spectral element method, see e.g. [2], [6], [4]. Let us explain in brief how the approximate eigenvalue problem can be obtained.

The rectangle D is divided into $n_r \times n_z$ concurrent rectangular elements D^{ij}, $i = 1, 2, \ldots, n_r$, $j = 1, 2, \ldots, n_z$, with side lengths $d_r = (R_2 - R_1)/n_r$ and $d_z = L/n_z$ in the discretization of the r-axis and z-axis, respectively. All quantities u_r, u_φ, u_z, h, w_{0r}, $w_{0\varphi}$ and w_{0z} are approximated by continuous piecewise polynomial functions, which are on every element D^{ij} polynomials of degree N uniquely determined by their values at nodes of the Gauss-Legendre-Lobatto (GLL) product quadrature formula of order $2N - 1$, see e.g. [6]. If the surface tension is considered, δ is similarly approximated by a continuous piecewise polynomial function, which is on every element D^{1j} polynomial of degree N uniquely determined by its values at nodes of the GLL quadrature formula.

Further, the variational formulation is derived. Let ψ_r, ψ_φ, ψ_z and ψ_h be test functions of the same type as corresponding piecewise polynomial approximations of u_r, u_φ, u_z, and h. Equations (15), (16), (17), and (18) are multiplied by ψ_r, ψ_φ, ψ_z and ψ_h and integrated over the domain D. The integral over the whole domain is expressed as a sum of integrals over individual elements D^{ij} and every from those integrals is computed by the GLL product quadrature formula. Variational forms connected with equations (20) and (21) are obtained similarly.

If we sum all equations and arrange unknown and free parameters in column vectors $\mathbf{u}$ and $\boldsymbol{\psi}$, respectively, we obtain

$$\boldsymbol{\psi}^T(\lambda\mathbf{B} - \mathbf{A})\mathbf{u} = 0\,,$$

and as the vector $\boldsymbol{\psi}$ is arbitrary, we arrive at the generalized eigenvalue problem

$$\mathbf{A}\mathbf{u} = \lambda \mathbf{B}\mathbf{u}\,. \tag{25}$$

The matrix $\mathbf{A}$ is regular, nonhermitian, real for $n = 0$ and complex for $n > 0$. The matrix $\mathbf{B}$ is diagonal and singular: diagonal coefficients corresponding to equations (18) and (21) are equal to zero, remaining diagonal coefficients are real and positive. Infinite eigenvalues have no influence on the stability examination and therefore we ignore them.

5. Numerical experiments

Example 1. Let us consider a hypothetical flow of water as if it was a solid body movement, set

$$\begin{array}{llll} R_1 = 0.015 & [\mathrm{m}] & R_2 = 0.15 & [\mathrm{m}] \\ L = 0.5 & [\mathrm{m}] & \varrho = 10^3 & [\mathrm{kg}\cdot\mathrm{m}^{-3}] \\ \sigma_p = 0.073 & [\mathrm{N}\cdot\mathrm{m}] & n = 0,1,2,3 & \end{array} \tag{26}$$

$w_{0r} = 0$, $w_{0\varphi} = 0$, $w_{0z} = C_0 = \text{const.}$ and experiment with values of $C_0 \geq 0$, Ω_0, n_r, n_z, and N. The flow of this type is stable and numerical results have confirmed this.

Remark. Through numerous numerical tests based on the data (26) we have found that the effects of surface tension are negligible. It is a good message saying that a steady state solution can be computed using any CFD software (whereas considering surface tension influences do not belong to standard equipment of commercial CFD software, setting the pressure on the boundary is a quite common instrument).

Example 2. We consider the case when the stationary velocity in the radial and axial direction are constant, $w_{0r} = 0$, $w_{0z} = C_0 \geq 0$, and the circumferential velocity is a function of the radial variable r,

$$w_{0\varphi} = r(ae^{-r/b} - \Omega_0)\,. \tag{27}$$

Constants $a > 0$, $b > 0$ are optional parameters: $c_{0\varphi} = w_{0\varphi} + \Omega_0 r = are^{-r/b}$ approaches its maximal value for $r = b$, a influences $\max|c_{0\varphi}(r)|$. We use again the data (26).

If we set $C_0 = 0$, $n = 0$ (which means that only axisymmetric perturbations were permitted) and instead of the boundary condition (23) we demanded $u_r = 0$ for $r = R_1$, we obtained the stability just when $R_2 < 2b$, which is in coincidence with the well known Rayleigh's criterion, see e.g. [5]. Further, an increase of C_0 caused an increase of the stability (i.e. $\max \mathrm{Re}(\lambda)$ of all finite λ was decreased).

For $C_0 = 0$, $n > 0$ we did not obtain the stability for neither a nor b. If C_0 was increased, the stability turned up (for appropriate values of a and b).

Another velocity profiles for $c_{0\varphi} = w_{0\varphi} + \Omega_0 r$ were created interactively. A given set of discrete points was interpolated by means of a cubic spline and then the dependence of the stability on the velocity $w_{0\varphi}$ and parameters C_0, Ω_0, n, N, n_r, n_z was examined. The obtained results fulfilled our expectancy.

Example 3. The steady state flow velocities w_{0r}, $w_{0\varphi}$, and w_{0z} were computed by the CFD package FLUENT with the data $R_1 = 0.015$, $R_2 = 0.15$, $L = 1$, $\varrho = 10^3$ and with the boundary conditions

$$\begin{aligned}
&\text{on the inlet } z = 0 : && w_{0r} = 0, \quad w_{0\varphi} = r(7.5e^{-r/0.05} - 5), \quad w_{0z} = 1,\\
&\text{on the outlet } z = L : && p_0 = p_{outlet},\\
&\text{on the interface } r = R_1 : && p_0 = p_{interface},\\
&\text{on the wall } r = R_2 : && w_{0r} = 0.
\end{aligned}$$

Here, p_{outlet} and $p_{interface}$ are the constant pressure invoking cavitating vortex rope and the saturated vapour pressure, respectively. The surface tension influences were not taken into account. The stability examination was performed for the following parameter values: $\Omega_0 = 5$, $n_r = 4$, $n_z = 1$, $N = 8$, and $n = 0, 1, 2$. The stability of the flow, resulting from the transient FLUENT modeling, was confirmed: all finite eigenvalues had negative real parts.

References

[1] M. Brdička, L. Samek, B. Sopko: *Mechanika kontinua.* Praha, Academia 2000.

[2] C. Canuto, M. Y. Hussaini, A. Quarteroni, T. A. Zang: *Spectral methods. Evolution to complex geometries and applications to fluid dynamics.* Berlin, Springer 2007.

[3] W. O. Criminale, T. L. Jackson, R. D. Joslin: *Theory and computation in hydrodynamic stability.* Cambridge, Cambridge University Press 2003.

[4] M. O. Deville, P. F. Fisher, E. H. Mund: *High-order methods for incompressible fluid flow.* Cambridge, Cambridge University Press 2002.

[5] P. G. Drazin, W. H. Reid: *Hydrodynamic stability.* Cambridge, Cambridge University Press 2004.

[6] G. E. Karniadakis, S. J. Sherwin: *Spectral/hp element methods for computational fluid dynamics.* Oxford, Oxford University Press 2005.

[7] F. Pochylý, V. Habán, P. Rudolf: The stability of the infinite cavitating vortex rope, (in Czech), *Research note* VUT-EU13303-QR-10-07. Faculty of Mechanical Engineering, Brno Univ. of Technology, 2007.

[8] F. Pochylý, P. Rudolf, V. Habán: Analytical method for stability analysis of steady flow in rotationally symmetrical domain, (in Czech), *Research note* VUT-EU13303-QR-17-07. Fac. of Mech. Engineering, Brno Univ. of Technology, 2007.

[9] V. Theofilis: Advances in global linear instability analysis of nonparalel and three-dimensional flows. Prog. Aero. Sci. **39** (2003), 249–315.

ADAPTIVE FRAME METHODS WITH CUBIC SPLINE-WAVELET BASES*

Dana Černá, Václav Finěk

In recent years, adaptive wavelet methods have been successfully used for solving operator equations [2, 3, 5]. It has been shown that these methods converge and that they are asymptotically optimal in the sense that storage and number of floating point operations, needed to resolve the problem with desired accuracy, remain proportional to the problem size when the resolution of the discretization is refined.

Suitable wavelet bases on bounded domains are needed for these methods. They are usually constructed in the following way: Wavelets on the real line are adapted to the interval and then by tensor product technique to the n-dimensional cube. Finally, by splitting the domain into nonoverlapping subdomains which are images of $(0,1)^n$ under appropriate parametric mappings, one can obtain wavelet bases on a fairly general domain. However, it can be very difficult to find these parametric mappings. For this reason, more general adaptive wavelet-frame methods were proposed in [7, 10]. These methods use frames instead of wavelet bases. A frame on a bounded domain can be obtained by a union of wavelet bases on the overlapping subdomains, which are lifted tensor products of a basis on the unit interval. Thus, the construction of wavelet frames is much simpler than the construction of wavelet bases.

The effectiveness of adaptive wavelet and frame methods is strongly influenced by the choice of the wavelet basis on the interval, in particular by its conditioning. However, the conditioning of the known spline-wavelet bases [8, 9] becomes bad for primal polynomial exactness of order $N > 3$, which causes problems in practical applications. In our contribution, we focus on the cubic case, i.e. $N = 4$, and we propose a construction of cubic spline-wavelet bases on the interval adapted for complementary boundary conditions of the first order. We show that these bases are well-conditioned and that the corresponding stiffness matrices have small condition numbers. Furthermore, we show that the adaptive wavelet frame method from [7] with bases constructed in our paper realizes the optimal convergence rate.

1. Construction of boundary adapted spline-wavelet bases

In this section, we introduce a construction of stable spline-wavelet bases on the interval satisfying complementary boundary conditions of the first order. It

[*] The research of the first author has been supported by the MSMT project no. LC06024, while the second author has been supported by the project IGS financed by Technical University in Liberec.

means that the primal wavelet basis is adapted for homogeneous Dirichlet boundary conditions of the first order, while the dual wavelet basis preserves the full degree of polynomial exactness. This construction is based on the spline-wavelet bases from [4]. Let $\tilde{N}$ be the order of polynomial exactness of the dual MRA.

Let $\Phi_j^{old} = \{\phi_{j,k}, k = -3, \ldots, 2^j - 1\}$ be the primal scaling basis on level j from [4]. The functions $\phi_{j,-3}$, $\phi_{j,2^j-1}$ are the only two functions which do not vanish at boundary points. Therefore, defining

$$\Phi_j := \left\{\phi_{j,k}, k = -2, \ldots, 2^j - 2\right\} \tag{1}$$

we obtain primal scaling bases satisfying the first order Dirichlet boundary conditions.

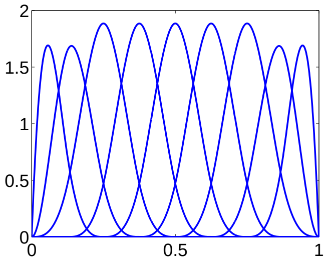


Fig. 1: *Cubic primal scaling basis for $\tilde{N} = 6$, $j = 3$ satisfying complementary boundary conditions of the first order.*

On the dual side, we also need to omit one scaling function at each boundary, because the number of the primal scaling functions must be the same as the number of the dual scaling functions. Let $\Theta_j^{old} = \left\{\theta_{j,k}^{old}, k = -3, \ldots, 2^j - 1\right\}$ be the dual scaling basis on level j before biorthogonalization from [4]. There are boundary functions of two types. The functions $\theta_{j,-3}^{old}$, $\ldots$, $\theta_{j,-4+\tilde{N}}^{old}$ are left boundary functions of the first type which are defined to preserve polynomial exactness of order $\tilde{N}$. The functions $\theta_{j,-3+\tilde{N}}^{old}$, $\ldots$, $\theta_{j,\tilde{N}-2}^{old}$ are left boundary functions of the second type. The right boundary scaling functions are then derived by reflection of the left boundary functions. Since we want to preserve the full degree of polynomial exactness, we omit one function of the second type at each boundary. Thus, we define

$$\begin{aligned} \theta_{j,k} &= \theta_{j,k-1}^{old}, \quad && k = -2, \ldots, -3 + \tilde{N}, \\ \theta_{j,k} &= \theta_{j,k}^{old}, \quad && k = -2 + \tilde{N}, \ldots, 2^j - \tilde{N} - 2, \\ \theta_{j,k} &= \theta_{j,k+1}^{old}, \quad && k = 2^j - \tilde{N} - 1, \ldots, 2^j - 2. \end{aligned}$$

Since the set $\Theta_j := \{\theta_{j,k} : k = -2, \ldots, 2^j - 2\}$ is not biorthogonal to Φ_j, we derive a new set $\tilde{\Phi}_j$ from Θ_j by biorthogonalization. Let $\mathbf{A}_j = \left(\langle \phi_{j,k}, \theta_{j,l} \rangle\right)_{k,l=-2}^{2^j-2}$, then viewing $\tilde{\Phi}_j$ and Θ_j as column vectors we define $\tilde{\Phi}_j := \mathbf{A}_j^{-T} \Theta_j$, assuming that $\mathbf{A}_j$ is invertible, which is the case of all choices of $\tilde{N}$ in our numerical experiments.

Our next goal is to determine the corresponding sets of wavelets at the scale j, i.e. $\Psi_j := \{\psi_{j,k}, k = 1, \dots 2^j\}$, $\tilde{\Psi}_j := \left\{\tilde{\psi}_{j,k}, k = 1, \dots 2^j\right\}$. We follow a general principle called stable completion as in [8] with some small changes. Since this construction is quite subtle we do not go into details here.

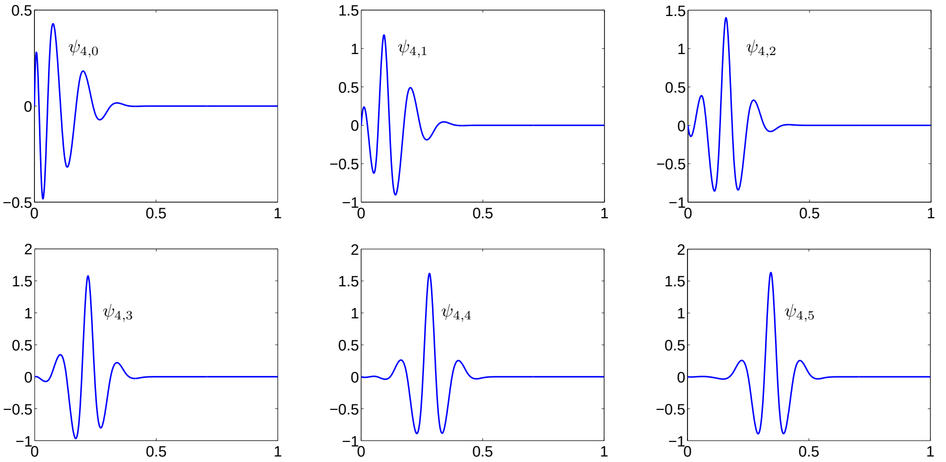


Fig. 2: *Some cubic primal wavelets for $\tilde{N} = 6$ satisfying the complementary boundary conditions of the first order.*

2. Quantitative properties of the constructed bases

In this section, quantitative properties of the constructed bases are presented. In order to further improve the condition we provide L^2-normalization of the primal functions. Then we multiply the dual functions by appropriate constants to preserve biorthogonality. The L^2-normalized bases are denoted by the superscript N. The conditioning of the resulting single-scale bases are listed in Table 1.

N	$\tilde{N}$	j	Φ_j	Φ_j^N	$\tilde{\Phi}_j$	$\tilde{\Phi}_j^N$	Ψ_j	Ψ_j^N	$\tilde{\Psi}_j$	$\tilde{\Psi}_j^N$
4	6	6	4.53	4.30	7.89	6.83	9.46	8.00	16.37	7.96
4	8	6	4.53	4.30	11.15	10.05	8.45	8.02	25.30	15.26
4	10	6	4.53	4.30	17.89	16.97	8.39	8.42	37.65	35.80

Tab. 1: *The conditioning of single-scale scaling and wavelet bases.*

The other criterion for the effectiveness of wavelet bases is the condition number of the corresponding stiffness matrix. Here, let us consider the stiffness matrix for the Poisson equation:

$$\mathbf{A}_{j_0,s} = \left(\langle \psi'_{j,k}, \psi'_{l,m} \rangle\right)_{\psi_{j,k},\psi_{l,m} \in \Psi_{j_0,s}}, \tag{2}$$

where $\Psi_{j_0,s} = \Phi_{j_0} \cup \bigcup_{j=j_0}^{j_0+s-1} \Psi_j$ denotes the multiscale basis. It is well-known that the condition number of $\mathbf{A}_{j_0,s}$ increases quadratically with the matrix size. To remedy this, we use a diagonal matrix for preconditioning

$$\mathbf{A}_{j_0,s}^{prec} = \mathbf{D}_{j_0,s}^{-1} \mathbf{A}_{j_0,s} \mathbf{D}_{j_0,s}^{-1}, \quad \mathbf{D}_{j_0,s} = \operatorname{diag}\left(\left\langle \psi_{j,k}', \psi_{j,k}' \right\rangle^{1/2} \right)_{\psi_{j,k} \in \Psi_{j_0,s}}. \tag{3}$$

To further improve the condition number of $\mathbf{A}_{j_0,s}^{prec}$ we apply orthogonal transformation to the scaling basis on the coarsest level as in [1] and then we use diagonal matrix for preconditioning. We denote the obtained matrix by $\mathbf{A}_{j_0,s}^{ort}$. Condition numbers of the resulting matrices are listed in Table 2.

N	$\tilde{N}$	j	s	M	$\mathbf{A}_{j,s}^{prec}$	$\mathbf{A}_{j,s}^{ort}$	N	$\tilde{N}$	j	s	M	$\mathbf{A}_{j,s}^{prec}$	$\mathbf{A}_{j,s}^{ort}$
4	4	4	1	33	47.02	15.38	4	8	5	1	65	205.56	15.92
			3	129	49.56	17.40				3	257	208.37	25.04
			5	513	50.17	18.52				5	1025	209.12	27.47
			7	2049	50.28	18.91				7	4097	209.31	27.69
4	6	4	1	33	48.98	15.25	4	10	5	1	65	224.22	22.51
			3	129	49.56	15.94				3	257	226.17	81.72
			5	513	50.17	16.24				5	1025	226.42	91.26
			7	2049	50.28	16.31				7	4097	226.63	92.17

Tab. 2: *The condition numbers of stiffness matrices* $\mathbf{A}_{j,s}^{prec}$, $\mathbf{A}_{j,s}^{ort}$ *of the size* $M \times M$.

3. Adaptive frame method with the constructed bases

Adaptive frame methods are designed in particular for solving operator equations on complicated domains. However, even in some one-dimensional numerical examples the optimal convergence rate was not realized, probably due to stability problems of the used bases. Our intention is to show that the optimal convergence rates of adaptive wavelet frame methods can be achieved also for the case of cubic spline wavelets. We should emphasize that we consider the one-dimensional example as a milestone on the way to higher-dimensional problems.

We consider the same test example as in [6], i.e. the Poisson equation

$$-u'' = f \quad \text{in} \quad \Omega = (0,1), \quad u(0) = u(1) = 0, \tag{4}$$

with the functional f defined by

$$f(v) = 4v\left(\frac{1}{2}\right) - \int_0^1 \left(9\pi^2 \sin(3\pi x) + 4\right) v(x)\, dx. \tag{5}$$

Then the solution u is given by

$$u(x) = \begin{cases} -\sin(3\pi x) + 2x^2, & x \in [0, 0.5), \\ -\sin(3\pi x) + 2(1-x)^2, & x \in [0.5, 1]. \end{cases} \tag{6}$$

To test our bases, we construct a wavelet frame on Ω simply as the union of interval wavelet bases on $\Omega_1 = (0, 0.7)$ and $\Omega_2 = (0.3, 1)$. Note that the singularity is contained in the overlapping part and thus the boundary scaling functions and wavelets, which may potentially cause instabilities, are more involved in the frame than in the wavelet approach. This is the reason why we use wavelet frames instead of wavelet bases directly.

Let us define

$$\mathbf{A} = \mathbf{D}^{-1} \left\langle \Psi', \Psi' \right\rangle \mathbf{D}^{-1}, \quad \mathbf{f} = \mathbf{D}^{-1} \left\langle f, \Psi \right\rangle, \quad \mathbf{D} = \operatorname{diag} \left(\left\langle \psi'_{j,k}, \psi'_{j,k} \right\rangle^{1/2} \right)_{\psi_{j,k} \in \Psi}. \tag{7}$$

Then the variational formulation of (4) is equivalent to

$$\mathbf{AU} = \mathbf{f}, \tag{8}$$

and the solution u is given by $u = \mathbf{UD}^{-1}\Psi$. We solve the infinite dimensional problem (8) by the inexact damped Richardson iterations, for details we refer to [7].

Since the solution u has limited Sobolev regularity, $u \in H^s(\Omega) \cap H_0^1(\Omega)$ only for $s < 1.5$, the linear methods can only converge with limited order. Let $B_q^s(L^p(\Omega))$ denote a Besov space of smoothness s over $L^p(\Omega)$ with additional index q. It can be shown that $u \in B_\tau^{s+1}(L^\tau(\Omega))$ for any positive s and $\tau = (s + 0.5)^{-1}$. Therefore,

$$\left\| \mathbf{U} - \mathbf{U}_k \right\|_{l^2} \leq C \left(\# \text{ supp } \mathbf{U}_k \right)^{-n}, \tag{9}$$

where $\mathbf{U}_k$ is the k-th approximate iterand. The theoretical rate of convergence n is limited only by the polynomial exactness of the underlying wavelet bases, in our case the relation (9) holds for any $n < 3$. Figure 3 shows the logarithmic plot of the realized convergence rate for the bases designed in this contribution with $\tilde{N} = 4$ and $\tilde{N} = 6$.

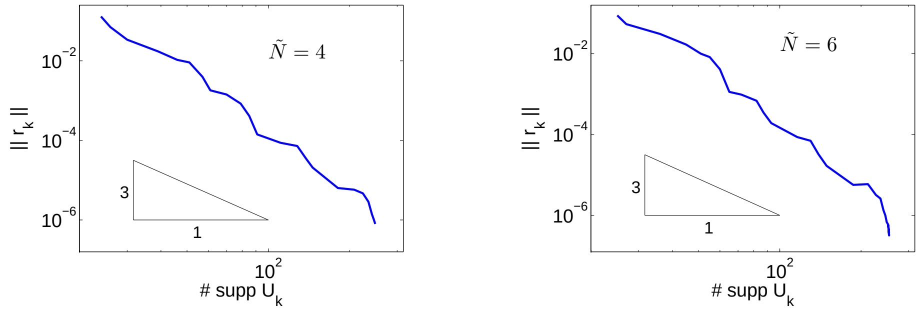


Fig. 3: *The l^2 norm of the residual $\mathbf{r}_k = \mathbf{f} - \mathbf{AU}_k$ versus the number of degrees of freedom.*

To conclude: We proposed a construction of cubic spline-wavelet bases on the interval adapted for complementary boundary conditions of the first order. As opposed to bases from [8, 9], bases constructed in our paper are well-conditioned, the corresponding stiffness matrices have small condition numbers and the adaptive wavelet frame method from [7] with our bases realizes the optimal convergence rate.

References

[1] C. Canuto, A. Tabacco, K. Urban: *The wavelet element method*, Part 1: Construction and Analysis. Appl. Comp. Harm. Anal. **6** (1999), 1–52.

[2] A. Cohen, W. Dahmen, R. DeVore: *Adaptive wavelet schemes for elliptic operator equations – convergence rates*. Math. Comput. **70** (2001), 27–75.

[3] A. Cohen, W. Dahmen, R. DeVore: *Adaptive wavelet methods II – beyond the elliptic case*. Found. Math. **2** (2002), 203–245.

[4] D. Černá, V. Finěk: *Optimized construction of biorthogonal spline-wavelets*. In: T.E. Simos et al. (Eds.), ICNAAM 2008. AIP Conference Proceedings 1048, American Institute of Physics, New York, 2008, pp. 134–137.

[5] S. Dahlke, W. Dahmen, K. Urban: *Adaptive wavelet methods for saddle point problems – optimal convergence rates*. SIAM J. Numer. Anal. **40** (2002), 1230–1262.

[6] S. Dahlke, M. Fornasier, M. Primbs, T. Raasch, M. Werner: *Nonlinear and adaptive frame approximation schemes for elliptic PDEs: Theory and numerical experiments*. Preprint, Philipps-Universität Marburg, 2007.

[7] S. Dahlke, M. Fornasier, T. Raasch: *Adaptive frame methods for elliptic operator equations*. Adv. Comp. Math. **27** (2007), 27–63.

[8] W. Dahmen, A. Kunoth, K. Urban: *Biorthogonal spline wavelets on the interval – stability and moment conditions*. Appl. Comp. Harm. Anal. **6** (1999), 132–196.

[9] M. Primbs: *New stable biorthogonal spline-wavelets on the interval*. Preprint, Universität Duisburg-Essen, 2007.

[10] R. Stevenson: *Adaptive solution of operator equations using wavelet frames*. SIAM J. Numer. Anal. **41** (2003), 1074–1100.

AN APPLICATION OF THE AVERAGED GRADIENT TECHNIQUE[*]

Jan Chleboun

Dedicated to Ivan Hlaváček on the occasion of his 75th birthday.

1. Introduction

Gradient averaging (also known as gradient recovery (GR)) is a technique for improving the accuracy of an approximate gradient obtained via a numerical method. Sensitivity analysis deals with analyzing the response of a function (or a functional) to a small perturbation of its input values. In this contribution, we limit ourselves to gradients originating from finite element solutions of boundary value problems (BVPs) and to criterion-functionals that evaluate these solutions. Our goal is to show that the use of a gradient recovery technique in sensitivity analysis formulae can result in a better assessment of the quality of approximate minimizers of criterion-functionals that appear in parameter identification problems or the worst scenario method, for example.

Let us finish this short introductory part with a few words to honor Ivan Hlaváček from the Institute of Mathematics of the Academy of Sciences of the Czech Republic who recently celebrated his 75th birthday. He has pioneered a mathematically rigorous analysis of the worst scenario problems since the mid-nineties and also contributed to the family of gradient averaging techniques. The following pages thus pay tribute to his scientific achievements.

2. Averaged gradient

The idea to improve the accuracy of an approximate gradient calculated by the finite element method is more than two decades old and has materialized in numerous applications. Take, for example, Zienkiewicz-Zhu error estimators stemming from [8]. The contribution of Czech mathematicians is not negligible, see [2, 3, 4, 5, 6], for instance.

Although various recovery techniques have been designed, let us confine ourselves to a simple averaging method proposed and analyzed in [3].

Let $\Omega \subset \mathbb{R}^2$ be a bounded domain with a polyhedral Lipschitz boundary. Let $\mathcal{F} = \{T_h\}_{h\to 0}$ be a family of triangulations of $\overline{\Omega}$, where h is the maximum diameter

[*]This research was supported by the Ministry of Education, Youth, and Sports of the Czech Republic through contract MSM 6840770003. The author's participation in the PANM 14 Conference was made possible through grant No. IAA100190803 from the Grant Agency of AS CR.

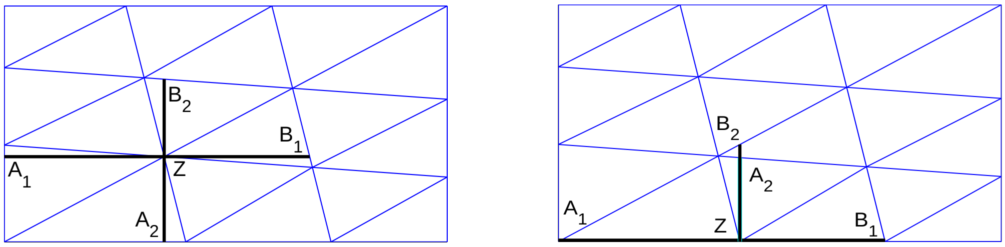


Fig. 1: *Auxiliary points: for an inner node (left), for a boundary node (right).*

of all elements $K \in T_h$. Let $V_h = \{v_h \in C(\overline{\Omega}) : \ v_h|_K \in P_1(K) \ \forall K \in T_h\}$, where $P_1(K)$ stands for the space of linear polynomials on K.

For a mesh node Z, we draw two lines parallel to the axes (see Figure 1), find their intersections with the edges of those triangles that share the vertex Z, and label these intersection points A_1, A_2, B_1, and B_2 as in Figure 1. We then set

$$a_i = (A_i - Z)_i, \quad b_i = (B_i - Z)_i, \quad i = 1, 2.$$

For $v_h \in V_h$, the components of the weighted averaged gradient $G_h v_h \in V_h$ at Z are defined as follows

$$(G_h v_h(Z))_i = \alpha_i v_h(A_i) - (\alpha_i + \beta_i) v_h(Z) + \beta_i v_h(B_i), \tag{1}$$

where $i = 1, 2$ and $\alpha_i = b_i/(a_i(b_i - a_i))$, $\beta_i = a_i/(b_i(a_i - b_i))$.

If $v \in C(\overline{\Omega})$, then (1) can also be applied (with v_h replaced by v) to define $G_h v \in V_h$, a continuous piece-wise linear approximation of ∇v.

We refer to [3] for details and a generalization to $\mathbb{R}^3$ as well as for situations that are not covered by Figure 1.

The features of $\mathcal{F}$ are substantial for the order of accuracy of $G_h v$. Let us recall that $\mathcal{F}$ is called a *strongly regular* family of triangulations if

$$\exists \varkappa > 0 \quad \forall T_h \in \mathcal{F} \quad \forall K \in T_h \quad \varkappa h \le \varrho_K,$$

where ϱ_K is the radius of the largest ball inscribed in K.

It is known, see [3, Theorem 3.8], that if $q \in (1, \infty)$ and $\mathcal{F}$ is strongly regular, then a constant $C > 0$ exists such that for any v belonging to the Sobolev space $W_q^3(\Omega)$, the following estimate holds (note the order h^2)

$$\|\nabla v - G_h v\|_{0,q,\Omega} \le C h^2 |v|_{3,q,\Omega} \quad \forall T_h \in \mathcal{F}. \tag{2}$$

Let us focus on the recovered gradient of a finite element (FE) solution.

We consider $u \in V_0 = H_0^1(\Omega)$ ($W_2^1(\Omega)$-functions with zero trace), a unique weak solution to the following V_0-elliptic BVP

$$-\operatorname{div}(\tilde{\lambda} \nabla u) = f \quad \text{in } \Omega, \tag{3}$$

$$u = 0 \quad \text{on } \partial\Omega, \tag{4}$$

where $\tilde{\lambda} \in W^2_{2+\varepsilon}(\Omega)$ for some $\varepsilon > 0$ and $f \in W^1_q(\Omega)$. Let us remark that $\tilde{\lambda}$ can be a symmetric matrix of functions, see [3]. It is assumed that $u \in W^3_q(\Omega)$, where $q > 2$.

Next, $u_h \in V^0_h = V_0 \cap V_h$, the piece-wise linear FE approximation of u, is defined through

$$\int_\Omega \tilde{\lambda}(x)\nabla u_h(x) \cdot \nabla v_h(x)\,\mathrm{d}x = \int_\Omega f(x)v_h(x)\,\mathrm{d}x \quad \forall v_h \in V^0_h. \tag{5}$$

To obtain a result similar to (2), we have to resort to more stringent assumptions about the meshes.

Let us consider $\mathcal{F}' \subset \mathcal{F}$, a special class of smoothly distorted uniform meshes, see [7, 3] for details and Figure 2 for an illustration of such a mesh.

For these meshes and for a fixed subdomain $\Omega_0 \subset\subset \Omega$, an analogue to (2) holds, see [3, page 22],

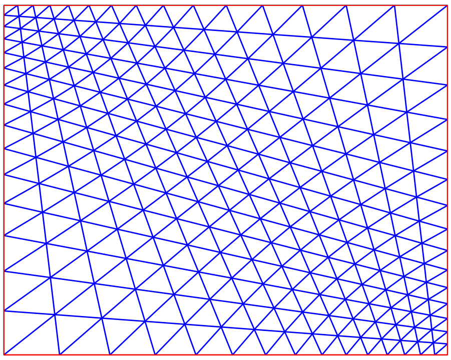


Fig. 2: *Distorted uniform mesh.*

$$\|\nabla u - G_h u_h\|_{0,2,\Omega_0} \leq C(u)h^2. \tag{6}$$

3. A parameter identification application

In this section, we will introduce a parameter identification problem.

For simplicity, let $\tilde{\lambda}$ be controlled by six parameters forming a vector λ, that is, $\lambda = (\lambda_1, \ldots, \lambda_6)$ and

$$\tilde{\lambda}(x) = \lambda_1 + \lambda_2 x_1 + \lambda_3 x_2 + \lambda_4 x_1^2 + \lambda_5 x_1 x_2 + \lambda_6 x_2^2, \quad x = (x_1, x_2) \in \Omega.$$

We define the criterion-functional (and its approximation) that appears in parameter identification problems:

$$\Psi(\lambda) = \int_\Omega |\nabla u(\lambda) - \nabla w|^2\,\mathrm{d}x \quad \text{and} \quad \Psi_h(\lambda) = \int_\Omega |\nabla u_h(\lambda) - \nabla w|^2\,\mathrm{d}x,$$

where Ψ evaluates $u(\lambda)$, the λ-dependent weak solution to (3)–(4), and Ψ_h evaluates $u_h(\lambda)$ determined by (5). In both cases, w is a given function.

Let $\lambda \in \mathcal{U}_{\mathsf{ad}} \subset \mathbb{R}^6$, where $\mathcal{U}_{\mathsf{ad}}$ is a set of admissible parameters. The specification of $\mathcal{U}_{\mathsf{ad}}$ is not necessary for the purposes of this paper; roughly speaking, $\mathcal{U}_{\mathsf{ad}}$ is a compact subset of $\mathbb{R}^6$ such that the BVP is uniformly V_0-elliptic with respect to $\lambda \in \mathcal{U}_{\mathsf{ad}}$.

We formulate a parameter identification problem and its approximation: Find $\lambda_0 \in \mathcal{U}_{\mathsf{ad}}$ and $\lambda^h_0 \in \mathcal{U}_{\mathsf{ad}}$ such that $\lambda_0 = \arg\min\limits_{\lambda \in \mathcal{U}_{\mathsf{ad}}} \Psi(\lambda)$ and $\lambda^h_0 = \arg\min\limits_{\lambda \in \mathcal{U}_{\mathsf{ad}}} \Psi_h(\lambda)$.

An elementary sensitivity analysis (see [1]) results in

$$\frac{\partial \Psi_h}{\partial \lambda_i}(\lambda) = -\int_\Omega \frac{\partial \tilde{\lambda}}{\partial \lambda_i} \nabla u_h \cdot \nabla z_h \, \mathrm{d}x, \quad i = 1, 2, \ldots, 6, \tag{7}$$

where $z_h \in V_h^0$ is the solution to the adjoint equation

$$\int_\Omega \tilde{\lambda} \nabla z_h \cdot \nabla v_h \, \mathrm{d}x = \int_\Omega 2(\nabla u_h - \nabla w) \cdot \nabla v_h \, dx \quad \forall v_h \in V_h^0(\Omega). \tag{8}$$

Computational experiments show that a gradient minimization procedure based on the derivative (7) is quite efficient in the search for λ_0^h. Nevertheless, the degree of accuracy of λ_0^h and $\Psi_h(\lambda_0^h)$ remains unknown.

To obtain at least an indicator of the (in)accuracy of the minimization results, we will apply the above-mentioned gradient recovery technique.

Let us define both a new functional

$$\Psi_h^G(\lambda) = \int_\Omega |G_h u_h(\lambda) - \nabla w|^2 \, \mathrm{d}x, \tag{9}$$

where $u_h = u_h(\lambda)$ solves (5), and a new equation determining $z_h^G \in V_h^0$ through

$$\int_\Omega \tilde{\lambda} \nabla z_h^G \cdot \nabla v_h \, \mathrm{d}x = \int_\Omega 2(G_h u_h - \nabla w) \cdot \nabla v_h \, \mathrm{d}x \quad \forall v_h \in V_h^0. \tag{10}$$

Strictly speaking, (10) is not the exact adjoint equation to (9) and (5) because the right-hand side of (10) is not the exact derivative of Ψ_h^G with respect to u_h; see [1] for the derivation of adjoint equations. As a consequence, z_h^G does not yield the exact derivative of Ψ_h^G. However, the approximation, i.e.,

$$\frac{\partial \Psi_h^G}{\partial \lambda_i}(\lambda) \approx -\int_\Omega \frac{\partial \tilde{\lambda}}{\partial \lambda_i} \nabla u_h(\lambda) \cdot \nabla z_h^G(\lambda) \, \mathrm{d}x, \quad i = 1, 2, \ldots, 6,$$

is sufficiently accurate to be used in solving the following minimization problem: Find

$$\lambda_0^{h,G} = \underset{\lambda \in \mathcal{U}_{\mathsf{ad}}}{\arg\min} \; \Psi_h^G(\lambda). \tag{11}$$

We end up with two approximate minimum points, that is, λ_0^h and $\lambda_0^{h,G}$, with two respective approximate state solutions $u_h(\lambda_0^h)$ and $u_h(\lambda_0^{h,G})$, and three approximate criterion-functional values, namely $\Psi_h(\lambda_0^h)$, $\Psi_h^G(\lambda_0^h)$, and $\Psi_h^G(\lambda_0^{h,G})$; the fourth value, $\Psi_h(\lambda_0^{h,G})$, is not relevant for our purposes.

The distances

$$\sigma_\lambda = \left|\lambda_0^h - \lambda_0^{h,G}\right|, \; \sigma_\Psi = \left|\Psi_h(\lambda_0^h) - \Psi_h^G(\lambda_0^{h,G})\right|, \text{ and } \sigma_\Psi^G = \left|\Psi_h^G(\lambda_0^h) - \Psi_h^G(\lambda_0^{h,G})\right|$$

can indicate an inaccuracy in the approximation of the exact solution pair λ_0 and $\Psi(\lambda_0)$. For $h \to 0_+$, it should be $\sigma_\lambda, \sigma_\Psi, \sigma_\Psi^G \to 0$.

Let us remark that there is no guarantee that $\lambda_0^{h,G}$ approximates λ_0 better than λ_0^h does.

Example
The problem is defined by $\Omega = (-3,3) \times (-3,3)$, $w = \exp(-x_1^2 - x_2^2)$, and $\Psi(\lambda) = 1000\|\nabla(u-w)\|_{0,2,\Omega}^2$. The right-hand side function f, see (3), is determined by w and a predefined polynomial $\widehat{\lambda}$. Although w does not comply with the homogeneous boundary condition, the difference is rather small and, for $\widehat{\lambda}$, the state solution u is close to w. A distorted mesh is formed by 800 triangles ($h = 0.3$).

The minimization process starts at $\lambda_{\mathrm{s}} = (2,2,2,2,2,2)$; we obtain $\Psi_h(\lambda_{\mathrm{s}}) = 1641$ (GR not used) and $\Psi_h^G(\lambda_{\mathrm{s}}) = 1693$ (GR is used).

The calculated results are as follows

$$\begin{aligned} \Psi_h(\lambda_0^h) &= 93.9 && \text{GR is not used in the minimization,} \\ \Psi_h^G(\lambda_0^h) &= 26.5 && \text{GR is applied just to } u_h(\lambda_0^h), \\ \Psi_h^G(\lambda_0^{h,G}) &= 3.9 && \text{GR is used in the minimization,} \\ \sigma_\lambda &= 0.21, \qquad \sigma_\Psi = 90.0, \qquad \sigma_\Psi^G = 22.6. \end{aligned}$$

The minimum value $\Psi_h(\lambda_0^h) = 93.9$ is the result of the minimization procedure where $\nabla u_h(\lambda)$ is piece-wise constant. There is no indication whether or not $\Psi_h(\lambda_0^h)$ is close to $\Psi(\lambda_0)$. If the averaged gradient of $u_h(\lambda_0^h)$ is evaluated by the criterion-functional, we obtain $\Psi_h^G(\lambda_0^h) = 26.5$, which shows that $G_h u_h$ is a better approximation of ∇w. This can be anticipated because ∇w is most significant in a subdomain $\Omega_0 \subset\subset \Omega$ and (6) holds. The minimization based on (9) leads to even lower minimum $\Psi_h^G(\lambda_0^{h,G}) = 3.9$, which says that the averaged gradient of the state solution $u_h(\lambda_0^{h,G})$ is so far the best approximation to ∇w. Nevertheless, $\nabla u_h(\lambda_0^{h,G})$ (piece-wise constant) does not outweigh $\nabla u_h(\lambda_0^h)$ because the latter results in the criterion minimum value without GR. As indicated by $\sigma_\lambda = 0.21$, the difference between λ_0^h and $\lambda_0^{h,G}$ is rather significant.

We can draw a few conclusions: (a) for the given h, $\Psi_h(\lambda_0^h)$ is not a good approximation of $\Psi(\lambda_0)$ (note that $\Psi_h(\lambda_{\mathrm{s}})$ seems to be a sufficient approximation of $\Psi(\lambda_{\mathrm{s}})$ because GR leads to a change of only 3%); (b) we can expect that $\Psi(\lambda_0)$ is "close" to zero; (c) the values $\Psi_h(\lambda_0^h)$, $\Psi_h^G(\lambda_0^h)$, and $\Psi_h^G(\lambda_0^{h,G})$ are significantly different; this means that both a refinement of the mesh as well as further minimization in the neighborhood of λ_0^h or $\lambda_0^{h,G}$ are necessary to gain confidence in the calculated minimum.

Acknowledgement: The author is indebted to Dr. Richard (Dick) Haas for many fruitful discussions.

References

[1] E.J. Haug, K.K. Choi, V. Komkov: *Design sensitivity analysis of structural systems*, Academic Press, Orlando, 1986.

[2] I. Hlaváček, M. Křížek: *On a superconvergent finite element scheme for elliptic systems*, Parts I–III. Appl. Math. **32** (1987), 131–154, 200–213, 276–289.

[3] I. Hlaváček, M. Křížek, V. Pištora: *How to recover the gradient of linear elements on nonuniform triangulations*, Appl. Math. **41** (1996), 241–267.

[4] I. Hlaváček, M. Křížek: *Optimal error and local error estimates of a recovered gradient of linear elements on nonuniform triangulations*, J. Comput. Math. **14** (1996), 345–362.

[5] M. Křížek, P. Neittaanmäki: *Superconvergence phenomenon in the finite element method arising from averaging gradients*, Numer. Math. **45** (1984), 105–116.

[6] M. Křížek, P. Neittaanmäki: *On a global superconvergence of the gradient of linear triangular elements*, J. Comput. Appl. Math. **18** (1987), 221–233.

[7] N. Levine: *Superconvergent recovery of the gradient from piecewise linear finite-element approximations*, IMA J. Numer. Anal. **5** (1985), 407–427.

[8] O. C. Zienkiewicz, J.Z. Zhu: *A simple error estimator and adaptive procedure for practical engineering analysis*, Int. J. Num. Meth. Eng. **24** (1987), 337–357.

INFLUENCE OF LINEARIZATION TO THE SOLUTION OF FISHER'S EQUATION IN A PLANE

Pavol Chocholatý

Abstract

Reaction-diffusion equations arise as mathematical models in a series of important applications. Some difference schemes to the solution of the Fisher's equation are presented.

1. Fisher's equation

We start our discussion of reaction-diffusion equations by considering a model arising in mathematical ecology. In order to understand the foundation of this model, we first recapture the model of population growth. This model states that the growth of a population facing limited resources is governed by an ordinary differential equation (ODE)

$$u'(t) = cu(t)(A - u(t))$$

for the population density. It is assumed that spatial variation in the density of the population is of little importance for the growth of the population. Thus, one simply assumes that the population is evenly distributed over some area $G \subset R^d$ for all time $t, c > 0$ is the growth rate, and $A > 0$ is the so-called carrying capacity of the environment. For real populations, this assumption is often quite dubious. In the next level of sophistication, it is common to take into account the tendency of a population to spread out over the area G where it is possible to live. This effect is incorporated by adding a Fickian diffusion term to the model.

Now, let $u(x, t)$ be a function of the population density in time t and a point x of an area $G \subset R^d$ with the boundary S . Then we get the following partial differential equation

$$u_t = \nabla(D\nabla u) + q. \tag{1}$$

Here, D is a diffusion coefficient. Equation (1) is a linear diffusion equation or the heat equation with nonhomogeneous forcing term, where q is assumed to be a continuous function in x and t. Assume now that the function q depends also on the population density u, i.e. $q = f(x, t, u)$, then equation (1) in the form

$$u_t - \nabla(D\nabla u) = f(x, t, u) \tag{2}$$

is a nonlinear reaction-diffusion equation.

Consider next a situation

$$f(x, t, u) = cu(A - u). \tag{3}$$

In mathematical ecology, model of population growth

$$u_t = \nabla(D\nabla u) + cu(A - u) \tag{4}$$

is called Fisher's equation. We mentioned that the first right-hand side term models the diffusion of the population. Similar terms arise in a lot of applications, where we want to capture the tendency of nature to smooth things out. For instance, if you drop a tiny amount of ink into a glass of water, you can watch how the ink spreads throughout the water by means of molecular diffusion. This situation is modeled by the diffusion equation, where Fick's law is used to state that there is a flux of ink from areas of high concentration to areas of low concentration. Similarly again, a Fickian diffusion term in a model of population density states that there is a migration from areas of high population density to areas of low population density. Human beings do not always obey this sound principle.

Fisher's equation (4) is usually studied in conjunction with Neumann-type boundary condition, i.e.

$$(\nabla u)^T \cdot \nu = 0 \quad \text{on} \quad S \times [0, T], \tag{5}$$

where ν is meant to be the out-side normal to S.

The reason for this boundary condition is that we assume the area G to be closed, so there is no migration from the domain. Assume that $g = g(x)$, $0 \leq g(x) \leq A$ denotes the initial distribution of the population, we have the initial condition

$$u(x, 0) = g(x), \quad x \in G. \tag{6}$$

Now, (4), (5), (6) represent the initial-boundary value problem for Fisher's equation.

Since we are interested in the qualitative behaviour of this model rather than the actual quantities, we simplify the situation by putting $D = c = A = 1$ and study the following problem

$$u_t = \Delta u + u(1 - u), \quad \frac{\partial u}{\partial \nu}\Big|_S = 0, \quad t \in [0, T], \quad u(x, 0) = g(x), \quad x \in G. \tag{7}$$

2. Finite difference schemes for Fisher's equation

First, we want to solve the initial-boundary value problem (7) and Cauchy problem by difference approximation. Let us start with Cauchy problem: We introduce a time-step $l = T/m$, $t_i = il$, $i = 0, 1, \ldots, m$ and denote by $P(x, t, u)$ the right-hand side of differential equation in (7). The $u_{\cdot,i}$ are meant to be approximations to $u(x, t_i)$, then an explicit finite difference scheme can be written as follows:

$$u_{\cdot,i+1} = u_{\cdot,i} + lP(x_\cdot, t_i, u_{\cdot,i}), \quad u_{\cdot,0} = g(x_\cdot), \tag{8}$$

which represents explicit Euler method for Cauchy problem by ODEs.

In order to define further integration procedures, we have to give a rule for determining $u_{\cdot,i+1}$ when $u_{\cdot,i}, u_{\cdot,i-1}$ and the differential equation are given. Such rule is based on Taylor's expansion and constitutes an explicit two-step formula for $u_{\cdot,i+1}$:

$$\begin{aligned} u_{\cdot,i+1} &= \frac{7}{4}u_{\cdot,i} - \frac{3}{4}u_{\cdot,i-1} + \frac{l}{4}P(x_{\cdot}, t_i, u_{\cdot,i}), && (9) \\ u_{\cdot,i+1} &= \frac{4}{3}u_{\cdot,i} - \frac{1}{3}u_{\cdot,i-1} + \frac{2l}{3}P(x_{\cdot}, t_i, u_{\cdot,i}), && (10) \\ u_{\cdot,i+1} &= u_{\cdot,i-1} + 2lP(x_{\cdot}, t_i, u_{\cdot,i}). && (11) \end{aligned}$$

We are now going to write down the approximation for the initial-boundary value problem:

A. For $d = 1$ the problem (7) can be written under the form

$$\begin{aligned} &u_t = u_{xx} + u(1-u), && x \in (0,L), \quad t \in (0,T), \\ &u_x(0,t) = u_x(L,t) = 0, && t \in [0,T], \\ &u(x,0) = g(x), && x \in [0,L], \end{aligned} \tag{12}$$

and a mesh width $h = L/n$, $x_j = jh$, $j = 0,1,2,\ldots,n$, n natural number, and divide the interval $[0,L]$ into subintervals of length h. For simplicity, put $L = 1$ and we have on $[0,1]$ $(n-1)$ inner grids $x_1, x_2, \ldots, x_{n-1}$, and two boundary grids $x_0 = 0$, $x_n = 0$. As usual, we assume that the solution u of (12) can be continued on the left side of boundary grid x_0 and on the right side of x_n, we use the approximation by symmetric differences to describe Neumann boundary conditions.

So, $P(x, t_i, u_j)$ can be written by using symmetric differences under the form

$$h^2P(x_j, t_i, u_{j,i}) = u_{j+1,i} + (h^2 - 2)u_{j,i} + u_{j-1,i} - h^2u_{j,i}^2, \quad j = 0,1,\ldots,n. \tag{13}$$

Denoting by u_i a vector function which is defined for all $t_i = il$, $i = 0,1,2,\ldots,m$, we approximate the Cauchy problem (8) by

$$\begin{aligned} u_{j,i+1} &= ru_{j+1,i} + (1 + rh^2 - 2r)u_{j,i} + ru_{j-1,i} - rh^2u_{j,i}^2, \\ u_{j,0} &= g(x_j), \quad u_{-1,i} = u_{1,i}, \quad u_{n-1,i} = u_{n+1,i}, \end{aligned} \tag{14}$$

where, as usual, $r = l/h^2$. Tveito and Winther [2] have presented the stability condition

$$l < \frac{h^2}{2 + h^2}, \tag{15}$$

this property provided that the mesh parameters satisfy the requirement, which is slightly more restrictive than the corresponding condition for the heat equation, $l \leq h^2/2$ or $r \leq 1/2$. In a paper [1] we have utilized the preceding result to construct a new stability condition

$$l < \frac{h^2}{2 + h^2/2}. \tag{16}$$

In order to verify this stability condition we have solved problem (12), $g(x) = \cos^2 \pi x$ for some given h and computed l such that

(C1) $$l < \frac{h^2}{2+h^2},$$

(C2) $$\frac{h^2}{2+h^2} \leq l < \frac{h^2}{2+h^2/2},$$

(C3) $$l \geq \frac{h^2}{2+h^2/2}.$$

In numerical experiments we have observed that the approximate solutions in cases (C1), (C2) always stayed within the unit interval and that they approached the state $u = 1$ as time increased, but in case (C3) the oscillatory solution about the state $u = 1$ as time increased was obtained.

Now, we approximate the Cauchy problem by explicit two-step formulas (9), (10), (11) with the operator (13)

$$4u_{j,i+1} = ru_{j-1,i} + (7 - 2r + rh^2)u_{j,i} + ru_{j+1,i} - 3u_{j,i-1} - rh^2u^2_{j,i}, \tag{17}$$

$$3u_{j,i+1} = 2ru_{j-1,i} + (4 - 4r + 2rh^2)u_{j,i} + 2ru_{j+1,i} - u_{j,i-1} - 2rh^2u^2_{j,i}, \tag{18}$$

$$u_{j,i+1} = 2ru_{j-1,i} + (2rh^2 - 4r)u_{j,i} + 2ru_{j+1,i} + u_{j,i-1} - 2rh^2u^2_{j,i}, \tag{19}$$

$$l = rh^2, \quad i = 0, 1, 2, \dots, m, \quad j = 0, 1, 2, \dots, n,$$

and described conditions

$$u_{j,0} = g(x_j), \quad u_{-1,i} = u_{1,i}, \quad u_{n-1,i} = u_{n+1,i}.$$

We now consider the question of an actual error at the discrete time points $t_i = il$, $i = 0, 1, 2, \dots, m$. The "good" dependence of actual error on $(i + p)$-th step to an error on i-th step is a motivation for the computational method used.

The following three tables present for $l = 0.0049875$ and $h = 0.1$ the propagation of the unit error at one vertex on time steps $i + p$ for (17), (18), (19). Notice that if the error is not unite, then the presented values should be scaled by its actual size ε.

(17)	x_0	x_1	x_2	x_3	x_4	x_5	x_6
i	0	0	0	1	0	0	0
i+1	0	0	0.125	1.5	0.125	0	0
i+2	0	0.016	0.374	1.532	0.374	0.016	0
i+3	0.002	0.070	0.662	1.266	0.662	0.070	0.002

(18)	x_0	x_1	x_2	x_3	x_4	x_5	x_6
i	0	0	0	1	0	0	0
i+1	0	0	0.333	0.668	0.333	0	0
i+2	0	0.111	0.445	0.335	0.445	0.111	0
i+3	0.037	0.222	0.337	0.299	0.337	0.222	0.037

(19)	x_0	x_1	x_2	x_3	x_4	x_5	x_6
i	0	0	0	1	0	0	0
i+1	0	0	0.998	-1.95	0.998	0	0
i+2	0	0.995	-3.98	6.911	-3.98	0.995	0
i+3	0.993	-5.95	16.62	-24.1	16.62	-5.95	0.993

So, it is shown that with formulas (17), (18) a good approximative solution for (12) can be computed and that the formula (19) leads to catastrophical spreading of actual error.

B. The computational technique outlined in the previous part can be applied formally to the problem (7) also in case $d = 2$ with domain $G = (0,1) \times (0,1)$, $\overline{G} = G + S$. In this chapter, we concentrate on the solution of Cauchy problem using implicit methods

$$u_{.,i+1} = u_{.,i} + lP(x_., t_{i+1}, u_{.,i+1}), \quad u_{.,0} = g(x_.).$$

For a grid function to $u(x, y, t)$, defined on the set of grid points $\overline{G}_h = \{(x_j, y_k); h = 1/n; 0 \leq j, k \leq n\}$ at a given time $t_i = il \in [0, T]$, $i = 0, 1, 2, \ldots, m$, $l = T/m$, we frequently write $u_{j,k,i}$. For such functions the finite difference operator approximating the differential operator $P(x, t, u)$ is now defined by

$$h^2 P(x_j, y_k, t_i, u_{j,k,i}) = u_{j+1,k,i} + u_{j-1,k,i} + u_{j,k-1,i} + u_{j,k+1,i} + (h^2 - 4)u_{j,k,i} + h^2 u^2_{j,k,i} \quad (20)$$

for all interior grid points (x_j, y_k, t_i), $j, k = 1, 2, \ldots, n-1$. For a Neumann condition defined on S, we frequently use forwards and backwards differences

$$u_{1,k,i} = u_{0,k,i}, \quad u_{j,1,i} = u_{j,0,i}, \quad u_{n,k,i} = u_{n-1,k,i}, \quad u_{j,n-1,i} = u_{j,n,i}, \quad j, k = 0, 1, \ldots, n. \quad (21)$$

A finite difference approximation is now started by Cauchy condition

$$u_{j,k,0} = g(x_j, y_k) \quad (22)$$

and is represented on each t_i, $i = 1, 2, \ldots, m$ by a system of nonlinear equations. But our computational approach is based on a sequential solution procedure, where the operator splitting is the key to obtain higher efficiency. This is the case for the alternating direction implicit (ADI) scheme first proposed for the implicit solution of heat flow in two geometric dimensions by Peaceman and Rachford. They have proved both stability and convergence for the method. The computational setup can be briefly described using linearization as follows: Consider the first finite-difference scheme in the linearization form

$$\begin{aligned} u_{j,k,z} - u_{j,k,i} &= r(u_{j-1,k,z} - 2u_{j,k,z} + u_{j+1,k,z}) + r(u_{j,k-1,i} - 2u_{j,k,i} + u_{j,k+1,i}) + \\ &\quad + rh^2 u_{j,k,z}(1 - u_{j,k,i}) \end{aligned} \quad (23)$$

and now formulate two approaches of linearization of the second finite-difference scheme

$$\begin{aligned} u_{j,k,i+1} - u_{j,k,z} &= r(u_{j,k-1,i+1} - 2u_{j,k,i+1} + u_{j,k+1,i+1}) + r(u_{j-1,k,z} - 2u_{j,k,z} + u_{j+1,k,z}) \\ &\quad + rh^2 u_{j,k,z}(1 - u_{j,k,i+1}), \qquad (24) \\ u_{j,k,i+1} - u_{j,k,z} &= r(u_{j,k-1,i+1} - 2u_{j,k,i+1} + u_{j,k+1,i+1}) + r(u_{j-1,k,z} - 2u_{j,k,z} + u_{j+1,k,z}) \\ &\quad + rh^2 u_{j,k,i+1}(1 - u_{j,k,z}) \qquad (25) \end{aligned}$$

with both Cauchy condition (22) and boundary condition (21), where $r = l/(2h^2)$ and $z = i + 1/2$, $i = 0, 1, \dots, m-1$.

A computational comparison of the above-mentioned approaches (23), (24) or (23), (25) for Fisher's equation with Cauchy condition $g(x, y) = \cos^2(\pi(x + y))/2$ and steps $h = 0.1$, $l = 0.01$ introduces on some $t_i = il$ at the point $(0.7, 0.5) \in G$ the following table:

t	0.01	0.1	1
(23),(24)	0.161929944	0.289809450	0.712927005
(23),(25)	0.162252848	0.290045491	0.714969435

References

[1] P. Chocholatý: *A numerical method to the solution of nonlinear reaction-diffusion equation.* In: M. Kováčová (Ed.), Proceedings of the 4th International Conference Aplimat 2005, Bratislava, Slovak University of Technology, 2005, pp. 63–66.

[2] A. Tveito and R. Winther: *Introduction to partial differential equations.* Springer-Verlag, 1998.

RESONANCE BEHAVIOUR OF THE SPHERICAL PENDULUM DAMPER*

Cyril Fischer, Jiří Náprstek

Abstract

The pendulum damper modelled as a two degree of freedom strongly non-linear auto-parametric system is investigated using two approximate differential systems. Uni-directional harmonic external excitation at the suspension point is considered. Semi-trivial solutions and their stability are analyzed. The thorough analysis of the non-linear system using less simplification than it is used in the paper [2] is performed. Both approaches are compared and conclusions are drawn.

1. Introduction

Many structures encountered in the civil and mechanical engineering are equipped with various devices for reducing dynamic response component due to external excitations. Among other low cost passive systems the pendulum dampers are still very popular for their reliability and simple maintenance, see e.g. [1]. However the dynamic behaviour of such a pendulum is significantly more complex than it is supposed by a widely used simple linear SDOF model working in the (xz) vertical plane only, see Figure 1. The conventional linear model is satisfactory only if the kinematic excitation $a(t)$ introduced at the suspension point is very small in amplitude and if its frequency remains outside a resonance frequency domain.

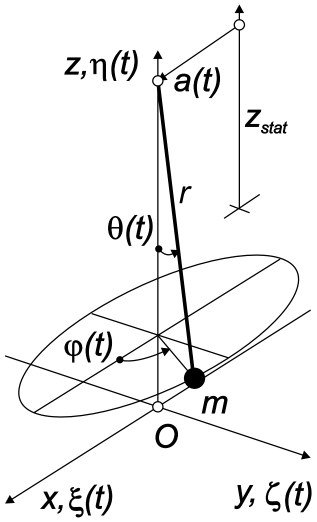


Fig. 1: *Sketch of the pendulum and coordinate systems used.*

2. Mechanical energy balance

Let us consider the kinematic excitation $a(t)$ at the suspension point in the x direction only. The natural choice of the coordinate system suitable for description of the movement of the pendulum would be the spherical coordinate system described by the angles θ (in the xz plane), φ (diversion from the xz plane) and radius $r = const$ (see Fig. 1). However, such a choice does not allow to consider the angle φ as a perturbation of the pure planar motion described by θ, r only. Indeed, even for a small transversal motion (in the y direction), the full range $\varphi \in \langle 0, 2\pi)$ occurs. Thus the mechanical energy balance has to be written in the Cartesian coordinates

[*] The support of the Grant Agency of the ASCR Nos. A2071401, IAA200710805 and AVOZ 20710524 research plan are acknowledged.

$(\xi(t) = \xi, \zeta(t) = \zeta, \eta(t) = \eta)$. The kinetic and potential energies T, V are described by:

$$\begin{aligned} T &= m(\dot{\xi}^2 + \dot{\zeta}^2 + \dot{\eta}^2 + 2\dot{a}\dot{\xi} + \dot{a}^2)/2, & (1)\\ V &= mg\eta & (2) \end{aligned}$$

and the geometric constraint of the suspension is expressed as:

$$\xi^2 + \zeta^2 + (1-\eta)^2 = r^2, \tag{3}$$

where m, r - mass and suspension length of the pendulum

$a = a(t)$ - kinematic excitation at the suspension point

From the relation between the spherical and Cartesian coordinates and the geometric constraint (3) it follows:

$$\eta = r(1-\cos\theta)\ ;\ \ \dot{\eta}^2 = r^2\dot{\theta}^2\sin^2\theta\ ;\ \ \sin\theta = \frac{\varrho}{r}, \quad \text{where} \quad \varrho^2 = \xi^2 + \zeta^2. \tag{4}$$

A hypothesis that the amplitude of $\theta(t)$ is small makes acceptable an approximation:

$$\theta = \arcsin\frac{\varrho}{r} \approx \frac{\varrho}{r} + \frac{1}{6}\frac{\varrho^3}{r^3} \quad \Rightarrow \quad \dot{\theta}^2 = \frac{\dot{\varrho}^2}{r^2}\left(1 + \frac{\varrho^2}{2r^2}\right)^2. \tag{5}$$

The equations of the motion follows from the Lagrangian principle:

$$\partial_t(\partial_{\dot{\chi}}T) - \partial_\chi T + \partial_\chi V = 0\ , \qquad \text{for } \chi \in \{\xi, \varphi\}. \tag{6}$$

Using (1), (2), (4), (5) and (6) an approximate Lagrangian system in the x, y coordinates for the components ξ, ζ on the level $O(\varepsilon^6)$; $\varepsilon^2 = (\xi^2+\zeta^2)/r^2$ can be obtained. The approximate linear damping with the relative scale ω_b equivalent in both components ξ, ζ will be included, giving the differential system:

$$\begin{aligned} \ddot{\xi} + 2\omega_b\dot{\xi} + \xi\left(1 + \frac{\xi^2+\zeta^2}{2r^2}\right)\left(\omega_0^2 + \frac{((\xi^2+\zeta^2)^{\bullet})^2}{4r^4} + \frac{\left(1+\frac{\xi^2+\zeta^2}{2r^2}\right)(\xi^2+\zeta^2)^{\bullet\bullet}}{2r^2}\right) &= -\ddot{a},\\ \ddot{\zeta} + 2\omega_b\dot{\zeta} + \zeta\left(1 + \frac{\xi^2+\zeta^2}{2r^2}\right)\left(\omega_0^2 + \frac{((\xi^2+\zeta^2)^{\bullet})^2}{4r^4} + \frac{\left(1+\frac{\xi^2+\zeta^2}{2r^2}\right)(\xi^2+\zeta^2)^{\bullet\bullet}}{2r^2}\right) &= 0, \end{aligned} \tag{7}$$

where $\omega_0^2 = g/r$. Taking into account the additional simplification,

$$\left(1 + \frac{(\xi^2+\zeta^2)}{2r^2}\right)^2 \approx 1, \ \ \frac{\chi}{r^4}\left(1 + \frac{(\xi^2+\zeta^2)}{2r^2}\right)\left((\xi^2+\zeta^2)^{\bullet}\right)^2 \approx 0 \ \text{ for } \chi \in \{\xi, \zeta\}, \tag{8}$$

the simplified form of the differential system can be obtained (see [2]):

$$\begin{aligned} \ddot{\xi} + \frac{1}{2r^2}\xi(\xi^2+\zeta^2)^{\bullet\bullet} + 2\omega_b\dot{\xi} + \omega_0^2(\xi + \frac{1}{2r^2}\xi(\xi^2+\zeta^2)) &= -\ddot{a},\\ \ddot{\zeta} + \frac{1}{2r^2}\zeta(\xi^2+\zeta^2)^{\bullet\bullet} + 2\omega_b\dot{\zeta} + \omega_0^2(\zeta + \frac{1}{2r^2}\zeta(\xi^2+\zeta^2)) &= 0. \end{aligned} \tag{9}$$

In both simplified and complete systems, neglecting the non-linear terms will result in two independent equations. Each of the components ξ, ζ can be considered as arbitrarily small and independently and continuously limited to zero. Therefore the system is auto-parametric and respective procedures can be applied [4].

3. Semi-trivial solution

To investigate the semi-trivial solution let us substitute $\zeta = 0$ into Eqs (7), (9) and specify the excitation to be harmonic (see [3] for details): $a(t) = a_0 \sin \omega t$.

The semi-trivial solution of Eqs (7) or (9) should be searched in the form:

$$\xi_0 = a_c \cos \omega t + a_s \sin \omega t \ ; \quad \zeta_0 = 0. \tag{10}$$

The coefficients a_c, a_s in general should be considered as functions of time: $a_c = a_c(t)$, $a_s = a_s(t)$. If a stationary solution exists for a given excitation frequency ω, then a_c, a_s should converge to constants for increasing $t \to \infty$. In such a case the coefficients a_c, a_s can be considered constant. Let us substitute (10) into Eq. (7) and (9), multiply them by $\sin(\omega t)$ or $\cos(\omega t)$ and integrate the resulting expressions over the interval $t \in (0, 2\pi/\omega)$. The described operation (so called *harmonic balance* operation) results for each of the equations (7) or (9) in an algebraic system consisting of two equations. For the simplified case of Eq. (9) it is:

$$\begin{aligned} a_c \left((\omega_0^2 - \omega^2) + \frac{1}{2r^2} \left(\frac{3}{4}\omega_0^2 - \omega^2 \right) (a_c^2 + a_s^2) \right) + 2\omega\omega_b \cdot a_s &= 0, \\ a_s \left((\omega_0^2 - \omega^2) + \frac{1}{2r^2} \left(\frac{3}{4}\omega_0^2 - \omega^2 \right) (a_c^2 + a_s^2) \right) - 2\omega\omega_b \cdot a_c &= a_0 \cdot \omega^2. \end{aligned} \tag{11}$$

If both equations are raised to the second power and summed together, then, finally, the equation for the amplitude of the response arises ($R_0^2 = a_c^2 + a_s^2$):

$$R_0^2 \left[4\omega^2\omega_b^2 + \left((\omega^2 - \omega_0^2) + \frac{R_0^2}{2r^2} \left(\omega^2 - \frac{3}{4}\omega_0^2 \right) \right)^2 \right] - 4\omega^4 a_0^2 = 0. \tag{12}$$

Applying the same procedure to the original system (7), one can get a similar equation for the amplitude:

$$R_0^2 \left[4\omega^2\omega_b^2 + \left((\omega^2 - \omega_0^2) + \frac{R_0^2}{2r^2} \left(\omega^2 - \frac{3}{4}\omega_0^2 \right) + \omega^2 \frac{R_0^4}{8r^4} \left(3 + \frac{5R_0^2}{8r^2} \right) \right)^2 \right] - \omega^4 a_0^2 = 0. \tag{13}$$

The Eqs (12) and (13) are known as *resonance curves.* They express the dependence of the amplitude R_0^2 of the solution (response) on the excitation frequency. Both curves are demonstrated in Figure 2. Depending on the parameters a_0, ω_b and ω, this relations can lose their unique character in some intervals of ω.

4. Perturbation of the semi-trivial solution

To assess the stability of the semi-trivial solution we will endow the semi-trivial solution (10) with small (in the meaning of a norm) perturbations u, v in both coordinates:

$$\begin{aligned} \xi &= \xi_0 + u, \qquad & u = u(t) = u_c \cos \omega t + u_s \sin \omega t. \\ \zeta &= 0 + v, \qquad & v = v(t) = v_c \cos \omega t + v_s \sin \omega t. \end{aligned} \tag{14}$$

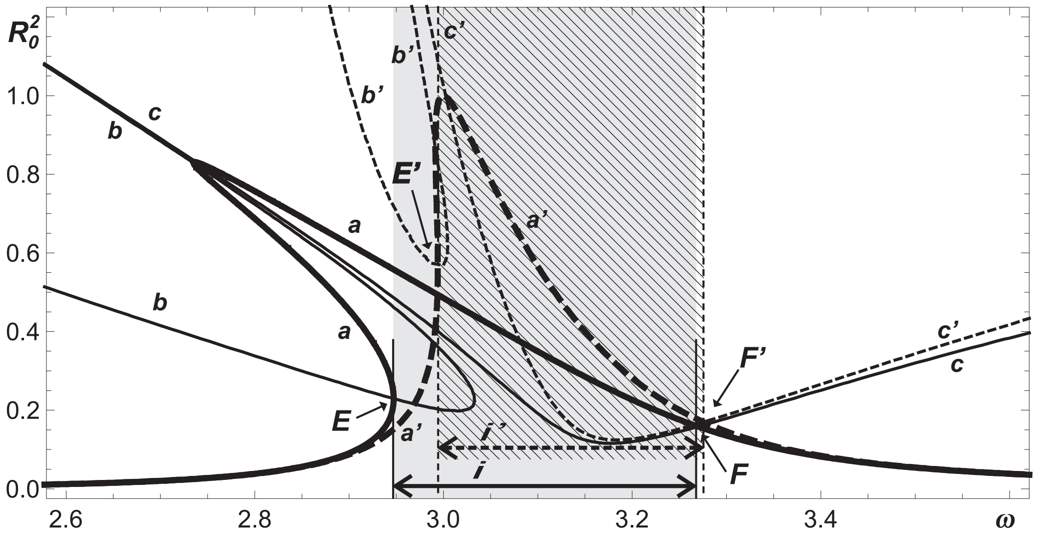


Fig. 2: *Resonance curves (thick lines a, a') and stability limits (thin lines b, b', c, c') of the semi-trivial solution computed using the original (solid lines a, b, c) and simplified (dashed lines a', b', c') equations.*
Curves (b, b'): in (xz) plane – ξ stability limit, Eqs (16), (18).
Curves (c, c'): out of (xz) plane – ζ stability limit, Eqs (17), (19).
Interval **i** *corresponds to the non-stability interval of the original formulas (16–17).*
Interval **i'** *corresponds to the non-stability interval of the simplified formulas (18–19).*
Values used: $r = 1$, $g = 9.81$, $\omega_b = 0.075$, $a_0 = 0.05$.

As the perturbations are expected to be small, only the first powers of u, v and their derivatives are kept after inserting expressions (14) into Eqs (7) and (9). After the harmonic balance operation and some algebra one obtains two linear algebraic systems for u_c, u_s and v_c, v_s. For the simplified case (9) it reads:

$$\begin{pmatrix} w_1 & w_2 \\ w_3 & w_1 \end{pmatrix} \begin{pmatrix} u_c \\ u_s \end{pmatrix} = 0 \ ; \quad \begin{pmatrix} z_1 & z_2 \\ z_3 & z_1 \end{pmatrix} \begin{pmatrix} v_c \\ v_s \end{pmatrix} = 0 \ ; \tag{15}$$

where it has been denoted:

$$w_1 = \left[2\omega\omega_b + \tfrac{1}{4r^2}\Omega_1 a_c a_s\right] ; \ w_2 = \left[2\omega\omega_b + \tfrac{1}{4r^2}\Omega_1 a_c a_s\right] ; \ w_3 = \left[\tfrac{1}{4r^2}\Omega_1 a_c a_s - 2\omega\omega_b\right]$$
$$z_1 = \left[\Omega_2 + \tfrac{1}{8r^2}(\Omega_1 a_c^2 + \Omega_3 a_s^2)\right] ; \ z_2 = \left[\tfrac{1}{4r^2}\Omega_4 a_c a_s + 2\omega\omega_b\right] ; \ z_3 = \left[\tfrac{1}{4r^2}\Omega_4 a_c a_s - 2\omega\omega_b\right]$$
$$\Omega_1 = 3\omega_0^2 - 4\omega^2; \ \Omega_2 = \omega_0^2 - \omega^2; \ \Omega_3 = \omega_0^2 + 4\omega^2; \ \Omega_4 = \omega_0^2 - 4\omega^2.$$

The both systems (15) are homogeneous and independent of excitation amplitude. Consequently to receive a non-trivial solution for u_c, u_s or v_c, v_s, the determinant of the systems (15) must equal zero. This rationale leads to two independent equations:

$$\frac{1}{2r^2}\Omega_1 R_0^2 \left(\Omega_2 + \frac{3}{32r^2}\Omega_1 R_0^2\right) + \Omega_2^2 + 4\omega^2\omega_b^2 = 0, \tag{16}$$

$$\frac{1}{2r^2} R_0^2 \left(\omega_0\Omega_2 + \frac{1}{32r^2}\Omega_1\Omega_3 R_0^2\right) + \Omega_2^2 + 4\omega^2\omega_b^2 = 0. \tag{17}$$

Similar equations can also be formulated for the original system (7)

$$\begin{aligned}\frac{175\omega^4 R_0^{12}}{4096 r^{12}}+\frac{45\omega^4 R_0^{10}}{128 r^{10}}+\frac{5\omega^2\left(56\omega^2-15\omega_0^2\right)R_0^8}{256 r^8}+\frac{\omega^2\left(17\omega^2-14\omega_0^2\right)R_0^6}{8r^6}+\\ +\frac{3\left(-3\omega^2\Omega_2+(\frac{1}{4}\Omega_1)^2\right)R_0^4}{4r^4}+\frac{\Omega_2\Omega_1 R_0^2}{2r^2}+\Omega_2^2+4\omega^2\omega_b^2=0,\end{aligned}\tag{18}$$

$$\begin{aligned}-\frac{5\omega^4 R_0^{12}}{4096 r^{12}}-\frac{\omega^4 R_0^{10}}{64 r^{10}}-\frac{\omega^2\left(24\omega^2+\omega_0^2\right)R_0^8}{256 r^8}-\frac{\omega^2\left(3\omega^2+\omega_0^2\right)R_0^6}{16 r^6}+\\ +\frac{\omega_0^2\left(3\omega_0^2-8\omega^2\right)R_0^4}{64 r^4}+\frac{\omega_0^2\Omega_2 R_0^2}{2r^2}+\Omega_2^2+4\omega^2\omega_b^2=0.\end{aligned}\tag{19}$$

Eqs (16)–(17) and (18)–(19) can be interpreted as limits dividing the plane (R_0^2,ω) into the stable and unstable domains. For given parameters r,ω_b,a_0 the unstable interval of excitation frequency is defined by the position of the intersections of the resonance curve with the corresponding stability limits (points **E**, **F** in Figure 2).

5. Post-critical response in the resonance domain

Let us try to assume a more general expressions as the basic solution:

$$\xi(t)=a_c(t)\cos\omega t+a_s(t)\sin\omega t\ ;\qquad \zeta(t)=b_c(t)\cos\omega t+b_s(t)\sin\omega t.\tag{20}$$

Increasing the number of unknown functions to four, one can exploit a possibility to formulate two arbitrarily selectable additional conditions. Then the following expressions for the first derivatives of the general solution (20) can be stated:

$$\dot\xi(t)=-a_c\omega\sin\omega t+a_s\omega\cos\omega t\ ;\qquad \dot\zeta(t)=-b_c\omega\sin\omega t+b_s\omega\cos\omega t,\tag{21}$$

where $a_c=a_c(t),a_s=a_s(t),b_c=b_c(t),b_s=b_s(t)$. Let us insert expressions (20), (21) in the simplified differential system (9) and apply the operation of the harmonic balance once again. After dull routine work one obtain the differential system for amplitudes a_c,a_s,b_c,b_s, whose system matrix $\mathbf{A}$ depends only on a_c,a_s,b_c,b_s,ω:

$$\mathbf{A}\begin{bmatrix}\dot a_c\\ \dot a_s\\ \dot b_c\\ \dot b_s\end{bmatrix}=-\frac{1}{2}\begin{bmatrix}a_c(8\Omega_2r^2+R_A^2\Omega_1)+2b_sS_A^2\Omega_4+4a_s\omega_b\omega r^2\\ a_s(8\Omega_2r^2+R_A^2\Omega_1)+2b_cS_A^2\Omega_4+4a_c\omega_b\omega r^2\ +8\omega^2a_0r^2\\ b_c(8\Omega_2r^2+R_A^2\Omega_1)+2a_sS_A^2\Omega_4+4b_s\omega_b\omega r^2\\ b_s(8\Omega_2r^2+R_A^2\Omega_1)+2a_cS_A^2\Omega_4+4b_c\omega_b\omega r^2\end{bmatrix},\tag{22}$$

where it has been denoted

$$R_A^2=a_c^2+a_s^2+b_c^2+b_s^2\ ;\qquad S_A^2=a_sb_c-a_cb_s.\tag{23}$$

The explicit solution of Eqs (22) is generally not possible in the resonance interval. However, from the numerical analysis can be seen that at least part of the resonance interval can be described by a steady state solution. The other part of the resonance interval, where the transient solution takes place, will not be discussed here.

The steady state response is characterized by constant amplitudes (for $t \to \infty$). This means that the time derivatives $\dot{a}_c, \dot{a}_s, \dot{b}_c, \dot{b}_s$ vanish for large t. The left-hand side of Eq. (22) vanishes and Eq. (22) reduces itself into the algebraic system. After tedious work, the relation between R_A^2, S_A^2 and ω can be deduced from Eq. (22), where the left hand side was substituted by the zero vector:

$$\begin{aligned} R_A^2\left((8\Omega_2 r^2 + R_A^2\Omega_1)^2 + 4\left(4\omega^2\omega_b^2 r^4 + S_A^4\Omega_4^2\right)\right) - 8S_A^4\left(8\Omega_2 r^2 + R_A^2\Omega_1\right)\Omega_4 &= 64 r^4 a_0^2\omega^4, \\ S_A^2\left(2R_A^2\left(8\Omega_2 r^2 + R_A^2\Omega_1\right)\Omega_4 - (8\Omega_2 r^2 + R_A^2\Omega_1)^2 - 16\omega^2\omega_b^2 r^4 - 4S_A^4\Omega_4^2\right) &= 0. \end{aligned} \tag{24}$$

Parameter R_A^2 can be interpreted as a generalized total or effective amplitude including both components (20). As regards the S_A^2, it represents a certain characteristics of their phase shift. If $S_A^2 = 0$ the vectors $[a_c, a_s]$, $[b_c, b_s]$ are co-linear. It represents the motion in the vertical plane. Indeed, putting $S_A^2 = 0$ into the first equation of (24) one obtains the formula for the semi-trivial resonance curve (12). The case $S_A^2 \neq 0$ implies motion out of the vertical plane. For this case, an analysis of the system (24) was carried out, but it is beyond the scope of this contribution.

Using the procedure described above, a similar relation was also derived for the original system (7). The resulting formulas are rather complicated which fact makes an analysis of the individual components hardly feasible. On the other hand, it brings no new qualitative results comparing to the simplified version (24).

6. Conclusion

Analytical and numerical investigations have shown that the widely used linear model of the damping pendulum is acceptable only in a very limited extent of parameters concerning pendulum characteristics and excitation properties. In the case of a harmonic kinematic external excitation at the suspension point, it is necessary to thoroughly investigate the dynamic stability limits and post-critical behaviour. To investigate the stability of the semi-trivial solution, it is necessary to use the approximate equations in the Cartesian coordinates. Using the harmonic balance method the resonance curves of a planar stationary response as well as the stability limits of the semi-trivial solution in both response components have been determined. Omitting the simplification (8) results in very complicated formulas and brings only quantitative specification.

References

[1] R.S. Haxton, A.D.S. Barr: *The auto-parametric vibration absorber*, ASME Jour. Applied Mechanics **94** (1974), 119–125.

[2] J. Náprstek, C. Fischer: *Auto-parametric post-critical behaviour of a spherical pendulum damper*. In: B. Topping (Ed.) Proc. 11th Conference on Civil, Struct. and Env. Eng. Computing, Malta, Civil-Comp Press, 2007.

[3] A. Tondl: *Quenching of self-excited vibrations*, Academia, Prague, 1991.

[4] A. Tondl, T. Ruijgrok, F. Verhulst, R. Nabergoj: *Auto-parametric resonance in mechanical systems*, Cambridge University Press, Cambridge, 2000.

THREE-DIMENSIONAL NUMERICAL MODEL OF NEUTRON FLUX IN HEX-Z GEOMETRY[*]

Milan Hanuš, Tomáš Berka, Marek Brandner, Roman Kužel, Aleš Matas

Abstract

We present a method for solving the equations of neutron transport with discretized energetic dependence and angular dependence approximated by the diffusion theory. We are interested in the stationary solution that characterizes neutron fluxes within the nuclear reactor core in an equilibrium state. We work with the VVER-1000 type core with hexagonal fuel assembly lattice and use a nodal method for numerical solution. The method effectively combines a whole-core coarse mesh calculation with a more detailed computation of fluxes based on the transverse integrated diffusion equations. By this approach, it achieves a good balance between accuracy and speed.

1. Multigroup diffusion theory

The set of steady-state neutron diffusion equations for G energy groups (the discrete ranges of neutron energies) can be written as follows:

$$\nabla \cdot \mathbf{j}^g(\mathbf{r}) + \Sigma_r^g(\mathbf{r})\phi^g(\mathbf{r}) = \sum_{\substack{g'=1\\ g'\neq g}}^{G} \Sigma_s^{g'\to g}(\mathbf{r})\phi^{g'}(\mathbf{r}) + \frac{\chi^g}{k_{\text{eff}}} \sum_{g'=1}^{G} \nu\Sigma_f^{g'}(\mathbf{r})\phi^{g'}(\mathbf{r}). \tag{1}$$

According to the usual notation,

$\mathbf{r} = (x, y, z)$ is the spatial variable,
$g = 1, 2, \ldots, G$ denotes the energy group,
ϕ^g is the neutron flux in group g (density of neutrons),
$\mathbf{j}^g$ is the neutron current in group g (flow of neutrons in specific direction),
Σ_r^g is the macroscopic removal cross section (characterizing losses of neutrons in given region and from group g),
$\Sigma_s^{g'\to g}$ is the macroscopic cross section characterizing scattering of neutrons from group g' into group g,
$\chi^g\nu\Sigma_f^{g'}$ characterizes the average number of neutrons that appear in group g due to fission induced by group g' neutrons,
k_{eff} is the reactor critical number.

Each of the G equations in (1) describes local conservation of flux of neutrons having energy within the respective group. We specifically consider a two group model in

[*] Supported by project 1M0545 and Research Plan MSM 4977751301.

which the energy threshold separating the groups is chosen such that $\chi^1 = 1,\ \chi^2 = 0$, and $\Sigma_s^{2\to 1} \equiv 0$ (for a physical explanation, see [3]).

Diffusion theory specifies the constitutive relation between neutron flux and neutron current by the Fick's law:

$$\mathbf{j}^g(\mathbf{r}) = -D^g(\mathbf{r})\nabla\phi^g(\mathbf{r}), \tag{2}$$

where D^g is the *diffusion coefficient.* Robin conditions apply on core boundary:

$$\gamma^g\phi^g(\mathbf{r}) - \mathbf{j}^g(\mathbf{r})\cdot\mathbf{n} = 0, \quad \gamma^g = \frac{1-\alpha^g}{2(1+\alpha^g)}, \quad \mathbf{r}\in\partial\Omega, \tag{3}$$

where $\mathbf{n}$ denotes the unit outward normal to the boundary $\partial\Omega$ at point $\mathbf{r}$ and physical properties of core surroundings are captured by the *albedo* coefficient α^g.

There is only one value of parameter k_{eff} for which the boundary value problem (1)–(3) admits a physically realistic solution ([4]). Physicists refer to this value as to the *reactor critical number* since it shows the deviation of the steady state of the core from its critical state (i.e. one in which there is a perfect balance between production and losses of neutrons). Mathematically, it is the largest eigenvalue of the problem and the corresponding eigenfunction, uniquely determined up to a multiple, represents the physical flux solution.

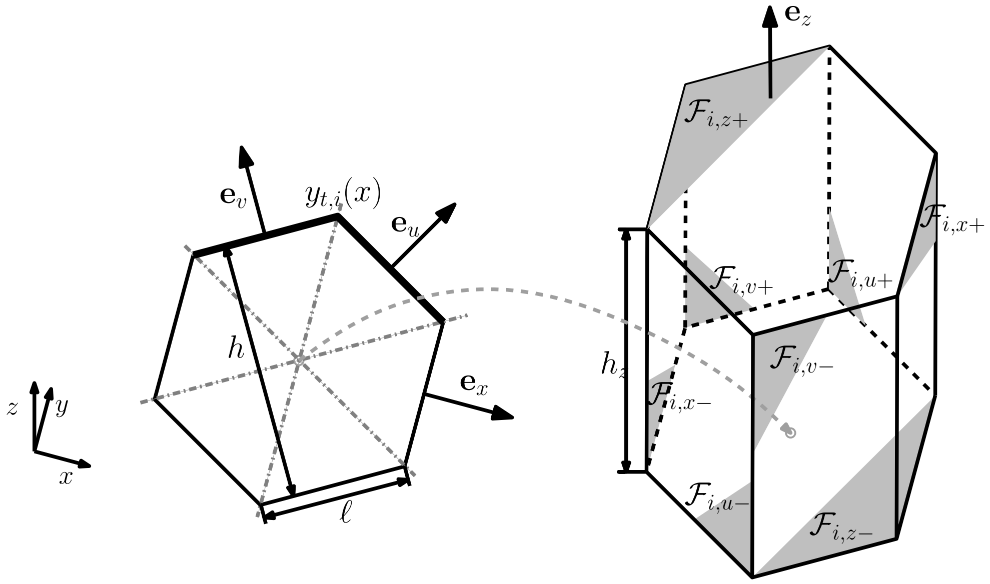
2. Nodal method

A proven numerical method for solving problems arising from conservation laws is the finite volume method (FVM). In order to obtain a computationally efficient number of discrete equations, each finite volume (here called *node*) is identified with a section of a real fuel assembly loaded into the core lattice. Extents of the node equal to full assembly width and height h_z such that $H_z = Mh_z$, where H_z is the total height of the core, and M a chosen number of its horizontal cuts. Nodal geometry is displayed in Fig. 1 and has the following metric properties:

$$\ell = \frac{h}{\sqrt{3}},\ F = \ell h_z,\ B = \frac{h^2\sqrt{3}}{2},\ V = Bh_z,$$

where B is the area of the nodal base, F the area of the other faces and V the nodal volume. Denoting by N the number of assemblies and by $\mathcal{V}_i$ individual nodes, the core domain is thus discretized as $\Omega = \bigcup_{i=1}^{NM}\mathcal{V}_i$. To each node we further associate a local coordinate system by unit vectors $\mathbf{e}_x$, $\mathbf{e}_u$, $\mathbf{e}_v$, $\mathbf{e}_z$ as shown in Fig. 1. Finally, we refer by symbols $\mathcal{V}_{i+\xi}$ and $\mathcal{V}_{i-\xi}$ to nodes adjacent to the reference node $\mathcal{V}_i$ to the right and left, respectively, with respect to coordinate direction $\xi \in \{x,u,v,z\}$. Their common face is denoted by $\mathcal{F}_{i,\xi\pm}$, i.e. $\mathcal{F}_{i,\xi\pm} = \mathcal{V}_i \cap \mathcal{V}_{i\pm\xi}$.

The equation expressing the local balance of group g of neutrons in node $\mathcal{V}_i$ can be formally obtained by integrating eq. (1) over $\mathcal{V}_i$, dividing by its volume and using the divergence theorem (for a rigorous derivation, see e.g. [3]):



Coordinates of radial cross section

Hex-Z prismatic node

Fig. 1: *Geometry of node $\mathcal{V}_i$.*

$$\frac{F}{V}\sum_{\xi\in\{x,u,v\}}\left(\bar{\bar{\mathcal{J}}}^g_{i,\xi+}-\bar{\bar{\mathcal{J}}}^g_{i,\xi-}\right)+\frac{B}{V}\left(\bar{\bar{\mathcal{J}}}^g_{i,z+}-\bar{\bar{\mathcal{J}}}^g_{i,z-}\right)+\Sigma^g_{i,r}\bar{\bar{\bar{\phi}}}^g_i=\sum_{\substack{g'=1\\ g'\neq g}}^{2}\Sigma^{g'\to g}_{i,s}\bar{\bar{\bar{\phi}}}^{g'}_i+\frac{\chi^g}{k_{\text{eff}}}\sum_{g'=1}^{2}\nu\Sigma^{g'}_{i,f}\bar{\bar{\bar{\phi}}}^{g'}_i\,. \tag{4}$$

The discrete unknowns in this equation are the *node-averaged neutron fluxes*:

$$\bar{\bar{\bar{\phi}}}^g_i := \frac{1}{V}\iiint_{\mathcal{V}_i}\phi^g(\mathbf{r})\,\mathrm{d}\mathbf{r}\,,$$

and the 6 *face-averaged radial* and 2 *base-averaged axial neutron currents*, respectively:

$$\bar{\bar{\mathcal{J}}}^g_{i,\xi\pm} := \frac{1}{F}\iint_{\mathcal{F}_{i,\xi\pm}}\mathbf{j}^g(\mathbf{r})\cdot\mathbf{e}_\xi\,\mathrm{d}\mathcal{F}\,,\ \text{for}\ \xi\in\{x,u,v\},\ \text{and}\ \bar{\bar{\mathcal{J}}}^g_{i,z\pm} := \frac{1}{B}\iint_{\mathcal{F}_{i,z\pm}}\mathbf{j}^g(\mathbf{r})\cdot\mathbf{e}_z\,\mathrm{d}\mathcal{F}\,.$$

We assume that physical properties of each fuel assembly are homogeneous, thus justifying the spatially constant Σ terms.

2.1. Coarse mesh, finite difference (CMFD) approximation

The standard FVM relation between discrete flux and current is obtained by replacing the flux derivative in eq. (2) by finite difference (FD) expressions and is thus usable only for small nodal sizes. The modification appropriate for relatively large nodal sizes characteristic for efficient reactor core models is called *nodal method.*

It is based on the following approximation of discrete average currents at interfaces $\mathcal{F}_{i,\xi+}$ and $\mathcal{F}_{i,\xi-}$, respectively:

$$\begin{aligned} \bar{\bar{J}}^g_{i,\xi+} &:= -D^g_{i,\xi+}\big(\bar{\bar{\bar{\Phi}}}^g_{i+\xi} - \bar{\bar{\bar{\Phi}}}^g_i\big) + {}^{\mathrm{C}}D^g_{i,\xi+}\big(\bar{\bar{\bar{\Phi}}}^g_i + \bar{\bar{\bar{\Phi}}}^g_{i+\xi}\big), \quad D^g_{i,\xi+} := \frac{2D^g_i D^g_{i+\xi}}{h_\xi\big(D^g_i + D^g_{i+\xi}\big)}, \\ \bar{\bar{J}}^g_{i,\xi-} &:= -D^g_{i,\xi-}\big(\bar{\bar{\bar{\Phi}}}^g_i - \bar{\bar{\bar{\Phi}}}^g_{i-\xi}\big) + {}^{\mathrm{C}}D^g_{i,\xi-}\big(\bar{\bar{\bar{\Phi}}}^g_{i-\xi} + \bar{\bar{\bar{\Phi}}}^g_i\big), \quad D^g_{i,\xi-} := \frac{2D^g_{i-\xi} D^g_i}{h_\xi\big(D^g_{i-\xi} + D^g_i\big)} \end{aligned} \tag{5}$$

(as a convention, capital letters will denote approximations of corresponding lowercase quantities, e.g. $\bar{\bar{J}}^g_{i,\xi+} \approx \bar{\bar{j}}_{i,\xi+}$). At core boundary, i.e. if $\mathcal{F}_{i,\xi\pm} \subset \partial\Omega$, expressions obtained by analogous discretization of the boundary condition (3) are used instead:

$$\bar{\bar{J}}^g_{i,\xi\pm} := D^g_{i,\xi\pm}\bar{\bar{\bar{\Phi}}}^g_i + {}^{\mathrm{C}}D^g_{i,\xi\pm}\bar{\bar{\bar{\Phi}}}^g_i, \quad D^g_{i,\xi\pm} := \pm\frac{2D^g_i\gamma_\xi}{h_\xi\gamma_\xi + 2D^g_i}, \quad \gamma^g_\xi := \frac{1-\alpha^g_\xi}{2(1+\alpha^g_\xi)}. \tag{6}$$

Different albedoes are defined for radial and axial boundaries which results in $\gamma_\xi = \gamma_{\mathrm{rad}}$ for $\xi \in \{x, u, v\}$ and $\gamma_z = \gamma_{\mathrm{ax}}$. Also note that $h_x = h_u = h_v \equiv h$.

This so called *CMFD approximation* alters the standard FD expressions by adding a term containing the *coupling correction factor* ${}^{\mathrm{C}}D^g$. This factor accounts for the coarse mesh spacing by forcing the rough estimate of current to match a more accurate value obtained by some appropriate refining calculation. We describe one possible method for getting such higher-quality solution in Section 2.2.

By inserting the CMFD expressions (5) or (6) into eq. (4), we obtain the following numerical approximation of the discrete balance relation for node $\mathcal{V}_i$ and group g:

$$\frac{2}{3}\sum_{\xi\in\{x,u,v\}} \bar{\bar{L}}^g_{i,\xi} + \bar{\bar{L}}^g_{i,z} + \Sigma^g_{i,r}\bar{\bar{\bar{\Phi}}}^g_i = \sum_{\substack{g'=1\\ g'\neq g}}^{2} \Sigma^{g'\to g}_{i,s}\bar{\bar{\bar{\Phi}}}^{g'}_i + \frac{\chi^g}{K_{\mathrm{eff}}}\sum_{g'=1}^{2}\nu\Sigma^{g'}_{i,f}\bar{\bar{\bar{\Phi}}}^{g'}_i, \tag{7}$$

where $\bar{\bar{L}}^g_{i,\xi} := \frac{\bar{\bar{J}}^g_{i,\xi+} - \bar{\bar{J}}^g_{i,\xi-}}{h_\xi}$ is the *neutron leakage* term expressing the average net neutron current through faces orthogonal to direction ξ. Using the lexicographic ordering of nodes inside the core, a matrix formulation follows:

$$\underbrace{\begin{bmatrix} \mathbf{L}^1 + {}^{\mathrm{C}}\mathbf{D}^1 + \boldsymbol{\Sigma}^1_r & \mathbf{0} \\ -\boldsymbol{\Sigma}^{1\to 2}_s & \mathbf{L}^2 + {}^{\mathrm{C}}\mathbf{D}^2 + \boldsymbol{\Sigma}^2_r \end{bmatrix}}_{\mathbf{M}} \cdot \underbrace{\begin{bmatrix} \bar{\bar{\bar{\boldsymbol{\Phi}}}}^1 \\ \bar{\bar{\bar{\boldsymbol{\Phi}}}}^2 \end{bmatrix}}_{\bar{\bar{\bar{\boldsymbol{\Phi}}}}} = \frac{1}{K_{\mathrm{eff}}} \underbrace{\begin{bmatrix} \nu\boldsymbol{\Sigma}^1_f & \nu\boldsymbol{\Sigma}^2_f \\ \mathbf{0} & \mathbf{0} \end{bmatrix}}_{\mathbf{F}} \cdot \underbrace{\begin{bmatrix} \bar{\bar{\bar{\boldsymbol{\Phi}}}}^1 \\ \bar{\bar{\bar{\boldsymbol{\Phi}}}}^2 \end{bmatrix}}_{\bar{\bar{\bar{\boldsymbol{\Phi}}}}}, \tag{8}$$

where $\bar{\bar{\bar{\boldsymbol{\Phi}}}}^g$ is a vector of NM average fluxes in group g, $\mathbf{L}^g$ and ${}^{\mathrm{C}}\mathbf{D}^g$ are matrices of finite-difference and correction factors, and $\boldsymbol{\Sigma}^g_r$, $\boldsymbol{\Sigma}^g_f$, $\boldsymbol{\Sigma}^{1\to 2}_s$ are diagonal matrices of reaction cross sections. All matrices are sparse of order NM.

Being a discrete analogue of the eigenvalue problem (1)–(3) for the neutron diffusion operator, eq. (8) expectedly constitutes a matrix eigenvalue problem. We also expect the same behaviour of its eigensolutions which allows us to use the standard power method for calculating the largest eigenvalue K_{eff} and the eigenvector $\bar{\bar{\mathbf{\Phi}}}^g$ of average nodal fluxes. The actual implementation of this so called *CMFD iteration* consists of two levels – the outer *source* iteration advancing the eigenvalue approximation as in the classical power method and the inner calculation of the fluxes vector according to eq. (8) with a fixed right-hand side.

2.2. Refinement of the CMFD approximation

After a few eigenvalue updates, the CMFD iteration is interrupted to refine the iteration matrix by determining new correction factors. For this purpose, one-dimensional diffusion equations are formed for each direction $\mathbf{e}_\xi$ by performing the transverse integration procedure. For a chosen node $\mathcal{V}_i$ and any of the radial directions, this amounts to integrating eq. (1) over a section slicing through the node orthogonally to the given direction. To illustrate the procedure for the x-direction, we integrate the 'flux version' of the equation (obtained by inserting (2) into eq. (1)) along the y and z axes and make an average over the cross-section area. This yields the following 1D diffusion equation (group index will be omitted in this section):

$$-D_i \frac{\mathrm{d}^2 \bar{\bar{\phi}}_i(x)}{\mathrm{d}x^2} + \Sigma_{i,r} \bar{\bar{\phi}}_i(x) = \bar{\bar{s}}_i(x) - \bar{\bar{l}}_i(x) \tag{9}$$

for the unknown *transverse-averaged flux*:

$$\bar{\bar{\phi}}_i(x) := \frac{1}{h_z} \int_{-h_z/2}^{h_z/2} \frac{1}{2y_i^t(x)} \int_{-y_i^t(x)}^{y_i^t(x)} \phi(x,y,z)\,\mathrm{d}y\ \mathrm{d}z\,, \tag{10}$$

where $y_i^t(x) = 1/\sqrt{3}(h - |x|)$ represents the radial transverse boundary of the node (the bold line in Fig. 1). The source term $\bar{\bar{s}}_i(x)$ abbreviates the transverse-averaged right-hand side of eq. (1), whereas the *transverse leakage term* $\bar{\bar{l}}_i(x)$ basically contains the normal components of neutron currents through transverse boundary faces $\mathcal{F}_{i,u\pm}$, $\mathcal{F}_{i,v\pm}$ and $\mathcal{F}_{i,z\pm}$. Since the CMFD solution provides only the average surface currents and their spatial dependence is not known until the transverse integrated equations in the remaining directions are solved, we need to approximate the transverse currents shapes to establish $\bar{\bar{l}}_i(x)$. By inserting (10) into eq. (9) and denoting the cusp of the boundary function $y_i^t(x)$, it becomes clear, however, that we also need to cope with the singularities that arise in $\bar{\bar{l}}_i(x)$ due to the differentiation of $y_i^t(x)$ and which form the other part of the transverse leakage term.

There are currently two approaches to the problem of hexagonal transverse leakage approximation investigated at our department. In the first, originally described in [1], the hexagonal problem is conformally mapped to a rectangular one, effectively eliminating the source of the singularities. In the other, invented by M.R. Wagner ([5]) and followed in this paper, singular terms in $\bar{\bar{l}}_i(x)$ are neglected. This gives the approximation $\bar{\bar{L}}_i(x) \approx \bar{\bar{l}}_i(x)$ which transforms eq. (9) into an approximate equation. We then seek its solution $\bar{\bar{\Phi}}_i(x) \approx \bar{\bar{\phi}}_i(x)$ so that the 1D problem remains consistent with the original 3D one in the sense of preserving the nodal averages of approximated quantities:

$$\frac{1}{V}\int_{-h/2}^{h/2} 2y_i^t(x)h_z\bar{\bar{\Phi}}_i(x)\,\mathrm{d}x = \bar{\bar{\bar{\Phi}}}_i, \quad \frac{1}{V}\int_{-h/2}^{h/2} 2y_i^t(x)h_z\bar{\bar{L}}_i(x)\,\mathrm{d}x = \bar{\bar{\bar{L}}}_i^{yz}. \tag{11}$$

Average nodal leakage $\bar{\bar{\bar{L}}}_i^{yz}$ through faces intersecting the y and z axes is obtained by integrating the 1D diffusion equation from $-h/2$ to $h/2$ and comparing the result with the original 3D eq. (7). This leads to:

$$\bar{\bar{\bar{L}}}_i^{yz} := \frac{2}{3}(\bar{\bar{L}}_{i,u} + \bar{\bar{L}}_{i,v}) - \frac{1}{3}\bar{\bar{L}}_{i,x} + \bar{\bar{L}}_{i,z}. \tag{12}$$

It is reasonable to assume that spatial variation of the transverse leakage function $\bar{\bar{L}}_i(x)$ is determined by the leakages through the transverse boundaries, i.e.

$$\bar{\bar{L}}_i(x) = f_i^{t,rad}(x) - \frac{1}{3}\bar{\bar{L}}_{i,x}, \tag{13}$$

where the shape function $f_i^{t,rad}(x)$ involves only $\bar{\bar{L}}_{i,u}$, $\bar{\bar{L}}_{i,v}$, $\bar{\bar{L}}_{i,z}$ and the second term ensures the consistency, whatever singularities the exact function $\bar{\bar{l}}_i(x)$ may contain.

Transverse integration in the axial direction leads to a z-direction diffusion equation formally identical to eq. (9). The transverse-averaged flux is now defined as

$$\bar{\bar{\phi}}_i(z) := \frac{1}{B}\int_{-h/2}^{h/2}\int_{-y_i^t(x)}^{y_i^t(x)} \phi(x,y,z)\,\mathrm{d}y\ \mathrm{d}x$$

and hence the transverse leakage term comprises only currents through faces orthogonal to the horizontal cross-section of the node. The approximate neutron balance relation may then be formally obtained by replacing $\bar{\bar{\phi}}_i(z)$ and $\bar{\bar{l}}_i(z)$ in the axial version of eq. (9) with their approximations $\bar{\bar{\Phi}}_i(z)$ and $\bar{\bar{L}}_i(z)$, respectively. They are again constructed so as to make the 1D and 3D equations consistent:

$$\frac{1}{h_z}\int_{-h_z/2}^{h_z/2} \bar{\bar{\Phi}}_i(z)\,\mathrm{d}z = \bar{\bar{\bar{\Phi}}}_i, \quad \frac{1}{h_z}\int_{-h_z/2}^{h_z/2} \bar{\bar{L}}_i(z)\,\mathrm{d}z = \bar{\bar{\bar{L}}}_i^{xy} := \frac{2}{3}\sum_{\xi\in\{x,u,v\}} \bar{\bar{L}}_{i,\xi} \tag{14}$$

Introducing the axial transverse profile function $f_i^{t,ax}(z)$ correspondingly to the radial case, we have $\bar{\bar{L}}_i(z) = f_i^{t,ax}(z)$ since there are no singularities in $\bar{\bar{l}}_i(z)$.

The simplest transverse profile function can be obtained by assuming a non-varying (*flat*) transverse leakage throughout the node, i.e. $f_i^{t,rad}(x) := \frac{2}{3}(\bar{\bar{L}}_{i,u} + \bar{\bar{L}}_{i,v})$ $+\ \bar{\bar{L}}_{i,z}$ and $f_i^{t,ax}(z) := \bar{\bar{\bar{L}}}_i^{xy}$. This approximation can be improved by taking into account the transverse leakages of two adjacent nodes $\mathcal{V}_{i-x}$, $\mathcal{V}_{i+x}$ and considering the consistency conditions for $\bar{\bar{L}}_i(x)$ also in their respective intervals. Assuming a parabolic transverse leakage profile, these three conditions form a system of three algebraic equations for the parabola's coefficients. Restriction of the resulting polynomial to $[-h/2, h/2]$ then defines $f_i^{t,rad}(x)$. An analogous approach is used in the axial direction.

2.3. Solution procedure

Equations (9) are cast into a group-matrix form and solved for both groups simultaneously by a semi-analytic method, first in the radial directions. Right-hand sides of the equations are defined using the results of the latest finished CMFD iteration. The algorithm sweeps through all pairs of nodes in given radial plane and direction, using the interface flux and current continuity conditions and the weighted residual method to solve the approximate 1D diffusion equations. The solution is expressed in terms of the face-averaged radial currents. Once these currents are known for all nodes in all planes, they are used to specify the transverse leakage term on the right-hand sides of the z-direction equations. Base-averaged axial currents can then be determined in the same manner (see [3] or [2] for further details). CMFD correction factors are finally obtained from eq. (5), (6) by equating their right-hand sides to the high-accuracy currents from the radial and axial sweeps. This completes the two-node subdomains solution and specifies new elements of the ${}^{\mathrm{C}}\mathbf{D}^g$ matrices, which in turn update the CMFD iteration matrix $\mathbf{M}$ (cf. eq. (8)). Another few CMFD iterations advancing the eigensolution are then performed and again followed by refinement sweeps. This procedure is repeated as long as the CMFD matrix changes considerably after each refining step.

The converged vector of node average fluxes may be analysed in various ways. Specifically for the benchmark problem presented in Section 3, we need the average nodal powers normalized to the average power of the whole core:

$$\bar{\bar{\bar{P}}}_i := \frac{1}{P}\big(\nu\Sigma^1_{i,f}\bar{\bar{\bar{\Phi}}}^1_i + \nu\Sigma^2_{i,f}\bar{\bar{\bar{\Phi}}}^2_i\big),\ i = 1, 2, \ldots, NM;\ P := \frac{1}{NM}\sum_{j=1}^{NM}\big(\nu\Sigma^1_{j,f}\bar{\bar{\bar{\Phi}}}^1_j + \nu\Sigma^2_{j,f}\bar{\bar{\bar{\Phi}}}^2_j\big),$$

and the axial offset, defined as the percentage difference between the average power generated in the upper and the lower halves of the core.

3. Numerical tests and conclusions

We tested the developed method by “Benchmark problem no. 6” from [1]. The investigated VVER-1000 type core is 200 cm high and has 163 fuel assemblies with radial pitch of 23.6 cm. As in [1], we divided the core into 10 axial layers, i.e. the height of each node was $h_z = 20$ cm. The calculation was finished by convergence of correction factors, indicated in step s by $\left\|{}^{\mathrm{C}}\mathbf{D}^{(s)} - {}^{\mathrm{C}}\mathbf{D}^{(s-1)}\right\|_\infty < \varepsilon = 10^{-6}$.

Table 1 shows deviations of the results from those of a fine-mesh finite-difference method DIF3D (reference is given in [1]), measured as the maximum and root mean square errors in average nodal powers ($\Delta\bar{\bar{\bar{P}}}_{max}$ and $\Delta\bar{\bar{\bar{P}}}_{rms}$, resp.), error in axial offset (ΔAO), and error in critical number (ΔK_{eff}). The table compares four versions of the method based on four combinations of transverse leakage approximation with two versions of the ANC code presented in [1]. ANC-HW uses Wagner's approach to hexagonal transverse leakage (as do we in this paper) while ANC-HM uses the conformal mapping technique. Neither of the ANC methods uses CMFD to advance

Method[†]	$\Delta\bar{\bar{P}}_{max}$ [%]	$\Delta\bar{\bar{P}}_{rms}$ [%]	ΔAO [%]	ΔK_{eff} [pcm=10^{-5}]
rFaF	10.3	4.5	−1.10	156.9
rFaQ	10.1	4.4	−1.15	138.1
rQaF	4.9	1.7	−0.41	7.6
rQaQ	4.7	1.7	−0.48	−12.3
ANC-HW	12.0	N/A[‡]	1.17	113.0
ANC-HM	0.9	N/A[‡]	0.07	13.0

[†] r ... approx. of $f_i^{t,rad}$, a ... approx. of $f_i^{t,ax}$; F ... flat, Q ... quadratic

[‡] result was not available

Tab. 1: *Benchmark results.*

the global solution, however, and both assume different transverse leakage shapes than do we in our four schemes.

The results indicate that it is the transverse leakage approximation used in radial direction that mostly determines the accuracy of results. Quadratic approximation proves to be superior to the flat one, although when the latter is used in the axial direction (in which the nodes are rectangular), it may give somewhat better core-wise average results. This results in a slightly better axial offset characterization and, in the case of sufficiently low maximum flux errors, also in a better critical number estimate. Either version of our method performs better than the ANC-HW method, which was developed under the same assumptions based on original Wagner's ideas, except for the transverse leakage profile. However, the superior accuracy of the conformal mapping technique for flux prediction still remains unmatched. This suggests that further research into integration of the CMFD and conformal mapping methods could yield fruitful results.

References

[1] Y.A. Chao, Y.A. Shatilla: *Conformal mapping and hexagonal nodal methods – II.* Nucl. Sci. Eng. **121** (1995), 210–225.

[2] X.D. Fu, N.Z. Cho: *Nonlinear analytic and semi-analytic nodal methods for multigroup neutron diffusion calculations.* J. Nucl. Sci. Technol. (Tokyo, Jpn.) **39** (2002), 1015–1025.

[3] M. Hanuš: *Numerical modelling of neutron flux in nuclear reactors.* Faculty of Applied Sciences, University of West Bohemia in Pilsen 2007. Bachelor's Thesis.

[4] E.L. Wachspress: *Iterative solution of elliptic systems and applications to the neutron diffusion equations of reactor physics.* Prentice-Hall, Inc., Englewood Cliffs, NJ, 1966.

[5] M.R. Wagner: *Three-dimensional nodal diffusion and transport theory methods for hexagonal-z geometry.* Nucl. Sci. Eng. **103** (1989), 377–391.

DISCONTINUOUS GALERKIN METHOD FOR THE SIMULATION OF 3D VISCOUS COMPRESSIBLE FLOWS*

Martin Holík, Vít Dolejší

1. Introduction

Our goal is to solve an unsteady viscous compressible flow which is described by the system of the *Navier-Stokes equations.* Our aim is to develop a sufficiently efficient, robust and accurate numerical method.

It is promising to use the *discontinuous Galerkin method* (DGM), which is based on a piecewise polynomial but discontinuous approximation, which is suitable for problems with discontinuities. We prefer the *discontinuous Galerkin finite element* (DGFE) method with *symmetric, nonsymmetric and/or incomplete* variant of stabilization and *interior* and *boundary penalty terms.* These schemes are usually denoted as SIPG, NIPG and IIPG, respectively.

The most usual approach for the time discretization is method of lines. Explicit methods such as the Runge-Kutta methods are very popular for their simplicity and a high order of accuracy. However, they suffer from strong time step restrictions. Fully implicit schemes lead to a system of highly nonlinear algebraic equations at each time step whose solution is complicated. To avoid this disadvantage we employ a semi-implicit method for the time discretization, which is based on a suitable linearization of the fluxes. The linear terms are treated implicitly and the nonlinear ones explicitly.

2. Compressible flow problem

For the description of motion of a viscous compressible flow we use the system of Navier-Stokes equations.

Let $\Omega \subset I\!R^3$ be a bounded domain and $T > 0$. We set $Q_T = \Omega \times (0, T)$ and by $\partial\Omega$ we denote the boundary of Ω which consists of several disjoint parts. We distinguish inlet Γ_I, outlet Γ_O and impermeable walls Γ_W on $\partial\Omega$. Using the Fourier law and some relations from physics, we can write these equations in the dimensionless form

$$\frac{\partial \boldsymbol{w}}{\partial t} + \nabla \cdot \vec{f}(\boldsymbol{w}) = \sum_{s=1}^{3} \frac{\partial}{\partial x_s} \left(\sum_{k=1}^{3} \boldsymbol{K}_{sk}(\boldsymbol{w}) \frac{\partial \boldsymbol{w}}{\partial x_k} \right) \quad \text{in } Q_T, \tag{1}$$

*This work is a part of the research project MSM 0021620839 financed by the Ministry of Education of the Czech Republic and was partly supported by grant No. 316/2006/B-MAT/MFF of the Grant Agency of the Charles University Prague.

where

$$\boldsymbol{w} = (w_1, \dots, w_5)^{\mathrm{T}} = (\rho,\ \rho v_1,\ \rho v_2, \rho v_3,\ e)^{\mathrm{T}} \tag{2}$$

is the so-called *state vector*, $\vec{f} = (\boldsymbol{f}_1, \boldsymbol{f}_2, \boldsymbol{f}_3)$,

$$\begin{aligned} \boldsymbol{f}_s(\boldsymbol{w}) &= (f_s^{(1)}(\boldsymbol{w}), \dots, f_s^{(5)}(\boldsymbol{w}))^{\mathrm{T}} \\ &= (\rho v_s,\ \rho v_s v_1 + \delta_{s1} p,\ \rho v_s v_2 + \delta_{s2} p,\ \rho v_s v_3 + \delta_{s3} p,\ (e+p)\, v_s)^{\mathrm{T}},\ \ s = 1,2,3, \end{aligned} \tag{3}$$

are the so-called *inviscid (Euler) fluxes*, where ρ, p, and e stand for the density, the pressure, and the total energy, respectively, and δ is the Kronecker's delta. For description of the matrix $\boldsymbol{K}_{sk}(\boldsymbol{w}) : I\!R^5 \to I\!R^5 \times I\!R^5,\ s,k = 1,2,3$ see [6].

In order to close the system we use the following thermodynamical relations: *the state equation for perfect gas* and *the relation for the total energy.* The system is of *hyperbolic-parabolic* type. It is equipped with initial and boundary conditions. For more details see [1].

2.1. Properties of inviscid fluxes

From the expression of the Euler fluxes $\boldsymbol{f}_s$, $s = 1,2,3$ we find that $\boldsymbol{f}_s$ can be written (see [1]) in the form

$$\boldsymbol{f}_s(\boldsymbol{w}) = \boldsymbol{A}_s(\boldsymbol{w})\boldsymbol{w}, \quad s = 1,2,3, \tag{4}$$

where $\boldsymbol{A}_s(\boldsymbol{w})$ are the Jacobi matrices of the mappings $\boldsymbol{f}_s$.

3. Discretization

For discretization we employ the *discontinuous Galerkin finite element method* (DGFEM), which takes advantages from finite element method as well as from finite volume method. DGFEM is based on piecewise polynomial approximation without any requirement on interelement continuity what is suitable for problems where shock waves and contact discontinuities appear.

Let $\mathcal{T}_h$ $(h > 0)$ be a partition of the domain Ω into a finite number of open three-dimensional mutually disjoint simplexes and/or parallelograms K i.e., $\overline{\Omega} = \bigcup_{K \in \mathcal{T}_h} \overline{K}$. We call $\mathcal{T}_h$ a *triangulation* of Ω and do not require the conforming properties from the finite element method. We define the set of faces $\mathcal{F}_h$, $\mathcal{F}_h^D$, $\mathcal{F}_h^{ID}$ and a unit normal vector $\boldsymbol{n}_\Gamma$, as can be seen in [7].

Over the triangulation $\mathcal{T}_h$ we define the *broken Sobolev space*

$$H^k(\Omega, \mathcal{T}_h) = \{v; v|_K \in H^k(K)\ \forall\, K \in \mathcal{T}_h\}, \tag{5}$$

where $H^k(K) = W^{k,2}(K)$ denotes the (classical) Sobolev space on element K.

We introduce the following notation $v|_\Gamma$, $\langle v \rangle_\Gamma$ and $[v]_\Gamma$ for trace, mean value and jump, respectively, of function v over the edge Γ, see [2].

There are several variant of DGFEM. A particular role is played by the *symmetric and nonsymmetric interior penalty Galerkin* variant, denoted by SIPG and NIPG, respectively. The main idea of SIPG and NIPG is to append artificial integral to each boundary integral

$$\int_\Gamma \langle \nabla u \cdot \vec{n} \rangle [\varphi] \, \mathrm{d}S, \qquad \nabla u \in [L^2(\Gamma)]^3, \ \varphi \in L^2(\Gamma), \tag{6}$$

arising from the use of Green's theorem in the case of linear diffusion simply by formal exchange of u and φ. We can see that this integrals vanish in case of regular solution. In our case we use the linearization formed by the terms $\boldsymbol{K}_{sk}$, see [6], or employ the so-called *incomplete interior penalty Galerkin* (IIPG) method, see [7].

Similarly, as in [3] for $\boldsymbol{w}, \boldsymbol{\varphi} \in [H^2(\Omega, \mathcal{T}_h)]^5$, we define the forms:

$$\begin{aligned}
\tilde{\boldsymbol{a}}_h(\boldsymbol{w}, \boldsymbol{\varphi}) &= \sum_{K \in \mathcal{T}_h} \int_K \sum_{s=1}^3 \left(\sum_{k=1}^3 \left(\boldsymbol{K}_{sk}(\boldsymbol{w}) \frac{\partial \boldsymbol{w}}{\partial x_k} \right) \frac{\partial \boldsymbol{\varphi}}{\partial x_s} \right) \mathrm{d}x \\
&- \sum_{\Gamma \in \mathcal{F}_h^{ID}} \int_\Gamma \sum_{s=1}^3 \left(\langle \sum_{k=1}^3 \boldsymbol{K}_{sk}(\boldsymbol{w}) \frac{\partial \boldsymbol{w}}{\partial x_k} \rangle_\Gamma n_s \right) \cdot [\boldsymbol{\varphi}]_\Gamma \, \mathrm{d}S \\
&- \Theta \sum_{\Gamma \in \mathcal{F}_h^{ID}} \int_\Gamma \sum_{s=1}^3 \left(\langle \sum_{k=1}^3 \boldsymbol{K}_{sk}(\boldsymbol{w}) \frac{\partial \boldsymbol{\varphi}}{\partial x_k} \rangle_\Gamma n_s \right) \cdot [\boldsymbol{w}]_\Gamma \, \mathrm{d}S \qquad (7) \\
&+ \Theta \sum_{\Gamma \in \mathcal{F}_h^{D}} \int_\Gamma \sum_{s=1}^3 \left((\sum_{k=1}^3 \boldsymbol{K}_{sk}(\boldsymbol{w}) \frac{\partial \boldsymbol{\varphi}}{\partial x_k}) n_s \right) \cdot \boldsymbol{w}_B(t) \, \mathrm{d}S, \\
\bar{\boldsymbol{b}}_h(\boldsymbol{w}, \boldsymbol{\varphi}) &= - \sum_{K \in \mathcal{T}_h} \int_K \sum_{s=1}^3 \boldsymbol{f}_s(\boldsymbol{w}) \cdot \frac{\partial \boldsymbol{\varphi}}{\partial x_s} \, \mathrm{d}x \\
&\quad + \sum_{\Gamma \in \mathcal{F}_h} \int_\Gamma H \left(\boldsymbol{w}|_\Gamma^{(p)}, \boldsymbol{w}|_\Gamma^{(n)}, \vec{n}_\Gamma \right) [\boldsymbol{\varphi}]_\Gamma \, \mathrm{d}S, \qquad (8) \\
J_h^\sigma(\boldsymbol{w}, \boldsymbol{\varphi}) &= \sum_{\Gamma \in \mathcal{F}_h^{ID}} \int_\Gamma \sigma [\boldsymbol{w}]_\Gamma \cdot [\boldsymbol{\varphi}]_\Gamma \, \mathrm{d}S - \sum_{\Gamma \in \mathcal{F}_h^{D}} \int_\Gamma \sigma \, \boldsymbol{w}_B(t) \cdot \boldsymbol{\varphi} \, \mathrm{d}S, \qquad (9)
\end{aligned}$$

where Θ is +1 in SIPG, -1 in NIPG and 0 in IIPG case and σ is a suitable coercive parameter and $\boldsymbol{w}_B(t)$ is the solution on the boundary, where Dirichlet condition is prescribed. $H \left(\boldsymbol{w}|_\Gamma^{(p)}, \boldsymbol{w}|_\Gamma^{(n)}, \vec{n}_\Gamma \right)$ is the so-called *numerical flux*, well-known in the finite volume method (see, e.g., [1, Section 3.2])

Now we can introduce the *semidiscrete* problem. The approximate solution of problem (1) with initial and boundary condition is sought at each instant time t in the space of discontinuous piecewise polynomial functions $\boldsymbol{S}_h$ defined by $\boldsymbol{S}_h \equiv [S_h]^5$, $\ S_h \equiv \{v; v|_K \in P^p(K) \ \forall K \in \mathcal{T}_h\}$, where p is a positive integer and $P^p(K)$ denotes the space of all polynomials on K of degree at most p.

In order to avoid the time step restriction and nonlinearity of the discretized problem, we carry out a linearization of the nonlinear forms $\tilde{\boldsymbol{a}}_h$ and $\bar{\boldsymbol{b}}_h$.

For $\bar{\boldsymbol{w}}_h, \boldsymbol{w}_h, \boldsymbol{\varphi}_h \in \boldsymbol{S}_h$ we define a new form $\boldsymbol{b}_h(\bar{\boldsymbol{w}}_h, \boldsymbol{w}_h, \boldsymbol{\varphi}_h)$ using (4) and (8), which is linear with respect to the second and the third variable and consistent with $\bar{\boldsymbol{b}}_h(\cdot,\cdot)$ by $\bar{\boldsymbol{b}}_h(\boldsymbol{w}_h,\boldsymbol{\varphi}_h) = \boldsymbol{b}_h(\boldsymbol{w}_h,\boldsymbol{w}_h,\boldsymbol{\varphi}_h) \quad \forall\, \boldsymbol{w}_h,\boldsymbol{\varphi}_h \in \boldsymbol{S}_h$. For more details see, e.g. [3].

In a similar way, as in the case of the form $\bar{\boldsymbol{b}}_h$, we define a new form $\boldsymbol{a}_h(\bar{\boldsymbol{w}}_h,\boldsymbol{w}_h,\boldsymbol{\varphi}_h)$ for $\bar{\boldsymbol{w}}_h,\boldsymbol{w}_h,\boldsymbol{\varphi}_h \in \boldsymbol{S}_h$ using properties of the forms $\boldsymbol{K}_{sk}$ and (7), which is also linear with respect to its second and third variable. Moreover, it is consistent with $\tilde{\boldsymbol{a}}_h(\cdot,\cdot)$ by $\tilde{\boldsymbol{a}}_h(\boldsymbol{w}_h,\boldsymbol{\varphi}_h) = \boldsymbol{a}_h(\boldsymbol{w}_h,\boldsymbol{w}_h,\boldsymbol{\varphi}_h) \quad \forall\, \boldsymbol{w}_h,\boldsymbol{\varphi}_h \in \boldsymbol{S}_h$. The definition of $\boldsymbol{a}_h(\cdot,\cdot,\cdot)$ can be found in [3].

Now we introduce the full space-time discrete problem. The main idea of the semi-implicit discretization is to treat the linear parts of forms $\boldsymbol{a}_h$ and $\boldsymbol{b}_h$ implicitly and their nonlinear parts explicitly. In order to obtain a sufficiently accurate approximation with respect to the time coordinate we use the so-called *backward difference formula* (BDF) for the solution of the ODE semidiscrete problem. Moreover, a suitable explicit higher order extrapolation is used in the nonlinear parts of $\boldsymbol{a}_h$ and $\boldsymbol{b}_h$.

Let $0 = t_0 < t_1 < \ldots < t_r = T$ be a partition of the interval $(0,T)$ and let $\tau_k \equiv t_{k+1} - t_k,\ k = 0,1,\ldots,r-1$, be the time steps.

Definition 1 *Functions* $\boldsymbol{w}_h^{k+1},\ k = 0,\ldots,r-1$ *are an approximate solution of problem (1) with some suitable initial and boundary conditions satisfying*

$$\begin{aligned}
&\textit{(a)} \quad && \boldsymbol{w}_h^{k+1} \in \boldsymbol{S}_h,\\
&\textit{(b)} \quad && \frac{1}{\tau_k}\left(\sum_{l=0}^{n}\alpha_l \boldsymbol{w}_h^{k+1-l},\boldsymbol{\varphi}_h\right) + \boldsymbol{a}_h\left(\sum_{l=1}^{n}\beta_l \boldsymbol{w}_h^{k+1-l},\boldsymbol{w}_h^{k+1},\boldsymbol{\varphi}_h\right)\\
& && \quad +\boldsymbol{b}_h\left(\sum_{l=1}^{n}\beta_l \boldsymbol{w}_h^{k+1-l},\boldsymbol{w}_h^{k+1},\boldsymbol{\varphi}_h\right) + \boldsymbol{J}_h\left(\boldsymbol{w}_h^{k+1},\boldsymbol{\varphi}_h\right) = 0 \qquad (10)\\
& && \quad \forall\, \boldsymbol{\varphi}_h \in \boldsymbol{S}_h,\ k = n-1,\ldots,r-1,\\
&\textit{(c)} \quad && \boldsymbol{w}_h^0 \ \textit{ is an } \boldsymbol{S}_h \textit{ approximation of initial condition } \boldsymbol{w}^0,\\
&\textit{(d)} \quad && \boldsymbol{w}_h^l \in \boldsymbol{S}_h,\ l = 1,\ldots,n-1 \textit{ are given by a suitable one-step method,}
\end{aligned}$$

where $n \geq 1$ *is the degree of the BDF scheme, the coefficients* $\alpha_l,\ l = 0,\ldots,n,$ *and* $\beta_l,\ l = 1,\ldots,n,$ *depend on time steps* $\tau_{k-l},\ l = 0,\ldots,n.$

The problem (10), (a)–(d) represents a system of linear algebraic equations for each $k = n-1,\ldots,r-1$, which is solved by a suitable iterative solver (e.g. GMRES).

4. Stabilization

Application of this numerical scheme to transsonic flow leads to spurious overshoots and undershoots in computed quantities near shock waves. We use the stabilization of the scheme similar to [5]. For each $K_i \in \mathcal{T}_h$ we define quantity $g_{K_i}(\boldsymbol{w}_h)$, what measures the interelement jump of the function ρ_h, what is piecewise polynomial approximation of ρ. Moreover, we define the forms d_h and J_h which represent artificial viscosity and interior penalty, respectively. Both terms vanish in region where $\boldsymbol{w}_h$ is smooth. In [5], the stabilization for 2D problems is derived. The choice of exponents and another changes in forms d_h and J_h for 3D is in development.

5. Adaptive time step

In order to achieve a steady-state solution in an efficient way it is necessary to adapt the time step during the computational process. In [4], an adaptive choice of the time step based on a comparison of two BDF formulae was presented. However, this approach does not seem to be very efficient for viscous compressible flow simulations. Therefore, we employ an heuristic choice of the time step which is based on a idea to increase the time step when the "steady-state residuum" is decreasing. Hence we put: $\tau_k = \sqrt{\frac{const}{||\boldsymbol{w}_k - \boldsymbol{w}_{k-1}||}}$, where $const$ is a suitable constant (e.g. 10^{-6}).

6. Implementation

The subject of this research is a part of the project ADIGMA supported by the European Commission. This project holds in the period 2006–2009 and is devoted to development, application and verification of higher order schemes for the simulation of viscous compressible flow.

The presented numerical method is now being implemented within the object oriented platform COOLFluid, developed at the Von Karman Institute in Brussel, see [8]. The main advantage of object oriented programing is in dynamic creation

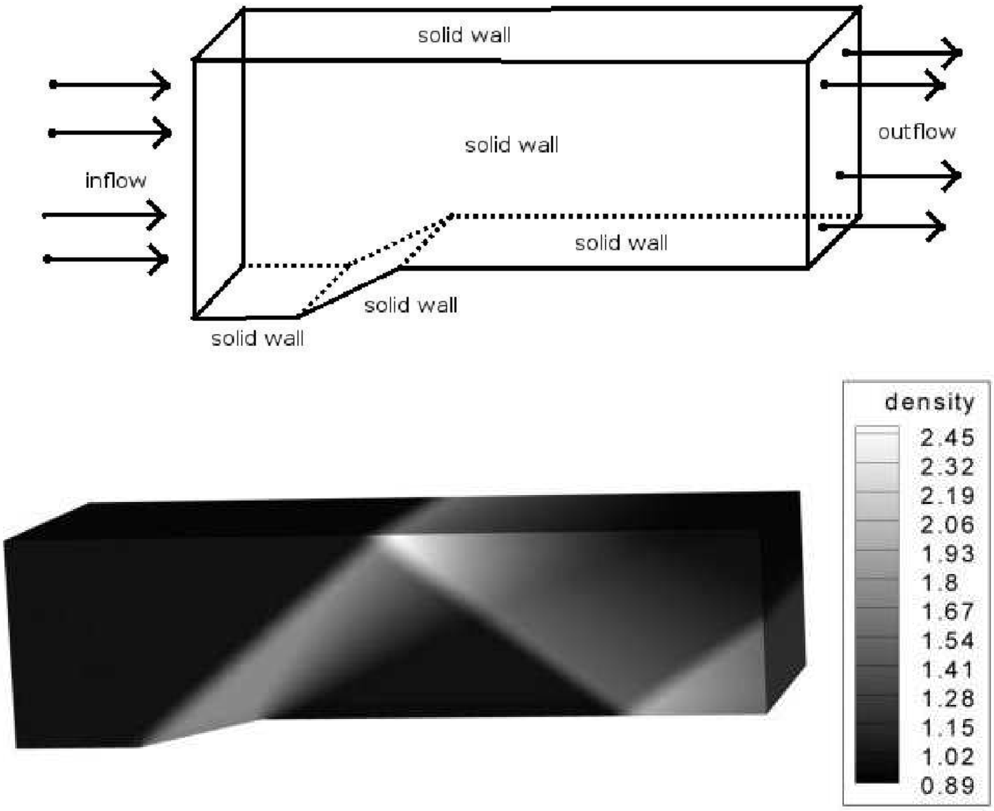


Fig. 1: *Distribution of density.*

of the object. There are templates of methods for discretization in time and space, for solving the system of equations, for any type of elements in 2D and 3D. But in the process of computation there are created only objects which are really needed for computation. It takes some time at the beginning for creation and initialization of objects, but the computer code is then shorter and simpler to write.

7. Example – Wedge 3D

Wedge 3D is one of COOLFluiD test cases. It is supersonic flow (MACH = 2.0) in channel forward facing oblique step. The initial condition is constant with values $\rho = 1.0, v = (2.366431913, 0.0, 0.0), e = 5.3$.

8. Conclusion

We described a numerical solution of the compressible Navier-Stokes equations by a combination of DGFEM and BDF. We presented SIPG, NIPG and IIPG variants of DGFEM. These schemes are theoretically unconditionally stable, have a high order of approximation with respect to space and time and lead to a linear algebraic systems at each time step.

References

[1] M. Feistauer, J. Felcman, I. Straškraba: *Mathematical and computational methods for compressible flow*, Oxford University Press, Oxford, 2003.

[2] V. Dolejší: *On the discontinuous Galerkin method for the numerical solution of the Navier-Stokes equations*, Internat. J. Numer. Methods Fluids **45** (2004), 1083–1106.

[3] V. Dolejší, J. Hozman: *Semi-implicit discontinuous Galerkin method for the solution of the compressible Navier-Stokes equations*, European Conference on Computational Fluid Dynamics, Egmond aan Zee, The Netherlands, 2006, pp. 1061–1080.

[4] V. Dolejší, P. Kůs: *Adaptive backward difference formula – discontinuous Galerkin finite element method for the solution of conservation laws*, Internat. J. Numer. Methods Engrg. **73** (2008), 1739–1766.

[5] V. Dolejší: *Discontinuous Galerkin method for the numerical simulation of unsteady compressible flow*, WSEAS Transactions on Systems **5** (2006), 1083–1090.

[6] V. Dolejší: *Semi-implicit interior penalty discontinuous Galerkin methods for viscous compressible flows*, Commun. Comput. Phys. **4** (2008), 231–274.

[7] V. Dolejší: *Analysis and application of IIPG method to quasilinear nonstationary convection–diffusion problems*, J. Comput. Appl. Math. **222** (2008), 251–273.

[8] Home page of COOLFluid: `https://coolfluidsrv.vki.ac.be/coolfluid/`

ANALYSIS OF THE DISCONTINUOUS GALERKIN FINITE ELEMENT METHOD APPLIED TO A SCALAR NONLINEAR CONVECTION-DIFFUSION EQUATION*

Jiří Hozman, Vít Dolejší

Abstract

We deal with a scalar nonstationary convection-diffusion equation with nonlinear convective as well as diffusive terms which represents a model problem for the solution of the system of the compressible Navier-Stokes equations describing a motion of viscous compressible fluids. We present a discretization of this model equation by the discontinuous Galerkin finite element method. Moreover, under some assumptions on the nonlinear terms, domain partitions and the regularity of the exact solution, we introduce a priori error estimates in the $L^\infty(0,T;L^2(\Omega))$-norm and in the $L^2(0,T;H^1(\Omega))$-seminorm. A sketch of the proof is presented.

1. Introduction

Our goal is to develop a sufficiently robust, accurate and efficient numerical method for the solution of the system of the compressible Navier-Stokes equations describing a motion of viscous compressible fluids. Due to the lack of the theory concerning an existence of the solution of the Navier-Stokes equations we consider the model problem represented by a nonstationary two-dimensional convection-diffusion equation with nonlinear convection as well as diffusion.

Among a wide class of numerical methods, the *discontinuous Galerkin finite element method* (DGFEM) seems to be a promising technique for the solution of convection-diffusion problems. DGFEM is based on a piecewise polynomial but discontinuous approximation, for a survey, see, e.g., [2], [3]. Within this paper we deal with the space semidiscretization of the model problem with the aid of the three variants of DGFEM. Namely nonsymmetric (NIPG), symmetric (SIPG) and incomplete interior penalty Galerkin (IIPG) techniques, see [1].

This article represents a generalization of research papers [5], [6], [7], and [8], where the linear diffusion term was considered. Moreover, let us cite works [4], [9], and [10], where simpler forms of nonlinear diffusion were analysed.

*This work is a part of the research project MSM 0021620839 financed by the Ministry of Education of the Czech Republic and it was partly supported by Grant No. 316/2006/B-MAT/MFF of the Grant Agency of the Charles University Prague.

2. Problem formulation

We consider the following unsteady nonlinear convection-diffusion problem. Let $\Omega \subset I\!\!R^2$ be a bounded polygonal domain and $T > 0$. We seek a function $u : Q_T \to I\!\!R$, $Q_T = \Omega \times (0, T)$, such that

$$\text{(a)} \qquad \frac{\partial u}{\partial t} + \sum_{s=1}^{2} \frac{\partial f_s(u)}{\partial x_s} = \operatorname{div}(I\!\!K(u)\,\nabla u) + g \quad \text{in } Q_T, \tag{1}$$

$$\text{(b)} \qquad u|_{\partial\Omega \times (0,T)} = u_D, \tag{2}$$

$$\text{(c)} \qquad u(x, 0) = u^0(x), \quad x \in \Omega, \tag{3}$$

where $g : Q_T \to I\!\!R$, $u_D : \partial\Omega \times (0, T) \to I\!\!R$, $u^0 : \Omega \to I\!\!R$ are given functions, $f_1, f_2 \in C^1(I\!\!R)$ represent convective terms and the matrix $I\!\!K(u) \in I\!\!R^{2,2}$ plays a role of nonlinear anisotropic diffusive coefficients. If $I\!\!K(u) = \varepsilon I\!\!I$, where ε is a positive constant and $I\!\!I \in I\!\!R^{2,2}$ the unit matrix, then problem (1) reduces to the equation considered in [5], [6], [7], [8]. For simplicity we prescribe the Dirichlet condition on the whole boundary but it is also possible to consider mixed Dirichlet-Neumann boundary conditions.

Formally, we introduce a weak solution u of the problem (1) by

$$\frac{\mathrm{d}}{\mathrm{d}t}(u(t), v) + b(u(t), v) + a(u(t), v) = (g(t), v) \quad \forall\, v \in H_0^1(\Omega), \text{ a.e. } t \in (0, T), \tag{4}$$

where $u(t)$ denotes the function on Ω such that $u(t)(x) = u(x, t)$, $x \in \Omega$. Further, $(\cdot, \cdot)$ denotes the L^2-scalar product, $a(\cdot, \cdot)$ and $b(\cdot, \cdot)$ are nonlinear forms representing the diffusive and convective terms, respectively. We also consider appropriate representation of initial and boundary conditions. For details see [4], [7].

3. Discretization

Let $\mathcal{T}_h$ $(h > 0)$ be a family of the partitions of the domain $\Omega \subset I\!\!R^2$ into triangular elements. We do not require the conformity of the mesh, i.e., the so-called hanging nodes are allowed. However, more general elements (even non-convex) can be considered within the frame of DGFEM, see [7]. By $\mathcal{F}_h$ we denote the smallest possible set of all edges of all elements $K \in \mathcal{T}_h$. Furthermore, let $\mathcal{F}_h^I$ and $\mathcal{F}_h^D$ represent the *interior* and the *boundary* edges of $\mathcal{T}_h$, respectively. Obviously $\mathcal{F}_h = \mathcal{F}_h^I \cup \mathcal{F}_h^D$. Finally, for each $\Gamma \in \mathcal{F}_h$ we define a unit normal vector $\vec{n}_\Gamma$. We assume that $\vec{n}_\Gamma$, $\Gamma \subset \partial\Omega$ has the same orientation as the outer normal of $\partial\Omega$. For $\vec{n}_\Gamma$, $\Gamma \in \mathcal{F}_I$ the orientation is arbitrary but fixed for each edge.

The approximate solution is sought in a space of piecewise polynomial but discontinuous functions

$$S_{hp} \equiv S_{hp}\,(\Omega, \mathcal{T}_h) = \{v; v|_K \in P_p(K)\ \forall\, K \in \mathcal{T}_h\}, \tag{5}$$

where $P_p(K)$ denotes the space of all polynomials on K of degree $\leq p$, $K \in \mathcal{T}_h$.

For each $\Gamma \in \mathcal{F}_h^I$ there exist two elements $K_L,\ K_R \in \mathcal{T}_h$ such that $\Gamma \subset K_L \cap K_R$. We use a convention that K_R lies in the direction of $\vec{n}_\Gamma$ and K_L in the opposite direction of $\vec{n}_\Gamma$. For $v \in S_{hp}$, by

$$v|_\Gamma^{(L)} = \text{ trace of } v|_{K_L} \text{ on } \Gamma, \quad v|_\Gamma^{(R)} = \text{ trace of } v|_{K_R} \text{ on } \Gamma \tag{6}$$

we denote the *traces* of v on edge Γ, which are different in general. Additionally,

$$[v]_\Gamma = v|_\Gamma^{(L)} - v|_\Gamma^{(R)}, \qquad \langle v \rangle_\Gamma = \frac{1}{2}\left(v|_\Gamma^{(L)} + v|_\Gamma^{(R)} \right), \tag{7}$$

denotes the *jump* and the *mean value* of function v over the edge Γ, respectively. For $\Gamma \subset \partial\Omega$ there exists an element $K_L \in \mathcal{T}_h$ such that $\Gamma \subset K_L \cap \partial\Omega$. Then for $v \in S_{hp}$, we put: $v|_\Gamma^{(L)} =$ trace of $v|_{K_L}$ on Γ, $\quad \langle v \rangle_\Gamma = [v]_\Gamma = v|_\Gamma^{(L)}$. In case that $[\cdot]_\Gamma$ and $\langle\,\cdot\,\rangle_\Gamma$ are arguments of $\int_\Gamma \ldots \,\mathrm{d}S$, $\Gamma \in \mathcal{F}_h$ we omit the subscript Γ and write simply $[\cdot]$ and $\langle\,\cdot\,\rangle$, respectively.

Similarly as in [5], it is possible to derive the space semi-discretization of (1). A particular attention should be paid to the nonlinear diffusive term. In order to replace the interelement continuity, we add some stabilization and penalty terms into formulation of the discrete problem. The convective term is approximated with the aid of a numerical flux $H(\cdot,\cdot,\cdot)$, known from the finite volume method.

Therefore, we say that $u_h \in C^1(0,T;S_{hp})$ is the *semi-discrete solution* of (1) if $(u_h(0), v_h) = (u^0, v_h)\ \forall v_h \in S_{hp}$ and

$$\left(\frac{\partial u_h(t)}{\partial t}, v_h \right) + b_h(u_h(t), v_h) + a_h^\Theta(u_h(t), v_h) + \alpha J_h^\sigma(u_h(t), v_h) = \ell_h^\Theta(u_h(t), v_h)\,(t) \tag{8}$$
$$\forall\, v_h \in S_{hp},\ \forall\, t \in (0,T),$$

where

$$\begin{aligned}
a_h^\Theta(u,v) &= \sum_{K\in\mathcal{T}_h} \int_K \mathbb{K}(u)\,\nabla u \cdot \nabla v \,\mathrm{d}x - \sum_{\Gamma\in\mathcal{F}_h} \int_\Gamma \langle \mathbb{K}(u)\,\nabla u \cdot \vec{n} \rangle [v] \,\mathrm{d}S \\
&+ \Theta \sum_{\Gamma\in\mathcal{F}_h} \int_\Gamma \langle \mathbb{K}(u)\,\nabla v \cdot \vec{n} \rangle [u] \,\mathrm{d}S, &(9)\\
b_h(u,v) &= -\sum_{K\in\mathcal{T}_h} \int_K \sum_{s=1}^{2} f_s(u) \frac{\partial v}{\partial x_s} \,\mathrm{d}x + \sum_{\Gamma\in\mathcal{F}_h} \int_\Gamma H(u|_\Gamma^{(L)}, u|_\Gamma^{(R)}, \vec{n}_\Gamma)\,[v] \mathrm{d}S, &(10)\\
J_h^\sigma(u,v) &= \sum_{\Gamma\in\mathcal{F}_h} \int_\Gamma \sigma [u]\,[v] \,\mathrm{d}S, &(11)\\
\ell_h^\Theta(u,v)(t) &= \int_\Omega g(t)\, v \,\mathrm{d}x + \sum_{\Gamma\in\mathcal{F}_h^D} \int_\Gamma \left(\Theta\, \mathbb{K}(u)\,\nabla v \cdot \vec{n}\, u_D(t) + \sigma\, u_D(t)\, v \right) \mathrm{d}S. &(12)
\end{aligned}$$

Nonlinear forms $a_h^\Theta(\cdot,\cdot)$ and $b_h(\cdot,\cdot)$ are the discrete variants of the forms $a(\cdot,\cdot)$ and $b(\cdot,\cdot)$, respectively. According to value of parameter Θ, we speak of SIPG ($\Theta = -1$), IIPG ($\Theta = 0$) or NIPG ($\Theta = 1$) variants of DGFEM. Penalty terms are represented by J_h^σ and the penalty parameter function σ in (11) is defined as $\sigma|_\Gamma = C_W \cdot |\Gamma|^{-1}$, $\Gamma \in \mathcal{F}_h$, where $C_W \geq 0$ is a suitable constant depending on the used variant of the scheme and on the degree of polynomial approximation. The value of multiplicative constant α before the penalty form J_h^σ will be specified in Section 4, assumption (14).

The problem (8) exhibits a system of ordinary differential equations for $u_h(t)$ which has to be discretized by a suitable ODE method.

If the numerical flux H is consistent with the convective fluxes f_1, f_2 (i.e., $H(u, u, \vec{n}) = f_1(u)n_1 + f_2(u)n_2 \; \forall u \in I\!R \; \forall \vec{n} = (n_1, n_2)$) then we find that the sufficiently regular exact solution u satisfies

$$\left(\frac{\partial u(t)}{\partial t}, v_h\right) + b_h(u(t), v_h) + a_h^\Theta(u(t), v_h) + \alpha J_h^\sigma(u(t), v_h) = \ell_h^\Theta(u(t), v_h)\,(t) \qquad (13)$$
$$\forall\, v_h \in S_{hp} \;\forall\, t \in (0, T),$$

4. Error analysis

To carry out the error analysis we need to specify the additional assumptions on mesh, nonlinear diffusion term and regularity of the solution u of the continuous problem. Therefore, we assume that:

(A1) The matrix $I\!K(v) = \{k_{ij}(v)\}_{i,j=1}^2$, $k_{ij}(v) : I\!R \to I\!R$, appearing in the diffusion terms satisfies

$$\begin{aligned} &\text{(a)} \quad \|I\!K(v)\|_\infty \leq C_U < \infty \;\forall\, v \in I\!R, \\ &\text{(b)} \quad \|I\!K(v_1) - I\!K(v_2)\|_\infty \leq C_L|v_1 - v_2| \;\forall\, v_1, v_2 \in I\!R, \\ &\text{(c)} \quad \xi^T I\!K(v)\xi \geq \alpha\|\xi\|^2,\; \alpha > 0,\; \forall\, v \in I\!R,\; \forall\, \xi \in I\!R^2, \end{aligned} \qquad (14)$$

where $\|\cdot\|_\infty$ represents the l^∞-matrix norm, i.e., $\|I\!K\|_\infty = \max_{1\leq i\leq n} \sum_{j=1}^n |k_{ij}|$.

(A2) The weak solution u is sufficiently regular, namely

$$\begin{aligned} &\text{(a)} \quad u \in L^2(0, T; H^{p+1}(\Omega)), \quad \frac{\partial u}{\partial t} \in L^2(0, T; H^p(\Omega)),\; p \geq 1 \\ &\text{(b)} \quad \|\nabla u(t)\|_{L^\infty(\Omega)} \leq C_D \quad \text{for a.a. } t \in (0, T), \end{aligned} \qquad (15)$$

where $p \geq 1$ denotes the given degree of the polynomial approximation.

(A3) The triangulations $\mathcal{T}_h$, $h \in (0, h_0)$ are *locally quasi-uniform* and *shape-regular* (for detailed definitions see [4]).

Now, we are ready to formulate the main result of this paper.

Theorem *Let assumptions (A1) be satisfied, let u be the exact solution of the continuous problem satisfying (A2). Let $\mathcal{T}_h$, $h \in (0, h_0)$ be a family of triangulations satisfying (A3) and let the numerical flux H from (10) be consistent, conservative* *and Lipschitz continuous. Let $u_h \in S_{hp}$ be the solution of the discrete problem given by (8). Then the discretization error $e_h = u_h - u$ satisfies*

$$\max_{t\in[0,T]} \|e_h(t)\|^2_{L^2(\Omega)} + \frac{\alpha}{2}\int_0^T |\!|\!|e_h(\vartheta)|\!|\!|^2 \mathrm{d}\vartheta \le Ch^{2p}, \tag{16}$$

where $|\!|\!|v|\!|\!|^2 := \sum_{K\in\mathcal{T}_h} |v|^2_{H^1(K)} + J^\sigma_h(v,v)$ and $C > 0$ is a constant independent of h.

***Sketch of the proof*:** Let $u \in H^{p+1}(\Omega,\mathcal{T}_h)$ be the solution of the continuous problem. For $v \in L^2(\Omega)$ we denote by $\Pi_h v$ the L^2-projection of v on S_{hp}. We set $\xi(t) = u_h(t) - \Pi_h u(t) \in S_{hp}$, $\eta(t) = \Pi_h u(t) - u(t)$, $e_h(t) = u_h(t) - u(t) = \xi(t) + \eta(t)$ for a.a. $t \in (0,T)$. We subtract (13) from (8), set $v_h := \xi$ and add terms $-a^\Theta_h(\Pi_h u,\xi) + \ell^\Theta_h(\Pi_h u,\xi)$ on both sides of this identity. Then we obtain

$$\begin{aligned}
&\left(\frac{\partial\xi}{\partial t},\xi\right) + \underbrace{a^\Theta_h(u_h(t),\xi) - a^\Theta_h(\Pi_h u,\xi) + \ell^\Theta_h(\Pi_h u,\xi) - \ell^\Theta_h(u_h,\xi) + \alpha J^\sigma_h(\xi,\xi)}_{=:\chi_1}\\
= \quad &\underbrace{-\left(\frac{\partial\eta}{\partial t},\xi\right) + b_h(u,\xi) - b_h(u_h,\xi) - \alpha J^\sigma_h(\eta,\xi)}_{=:\chi_2} \qquad\qquad (17)\\
&+\underbrace{a^\Theta_h(u,\xi) - a^\Theta_h(\Pi_h u,\xi) + \ell^\Theta_h(\Pi_h u,\xi) - \ell^\Theta_h(u,\xi)}_{=:\chi_3}.
\end{aligned}$$

With the aid of the *multiplicative trace inequality*, *inverse inequality* and *approximation properties of the space S_{hp}*, (see [7, Lemmas 4.2–4.4]), we estimate terms χ_1, χ_2 and χ_3:

$$|\chi_2| \le C\Big\{h^p|\partial u/\partial t|_{H^p(\Omega)}\|\xi\|_{L^2(\Omega)} + |\!|\!|\xi|\!|\!|(h^{p+1}|u|_{H^{p+1}(\Omega)} + \alpha h^p|u|_{H^{p+1}(\Omega)} + \|\xi\|_{L^2(\Omega)})\Big\}. \tag{18}$$

Analogously as in [9] we derive

$$|\chi_3| \le C\Big(C_U h^p|u|_{H^{p+1}(\Omega)} + C_D C_L h^{p+1}|u|_{H^{p+1}(\Omega)}\Big)|\!|\!|\xi|\!|\!|. \tag{19}$$

Finally, by a particular choice of the constant C_W we obtain

$$\chi_1 \ge \frac{\alpha}{2}|\!|\!|\xi|\!|\!|^2 - C\left(C_U h^p|u|_{H^{p+1}(\Omega)} + C_D C_L\|\xi\|_{L^2(\Omega)})\right)|\!|\!|\xi|\!|\!|. \tag{20}$$

In consequence, we use inequalities (18)–(20) in identity (17), apply Young's inequality, and integrate from 0 to t, $t \in [0,T]$. Finally application of the Gronwall's lemma leads to desirable error estimate (16). □

5. Conclusion

We presented a space semi-discretization of a scalar nonstationary convection-diffusion equation (with nonlinear convection as well as diffusion) with the aid of SIPG, IIPG and NIPG variants of DGFEM. We presented a priori error estimates which are optimal in the $L^2(0,T,H^1(\Omega))$-seminorm but suboptimal in the $L^\infty(0,T,L^2(\Omega))$-norm.

References

[1] D.N. Arnold, F. Brezzi, B. Cockburn, L.D. Marini: *Unified analysis of discontinuous Galerkin methods for elliptic problems.* SIAM J. Numer. Anal. **39** (2001/02), 1749–1779.

[2] B. Cockburn: *Discontinuous Galerkin methods for convection dominated problems.* In T.J. Barth and H. Deconinck (Eds.), High-Order Methods for Computational Physics, Lecture Notes in Computational Science and Engineering 9, Springer, Berlin, 1999, pp. 69–224.

[3] B. Cockburn, G.E. Karniadakis, C.-W. Shu, (Eds.): *Discontinuous Galerkin methods.* Springer, Berlin, 2000.

[4] V. Dolejší: *Analysis and application of IIPG method to quasilinear nonstationary convection-diffusion problems.* J. Comp. Appl. Math., in press, doi: 10.1016/ j.cam.2007.10.055, 2007.

[5] V. Dolejší, M. Feistauer: *Error estimates of the discontinuous Galerkin method for nonlinear nonstationary convection-diffusion problems.* Numer. Funct. Anal. Optim. **26** (2005), 2709–2733.

[6] V. Dolejší, M. Feistauer, V. Kučera, V. Sobotíková: *An optimal $L^\infty(L^2)$-error estimate for the discontinuous Galerkin approximation of a nonlinear nonstationary convection-diffusion problem.* IMA J. Numer. Anal. **28** (2008), 496–521.

[7] V. Dolejší, M. Feistauer, V. Sobotíková: *Analysis of the discontinuous Galerkin method for nonlinear convection-diffusion problems.* Comput. Methods Appl. Mech. Engrg. **194** (2005), 2709–2733.

[8] M. Feistauer, V. Dolejší, V. Kučera, V. Sobotíková: *An optimal $L^\infty(L^2)$ error estimates for the DG approximation of a nonlinear nonstationary convection-diffusion problem on nonconforming meshes.* *M^2AN*, (submitted).

[9] V. Kučera: *Higher order methods for the solution of compressible flows.* PhD thesis, Charles University Prague, Faculty of Mathematics and Physics, 2008.

[10] B. Rivière, M.F. Wheeler: *A discontinuous Galerkin method applied to nonlinear parabolic equations.* In: B. Cockburn, G.E. Karniadakis, and C.-W. Schu (Eds.), Discontinuous Galerkin methods. Theory, computation and applications., volume 11 of Lect. Notes Comput. Sci. Eng., Springer, Berlin, 2000, 231–244.

FINITE ELEMENT MODELING OF WOOD STRUCTURE[*]

Petr Koňas

Abstract

This work is focused on a weak solution of a coupled physical task of the microwave wood drying process with stress-strain effects and moisture/temperature dependency. Due to the well known weak solutions for the individual physical fields, the author concerns with the coupled stress-strain relation coupled with the moisture and temperature distributions. For the scale dependency the subgrid upscaling method was used. The solved region is assumed to be divided into discontinuous subregions according to the investigated scale. This approach suggests sequential type of solution for highly coupled tasks. This way, very huge structures (huge with regard to the geometry and also physics) can be solved in the reasonable time and with reasonable memory consumptions. Main emphasis was put on evaluation of the structural response of the whole complex. Due to the influence of the moisture, temperature, and time, the coupled physical task of the structural response is solved. Suggested approach is of course usable not only for the structural response, but also for the other physical fields, which were taken into account. The weak solution is based on a slight modification of the Ritz-Galerkin method.

Keywords: FEM, multiphysics, microwave wood drying, upscaling, homogenization

1. Introduction

Microwave drying of wood is one of the most difficult problems of Wood Science. The problem is coupled according to the following variables: moisture w, temperature T, velocity of free water within the conductive wood elements $\mathbf{v}$, intensity of electric field $\mathbf{E}$, intensity of magnetic field $\mathbf{B}$, static pressure p and displacement of structural parts $\mathbf{u}$. Parabolic equations arise as models of many partial physical processes which occur during the drying process. The time-dependency affects most of these processes. Generally, diffusive partial differential equations (PDE) represent usually the base constitutive relationships. The electromagnetic field is sufficiently described by the reduced system of Maxwell's equations. To solve the coupled system, we evaluate the following unknowns: $T, w, p, \mathbf{u}, \mathbf{v}, \mathbf{E}, \mathbf{B}, \mathbf{H}, \mathbf{J}, \mathbf{D}$. These quantities are considered as elements of appropriate Hilbert spaces.

The first set of equations in the coupled system consists of the Maxwell's equations

$$\begin{aligned} \nabla \times \mathbf{E} &= \frac{\partial \mathbf{B}}{\partial t}, & \nabla \cdot \mathbf{D} &= \rho_e, \\ \nabla \times \mathbf{H} &= \mathbf{J} + \frac{\partial \mathbf{D}}{\partial t}, & \nabla \cdot \mathbf{B} &= 0, \end{aligned} \tag{1}$$

[*]This work was supported by grant No. GP106/06/P363 of the Czech Science Foundation.

where t is the time, $\mathbf{B}$ is the magnetic flux density, $\mathbf{D}$ is the electric flux density, $\mathbf{H}$ is the magnetic field intensity, $\mathbf{E}$ is the electric field intensity, $\mathbf{J}$ is the current density, ρ_e is the electric charge density. Due to the anisotropy of wood, we can itemize these variables to $\mathbf{D} = \varepsilon\mathbf{E}$, $\mathbf{B} = \mu\mathbf{H}$, $\mathbf{J} = \sigma\mathbf{E}$, where ε is permittivity, μ is permeability, and σ is the electric conductivity of the material.

Natural requirement for continuous charge is satisfied by the equation of continuity

$$\nabla \cdot \mathbf{J} = -\frac{\partial \rho_e}{\partial t}. \tag{2}$$

However, the electro-dynamical effects are not alone. Also the influence of the moisture and pressure changes in wood should be included. Content of water in the material is obviously separated into free water and water bonded through H-bridges (the chemical bonded water by stronger types of bindings are omitted). Bonded water keeps diffusive character. Interaction of moisture, temperature and static pressure can be described by system of equations (6). Widely disputed is the diffusive character of the static pressure. For this reason, the last equation is often omitted in the following system

$$\begin{aligned}
\rho C\frac{\partial T}{\partial t} - \nabla\mathbf{k_{Tw}}\nabla w - \nabla\mathbf{k_{Tp}}\nabla p - \nabla\mathbf{k_{TT}}\nabla T &= q_{abs} + \mathbf{k_{b_T}}\left(T_{ext} - T\right), \\
\frac{\partial w}{\partial t} - \nabla\mathbf{k_{ww}}\nabla w - \nabla\mathbf{k_{wp}}\nabla p - \nabla\mathbf{k_{wT}}\nabla T &= \mathbf{k_{b_w}}\left(w_{ext} - w\right), \\
\frac{\partial p}{\partial t} - \nabla\mathbf{k_{pw}}\nabla w - \nabla\mathbf{k_{pp}}\nabla p - \nabla\mathbf{k_{pT}}\nabla T &= \mathbf{k_{b_p}}\left(p_{ext} - p\right),
\end{aligned} \tag{3}$$

where ρ is the density, C is the heat capacity, q_{abs} is the density of energy, T_{ext} is the temperature in the surroundings, w is the mass concentration (moisture content), w_{ext} and p_{ext} are moisture content, and static pressure in the surroundings, respectively, $\mathbf{k_{b_w}}, \mathbf{k_{b_T}}, \mathbf{k_{b_p}}$ are convective coefficients and $\mathbf{k_{Tw}}, \mathbf{k_{Tp}}, \mathbf{k_{TT}}, \mathbf{k_{ww}}$, $\mathbf{k_{wp}}, \mathbf{k_{wT}}, \mathbf{k_{pw}}, \mathbf{k_{pp}}$, $\mathbf{k_{pT}}$ are the matrices of diffusion coefficients.

Structural response of the wood structure is described by parabolic equation

$$\begin{aligned}
\rho\frac{\partial \mathbf{u}}{\partial t^2} - \left(\nabla\mathbf{c_{EG}} + (w - w_{ext})\nabla\mathbf{c_{K_{bw}}} + (T - T_{ext})\nabla\mathbf{c_{K_{bT}}}\right)\nabla\mathbf{u} - \nabla\mathbf{c}_{\lambda_{\mathbf{w,T}}}\nabla\frac{\partial \mathbf{u_{vel}}}{\partial t} \\
+ \mathbf{C_w}\cdot w + \mathbf{C_{w^2}}\cdot w^2 + \mathbf{C_T}\cdot T + \mathbf{C_{T^2}}\cdot T^2 + \mathbf{C_{wT}}\cdot wT + \mathbf{C} = \mathbf{F}.
\end{aligned} \tag{4}$$

Definition of individual coefficients for equation (4) was described in [5], where generally orthotropic elastic properties related to moisture and temperature are defined. This described model is valid for diffusive transport of moisture and temperature. It is not appropriate (due to the physical nature of the phenomenon) for the free water movement. This transport is allocated into inter-cellular spaces and cell lumen. Description of this process can be done with Navier-Stokes equation

$$\frac{\partial \nu}{\partial t} + (\nabla \times \nu) \times \nu + \frac{1}{2}\nabla \nu^2 = -\frac{\nabla p^{fl}}{\rho} - \nabla U + \frac{\eta}{\rho}\nabla^2 \nu, \tag{5}$$

where ∇U is the potential and ν the velocity of the fluid.

Since the weak form of equations (1)–(3) and (5) is well known, see e.g. [2], [1], we focus on equation (4), where we will outline the variational formulation for the mixed type of elements in numerical subgrid upscaling method ([8], [6], [7], and [4]). It should be noted that we will suppose the sequential type of mentioned equations. This assumption leads to simplification of equation (4), where T and w are constant in one time-step.

2. Methods

Assembling of the weak form of equation (4) is realized with regard to the schema of the Ritz-Galerkin method by the following quadratic functional, which should be minimal:

$$\begin{aligned} G(\mathbf{u}) = \left(\rho \frac{\partial^2 \mathbf{u}}{\partial t^2}, \xi\right) - \Bigg(\Big(\nabla \mathbf{c}_{\mathbf{EG}} + (w - w_{ext})\, \nabla \mathbf{c}_{\mathbf{K_{bw}}} \\ + (T - T_{ext})\, \nabla \mathbf{c}_{\mathbf{K_{bT}}}\Big)\nabla \mathbf{u}, \xi\Bigg) - \left(\nabla \mathbf{c}_{\lambda_{\mathbf{w},\mathbf{T}}} \nabla \frac{\partial \mathbf{u}}{\partial t}, \xi\right) \\ - 2\left(\mathbf{F} - \left(\mathbf{C}_w \cdot w + \mathbf{C}_{w^2} \cdot w^2 + \mathbf{C}_T \cdot T + \mathbf{C}_{T^2} \cdot T^2 + \mathbf{C}_{wT} \cdot wT + \mathbf{C}\right), \xi\right) = 0. \end{aligned} \tag{6}$$

This functional is well-defined for all $\xi \in H(\Omega)$ and $(\cdot, \cdot)$ stands for the scalar product on this Hilbert space. By the above mentioned simplifications, we obtain the integral form. Let the functional equation (6) be defined on the vector space V, which is a finite dimensional subspace of $H(\Omega)$. Let us assume the Ω to be partitioned into a finite number of subregions on very fine scale δ_1. Further, we assume that m_1 of these subregions are covered by mesh on this scale (subgrids). Functional from equation (6) is then considered on vector subspaces $V_1^{\delta_1}, V_2^{\delta_1}, \ldots, V_{m_1}^{\delta_1} \subseteq V$, where $V_j^{\delta_j}$ for $j = 1, \ldots, m_1$ are Raviart-Thomas (RT) spaces. Subspaces may not fill the full space V. It means that $V_1^{\delta_1} \cup V_2^{\delta_1} \cup V_3^{\delta_1} \cup \ldots \cup V_{m_1}^{\delta_1} \equiv V^{\delta_1} \subseteq V$. Simultaneously, we declare mentioned vector subspaces with bases

$$\left\{\varphi_{V_j,1}^{\delta_j}, \varphi_{V_j,2}^{\delta_j}, \ldots, \varphi_{V_j,n_1}^{\delta_j}\right\} \subseteq V_j^{\delta_j}, \quad j = 1, 2, \ldots, m_1.$$

Complete basis

$$\varphi^{\delta_1} \equiv \bigcup_{j=1}^{m_1} \left\{\varphi_{V_j,1}^{\delta_1}, \varphi_{V_j,2}^{\delta_1}, \ldots, \varphi_{V_j,n_j}^{\delta_1}\right\} \subset V^{\delta_1}.$$

on vector space V^{δ_1} is derived from the fine mesh of subgrids, where linear basis functions are used. Similarly, let us partition Ω by further linear meshes $\Psi_{\delta_2}, \Psi_{\delta_3}, \ldots, \Psi_{\delta_i}$

for different scales $\delta_1 < \delta_2 < \ldots < \delta_i$, where again regions $m_2, m_3, \ldots, m_i$ cover some parts of Ω on the specific scale. Consequently, similar vector subspaces can be distinguished $V_1^{\delta_k}, V_2^{\delta_k}, \ldots, V_{m_k}^{\delta_k} \subseteq V$, $k = 2, 3, \ldots, i$ with the same requirements:

$$\begin{aligned} V_1^{\delta_2} \cup V_2^{\delta_2} \cup V_3^{\delta_2} \cup \ldots \cup V_{m_2}^{\delta_2} &\equiv V^{\delta_2} \subseteq V, \\ V_1^{\delta_3} \cup V_2^{\delta_3} \cup V_3^{\delta_3} \cup \ldots \cup V_{m_3}^{\delta_3} &\equiv V^{\delta_3} \subseteq V, \\ &\vdots \\ V_1^{\delta_i} \cup V_2^{\delta_i} \cup V_3^{\delta_i} \cup \ldots \cup V_{m_i}^{\delta_i} &\equiv V^{\delta_i} \subseteq V. \end{aligned} \tag{7}$$

In addition, we will tie subspaces by these important rules:

$$V^{\delta_1} \subseteq V^{\delta_2} \subseteq V^{\delta_3} \subseteq \ldots \subseteq V^{\delta_i} \equiv V, \tag{8}$$

where δ_i is maximal scale $V^{\delta_i} \equiv V$ and V is defined on the entire Ω.

All unknown functions can be decomposed to individual scales, e.g.:

$$\mathbf{u} = \mathbf{u}^{\delta_1} + \mathbf{u}^{\delta_2} + \ldots + \mathbf{u}^{\delta_i}, \tag{9}$$

on some Ω_0. This decomposition of unknowns to individual scales affects the solution in the sense of finite elements and the minimization of functional equation (6) does not provide the common appearance of Ritz system.

Let us consider PDE $A\mathbf{u} = f$, $\mathbf{u} \in V$ with a differential operator A and let us follow the common steps in the solution of this task for multi-scale problems.

Functional which will be minimized has the standard form ([3]):

$$G(\mathbf{u}) = (\mathbf{u}, \mathbf{u})_A - 2\,(\mathbf{f}, \mathbf{u})\,. \tag{10}$$

Equation (9) will be substituted into the first part of equation (10):

$$L(\mathbf{u}) = \left(\mathbf{u}^{\delta_1} + \mathbf{u}^{\delta_2} + \ldots + \mathbf{u}^{\delta_i}, \mathbf{u}^{\delta_1} + \mathbf{u}^{\delta_2} + \ldots + \mathbf{u}^{\delta_i}\right)_A. \tag{11}$$

It can be expanded due to the rules of the bilinear form in the following way:

$$L(\mathbf{u}) = \sum_{k=1}^{i} \sum_{j=1}^{i} \left(\mathbf{u}^{\delta_k}, \mathbf{u}^{\delta_j}\right)_A. \tag{12}$$

Our problem is reduced into the task of finding such function $\mathbf{u}$ which minimizes the functional (10). This functional is minimized by a function in the form:

$$\widetilde{\mathbf{u}}_n^{\delta_j} = \sum_{k=1}^{s_j} b_k^{\delta_j} \varphi_k^{\delta_j}, \tag{13}$$

where $s_j = \dim V^{\delta_j}$, $\varphi_k^{\delta_j}$ denote the basis functions in V^{δ_j} and $b_k^{\delta_j}$ represent variables which will be evaluated in the point of the approximate minimum $a_k^{\delta_j}$. As the first step, we estimate the functional in the subgrid on the scale δ_1. The minimizing function is denoted as:

$$\mathbf{u}_n^{\delta_j} = \sum_{k=1}^{s_j} a_k^{\delta_j} \varphi_k^{\delta_j}. \tag{14}$$

To evaluate the coefficients $a_k^{\delta_j}$, we compute the partial derivatives of $L(\mathbf{u}_n)$ with respect to all coefficients on all scales and equal them to zero:

$$\frac{\partial L(\mathbf{u}_n)}{\partial b_l^{\delta_k}}\Big|_{\mathbf{b}_{\delta_1}=\mathbf{a}_{\delta_1},\ldots,\mathbf{b}_{\delta_i}=\mathbf{a}_{\delta_i}} \quad \text{for } k = 1,\ldots,i;\ l = 1,\ldots,s_k, \tag{15}$$

where $\mathbf{a}_{\delta_j} = \left(a_1^{\delta_j},\ldots,a_{s_j}^{\delta_j}\right)^T$ and $\mathbf{b}_{\delta_j} = \left(b_1^{\delta_j},\ldots,b_{s_j}^{\delta_j}\right)^T$ for $j = 1,\ldots,i$. The partial derivatives of $L(\mathbf{u}_n)$ with respect to coefficients $\mathbf{b}_{\delta_k}$ can be computed as follows

$$\frac{\partial\left(\widetilde{\mathbf{u}}_n^{\delta_j}, \widetilde{\mathbf{u}}_n^{\delta_k}\right)_A}{\partial b_1^{\delta_k}\ldots\partial b_{s_k}^{\delta_k}} = \mathbf{R}^L_{A_{\delta_j\delta_k}}\mathbf{a}_{\delta_j} \quad \text{for } j \neq k, \tag{16}$$

where the symbol on the left-hand side stands for the vector of partial derivatives with respect to $b_1^{\delta_k},\ldots,b_{s_k}^{\delta_k}$ and $\mathbf{R}^L_{A_{\delta_j\delta_k}}$ denotes the modified lower triangular matrix of Ritz system:

$$\mathbf{R}^L_{A_{\delta_j\delta_k}} = \begin{pmatrix} \left(\varphi_1^{\delta_j},\varphi_1^{\delta_k}\right)_A & 0 & \cdots & 0 \\ 2\left(\varphi_1^{\delta_j},\varphi_2^{\delta_k}\right)_A & \left(\varphi_2^{\delta_j},\varphi_2^{\delta_k}\right)_A & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 2\left(\varphi_1^{\delta_j},\varphi_{s_k}^{\delta_k}\right)_A & 2\left(\varphi_2^{\delta_j},\varphi_{s_k}^{\delta_k}\right)_A & \cdots & \left(\varphi_{s_j}^{\delta_j},\varphi_{s_k}^{\delta_k}\right)_A \end{pmatrix}. \tag{17}$$

Similarly, partial derivatives of the first part of equation (10) with respect to coefficients $\mathbf{b}_{\delta_j}$ can be computed as

$$\frac{\partial\left(\widetilde{\mathbf{u}}_n^{\delta_j}, \widetilde{\mathbf{u}}_n^{\delta_k}\right)_A}{\partial b_1^{\delta_j}\ldots\partial b_{s_j}^{\delta_j}} = \mathbf{R}^U_{A_{\delta_j\delta_k}}\mathbf{a}_{\delta_k} \quad \text{for } j \neq k, \tag{18}$$

where $\mathbf{R}^U_{A_{\delta_j\delta_k}}$ is modified upper triangular matrix of Ritz system:

$$\mathbf{R}^U_{A_{\delta_j\delta_k}} = \begin{pmatrix} \left(\varphi_1^{\delta_j},\varphi_1^{\delta_k}\right)_A & 2\left(\varphi_1^{\delta_j},\varphi_2^{\delta_k}\right)_A & \cdots & 2\left(\varphi_1^{\delta_j},\varphi_{s_k}^{\delta_k}\right)_A \\ 0 & \left(\varphi_2^{\delta_j},\varphi_2^{\delta_k}\right)_A & \cdots & 2\left(\varphi_2^{\delta_j},\varphi_{s_k}^{\delta_k}\right)_A \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \left(\varphi_{s_j}^{\delta_j},\varphi_{s_k}^{\delta_k}\right)_A \end{pmatrix}. \tag{19}$$

Finally, partial derivatives of the first part of equation (10) with respect to coefficients $\mathbf{b}_{\delta_j}$ on the same scale δ_j are given by:

$$\frac{\partial\left(\widetilde{\mathbf{u}}_n^{\delta_j}, \widetilde{\mathbf{u}}_n^{\delta_j}\right)_A}{\partial b_1^{\delta_j} \ldots \partial b_{s_j}^{\delta_j}} = \mathbf{R}_{A_{\delta_j \delta_j}} \mathbf{a}_{\delta_j}, \tag{20}$$

where $\mathbf{R}_{A_{\delta_j \delta_j}}$ is the well known matrix of the Ritz system:

$$\mathbf{R}_{A_{\delta_j \delta_j}} = 2 \begin{pmatrix} \left(\varphi_1^{\delta_j}, \varphi_1^{\delta_j}\right)_A & \left(\varphi_1^{\delta_j}, \varphi_2^{\delta_j}\right)_A & \cdots & \left(\varphi_1^{\delta_j}, \varphi_{s_k}^{\delta_j}\right)_A \\ \left(\varphi_1^{\delta_j}, \varphi_2^{\delta_j}\right)_A & \left(\varphi_2^{\delta_j}, \varphi_2^{\delta_j}\right)_A & \cdots & \left(\varphi_2^{\delta_j}, \varphi_{s_j}^{\delta_j}\right)_A \\ \vdots & \vdots & \ddots & \vdots \\ \left(\varphi_1^{\delta_j}, \varphi_{s_j}^{\delta_j}\right)_A & \left(\varphi_2^{\delta_j}, \varphi_{s_j}^{\delta_j}\right)_A & \cdots & \left(\varphi_{s_j}^{\delta_j}, \varphi_{s_j}^{\delta_j}\right)_A \end{pmatrix}. \tag{21}$$

3. Results and discussion

The main result of this work is the minimization of the quadratic functional with respect to the mentioned subspace division using the common Ritz system and triangular matrices of the Ritz system. Detail reordering of individual equations is beyond the scope of this contribution and also huge and inappropriate for the proceedings. For this reason, just final results are introduced. Using the above mentioned relations, the partial derivatives (15) can be expressed in the following form

$$\begin{aligned} \frac{\partial L(\mathbf{u}_n)}{\partial b_1^{\delta_1} \ldots \partial b_{s_1}^{\delta_1}} &= \mathbf{R}_{A_{\delta_1 \delta_1}} \mathbf{a}_{\delta_1} + 2\mathbf{R^U}_{A_{\delta_1 \delta_2}} \mathbf{a}_{\delta_2} + \ldots + 2\mathbf{R^U}_{A_{\delta_1 \delta_i}}, \\ \frac{\partial L(\mathbf{u}_n)}{\partial b_1^{\delta_2} \ldots \partial b_{s_2}^{\delta_2}} &= 2\mathbf{R^L}_{A_{\delta_1 \delta_2}} \mathbf{a}_{\delta_1} + \mathbf{R}_{A_{\delta_2 \delta_2}} \mathbf{a}_{\delta_2} + 2\mathbf{R^U}_{A_{\delta_2 \delta_3}} \mathbf{a}_{\delta_3} + \ldots + 2\mathbf{R^U}_{A_{\delta_2 \delta_i}} \mathbf{a}_{\delta_i}, \\ \frac{\partial L(\mathbf{u}_n)}{\partial b_1^{\delta_3} \ldots \partial b_{s_3}^{\delta_3}} &= 2\mathbf{R^L}_{A_{\delta_1 \delta_3}} \mathbf{a}_{\delta_1} + 2\mathbf{R^L}_{A_{\delta_2 \delta_3}} \mathbf{a}_{\delta_2} + \mathbf{R}_{A_{\delta_3 \delta_3}} \mathbf{a}_{\delta_3} + 2\mathbf{R^U}_{A_{\delta_3 \delta_4}} \mathbf{a}_{\delta_4} + \ldots \\ &\qquad \ldots + 2\mathbf{R^U}_{A_{\delta_3 \delta_i}} \mathbf{a}_{\delta_i}, \\ &\vdots \\ \frac{\partial L(\mathbf{u}_n)}{\partial b_1^{\delta_i} \ldots \partial b_{s_i}^{\delta_i}} &= 2\mathbf{R^L}_{A_{\delta_1 \delta_i}} \mathbf{a}_{\delta_1} + 2\mathbf{R^L}_{A_{\delta_2 \delta_i}} \mathbf{a}_{\delta_2} + \ldots + \mathbf{R}_{A_{\delta_i \delta_i}} \mathbf{a}_{\delta_i}. \end{aligned} \tag{22}$$

This complex system can be rewritten in more readable form (T means vector transposition):

$$S_A(\mathbf{u}_n) = \mathbf{R}_{A_{\delta_j \delta_j}} \mathbf{a}_{\delta_j} + 2 \sum_{k=j+1}^{i} \mathbf{R^U}_{A_{\delta_j \delta_k}} \mathbf{a}_{\delta_k} + 2 \sum_{k=1}^{j-1} \mathbf{R^L}_{A_{\delta_k \delta_j}} \mathbf{a}_{\delta_k}, \tag{23}$$

where $S_A(\mathbf{u}_n) = \left(\frac{\partial L(\mathbf{u}_n)}{\partial b_1^{\delta_j}}, \frac{\partial L(\mathbf{u}_n)}{\partial b_2^{\delta_j}}, \ldots, \frac{\partial L(\mathbf{u}_n)}{\partial b_{s_j}^{\delta_j}} \right)^T$. In this way, we obtained a numerical approximation of the first part of equation (10).

Minimization of functional equation (10) is done by the relation:

$$\frac{\partial L(\mathbf{u}_n)}{\partial b_l^{\delta_k}} \Big|_{\mathbf{b}_{\delta_1}=\mathbf{a}_{\delta_1},\ldots,\mathbf{b}_{\delta_i}=\mathbf{a}_{\delta_i}} = 0 \quad \text{for } k = 1, \ldots, i;\ l = 1, \ldots, s_k. \tag{24}$$

Thus, the approximate solution $\mathbf{u}_n$ can be computed by evaluation of $\mathbf{a}_{\delta_j}$ in the following equation:

$$S_A(\mathbf{u}_n) = \left(\left(f, \varphi_1^{\delta_j} \right), \left(f, \varphi_2^{\delta_j} \right), \ldots, \left(f, \varphi_{s_j}^{\delta_j} \right) \right)^T, \tag{25}$$

where $S_A(\mathbf{u}_n)$ is given by (23).

By analogy, the solution of equation (6) with applying of $S_A(\mathbf{u}_n)$ derivation can be rewritten in this form:

$$S_A(\mathbf{u}_n) - S_B(\mathbf{u}_n) - S_C(\mathbf{u}_n) = 2 \left(\left(f_c, \varphi_1^{\delta_j} \right), \left(f_c, \varphi_2^{\delta_j} \right), \ldots, \left(f_c, \varphi_{s_j}^{\delta_j} \right) \right)^T, \tag{26}$$

where we use the following differential operators

$$\begin{aligned}
S_A(\mathbf{u}) &= \rho \frac{\partial^2}{\partial t^2} \mathbf{u}, \\
S_B(\mathbf{u}) &= \left(\nabla \mathbf{c}_{\mathbf{EG}} + (w - w_{ext}) \nabla \mathbf{c}_{\mathbf{K_{bw}}} + (T - T_{ext}) \nabla \mathbf{c}_{\mathbf{K_{bT}}} \right) \nabla \mathbf{u}, \\
S_C(\mathbf{u}) &= \nabla \mathbf{c}_{\lambda_{\mathbf{w,T}}} \nabla \frac{\partial}{\partial t} \mathbf{u},
\end{aligned}$$

and $f_c = \mathbf{F} - \left(\mathbf{C_w} \cdot w + \mathbf{C_{w^2}} \cdot w^2 + \mathbf{C_T} \cdot T + \mathbf{C_{T^2}} \cdot T^2 + \mathbf{C_{wT}} \cdot w \cdot T + \mathbf{C} \right)$.

If finite elements with linear basis functions are used, then system equation (26) is uniquely solvable. Solution is realized in i consequent steps. In the first step, equation (26) is formed for $j = 1$. Since the results of higher scales are unknown (in Ritz or modified Ritz system), the solution on higher scales in individual nodes is expressed by value of $\mathbf{a}_{\delta_1}$ or other appropriate lower scales. From this step, we obtain suitable extrapolation in some nodes on higher scale(s) which include the region of element on this solved scale. In the next step, we calculate the same equation, but on the following higher scale. At the same time, some nodes on this scale are strictly derived from previous step. This idea is repeated until the highest scale is reached.

4. Conclusions

Advantage of this type of solution is the null requirement of results enumeration on lower scales. Simultaneously, just results on last scale can be enumerated, whereas results on the scale are derived from lower scales. The solution can be simplified by this statement:

If a position of a node for higher scale is in some region of a lower scale mesh, then $\mathbf{a}_{\delta_{\mathbf{j-k}}}$ can be mapped directly to results on higher scale a_{δ_j}, $\left(a_{\delta_{j-k}} \rightarrow a_{\delta_j}\right)$. Let each node of element on some higher scale E^{δ_j} coincides with node in element on lower scale $E^{\delta_{j-k}}$. All contributions of higher scales $\mathbf{a}_{\delta_{\mathbf{j>1}}}$ to subgrid can be derived from consequent mapping of $\mathbf{a}_{\delta_{\mathbf{1}}}, \mathbf{a}_{\delta_{\mathbf{2}}}, \ldots, \mathbf{a}_{\delta_{\mathbf{j-1}}}$ to required $\mathbf{a}_{\delta_{\mathbf{j}}}$.

5. Summary

The weak solution of coupled stress-strain task with moisture/temperature dependency of material model was obtained in this project. The subgrid upscaling homogenization method for large scale hierarchical structure, which is typical for wood structure, was used. Modified Ritz-Galerkin method for simple solution was derived. The coefficient form of the PDE suitable for nowadays numerical solvers was used ([5]). Suggested weak solution offers unique and relatively accurate solution of large scale problems with dependency on low scale. The solution is very general and slight modification of the approach allows solution of a lot of common tasks in the field of bio-mechanics.

References

[1] G.A. Kriegsmann: *Hot spot formation in microwave heated ceramic fibres*, IMA Journal of Applied Mathematics **59** (1997), no. 2, 123–148.

[2] J. Bodig, B.A. Jayne: *Mechanics of wood and wood composites*, Van Nostrand Reinhold, New York, 1982.

[3] K. Rektorys: *Variační metody v inženýrských problémech a v problémech matematické fyziky*, Academia, Praha, 1999.

[4] O. Korostyschevskaya, S.E. Minkoff: *A matrix analysis of operator-based upscaling for the wave equation*, SIAM Journal of Numerical Analysis **44** (2006), no. 2, 586–612.

[5] P. Koňas: *General model of wood in typical coupled tasks*, ACTA Universitatis agriculturae et silviculturae Mendelianae Brunensis **LVI** (2008), no. 4, 10–27.

[6] T. Arbogast: *Numerical subgrid upscaling of two-phase flow in porous media*, Lecture Notes in Physics (2000), no. 2, 250–260.

[7] T. Arbogast, S. Bryant: *A two-scale numerical subgrid technique for waterflood simulations*, Lecture Notes in Physics (2002), no. 27, 446–457.

[8] T. Arbogast, S. Minkoff, P. Keenan: *An operator-based approach to upscaling the pressure equation*, Computational Methods in Water Resource **XII** (1998), no. 1, 405–412.

INTERIOR POINT ALGORITHMS FOR 3D CONTACT PROBLEMS

Radek Kučera[*], Jitka Machalová, Pavel Ženčák

1. Introduction

We shall be concerned with solving

$$\begin{array}{ll} \text{minimize} & \tfrac{1}{2}\, x^\top A x - x^\top b, \\ \text{subject to} & x_{1,i} \geq l_i,\ x_{2,i}^2 + x_{3,i}^2 \leq g_i^2,\ i = 1, \ldots, m, \\ & x = (x_1^\top, x_2^\top, x_3^\top)^\top \in \mathbb{R}^n, \end{array} \tag{1}$$

where $|x_1| = |x_2| = |x_3| = m$, $n = 3m$, $A \in \mathbb{R}^{n\times n}$ is the symmetric, positive definite Hessian matrix, $b \in \mathbb{R}^n$, and $l, g \in \mathbb{R}^m$. This problem arises, e.g., in duality based methods for the solution of 3D contact problems of linear elasticity with Tresca friction. As a widely used approach of contact problems with (more realistic) Coulomb friction is based on a sequence of Tresca friction problems [2], an efficient solver for (1) is of crucial importance. In this contribution we shall test algorithms based on an "interior point" idea.

2. Description of algorithms

The solution to (1) exists and it is necessarily unique. We denote it by x^*. It is well-known [1] that x^* is fully determined by the Karush-Kuhn-Tucker (KKT) conditions. The basic idea of interior point methods consists in applying Newton iterations to equalities in the KKT conditions while inequalities are satisfied strictly by damping Newton steps.

Let us introduce the Lagrangian $\mathcal{L} : \mathbb{R}^n \times \mathbb{R}^{2m} \mapsto \mathbb{R}$ associated with (1) by

$$\mathcal{L}(x, \lambda, \mu) = \frac{1}{2} x^\top A x - x^\top b + \lambda^\top (l - x_1) + \mu^\top (X_2^2 + X_3^2 - G^2) e,$$

where $X_2, X_3, G \in \mathbb{R}^{m\times m}$ are defined by $X_2 = diag(x_2)$, $X_3 = diag(x_3)$, $G = diag(g)$, and $e = (1, \ldots, 1)^\top \in \mathbb{R}^m$. There is $y^* := (\lambda^*, s^*, \mu^*, d^*) \in \mathbb{R}^{4m}$ so that the pair (x^*, y^*) is the unique solution to the following system:

$$\frac{\partial \mathcal{L}}{\partial x}(x, \lambda, \mu) = 0,\ \frac{\partial \mathcal{L}}{\partial \lambda}(x, \lambda, \mu) + s = 0,\ \lambda^\top s = 0,\ \frac{\partial \mathcal{L}}{\partial \mu}(x, \lambda, \mu) + d = 0,\ \mu^\top d = 0, \tag{2}$$

$$\lambda \geq 0,\ s \geq 0,\ \mu \geq 0,\ d \geq 0. \tag{3}$$

[*] Supported by grants GAČR 101/08/0574, and MSM6198910027.

Here, λ, and μ are the *Lagrange multipliers* while s, and d are the *slack variables.* The classical KKT conditions can be derived from (2), (3) by eliminating the slack variables.

Let us divide A, b into blocks A_{ij}, b_i, $i, j \in \{1, 2, 3\}$ consistently with the partition of x onto x_1, x_2, x_3. We can equivalently rewrite (2), (3) as

$$F(x, y) = 0, \quad y \geq 0, \tag{4}$$

where $y := (\lambda^\top, s^\top, \mu^\top, d^\top)^\top$, and $F : \mathbb{R}^{n+4m} \mapsto \mathbb{R}^{n+4m}$,

$$F(x, y) := \left(\begin{array}{l} A_{11}x_1 + A_{12}x_2 + A_{13}x_3 - \lambda - b_1 \\ A_{21}x_1 + (A_{22} + 2M)x_2 + A_{23}x_3 - b_2 \\ A_{31}x_1 + A_{32}x_2 + (A_{33} + 2M)x_3 - b_3 \\ \hline -x_1 + s + l \\ \Lambda S e \\ (X_2^2 + X_3^2 - G^2)e + d \\ MDe \end{array} \right)$$

with $\Lambda, S, M, D \in \mathbb{R}^{m \times m}$, $\Lambda = diag(\lambda)$, $S = diag(s)$, $M = diag(\mu)$, $D = diag(d)$. The Jacobi matrix to F reads as follows:

$$J(x, y) = \left(\begin{array}{rrr|rrrr} A_{11} & A_{12} & A_{13} & -I & 0 & 0 & 0 \\ A_{21} & A_{22} + 2M & A_{23} & 0 & 0 & 2X_2 & 0 \\ A_{31} & A_{32} & A_{33} + 2M & 0 & 0 & 2X_3 & 0 \\ \hline -I & 0 & 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & S & \Lambda & 0 & 0 \\ 0 & 2X_2 & 2X_3 & 0 & 0 & 0 & I \\ 0 & 0 & 0 & 0 & 0 & D & M \end{array} \right). \tag{5}$$

Let $(x^{(k)}, y^{(k)})$, $y^{(k)} > 0$ be a known approximation of (x^*, y^*). The Newton direction is the solution to the linear system

$$J(x^{(k)}, y^{(k)}) \left(\begin{array}{c} \Delta x^{(k+1)} \\ \Delta y^{(k+1)} \end{array} \right) = -F(x^{(k)}, y^{(k)}), \tag{6}$$

and the new iterate is

$$(x^{(k+1)}, y^{(k+1)}) = (x^{(k)}, y^{(k)}) + \alpha^{(k)} (\Delta x^{(k+1)}, \Delta y^{(k+1)}), \tag{7}$$

where $\alpha^{(k)} = \min_{\Delta y_i^{(k+1)} < 0} \{1, -\delta y_i^{(k)} / \Delta y_i^{(k+1)}\}$ with $\delta \in (0, 1]$ providing $y^{(k+1)} > 0$ (typically $\delta = 0.999$).

The computations based on (6), (7) can take short steps before violating $y^{(k+1)} > 0$ so that the convergence rate can be slow. Therefore we use two modifications [7] called *path following method*, and (Mehrotra's) *predictor-corrector method* keeping iterates deeper in the feasible region so that it enables us to perform longer steps.

Let us replace (4) by

$$F(x,y) = (0^\top, 0^\top, 0^\top, 0^\top, \tau e^\top, 0^\top, \tau e^\top)^\top, \quad y > 0, \tag{8}$$

where $\tau > 0$. Solutions (x^τ, y^τ) to (8) define in $\mathbb{R}^n \times \mathbb{R}_+^{4m}$ a curve $\mathcal{C}(\tau)$ called *central path* that leads to (x^*, y^*) as τ tends to zero. The next algorithm combines Newton iterations for the equation in (8) with changes of τ so that the (modified) Newton steps follow $\mathcal{C}(\tau)$.

Algorithm PF: Given $x^{(0)} \in \mathbb{R}^n$, $y^{(0)} \in \mathbb{R}_+^{4m}$, $\sigma_l, \sigma_q, \delta \in [0,1]$, and $\epsilon \geq 0$. Set $k := 0$.

(1) Compute $\beta_l^{(k)} = {\lambda^{(k)}}^\top s^{(k)}/m$, $\beta_q^{(k)} = {\mu^{(k)}}^\top d^{(k)}/m$, and $\tau_l^{(k)} = \sigma_l \beta_l^{(k)}$, $\tau_q^{(k)} = \sigma_q \beta_q^{(k)}$.

Solve

$$J(x^{(k)}, y^{(k)}) \begin{pmatrix} \Delta x^{(k+1)} \\ \Delta y^{(k+1)} \end{pmatrix} = -F(x^{(k)}, y^{(k)}) + (0^\top, 0^\top, 0^\top, 0^\top, \tau_l^{(k)} e^\top, 0^\top, \tau_q^{(k)} e^\top)^\top \tag{9}$$

and generate $(x^{(k+1)}, y^{(k+1)})$ by (7).

(2) If $\|(\Delta x^{(k+1)}, \Delta y^{(k+1)})\|_{\mathbb{R}^{n+4m}} \leq \epsilon$, return $(\bar{x}, \bar{y}) = (x^{(k+1)}, y^{(k+1)})$, else set $k := k+1$, and go to step (1).

The parameters $\beta_l^{(k)}, \beta_q^{(k)}$, and σ_l, σ_q are called *duality measures*, and *centering parameters*, respectively. Let us note that $\sigma_l = \sigma_q = 0$ reduces Algorithm PF to the standard (damped) Newton method. Our choices of σ_l, and σ_q are based on the rule proposed in [4]:

$$\sigma_l = (\min\{2 \cdot 10^{-3}, 5 \cdot 10^{-5}(1-\xi_l)/\xi_l\})^3,$$

where $\xi_l = \min_{i=1,\ldots,m}\{\lambda_i^{(k)} s_i^{(k)}\}/\beta_l^{(k)}$, and analogously for σ_q.

The second algorithm calculates centering parameters adaptively using second order information (curvature) of the central path $\mathcal{C}(\tau)$. First, in the predictor stage, we compute duality measures β_l^P, β_q^P for the longest step of the (standard) Newton direction. Then, in the corrector stage, we set σ_l, σ_q near 0, when the good progress along the predicted direction is made or near 1 conversely.

Algorithm PC: Given $x^{(0)} \in \mathbb{R}^n$, $y^{(0)} \in \mathbb{R}_+^{4m}$, $\delta \in (0,1)$, and $\epsilon \geq 0$. Set $k := 0$.

(1) Solve

$$J(x^{(k)}, y^{(k)}) \begin{pmatrix} \Delta x^P \\ \Delta y^P \end{pmatrix} = -F(x^{(k)}, y^{(k)}),$$

compute $\alpha^P = \min_{\Delta y_i^P < 0}\{1, -y_i^{(k)}/\Delta y_i^P\}$, and

$$\begin{aligned} \beta_l^P &= (\lambda^{(k)} + \alpha^P \Delta\lambda^P)^\top (s^{(k)} + \alpha^P \Delta s^P)/m, \\ \beta_q^P &= (\mu^{(k)} + \alpha^P \Delta\mu^P)^\top (d^{(k)} + \alpha^P \Delta d^P)/m. \end{aligned}$$

(2) Set $\sigma_l = \left(\beta_l^P/\beta_l^{(k)}\right)^3$, $\sigma_q = \left(\beta_q^P/\beta_q^{(k)}\right)^3$, compute $(\Delta x^{(k+1)}, \Delta y^{(k+1)})$ solving (9) with the right-hand-side replaced by

$$-F(x^{(k)}, y^{(k)}) + (0^\top, 0^\top, 0^\top, 0^\top, -e^\top \Delta\Lambda^P \Delta S^P + \tau_l^{(k)} e^\top, 0^\top, -e^\top \Delta M^P \Delta D^P e + \tau_q^{(k)} e^\top)^\top,$$

and generate $(x^{(k+1)}, y^{(k+1)})$ by (7).

(3) If $\|(\Delta x^{(k+1)}, \Delta y^{(k+1)})\|_{\mathbb{R}^{n+4m}} \leq \epsilon$, return $(\bar{x}, \bar{y}) = (x^{(k+1)}, y^{(k+1)})$, else set $k := k+1$, and go to step (1).

3. Numerical experiments

3.1. Model problem

Let us consider a steel brick in $\mathbb{R}^3$ lying on a rigid foundation. The brick occupies the domain $\Omega = (0,3) \times (0,1) \times (0,1)$, whose boundary $\partial\Omega$ split into three nonempty disjoint parts $\Gamma_u = \{0\} \times (0,1) \times (0,1)$, $\Gamma_c = (0,3) \times (0,1) \times \{0\}$, and $\Gamma_p = \partial\Omega \backslash (\bar{\Gamma}_u \cup \bar{\Gamma}_c)$ with different boundary conditions. The zero displacements are prescribed on Γ_u, whereas the surface tractions act on Γ_p. On Γ_c we consider the contact conditions, i.e., the non-penetration, and the effect of friction. The elastic behavior of the brick is described by Lamé equations that, after finite element discretization, lead to a symmetric positive definite stiffness matrix $K \in \mathbb{R}^{3n_c \times 3n_c}$ and to a load vector $f \in \mathbb{R}^{3n_c}$. Moreover, we introduce full rank matrices $N, T_1, T_2 \in \mathbb{R}^{m \times 3n_c}$ projecting displacements at contact nodes to normal and tangential directions, respectively, and we denote $B = \left(N^\top, T_1^\top, T_2^\top\right)^\top \in \mathbb{R}^{3m \times 3n_c}$. For more details about this model problem we refer to [2].

Here, we shall use the dual formulation in terms of contact stresses. Considering only *Tresca friction*, our model problem reduces directly to (1), where $A = BK^{-1}B^\top$, $b = BK^{-1}f$, $l = 0$, and $g_i \geq 0$ are given slip bound values at contact nodes. Let us note that unknowns x_1, and x_2, x_3 represent the normal, and tangential contact stresses, respectively.

3.2. Inner solver

Our algorithms require repeatedly to solve the linear systems

$$J(x,y) \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \begin{pmatrix} r_x \\ r_y \end{pmatrix} \tag{10}$$

with the Jacobi matrix given by (5), and $r_x \in \mathbb{R}^n$, $r_y = (r_\lambda^\top, r_s^\top, r_\mu^\top, r_d^\top)^\top \in \mathbb{R}^{4m}$, $\Delta y = (\Delta\lambda^\top, \Delta s^\top, \Delta\mu^\top, \Delta d^\top)^\top \in \mathbb{R}^{4m}$. First we compute the solution to the reduced system arising from (10) by eliminating increments with respect to the slack variables:

$$J_R(x,y) \begin{pmatrix} \Delta x \\ \Delta z \end{pmatrix} = \begin{pmatrix} r_x \\ r_z \end{pmatrix}, \tag{11}$$

where

$$J_R(x,y) = \left(\begin{array}{ccc|cc} A_{11} & A_{12} & A_{13} & -I & 0 \\ A_{21} & A_{22}+2M & A_{23} & 0 & 2X_2 \\ A_{31} & A_{32} & A_{33}+2M & 0 & 2X_3 \\ \hline -I & 0 & 0 & -\Lambda^{-1}S & 0 \\ 0 & 2X_2 & 2X_3 & 0 & -M^{-1}D \end{array} \right),$$

and $r_z = (r_\lambda^\top - r_s^\top \Lambda^{-1}, r_\mu^\top - r_d^\top M^{-1})^\top \in \mathbb{R}^{2m}$, $\Delta z = (\Delta\lambda^\top, \Delta\mu^\top)^\top \in \mathbb{R}^{2m}$. Then we obtain the eliminated components by $\Delta s = \Lambda^{-1}(r_s - S\Delta\lambda)$, and $\Delta d = M^{-1}(r_d - D\Delta\mu)$. It is easy to prove that the Schur complement with respect to the second diagonal block in $J_R(x,y)$ is positive definite provided $y > 0$. Therefore $J_R(x,y)$ is non-singular but indefinite. In order to solve (11), one can use direct [6] or iterative methods. In this paper we apply the conjugate gradient method with an appropriate indefinite preconditioning [5].

3.3. Tests

We compare the algorithms PF, and PC with the one presented in [3], here denoted by QPC. For various numbers of the primal, and dual degrees of freedoms ($3n_c/3m$), we report the computational time (*time*), the total number of the matrix-vector multiplications (n_A), and, in case of the interior point algorithms, the number of the outer iterations (*out*), and the number of the full steps (*full*), i.e. the steps with $\alpha^{(k)} = 1$. All computations are performed by Matlab 7 on Pentium 4, 2.8 GHz with 1GB RAM.

The first experiments in Table 1 demonstrate the computational strategy in which the Hessian matrix A is assembled (only in PF, and PC). As the time consumed by assembling A predominates for larger $3n_c/3m$, this strategy seems to be non-acceptable form more realistic contact problems.

In Tables 2, and 3 we present the computational efficiency of PF, and PC with non-assembled A. We test two preconditioners of indefinite type for (10). The computation of Hessian matrix is based on an approximation of a Schur complement

	QPC		PF		PC	
$3n_c/3m$	*time*	n_A	*time*	*time*$_A$	*time*	*time*$_A$
162/54	0.29	203	0.07	0.03 (45%)	0.12	0.03 (26%)
900/180	2.08	311	0.68	0.34 (50%)	1.07	0.34 (31%)
2646/378	12.91	347	5.85	3.46 (59%)	7.00	3.26 (47%)
5832/648	53.4	384	27.1	18.1 (67%)	27.0	15.8 (59%)
10890/990	126.2	408	79.7	58.5 (73%)	90.0	60.5 (67%)
18252/1404	361.9	493	246.2	192.5 (78%)	274.0	184 (67%)
28350/1890	809.4	478	620.5	493.0 (79%)	677.6	493.5 (73%)

Tab. 1: *A is assembled in PF, and PC (time$_A$ is consumed by assembling A).*

	QPC		PF, precond. 1				PC, precond. 1			
$3n_c/3m$	*time*	n_A	*time*	n_A	*out*	*full*	*time*	n_A	*out*	*full*
162/54	0.29	203	0.25	97	19	11	0.30	195	27	16
900/180	2.08	311	0.94	112	20	12	1.20	154	17	11
2646/378	12.91	347	6.69	139	22	13	6.33	147	13	8
5832/648	53.4	384	29.33	173	25	13	23.8	157	13	7
10890/990	126.2	408	109.5	233	30	13	68.3	159	13	6
18252/1404	361.9	493	265.3	244	31	12	177.7	183	14	7
28350/1890	809.4	478	644.2	282	36	13	420.9	209	16	7

Tab. 2: *A is not assembled, the preconditioner 1 is assembled.*

	QPC		PF, precond. 2				PC, precond. 2			
$3n_c/3m$	*time*	n_A	*time*	n_A	*out*	*full*	*time*	n_A	*out*	*full*
162/54	0.29	203	0.28	154	27	12	0.23	148	16	12
900/180	2.08	311	0.98	112	20	14	1.05	138	13	8
2646/378	12.91	347	6.61	133	22	13	5.5	122	12	8
5832/648	53.4	384	26.6	150	24	13	22.6	145	13	8
10890/990	126.2	408	105.4	218	30	13	66.4	151	13	7
18252/1404	361.9	493	253.2	224	31	13	169.8	169	14	7
28350/1890	809.4	478	584.7	251	33	12	397.1	190	15	7

Tab. 3: *A is not assembled, the preconditioner 2 is assembled.*

to the stiffness matrix K. While the first preconditioner uses diagonal scaling, the second one requires more expensive computations. Let us note that both preconditioners are assembled.

4. Conclusions

In the contribution we present our first experience with solving contact problems by the interior point algorithms. As the results seem promising many questions are still open, namely the convergence proof of both algorithms. The future work consists also of applying non-assembled preconditioners, and of implementing theoretically supported inner solvers.

References

[1] S. Boyd, L. Vandenberghe: *Convex optimization.* Cambridge University Press, Cambridge 2004.

[2] J. Haslinger, R. Kučera, Z. Dostál: *An algorithm for the numerical realization of 3D contact problems with Coulomb friction.* J. Comput. Appl. Math., **164–165** (2004), 387–408.

[3] R. Kučera: *Convergence rate of an optimization algorithm for minimizing quadratic functions with separable convex constraints.* SIAM J. Optim., **19** (2008), 846–862.

[4] J. Nocedal, A. Wächter, R.A. Waltz: *Adaptive barrier strategies for nonlinear interior methods.* TR RC 23563, IBM T.J. Watson Research Center, 2005.

[5] L. Lukšan, C. Matonoha, J. Vlček: *Interior point methods for large-scale nonlinear programming.* TR No. 917, Academy of Sciences of the Czech Republic 2004.

[6] J. Machalová, P. Ženčák, R. Kučera: *Metody vnitřních bodů pro řešení úlohy nelineárního programování.* ODAM 2007, pp. 4–17. Can be found online at: `http://mant.upol.cz/soubory/odam/odam07sb.pdf`

[7] J. Nocedal, S. J. Wright: *Numerical optimization.* Springer, New York 1999.

THE NUMERICAL SOLUTION OF COMPRESSIBLE FLOWS IN TIME DEPENDENT DOMAINS[*]

Václav Kučera, Jan Česenek

Abstract

This work is concerned with the numerical solution of inviscid compressible fluid flow in moving domains. Specifically, we assume that the boundary part of the domain (impermeable walls) are time dependent. We consider the Euler equations, which describe the movement of inviscid compressible fluids. We present two formulations of the Euler equations in the ALE (Arbitrary Lagrangian-Eulerian) form. These two formulations are discretized in space by the discontinuous Galerkin method. We apply a semi-implicit linearization with respect to time to obtain a numerical scheme requiring the solution of only one linear system on each time level. We apply the method to the compressible flow around a moving (vibrating) profile.

1. Continuous problem

In this paper we shall be concerned with two-dimensional inviscid compressible flow in a bounded domain $\Omega_t \subset \mathbb{R}^2$ depending on time $t \in [0, T]$. We assume that the boundary of Ω_t consists of three disjoint parts $\Gamma_I, \Gamma_O, \Gamma_{W_t}$: $\partial\Omega_t = \Gamma_I \cup \Gamma_O \cup \Gamma_{W_t}$, where Γ_I and Γ_O represent the time-independent inlet and outlet, respectively, and Γ_{W_t} represents moving impermeable walls.

As the governing equations we take the Euler equations written in the conservative form

$$\frac{\partial \boldsymbol{w}}{\partial t} + \sum_{s=1}^{2} \frac{\partial \boldsymbol{f}_s(\boldsymbol{w})}{\partial x_s} = 0 \quad \text{in } \Omega_t, \quad t \in (0, T), \tag{1}$$

where

$$\boldsymbol{w} = (w_1, \dots, w_4)^{\mathrm{T}} = (\rho,\, \rho v_1,\, \rho v_2,\, E)^{\mathrm{T}}$$

is the so-called *state vector* and

$$\boldsymbol{f}_s(\boldsymbol{w}) = (\rho v_s,\, \rho v_s v_1 + \delta_{s1} p,\, \rho v_s v_2 + \delta_{s2} p,\, (E + p)\, v_s)^{\mathrm{T}}$$

are the *Euler inviscid fluxes* of the quantity $\boldsymbol{w}$ in the directions x_s, $s = 1, 2$. We use the following notation: ρ - density, p - pressure, E - total energy, $\boldsymbol{v} = (v_1, v_2)$ - velocity vector, $\gamma > 1$ - Poisson adiabatic constant (we take $\gamma = 1.4$ for air), $a = \sqrt{\gamma p/\rho}$ - local speed of sound.

[*] This work is a part of the research project No. MSM 0021620839 of the Ministry of Education of the Czech Republic. The research was also partly supported by the Grant No. 201/08/0012 of the Czech Grant Agency.

System (1) is completed by the thermodynamical relation arising from the equation of state

$$p = (\gamma - 1)(E - \rho \left|\boldsymbol{v}\right|^2 /2),$$

furthermore by the initial condition

$$\boldsymbol{w}(\boldsymbol{x}, 0) = \boldsymbol{w}^0(x), \quad \boldsymbol{x} \in \Omega, \tag{2}$$

and boundary conditions, which are treated in Section 4

As in [6] we define the flux of the quantity $\boldsymbol{w}$ in the direction $\boldsymbol{n} = (n_1, n_2) \in \mathbb{R}^2$, $n_1^2 + n_2^2 = 1$, by

$$\boldsymbol{F}(\boldsymbol{w}, \boldsymbol{n}) = \sum_{s=1}^{2} \boldsymbol{f}_s(\boldsymbol{w}) n_s$$

and its Jacobi matrix

$$\mathbb{P}(\boldsymbol{w}, \boldsymbol{n}) = \frac{D\boldsymbol{F}(\boldsymbol{w}, \boldsymbol{n})}{D\boldsymbol{w}} = \sum_{s=1}^{2} \boldsymbol{A}_s(\boldsymbol{w}) n_s, \tag{3}$$

where

$$\boldsymbol{A}_s(\boldsymbol{w}) = \frac{D\boldsymbol{f}_s(\boldsymbol{w})}{D\boldsymbol{w}}, \ s = 1, 2,$$

are the Jacobi matrices of the mappings $\boldsymbol{f}_s$. It is possible to show that $\boldsymbol{f}_s, \ s = 1, 2,$ are homogeneous mappings of order one, which implies that

$$\boldsymbol{f}_s(\boldsymbol{w}) = \boldsymbol{A}_s(\boldsymbol{w})\boldsymbol{w}, \ s = 1, 2, \tag{4}$$

and

$$\boldsymbol{F}(\boldsymbol{w}, \boldsymbol{n}) = \mathbb{P}(\boldsymbol{w}, \boldsymbol{n})\boldsymbol{w}. \tag{5}$$

The matrix $\mathbb{P}$ is diagonalizable, i.e.

$$\mathbb{P} = \mathbb{T}\Lambda\mathbb{T}^{-1}, \tag{6}$$

where $\mathbb{T} = \mathbb{T}(\boldsymbol{w}, \boldsymbol{n})$ is a nonsingular matrix and

$$\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_4) \tag{7}$$

is the diagonal matrix with entries

$$\lambda_1 = \boldsymbol{v} \cdot \boldsymbol{n} - a, \ \lambda_2 = \lambda_3 = \boldsymbol{v} \cdot \boldsymbol{n}, \ \lambda_4 = \boldsymbol{v} \cdot \boldsymbol{n} + a, \tag{8}$$

which are the eigenvalues of the matrix $\mathbb{P}$. (See, e.g. [6], Section 3.1.5.)

2. ALE formulation

In order to treat the time dependance of the domain, we use the so-called *arbitrary Lagrangian-Eulerian* ALE technique. We define a reference domain Ω_0 and regular one-to-one ALE mapping of Ω_0 onto Ω_t (cf. [8], [9] and [10])

$$\mathcal{A}_t : \overline{\Omega}_0 \longrightarrow \overline{\Omega}_t, \text{ i.e. } \boldsymbol{X} \in \overline{\Omega}_0 \longmapsto \boldsymbol{x} = \boldsymbol{x}(\boldsymbol{X}, t) = \mathcal{A}_t(\boldsymbol{X}) \in \overline{\Omega}_t.$$

Here we use the notation $\boldsymbol{X}$ for points in $\overline{\Omega}_0$ and $\boldsymbol{x}$ for points in $\overline{\Omega}_t$.

Further, we define the ALE velocity:

$$\begin{aligned} \tilde{\boldsymbol{z}}(\boldsymbol{X}, t) &= \frac{\partial}{\partial t}\mathcal{A}_t(\boldsymbol{X}), \quad t \in [0, T], \ \boldsymbol{X} \in \Omega_0, \\ \boldsymbol{z}(\boldsymbol{x}, t) &= \tilde{\boldsymbol{z}}(\mathcal{A}^{-1}(\boldsymbol{x}), t), \quad t \in [0, T], \ \boldsymbol{x} \in \Omega_t, \end{aligned}$$

and the ALE derivative of a function $f = f(\boldsymbol{x}, t)$ defined for $\boldsymbol{x} \in \Omega_t$ and $t \in [0, T]$:

$$\frac{D^A}{Dt} f(\boldsymbol{x}, t) = \frac{\partial \tilde{f}}{\partial t}(\boldsymbol{X}, t), \tag{9}$$

where

$$\tilde{f}(\boldsymbol{X}, t) = f(\mathcal{A}_t(\boldsymbol{X}), t), \ \boldsymbol{X} \in \Omega_0, \ \boldsymbol{x} = \mathcal{A}_t(\boldsymbol{X}).$$

The following relations are a direct consequence of the chain rule:

$$\frac{D^A f}{Dt} = \frac{\partial f}{\partial t} + \boldsymbol{z} \cdot \nabla f = \frac{\partial f}{\partial t} + \operatorname{div}(\boldsymbol{z} f) - f \operatorname{div} \boldsymbol{z}.$$

This leads to two different formulations of the Euler equations in ALE form.

Formulation 1:

$$\frac{D^A \boldsymbol{w}}{Dt} + \sum_{s=1}^{2} \frac{\partial \boldsymbol{f}_s(\boldsymbol{w})}{\partial x_s} - \boldsymbol{z} \cdot \nabla \boldsymbol{w} = 0. \tag{10}$$

Formulation 2:

$$\frac{D^A \boldsymbol{w}}{Dt} + \sum_{s=1}^{2} \frac{\partial \boldsymbol{g}_s(\boldsymbol{w})}{\partial x_s} + \boldsymbol{w} \operatorname{div}\boldsymbol{z} = 0, \tag{11}$$

where $\boldsymbol{g}_s$, $s = 1, 2$, are modified inviscid fluxes

$$\boldsymbol{g}_s(\boldsymbol{w}) := \boldsymbol{f}_s(\boldsymbol{w}) - z_s \boldsymbol{w}.$$

3. Discretization of the problem in the time dependent domain

In this section, we shall describe the discretization of the initial-boundary value problem for the Euler equations written in the ALE forms (10) and (11). In the presented work we shall use the discontinuous Galerkin finite element method (DGFEM) for space semi-discretization. For an overview of various applications of the discontinuous Galerkin methods cf. [2].

3.1. Notation

In what follows we shall assume that Ω_t is a polygonal domain for all t. Let $\mathcal{T}_{ht}$ be a partition of the closure $\overline{\Omega}_t$ into a finite number of closed triangles with mutually disjoint interiors. We shall call $\mathcal{T}_{ht}$ a triangulation of Ω_t. We do not require the standard conforming properties of $\mathcal{T}_{ht}$ used in the finite element method. This means that we admit the so-called hanging nodes. We shall use the following notation. By ∂K we denote the boundary of an element $K \in \mathcal{T}_{ht}$ and set $h_K = \operatorname{diam}(K)$, $h = \max_{K\in\mathcal{T}_{ht}} h_K$. By ρ_K we denote the radius of the largest circle inscribed into K and by $|K|$ we denote the area of K.

Let $K, K' \in \mathcal{T}_{ht}$. We say that K and K' are *neighbours*, if the set $\partial K \cap \partial K'$ has positive length. We say that $\Gamma \subset K$ is a *face* of K, if it is a maximal connected open subset either of $\partial K \cap \partial K'$, where K' is a neighbour of K, or of $\partial K \cap \partial\Omega_t$. By $\mathcal{F}_{ht}$ we denote the system of all faces of all elements $K \in \mathcal{T}_{ht}$. Further, we define the set of all inner faces by

$$\mathcal{F}^I_{ht} = \{\Gamma \in \mathcal{F}_{ht};\ \Gamma \subset \Omega_t\}$$

and the set of all boundary faces by

$$\mathcal{F}^B_{ht} = \{\Gamma \in \mathcal{F}_{ht};\ \Gamma \subset \partial\Omega_t\}\,.$$

Obviously, $\mathcal{F}_{ht} = \mathcal{F}^I_{ht} \cup \mathcal{F}^B_{ht}$.

For each $\Gamma \in \mathcal{F}_{ht}$ we define a unit normal vector $\mathbf{n}_\Gamma$. We assume that for $\Gamma \in \mathcal{F}^B_{ht}$ the normal $\mathbf{n}_\Gamma$ has the same orientation as the outer normal to $\partial\Omega$. For each face $\Gamma \in \mathcal{F}^I_{ht}$ the orientation of $\mathbf{n}_\Gamma$ is arbitrary but fixed cf. Figure 1. Finally, by $d(\Gamma)$ we denote the length of $\Gamma \in \mathcal{F}_{ht}$.

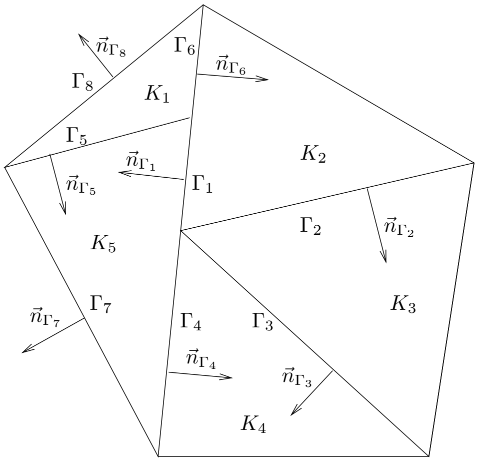


Fig. 1: *Typical DG triangulation with possible hanging nodes.*

3.2. Spaces of discontinuous functions

For each face $\Gamma \in \mathcal{F}_{ht}^I$ there exist two neighbours $K_\Gamma^{(L)}$, $K_\Gamma^{(R)} \in \mathcal{T}_{ht}$ such that $\Gamma \subset K_\Gamma^{(L)} \cap K_\Gamma^{(R)}$. We use the convention that $\mathbf{n}_\Gamma$ is the outer normal to the element $K_\Gamma^{(L)}$ and the inner normal to the element $K_\Gamma^{(R)}$, see Figure 2. Let $p \geq 1$ be an integer. The approximate solution will be sought in the space of discontinuous piecewise polynomial functions

$$\boldsymbol{S}_{ht} = \{v;\, v|_K \in P^p(K), \forall K \in \mathcal{T}_{ht}\}^4,$$

where $P^p(K)$ denotes the space of all polynomials on K of degree $\leq p$. For $v \in \boldsymbol{S}_{ht}$ and $\Gamma \in \mathcal{F}_{ht}^I$ we introduce the following notation:

$$\begin{aligned} v|_\Gamma^{(L)} &= \text{ the trace of } v|_{K_\Gamma^{(L)}} \text{ on } \Gamma, \qquad & v|_\Gamma^{(R)} &= \text{ the trace of } v|_{K_\Gamma^{(R)}} \text{ on } \Gamma, \\ \langle v \rangle_\Gamma &= \tfrac{1}{2}\big(v|_\Gamma^{(L)} + v|_\Gamma^{(R)}\big), & [v]_\Gamma &= v|_\Gamma^{(L)} - v|_\Gamma^{(R)}. \end{aligned}$$

Now, let $\Gamma \in \mathcal{F}_{ht}^B$ and $K_\Gamma^{(L)} \in \mathcal{T}_{ht}$ be such an element that $\Gamma \subset \partial K_\Gamma^{(L)} \cap \partial\Omega_t$. For $v \in \boldsymbol{S}_{ht}$ we set

$$v_\Gamma = v|_\Gamma^{(L)} = v|_\Gamma^{(R)} = \text{ the trace of } v|_{K_\Gamma^{(L)}} \text{ on } \Gamma,$$

i.e. we define $v|_\Gamma^{(R)}$ by extrapolation.

If $[\cdot]_\Gamma$ and $\langle\cdot\rangle_\Gamma$ appear in an integral of the form $\int_\Gamma \ldots \,\mathrm{d}S$, we omit the subscript Γ and write simply $[\cdot]$ and $\langle\cdot\rangle$.

3.3. Space semidiscretization

3.3.1. Formulation 1

In order to derive the discrete problem, we assume that $\boldsymbol{w}$ is a sufficiently regular solution of system (10), multiply (10) by a test function $\boldsymbol{\varphi} \in \boldsymbol{S}_{ht}$, integrate over any element K apply Green's theorem and sum over all $K \in \mathcal{T}_{ht}$. We get the relation

$$\begin{aligned} \sum_{K\in\mathcal{T}_{ht}} \int_K \frac{D^{\mathcal{A}}\boldsymbol{w}(t)}{Dt} \cdot \boldsymbol{\varphi}\,\mathrm{d}\boldsymbol{x} &= \sum_{K\in\mathcal{T}_{ht}} \int_K \sum_{s=1}^2 \boldsymbol{f}_s(\boldsymbol{w}(t)) \cdot \frac{\partial\boldsymbol{\varphi}}{\partial x_s}\,\mathrm{d}\boldsymbol{x} \\ &- \sum_{\Gamma\in\mathcal{F}_{ht}} \int_\Gamma \sum_{s=1}^2 \boldsymbol{f}_s(\boldsymbol{w}(t))n_s \cdot [\boldsymbol{\varphi}]\,\mathrm{d}S - \sum_{K\in\mathcal{T}_{ht}} \int_K \sum_{s=1}^2 z_s \frac{\partial\boldsymbol{w}}{\partial x_s} \cdot \boldsymbol{\varphi}\,\mathrm{d}\boldsymbol{x}. \end{aligned}$$

Now the exact solution $\boldsymbol{w}(t)$ is approximated by an element $\boldsymbol{w}_h(t) \in \boldsymbol{S}_{ht}$ and the fluxes through the faces Γ are approximated as in the finite volume method with the aid of a numerical flux $\boldsymbol{H}_f = \boldsymbol{H}_f(\boldsymbol{u}, \boldsymbol{w}, \boldsymbol{n})$. It means that on edge Γ

$$\sum_{s=1}^2 \boldsymbol{f}_s(\boldsymbol{w}(t))n_s \approx \boldsymbol{H}_f(\boldsymbol{w}_h|_\Gamma^{(L)}(t), \boldsymbol{w}_h|_\Gamma^{(R)}(t), \boldsymbol{n}).$$

In this work we take $\boldsymbol{H}_f$ as the Vijayasundaram numerical flux consistent with the fluxes $\boldsymbol{f}_s$, $s = 1, 2$, cf. [11]. Taking into account (3) we define the "positive" and "negative" parts of the matrix $\mathbb{P}$ as

$$\mathbb{P}^{\pm} = \mathbb{T}\Lambda^{\pm}\mathbb{T}^{-1},$$

where

$$\Lambda^{\pm} = \mathrm{diag}(\lambda_1^{\pm}, \ldots, \lambda_4^{\pm}),$$

and $\lambda^+ = \max(\lambda, 0)$, $\lambda^- = \min(\lambda, 0)$. Using the above concepts, we introduce the Vijayasundaram numerical flux

$$\boldsymbol{H}_f(\boldsymbol{w}^{(L)}, \boldsymbol{w}^{(R)}, \boldsymbol{n}) = \mathbb{P}^+ \left(\langle \boldsymbol{w} \rangle, \boldsymbol{n}\right) \boldsymbol{w}^{(L)} + \mathbb{P}^- \left(\langle \boldsymbol{w} \rangle, \boldsymbol{n}\right) \boldsymbol{w}^{(R)}.$$

Finally we can define the discrete forms defining the discrete form of formulation 1:

$$\begin{aligned}
\left(\frac{D^{\mathcal{A}}\boldsymbol{w}_h}{Dt}, \boldsymbol{\varphi}_h\right)_{ht} &= \int_{\Omega_{ht}} \frac{D^{\mathcal{A}}\boldsymbol{w}_h}{Dt} \cdot \boldsymbol{\varphi}_h \,\mathrm{d}\boldsymbol{x}, \\
\widetilde{b}_h^{(1)}(\boldsymbol{w}_h, \boldsymbol{\varphi}_h) &= -\sum_{K \in \mathcal{T}_{ht}} \int_K \sum_{s=1}^{2} \boldsymbol{f}_s(\boldsymbol{w}_h) \cdot \frac{\partial \boldsymbol{\varphi}}{\partial x_s} \,\mathrm{d}\boldsymbol{x} \\
&\quad + \sum_{\Gamma \in \mathcal{F}_{ht}} \int_\Gamma \boldsymbol{H}_f(\boldsymbol{w}_h|_\Gamma^{(L)}, \boldsymbol{w}_h|_\Gamma^{(R)}, \boldsymbol{n}_{ij}) \cdot [\boldsymbol{\varphi}_h] \,\mathrm{d}S, \\
d_h^{(1)}\left(\boldsymbol{w}_h, \boldsymbol{\varphi}_h\right) &= -\sum_{K \in \mathcal{T}_{ht}} \int_K \sum_{s=1}^{2} z_s \frac{\partial \boldsymbol{w}}{\partial x_s} \cdot \boldsymbol{\varphi} \,\mathrm{d}\boldsymbol{x}.
\end{aligned}$$

3.3.2. Formulation 2

We proceed similarly as in the preceding section. We multiply (11) by a test function $\boldsymbol{\varphi} \in \boldsymbol{S}_{ht}$, integrate over any element K, apply Green's theorem and sum over all $K \in \mathcal{T}_{ht}$. We get the relation

$$\begin{aligned}
\sum_{K \in \mathcal{T}_{ht}} \int_K \frac{D^{\mathcal{A}}\boldsymbol{w}(t)}{Dt} \cdot \boldsymbol{\varphi} \,\mathrm{d}\boldsymbol{x} &= \sum_{K \in \mathcal{T}_{ht}} \int_K \sum_{s=1}^{2} \boldsymbol{g}_s(\boldsymbol{w}(t)) \cdot \frac{\partial \boldsymbol{\varphi}}{\partial x_s} \,\mathrm{d}\boldsymbol{x} \\
- \sum_{\Gamma \in \mathcal{F}_{ht}} \int_\Gamma \sum_{s=1}^{2} \boldsymbol{g}_s(\boldsymbol{w}(t)) n_s \cdot [\boldsymbol{\varphi}] \,\mathrm{d}S &- \sum_{K \in \mathcal{T}_{ht}} \int_K \mathrm{div}\boldsymbol{z}\left(\boldsymbol{w} \cdot \boldsymbol{\varphi}\right) \,\mathrm{d}\boldsymbol{x}.
\end{aligned}$$

Now the exact solution $\boldsymbol{w}(t)$ is approximated by an element $\boldsymbol{w}_h(t) \in \boldsymbol{S}_{ht}$ and the fluxes through the faces Γ are approximated with the aid of a numerical flux $\boldsymbol{H}_g = \boldsymbol{H}_g(\boldsymbol{u}, \boldsymbol{w}, \boldsymbol{n})$. It means that on edge Γ

$$\sum_{s=1}^{2} \boldsymbol{g}_s(\boldsymbol{w}(t)) n_s \approx \boldsymbol{H}_g(\boldsymbol{w}_h|_\Gamma^{(L)}(t), \boldsymbol{w}_h|_\Gamma^{(R)}(t), \boldsymbol{n}). \tag{12}$$

Here $\boldsymbol{H}_g$ is an analogy to the Vijayasundaram numerical flux consistent with the fluxes $\boldsymbol{g}_s,\ s=1,2$. We have

$$\frac{D\boldsymbol{g}_s(\boldsymbol{w})}{D\boldsymbol{w}} = \frac{D\boldsymbol{f}_s(\boldsymbol{w})}{D\boldsymbol{w}} - z_s\mathbb{I} = \boldsymbol{A}_s - z_s\mathbb{I}$$

and can write

$$\widetilde{\mathbb{P}}(\boldsymbol{w},\boldsymbol{n}) = \sum_{s=1}^{2}\frac{D\boldsymbol{g}_s(\boldsymbol{w})}{D\boldsymbol{w}}n_s = \sum_{s=1}^{2}(\boldsymbol{A}_s n_s - z_s n_s\mathbb{I}) = \mathbb{P}(\boldsymbol{w},\boldsymbol{n}) - (\boldsymbol{z}\cdot\boldsymbol{n})\mathbb{I}.$$

This, (6), (7) and (8) imply that

$$\widetilde{\mathbb{P}} = \mathbb{T}\widetilde{\Lambda}\mathbb{T}^{-1}, \quad \widetilde{\Lambda} = \operatorname{diag}(\widetilde{\lambda}_1,\dots,\widetilde{\lambda}_4), \quad \widetilde{\lambda}_i = \lambda_i - \boldsymbol{z}\cdot\boldsymbol{n},\ i=1,\dots,4.$$

Now we define the "positive" and "negative" parts of the matrix $\widetilde{\mathbb{P}}$ as

$$\widetilde{\mathbb{P}}^{\pm} = \mathbb{T}\widetilde{\Lambda}^{\pm}\mathbb{T}^{-1}$$

and we introduce the modification of the Vijayasundaram numerical flux

$$\boldsymbol{H}_g(\boldsymbol{w}^{(L)},\boldsymbol{w}^{(R)},\boldsymbol{n}) = \tilde{\mathbb{P}}^{+}\left(\langle\boldsymbol{w}\rangle,\boldsymbol{n}\right)\boldsymbol{w}^{(L)} + \tilde{\mathbb{P}}^{-}\left(\langle\boldsymbol{w}\rangle,\boldsymbol{n}\right)\boldsymbol{w}^{(R)}.$$

Finally we can define the discrete forms of formulation 2:

$$\begin{aligned}
\widetilde{b}_h^{(2)}(\boldsymbol{w}_h,\boldsymbol{\varphi}_h) &= -\sum_{K\in\mathcal{T}_{ht}}\int_K\sum_{s=1}^{2}\boldsymbol{g}_s(\boldsymbol{w}_h)\cdot\frac{\partial\boldsymbol{\varphi}}{\partial x_s}\,\mathrm{d}\boldsymbol{x}\\
&\quad+\sum_{\Gamma\in\mathcal{F}_{ht}}\int_\Gamma\boldsymbol{H}_g(\boldsymbol{w}_h|_\Gamma^{(L)},\boldsymbol{w}_h|_\Gamma^{(R)},\boldsymbol{n}_{ij})\cdot[\boldsymbol{\varphi}_h]\,\mathrm{d}S,\\
d_h^{(2)}(\boldsymbol{w}_h,\boldsymbol{\varphi}_h) &= -\sum_{K\in\mathcal{T}_{ht}}\int_K\operatorname{div}\boldsymbol{z}\,(\boldsymbol{w}_h\cdot\boldsymbol{\varphi})\,\mathrm{d}\boldsymbol{x}.
\end{aligned}$$

Finally we define an *approximate solution* of (10) and (11), respectively, as a function $\boldsymbol{w}_h=\boldsymbol{w}_h(t)$ satisfying the conditions

$$\begin{aligned}
&\text{(a)} \quad \boldsymbol{w}_h(t)\in\boldsymbol{S}_{ht},\ \forall t\in[0,T],\\
&\text{(b)} \quad \left(\frac{D^{\mathcal{A}}\boldsymbol{w}_h(t)}{Dt},\boldsymbol{\varphi}_h\right)_h + \widetilde{b}_h(\boldsymbol{w}_h(t),\boldsymbol{\varphi}_h) - d_h\left(\boldsymbol{w}_h(t),\boldsymbol{\varphi}_h\right) = 0,\\
&\qquad \forall\boldsymbol{\varphi}_h\in\boldsymbol{S}_{ht},\ \forall t\in(0,T),\\
&\text{(c)} \quad \boldsymbol{w}_h(0) = \Pi_h\boldsymbol{w}^0,
\end{aligned}\tag{13}$$

where $\Pi_h\boldsymbol{w}^0$ is the $L^2(\Omega_0)$-projection of $\boldsymbol{w}^0$ from the initial condition (2) on the space $\boldsymbol{S}_{h0}$. Relation (13) represents a discrete formulation of (10) and (11), when setting $\widetilde{b}_h := \widetilde{b}_h^{(1)}$, $d_h := d_h^{(1)}$ and $\widetilde{b}_h := \widetilde{b}_h^{(2)}$, $d_h := d_h^{(2)}$, respectively.

3.4. Time discretization

In this section we shall introduce the time discretization of problem (13). Due to the similarity of the two formulations we treat here only the second formulation. The discrete problem (13) is equivalent to a large system of ordinary differential equations. In order to avoid a CFL-stability condition we apply a semi-implicit scheme, which is a generalization of the techniques proposed in [4] and [7].

We introduce the partition $0 = t_0 < t_1 < \ldots$ of the time interval $[0, T]$ and set $\tau_j = t_{j+1} - t_j$. The function $\boldsymbol{w}_h(\cdot, t_j)$ will be approximated by $\boldsymbol{w}^j$, defined in Ω_{t_j}. Let us assume that the approximate solution $\boldsymbol{w}_h^j$ has already been computed for $j = 0, \ldots, k$. We are interested in the computation of the approximate solution $\boldsymbol{w}_h^{k+1}$ at time instant t_{k+1}. If we set

$$\hat{\boldsymbol{w}}_h^j(\boldsymbol{x}) = \boldsymbol{w}^j\left(\mathcal{A}_{t_j}\left(\mathcal{A}_{t_{k+1}}^{-1}\right)(\boldsymbol{x})\right), \quad \boldsymbol{x} \in \Omega_{ht_{k+1}},$$

then, on the basis of (9), we can approximate the ALE derivative using the first order backward difference:

$$\left(\frac{D^{\mathcal{A}}\boldsymbol{w}_h(\boldsymbol{x}, t)}{Dt}, \boldsymbol{\varphi}_h\right)\Bigg|_{t_{k+1}} \approx \left(\frac{\boldsymbol{w}^{k+1}(\boldsymbol{x}) - \hat{\boldsymbol{w}}_h^k(\boldsymbol{x})}{\tau_k}, \boldsymbol{\varphi}_h\right), \quad \boldsymbol{x} \in \Omega_{ht_{k+1}}.$$

Further, on the basis of (4), (5) and the definition of the modified Vijayasundaram numerical flux we define a partial linearization $b_h^{(2)}$ of the form $\tilde{b}_h^{(2)}$:

$$b_h^{(2)}(\hat{\boldsymbol{w}}_h^k, \boldsymbol{w}_h^{k+1}, \boldsymbol{\varphi}_h) = -\sum_{K \in \mathcal{T}_{ht+1}} \int_K \sum_{s=1}^{2} (\boldsymbol{A}_s(\hat{\boldsymbol{w}}^k) - z_s)\mathbb{I})\boldsymbol{w}^{k+1}) \cdot \frac{\partial \boldsymbol{\varphi}_h}{\partial x_s} \, \mathrm{d}\boldsymbol{x} \tag{14}$$

$$+ \sum_{\Gamma \in \mathcal{F}_{ht+1}} \int_\Gamma \left[\tilde{\mathbb{P}}^+ \left(\langle \hat{\boldsymbol{w}}_h^k \rangle, \boldsymbol{n}_{ij}\right) \boldsymbol{w}_h^{k+1}|_{\Gamma_{ij}} + \tilde{\mathbb{P}}^- \left(\langle \hat{\boldsymbol{w}}_h^k \rangle, \boldsymbol{n}_{ij}\right) \boldsymbol{w}_h^{k+1}|_{\Gamma_{ji}}\right] \cdot \boldsymbol{\varphi}_h \, \mathrm{d}S.$$

The term $d_h^{(2)}$ linear with respect to $\boldsymbol{w}_h$ will be treated implicitly.

These considerations lead us to the following *semi-implicit scheme*: For $k = 0, 1, \ldots$ find $\boldsymbol{w}_h^{k+1}$ such that

$$\begin{aligned}
&\text{(a)} \qquad \boldsymbol{w}_h^{k+1} \in \boldsymbol{S}_{ht_{k+1}}, \\
&\text{(b)} \qquad \left(\frac{\boldsymbol{w}_h^{k+1} - \hat{\boldsymbol{w}}_h^k}{\tau_k}, \boldsymbol{\varphi}_h\right) + b_h^{(2)}(\hat{\boldsymbol{w}}_h^k, \boldsymbol{w}_h^{k+1}, \boldsymbol{\varphi}_h) - d_h^{(2)}\left(\boldsymbol{w}_h^{k+1}, \boldsymbol{\varphi}_h\right) = 0, \\
&\qquad\qquad \forall \boldsymbol{\varphi}_h \in \boldsymbol{S}_{ht_{k+1}}, \\
&\text{(c)} \qquad \boldsymbol{w}_h^0 = \Pi_h \boldsymbol{w}^0.
\end{aligned}$$

This relation represents a system of linear algebraic equations on each time level which is solved either iteratively using the block-Jacobi preconditioned GMRES or a direct method (e.g. the direct unsymmetric multifrontal solver UMFPACK cf. [3]).

4. Boundary conditions

If $\Gamma \subset \partial\Omega_{ht}$, it is necessary to specify the boundary state $\boldsymbol{w}|_{\Gamma}^{(R)}$ appearing in the numerical flux $\boldsymbol{H}$ in the definition of the inviscid form b_h.

On the inlet and outlet, which are fixed, we proceed in the same way as in [7], Section 4, where we prescribe the state $\boldsymbol{w}|_{\Gamma}^{(R)}$ in such a way that the locally linearized Euler equations are well posed. On the impermeable moving wall we prescribe the normal component of the velocity

$$\boldsymbol{v} \cdot \boldsymbol{n} = \boldsymbol{z} \cdot \boldsymbol{n}, \tag{15}$$

where $\boldsymbol{n}$ is unit outer normal to Γ_{W_t} and $\boldsymbol{z}$ is the wall velocity. This is done by prescribing the numerical flux $\boldsymbol{H}$ on Γ_{W_t}. Again we shall treat only formulation 2. We define the numerical flux as the physical flux through the boundary with the assumption (15) taken into account. We write:

$$\sum_{s=1}^{2} \boldsymbol{g}_s(\boldsymbol{w}) n_s = (\mathbf{v} \cdot \mathbf{n} - \boldsymbol{z} \cdot \mathbf{n})\boldsymbol{w} + p\,(0, n_1, n_2, \mathbf{v} \cdot \mathbf{n})^T = p\,(0, n_1, n_2, \mathbf{v} \cdot \mathbf{n})^T =: \boldsymbol{H}_g$$

on Γ_{W_t}. We proceed similarly in formulation 1.

5. Limiting procedure at discontinuities

For high speed flows with shock waves and contact discontinuities it is necessary to avoid the Gibbs phenomenon manifested by spurious overshoots and undershoots in computed quantities near discontinuities. In order to avoid the Gibbs phenomenon, we apply the limiting procedure from [7] based on the discontinuity indicator

$$g^k(K) = \int_{\partial K} [\hat{\rho}_h^k]^2 \, \mathrm{d}S / (h_K |K|^{3/4}), \quad K \in \mathcal{T}_{ht_{k+1}},$$

introduced in [5]. The density $\hat{\rho}_h^k$ represents the first component of the state vector $\hat{\boldsymbol{w}}_h^k$. Then we define the discrete discontinuity indicator

$$G^k(K) = \begin{cases} 0 & \text{if } g^k(K) < 1, \\ 1 & \text{if } g^k(K) \geq 1, \end{cases}$$

and the artificial viscosity forms

$$\beta_h(\hat{\boldsymbol{w}}_h^k, \boldsymbol{w}_h^{k+1}, \boldsymbol{\varphi}_h) = \nu_1 \sum_{K \in \mathcal{T}_{ht}} h_K G^k(K) \int_K \nabla \boldsymbol{w}_h^{k+1} \cdot \nabla \boldsymbol{\varphi}_h \, \mathrm{d}\boldsymbol{x},$$

and

$$J_h(\hat{\boldsymbol{w}}_h^k, \boldsymbol{w}_h^{k+1}, \boldsymbol{\varphi}_h) = \nu_2 \sum_{\Gamma \in \mathcal{F}_{ht}} \frac{1}{2}\big(G^k(K) + G^k(K)\big) \int_\Gamma [\boldsymbol{w}_h^{k+1}] \cdot [\boldsymbol{\varphi}_h] \, \mathrm{d}S,$$

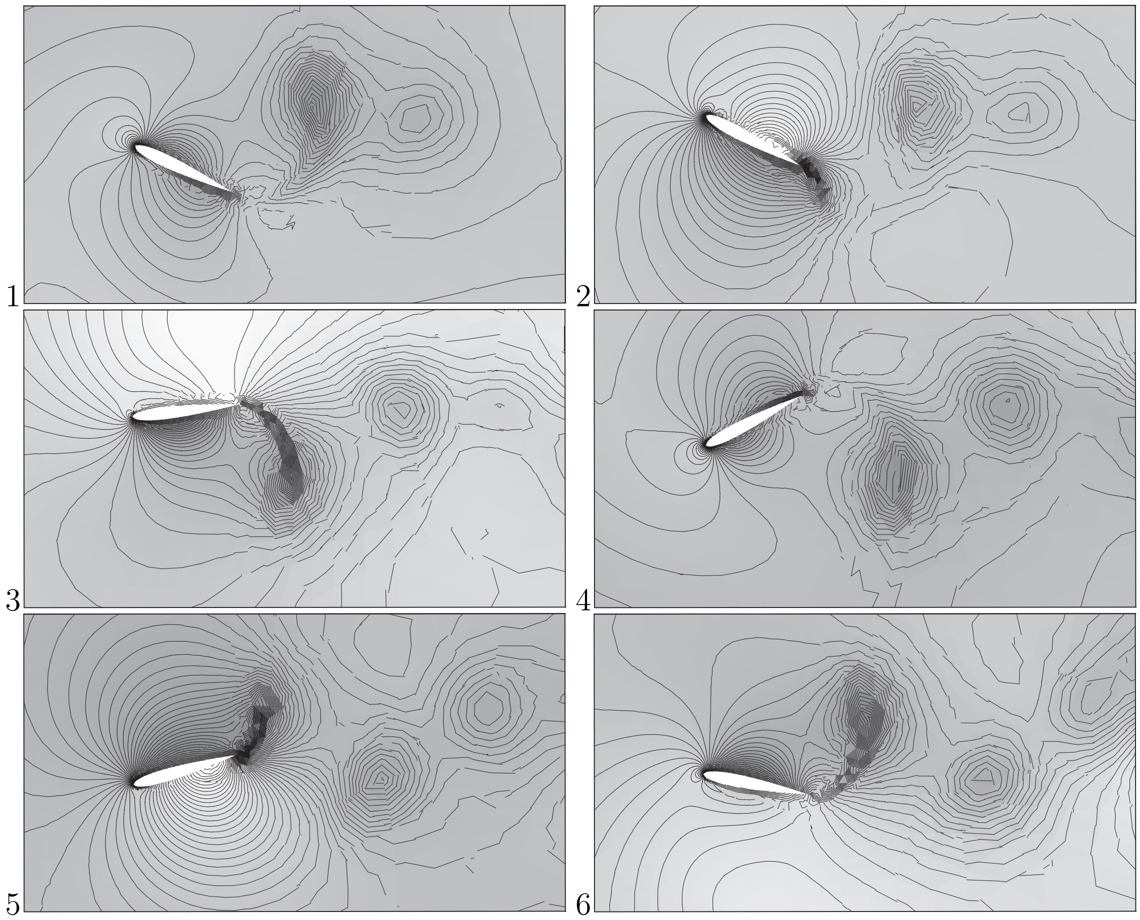


Fig. 2: *Density isolines for the periodically oscillating NACA0012 profile.*

with ν_1, $\nu_2 = O(1)$. Then the resulting scheme obtained by limiting of (14), (b) has the form

$$\begin{aligned}
&\text{(a)} \qquad \boldsymbol{w}_h^{k+1} \in \boldsymbol{S}_{ht_{k+1}},\\
&\text{(b)} \qquad \left(\frac{\boldsymbol{w}_h^{k+1} - \hat{\boldsymbol{w}}_h^k}{\tau_k}, \boldsymbol{\varphi}_h\right)_h + b_h(\hat{\boldsymbol{w}}_h^k, \boldsymbol{w}_h^{k+1}, \varphi_h) + \beta_h(\hat{\boldsymbol{w}}_h^k, \boldsymbol{w}_h^{k+1}, \boldsymbol{\varphi}_h)\\
&\qquad\qquad + J_h(\hat{\boldsymbol{w}}_h^k, \boldsymbol{w}_h^{k+1}, \boldsymbol{\varphi}_h) = 0, \ \forall\, \varphi_h \in \boldsymbol{S}_{ht_{k+1}}, \ k = 0, 1, \dots,\\
&\text{(c)} \qquad \boldsymbol{w}_h^0 = \Pi_h \boldsymbol{w}^0.
\end{aligned}$$

Remark. In practical computations, the integrals appearing in the definition of the approximated solution are evaluated with the aid of quadrature formulae. In order to obtain an accurate, physically admissible solution, it is necessary to use isoparametric elements near curved boundaries (see [1] or [6], Section 4.6.8). In our computations we proceed in such a way that a reference triangle is transformed by a bilinear mapping onto the approximation of a curved triangle adjacent to the boundary $\partial\Omega$.

6. Numerical experiments

We consider inviscid compressible flow around the NACA0012 profile, which is periodically moving in the vertical direction with a periodically varying angle of attack. Figure 2 shows density contours plotted at six time frames distributed over one period of the flow. The presented plots represent a part of the computational domain, which is chosen large in order to eliminate the role of artificial boundaries.

The ALE mapping is defined using two concentric circles with a center at the center of gravity of the profile. Outside the outer circle the ALE mapping is chosen as the identity mapping, i.e. no motion of the computational domain takes place. Inside the inner circle the ALE mapping is defined so that the motion of the computational domain coincides with the prescribed movement of the profile. In the space between the two circles the movement of the ALE mapping is linearly interpolated to yield a globally regular mapping.

In the solution we can observe the formation of vortices in the wake of the airfoil due to the vibrating motion, which introduces vorticity into the fluid flow. These vortices are then convected out of the computational domain. The Mach number at infinity of the flow is $M_\infty = 0.1$.

References

[1] F. Bassi, S. Rebay: *High-order accurate discontinuous finite element solution of the 2D Euler equations*, J. Comput. Phys. **138** (1997), 251–285.

[2] B. Cockburn, G.E. Karniadakis, C.-W.Shu (Eds.): *Discontinuous Galerkin methods*, Lecture Notes in Computational Science and Engineering 11, Springer, Berlin (2000).

[3] T.A. Davis, I.S. Duff: *A combined unifrontal/multifrontal method for unsymmetric sparse matrices*, ACM Trans. Math. Software, **25** (1999), 1–19.

[4] V. Dolejší, M. Feistauer: *A Semi-implicit discontinuous Galerkin finite element method for the numerical solution of inviscid compressible flow*, J. Comput. Phys. **198** (2004), 727–746.

[5] V. Dolejší, M. Feistauer, C. Schwab: *On some aspects of the discontinuous Galerkin finite element method for conservation laws.* Math. Comput. Simul. **61** (2003), 333–346.

[6] M. Feistauer, J. Felcman, I. Straškaraba: *Mathematical and computational methods for compressible flow*, Clarendon Press, Oxford (2003).

[7] M. Feistauer, V. Kučera: *On a robust discontinuous Galerkin technique for the solution of compressible flow*, J. Comput. Phys. **224** (2007), 208–221.

[8] T. Nomura, T.J.R. Hughes: *An arbitrary Lagrangian-Eulerian finite element method for interaction of fluid and a rigid body*, Comput. Methods Appl. Mech. Engrg. **95** (1992), 115–138.

[9] P. Sváček, M. Feistauer, J. Horáček: *Numerical simulation of flow induced airfoil vibrations with large amplitudes*, J. Fluids Struct. **23** (2007), 391–411.

[10] J.J.W. van der Vegt, H. van der Ven: *Space-time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flow*, J. Comput. Phys. **182** (2002), 546–585.

[11] G. Vijayasundaram: *Transonic flow simulation using upstream centered scheme of Godunov type in finite elements*, J. Comput. Phys. **63** (1986), 416–433.

ON LAGRANGE MULTIPLIERS OF TRUST-REGION SUBPROBLEMS*

Ladislav Lukšan, Ctirad Matonoha, Jan Vlček

1. Introduction

Consider the problem

$$\min F(x), \quad x \in \mathcal{R}^n,$$

where $F : \mathcal{R}^n \to \mathcal{R}$ is a twice continuously differentiable objective function. Basic optimization methods (trust-region and line-search methods) [6] generate points $x_i \in \mathcal{R}^n, i \in \mathcal{N}$, in such a way that x_1 is arbitrary and

$$x_{i+1} = x_i + \alpha_i d_i, \quad i \in \mathcal{N},$$

where $d_i \in \mathcal{R}^n$ are direction vectors and $\alpha_i > 0$ are step sizes.

Trust-region methods [1] are globally convergent techniques widely used, for example, in connection with the Newton's method for unconstrained optimization. They can be advantageously used when the Hessian matrix (or its approximation) of the objective function is indefinite, ill-conditioned or singular. This situation often arises in connection with the Newton's method for general objective function (indefiniteness) or with the Gauss-Newton's method for nonlinear least-squares problems (near singularity).

For a description of trust-region methods we define the quadratic function

$$Q_i(d) = \frac{1}{2}\, d^T B_i d + g_i^T d$$

which locally approximates the difference $F(x_i + d) - F(x_i)$, the vector

$$\omega_i(d) = \frac{(B_i d + g_i)}{\|g_i\|}$$

for the accuracy of computed direction, and the number

$$\rho_i(d) = \frac{F(x_i + d) - F(x_i)}{Q_i(d)}$$

*This work was supported by the Grant Agency of the Czech Academy of Sciences, project No. IAA1030405, the Grant Agency of the Czech Republic, project No. 201/06/P397, and the institutional research plan No. AV0Z10300504.

for the ratio of actual and predicted decrease of the objective function. Here $g_i = g(x_i) = \nabla F(x_i)$ and $B_i \approx \nabla^2 F(x_i)$ is an approximation of the Hessian matrix of the function F at the point $x_i \in \mathcal{R}^n$.

Trust-region methods are based on approximate minimizations of $Q_i(d)$ on the balls $\|d\| \le \Delta_i$ followed by updates of radii $\Delta_i > 0$. Thus direction vectors $d_i \in \mathcal{R}^n$ are chosen to satisfy the conditions

$$\begin{aligned} \|d_i\| &\le \Delta_i, & (1)\\ \|d_i\| &< \Delta_i \quad \Rightarrow \quad \|\omega_i(d_i)\| \le \overline{\omega}, & (2)\\ -Q_i(d_i) &\ge \underline{\sigma}\|g_i\| \min(\|d_i\|, \|g_i\|/\|B_i\|), & (3) \end{aligned}$$

where $0 \le \overline{\omega} < 1$ and $0 < \underline{\sigma} < 1$. Step sizes $\alpha_i \ge 0$ are selected so that

$$\begin{aligned} \rho_i(d_i) \le 0 \quad &\Rightarrow \quad \alpha_i = 0, & (4)\\ \rho_i(d_i) > 0 \quad &\Rightarrow \quad \alpha_i = 1. & (5) \end{aligned}$$

Trust-region radii $0 < \Delta_i \le \overline{\Delta}$ are chosen in such a way that $0 < \Delta_1 \le \overline{\Delta}$ is arbitrary and

$$\begin{aligned} \rho_i(d_i) < \underline{\rho} \quad &\Rightarrow \quad \underline{\beta}\|d_i\| \le \Delta_{i+1} \le \overline{\beta}\|d_i\|, & (6)\\ \rho_i(d_i) \ge \underline{\rho} \quad &\Rightarrow \quad \Delta_i \le \Delta_{i+1} \le \overline{\Delta}, & (7) \end{aligned}$$

where $0 < \underline{\beta} \le \overline{\beta} < 1$ and $0 < \underline{\rho} < 1$.

2. Direction determination

A crucial part of each trust-region method is the direction determination. There are various commonly known methods for computing direction vectors satisfying conditions (1)-(3). To simplify the notation, we omit the major index i and use the inner index j.

The most sophisticated method is based on a computation of the optimal locally constrained step. In this case, the vector $d \in \mathcal{R}^n$ is obtained by solving the subproblem

$$\min\ Q(d) = \frac{1}{2} d^T B d + g^T d \quad \text{subject to} \quad \|d\| \le \Delta. \tag{8}$$

Necessary and sufficient conditions for this solution are

$$\|d\| \le \Delta, \quad (B + \lambda I)d = -g, \quad B + \lambda I \succeq 0, \quad \lambda \ge 0, \quad \lambda(\Delta - \|d\|) = 0,$$

where λ is the optimal Lagrange multiplier. The Moré-Sorensen method [5] is based on solving the nonlinear equation $1/\|d(\lambda)\| = 1/\Delta$ with $(B + \lambda I)d(\lambda) + g = 0$ by the Newton's method, possibly the modified Newton's method [8] using the Choleski decomposition of $B + \lambda I$.

Steihaug [7] and Toint [9] proposed a technique for finding an approximate solution of (8). This implementation is based on the conjugate gradient algorithm [6] for solving the linear system $Bd = -g$. We either obtain an unconstrained solution with a sufficient precision or stop on the trust-region boundary. The latter possibility occurs if either a negative curvature is encountered or the constraint is violated. This method is based on the fact that $Q(d_{j+1}) < Q(d_j)$ and $\|d_{j+1}\| > \|d_j\|$ hold in the subsequent CG iterations if the CG coefficients are positive and no preconditioning used. For a general preconditioner C (symmetric and positive definite), we have $\|d_{j+1}\|_C > \|d_j\|_C$, where $\|d_j\|_C^2 = d_j^T C d_j$.

There are two possibilities how the Steihaug-Toint method can be preconditioned. The first way uses the norms $\|d_i\|_{C_i}$ (instead of $\|d_i\|$) in (1)–(7), where C_i are preconditioners chosen. This possibility is not always efficient because the norms $\|d_i\|_{C_i}$, $i \in \mathcal{N}$, vary considerably in the major iterations and preconditioners C_i, $i \in \mathcal{N}$, can be ill-conditioned. The second way uses Euclidean norms in (1)–(7), even if arbitrary preconditioners C_i, $i \in \mathcal{N}$, are used. In this case, the trust-region can be leaved prematurely and the direction vector obtained can be farther from the optimal locally-constrained step than that obtained without preconditioning. This shortcoming is usually compensated by the rapid convergence of the preconditioned CG method. Our computational experiments indicate that the second way is more efficient in general.

Another approach, the GLRT method [2], approximately solves (8) iteratively by using the symmetric Lanczos process. A vector d_j which is the j-th approximation of optimal d is contained in the Krylov subspace $\mathcal{K}_j = \operatorname{span}\{g, Bg, \dots, B^{j-1}g\}$ of dimension j defined by the matrix B and the vector g. In this case, $d_j = Z\tilde{d}$, where $\tilde{d}$ is obtained by minimizing the quadratic function

$$\frac{1}{2}\,\tilde{d}^T T \tilde{d} + \|g\| e_1^T \tilde{d}$$

subject to $\|\tilde{d}\| \le \Delta$. Here, $T = Z^T B Z$ (with $Z^T Z = I$) is the Lanczos tridiagonal matrix and e_1 is the first column of the unit matrix. Using preconditioner C, the preconditioned Lanczos method generates basis such that $Z^T C Z = I$. Thus, we have to use the norms $\|d_i\|_{C_i}$ in (1)–(7), i.e. the first way of preconditioning, which can be inefficient when C_i vary considerably in the trust-region iterations or are ill-conditioned.

3. A shifted Steihaug-Toint method

In this contribution, we consider a sequence of subproblems

$$d_j = \arg\min_{d \in \mathcal{K}_j} Q(d) \quad \text{subject to} \quad \|d\| \le \Delta, \quad \text{where} \quad Q(d) = \frac{1}{2}\, d^T B d + g^T d,$$

with corresponding Lagrange multipliers λ_j, $j \in \{1, \dots, n\}$. The method [3] uses the conjugate gradient method applied to the linear system $(B + \tilde{\lambda} I)d + g = 0$, where $\tilde{\lambda}$ is an approximation to the optimal Lagrange multiplier λ. For this reason, we need to investigate the properties of the Lagrange multipliers corresponding to the trust-region subproblems used.

Before showing the main result, we first present several lemmas, which lead to the main theorem. The first lemma, coming from [7], shows a simple property of the conjugate gradient method, the second one compares Krylov subspaces of the matrices B and $B + \lambda I$, and the last one states a relation between the values of the Lagrange multipliers and the norms of the direction vectors.

Lemma 1. *Let B be a symmetric and positive definite matrix, let*

$$\mathcal{K}_j = \operatorname{span}\{g, Bg, \ldots, B^{j-1}g\}, \quad j \in \{1, \ldots, n\},$$

be the j-th Krylov subspace given by the matrix B and the vector g. Let

$$d_j = \arg\min_{d \in \mathcal{K}_j} Q(d), \quad \textit{where} \quad Q(d) = \frac{1}{2}\, d^T B d + g^T d.$$

If $1 \le k \le l \le n$, then

$$\|d_k\| \le \|d_l\|.$$

Especially,

$$\|d_k\| \le \|d_n\|, \quad \textit{where} \quad d_n = \arg\min_{d \in \mathcal{R}^n} Q(d).$$

Lemma 2. *Let $\lambda \in \mathcal{R}$ and*

$$\mathcal{K}_j(\lambda) = \operatorname{span}\{g, (B + \lambda I)g, \ldots, (B + \lambda I)^{j-1}g\}, \quad j \in \{1, \ldots, n\},$$

be the j-dimensional Krylov subspace generated by the matrix $B + \lambda I$ and the vector g. Then

$$\mathcal{K}_j(\lambda) = \mathcal{K}_j(0).$$

Lemma 3. *Let $Z_j^T B Z_j + \lambda_k I$, $\lambda_k \in \mathcal{R}$, $k \in \{1, 2\}$, where $Z_j \in \mathcal{R}^{n \times j}$ is a matrix whose columns form an orthonormal basis for $\mathcal{K}_j$, be symmetric and positive definite. Let*

$$d_j(\lambda_k) = \arg\min_{d \in \mathcal{K}_j} Q_{\lambda_k}(d), \quad \textit{where} \quad Q_\lambda(d) = \frac{1}{2}\, d^T (B + \lambda I) d + g^T d.$$

Then

$$\lambda_2 \le \lambda_1 \quad \Leftrightarrow \quad \|d_j(\lambda_2)\| \ge \|d_j(\lambda_1)\|.$$

Now we are in a position to present the main theorem.

Theorem 1. *Let d_j, $j \in \{1, \ldots, n\}$, be solutions of the minimization problems*

$$d_j = \arg\min_{d \in \mathcal{K}_j} Q(d) \quad \textit{subject to} \quad \|d\| \le \Delta, \quad \textit{where} \quad Q(d) = \frac{1}{2}\, d^T B d + g^T d,$$

with corresponding Lagrange multipliers λ_j, $j \in \{1, \ldots, n\}$. If $1 \le k \le l \le n$, then

$$\lambda_k \le \lambda_l.$$

4. Applications

The result of Theorem 1 can be applied to the following idea. We apply the Steihaug-Toint method to a shifted subproblem

$$\min\ \tilde{Q}(d) = Q_{\tilde{\lambda}}(d) = \frac{1}{2}\,d^T(B+\tilde{\lambda}I)d + g^Td \quad \text{subject to} \quad \|d\| \le \Delta, \tag{9}$$

where $\tilde{\lambda}$ is an approximation to the optimal λ. If we set $\tilde{\lambda} = \lambda_j$ for some $j \le n$, then Theorem 1 implies that $0 \le \tilde{\lambda} = \lambda_j \le \lambda_n = \lambda$. As a consequence of this inequality, one has that $\lambda = 0$ implies $\tilde{\lambda} = 0$ so that $\|\tilde{d}\| < \Delta$ implies $\tilde{\lambda} = 0$. Thus, the shifted Steihaug-Toint method proposed in [3] reduces to the standard Steihaug-Toint method in this case. At the same time, if B is positive definite and $0 < \tilde{\lambda} \le \lambda$, then one has $\Delta = \|(B+\lambda I)^{-1}g\| \le \|(B+\tilde{\lambda}I)^{-1}g\| < \|B^{-1}g\|$ by Lemma 3. Thus, the unconstrained minimizer of (9) is closer to the trust-region boundary than the unconstrained minimizer of (8) and we can expect that $d(\tilde{\lambda})$ is closer to the optimal locally constrained step than $d(0)$. Finally, if B is positive definite and $\tilde{\lambda} > 0$, then the matrix $B + \tilde{\lambda}I$ is better conditioned than B and we can expect that the shifted Steihaug-Toint method will converge more rapidly than the standard Steihaug-Toint method.

The shifted Steihaug-Toint method for solving subproblem (8) consists of the three major steps.

1. Carry out $j \ll n$ steps (usually $j = 5$) of the unpreconditioned Lanczos method to obtain the tridiagonal matrix $T \equiv T_j = Z_j^T B Z_j$.

2. Solve the subproblem

   $$\min\ \frac{1}{2}\,\tilde{d}^T T \tilde{d} + \|g\| e_1^T \tilde{d} \quad \text{subject to} \quad \|\tilde{d}\| \le \Delta,$$

   using the method of Moré and Sorensen, to obtain the Lagrange multiplier $\tilde{\lambda}$.

3. Apply the (preconditioned) Steihaug-Toint method to the shifted subproblem

   $$\min\ \tilde{Q}(d) \quad \text{subject to} \quad \|d\| \le \Delta$$

   to obtain the direction vector $d = d(\tilde{\lambda})$, a suitable approximation to the solution of problem (8).

5. Numerical experiments

We present a numerical comparison of methods for computing direction vectors satisfying conditions (1)-(3):

- MS – the method of Moré and Sorensen [5] for computing the optimal locally constrained step.

- ST – the basic (unpreconditioned) Steihaug [7] and Toint [9] method.
- SST – the basic (unpreconditioned) shifted Steihaug-Toint method [3].
- GLRT – the method of Gould, Lucidi, Roma, and Toint [2] which combines CG method with the Lanczos process to give a good approximation to the optimal locally constrained step.
- PST – the preconditioned Steihaug-Toint method.
- PSST – the preconditioned shifted Steihaug-Toint method.

Note that the incomplete Choleski preconditioner is used for methods PST and PSST, the number of extra CG or Lanczos steps in SST and PSST methods is equal to 5, and the number of Lanczos vectors in the GLRT method is bounded from above by 100.

The methods were tested by using two collections of 22 sparse test problems with 1000 and 5000 variables (subroutines `TEST14` and `TEST15` described in [4], which can be downloaded from `www.cs.cas.cz/luksan/test.html`). The results are given in Tables 1 and 2, where `NIT` is the total number of iterations, `NFV` is the total number of function evaluations, `NFG` is the total number of gradient evaluations, `NDC` is the total number of Choleski-type decompositions (complete for method MS and incomplete for methods PST, PSST), `NMV` is the total number of matrix-vector multiplications, and `Time` is the total computational time in seconds. Note that `NFG` is much greater than `NFV` in Table 1, since the Hessian matrices are computed by using gradient differences. At the same time, the problems referred in Table 2 are the sums of squares having the form $F(x) = (1/2)f^T(x)f(x)$ and `NFV` denotes the total number of vector $f(x)$ evaluations. Since $f(x)$ is used in the expression $g(x) = J^T(x)f(x)$, where $J(x)$ is the Jacobian matrix of $f(x)$, `NFG` is comparable with `NFV` in this case.

Results in Tables 1 and 2 require several comments. All problems are sparse with a simple sparsity pattern. For this reason, the MS method based on complete Choleski-type decompositions (CD) is very efficient, much better than unpreconditioned methods based on matrix-vector multiplications (MV). Since `TEST14` contains reasonably conditioned problems, the preconditioned MV methods are competitive with the CD method. On the contrary, `TEST15` contains several very ill-conditioned problems (one of them had to be removed) and thus, the CD method works better than the MV methods.

6. Conclusion

Our conclusions concern large-scale problems where the sparsity pattern plays a considerable role. The Moré-Sorensen method is very efficient for ill-conditioned but reasonably sparse problems. If the problems do not have sufficiently sparse Hessian matrices, then this method can be much worse than the Steihaug-Toint method whose efficiency also strongly depends on suitable preconditioning. There are two possibilities of preconditioning mentioned in Section 2. The first one changes the trust-region problem whereas the second one deforms the trust-region path in the

N	Method	NIT	NFV	NFG	NDC	NMV	Time
1000	MS	1911	1952	8724	3331	1952	3.13
	ST	3475	4021	17242	0	63016	5.44
	SST	3149	3430	15607	0	75044	5.97
	GLRT	3283	3688	16250	0	64166	5.40
	PST	2608	2806	12802	2609	5608	3.30
	PSST	2007	2077	9239	2055	14440	2.97
5000	MS	8177	8273	34781	13861	8272	49.02
	ST	16933	19138	84434	0	376576	134.52
	SST	14470	15875	70444	0	444142	146.34
	GLRT	14917	16664	72972	0	377588	132.00
	PST	11056	11786	53057	11057	23574	65.82
	PSST	8320	8454	35629	8432	59100	45.57

Tab. 1: *Comparison of methods using TEST14.*

N	Method	NIT	NFV	NFG	NDC	NMV	Time
1000	MS	1946	9094	9038	3669	2023	5.86
	ST	2738	13374	13030	0	53717	11.11
	SST	2676	13024	12755	0	69501	11.39
	GLRT	2645	12831	12547	0	61232	11.30
	PST	3277	16484	16118	3278	31234	11.69
	PSST	2269	10791	10613	2446	37528	8.41
5000	MS	7915	33607	33495	14099	8047	89.69
	ST	11827	54699	53400	0	307328	232.70
	SST	11228	51497	50333	0	366599	231.94
	GLRT	10897	49463	48508	0	300580	214.74
	PST	9360	41524	41130	9361	179166	144.40
	PSST	8634	37163	36881	8915	219801	140.44

Tab. 2: *Comparison of methods using TEST15.*

original trust-region problem. Note that the GLRT method cannot be preconditioned in the second way since the preconditioned Lanczos process does not generate the orthonormal basis related to the original trust-region subproblem. Our preliminary tests have shown that the first preconditioning technique is less efficient because it failed in many cases.

To sum up, the shifted Steihaug-Toint method combines good properties of the Moré-Sorensen and the Steihaug-Toint methods. Our computational experiments indicate that this method works well in connection with the second way of preconditioning. The point on the trust-region boundary obtained by this method is usually closer to the optimal solution in comparison with the point obtained by the original Steihaug-Toint method.

References

[1] A.R. Conn, N.I.M. Gould, P.L. Toint: *Trust-region methods.* SIAM, Philadelphia, 2000.

[2] N.I.M. Gould, S. Lucidi, M. Roma, P.L. Toint: *Solving the trust-region subproblem using the Lanczos method.* Report No. RAL-TR-97-028, Rutherford Appleton Laboratory, 1997.

[3] L. Lukšan, C. Matonoha, J. Vlček: *A shifted Steihaug-Toint method for computing a trust-region step.* Report No. V914-04, Institute of Computer Science, Academy of Sciences, Czech Republic, 2004.

[4] L. Lukšan, J. Vlček: *Sparse and partially separable test problems for unconstrained and equality constrained optimization.* Report No. V767-98, Institute of Computer Science, Academy of Sciences, Czech Republic, 1998.

[5] J.J. Moré, D.C. Sorensen: *Computing a trust region step.* SIAM J. Sci. Statist. Comput. **4** (1983), 553–572.

[6] J. Nocedal, S.J. Wright: *Numerical optimization.* Springer, New York, 1999.

[7] T. Steihaug: *The conjugate gradient method and trust regions in large-scale optimization.* SIAM J. Numer. Anal. **20** (1983), 626–637.

[8] D.C. Sorensen: *Newton's method with a model trust region modification.* SIAM J. Numer. Anal. **19(2)** (1982), 409–426.

[9] P.L. Toint: *Towards an efficient sparsity exploiting Newton method for minimization.* In: I.S. Duff (Ed.), Sparse Matrices and Their Uses. Academic Press, London, 1981, pp. 57–88.

PRIMAL INTERIOR POINT METHOD FOR GENERALIZED MINIMAX FUNCTIONS[*]

Ladislav Lukšan, Ctirad Matonoha, Jan Vlček

Introduction

Generalized minimax optimization covers many practical problems, e.g., l_1 and l_∞ approximation or classic minimax optimization. In this contribution, we summarize new results described in our previous works [1]–[4], which can be downloaded from `http://www.cs.cas.cz/luksan/reports.html`. In these works, a connection with the current research and additional references are shown.

Definition 1 *We say that $F(x)$ is a generalized minimax function if*

$$F(x) = h(F_1(x), \dots, F_m(x)), \quad F_i(x) = \max_{1 \le j \le n_i} f_{ij}(x), \quad 1 \le i \le m,$$

where $h : R^m \to R$ and $f_{ij} : R^n \to R$, $1 \le i \le m$, $1 \le j \le n_i$, are smooth functions satisfying the following assumptions.

Assumption 1. Functions $F_i(x)$, $1 \le i \le m$, are bounded from below on R^n: there are $\underline{F}_i \in R$ such that $F_i(x) \ge \underline{F}_i$, $1 \le i \le m$, for all $x \in R^n$.

Assumption 2. Function $h(z)$ is twice continuously differentiable and convex satisfying

$$\partial h(z)/\partial z_i \ge \underline{h}_i > 0, \quad 1 \le i \le m,$$

for every $z \in Z = \{z \in R^m : z_i \ge \underline{F}_i,\ 1 \le i \le m\}$ (vector $z \in R^m$ will be called the minimax vector).

Assumption 3. Functions $f_{ij}(x)$, $1 \le i \le m$, $1 \le j \le n_i$, are twice continuously differentiable on the convex hull of the level set

$$\mathcal{L}(\overline{F}) = \{x \in R^n : F_i(x) \le \overline{F},\ 1 \le i \le m\}$$

for a sufficiently large upper bound $\overline{F}$ and they have bounded the first and second-order derivatives on conv$\mathcal{L}(\overline{F})$: there are $\overline{g}$ and $\overline{G}$ such that $\|\nabla f_{ij}(x)\| \le \overline{g}$ and $\|\nabla^2 f_{ij}(x)\| \le \overline{G}$ for all $1 \le i \le m$, $1 \le j \le n_i$ and $x \in \text{conv}\mathcal{L}(\overline{F})$.

[*]This work was supported by the Grant Agency of the Czech Academy of sciences, project No. IAA1030405, the Grant Agency of the Czech Republic, project No. 201-06-P397, and the institutional research plan No. AV0Z10300504.

Unconstrained minimization of function $F(x)$ is equivalent to the nonlinear programming problem: Minimize the function

$$h(z_1, \dots, z_m)$$

with constraints

$$f_{ij}(x) \le z_i, \quad 1 \le i \le m, \quad 1 \le j \le n_i,$$

(conditions $\partial h(z)/\partial z_i \ge \underline{h}_i > 0$, $1 \le i \le m$, for $z \in Z$ are sufficient for satisfying equalities $z_i = F_i(x)$, $1 \le i \le m$, at the minimum point). The necessary first-order (KKT) conditions for a solution of this problem have the form

$$\sum_{i=1}^{m} \sum_{j=1}^{n_i} u_{ij} \nabla f_{ij}(x) = 0, \quad \sum_{j=1}^{n_i} u_{ij} = \frac{\partial h(z)}{\partial z_i},$$

$$u_{ij} \ge 0, \quad z_i - f_{ij}(x) \ge 0, \quad u_{ij}(z_i - f_{ij}(x)) = 0, \quad 1 \le j \le n_i,$$

where u_{ij}, $1 \le i \le m$, $1 \le j \le n_i$, are Lagrange multipliers.

Nonlinear programming problem can be solved by using the primal interior point method. For this reason we apply the Newton minimization method to the sequence of barrier functions

$$B_\mu(x, z) = h(z) + \mu \sum_{i=1}^{m} \sum_{j=1}^{n_i} \varphi(z_i - f_{ij}(x)),$$

assuming $0 < \mu \le \overline{\mu}$ and $\mu \to 0$, where $\varphi : (0, \infty) \to R$ is a barrier which satisfies the following conditions.

Condition 1. $\varphi(t)$, $t \in (0, \infty)$, is a twice continuously differentiable function such that $\varphi(t)$ is decreasing, strictly convex, with $\lim_{t \to 0} \varphi(t) = \infty$, $\varphi'(t)$ is increasing, strictly concave, with $\lim_{t \to \infty} \varphi'(t) = 0$, and $t\varphi'(t)$ is bounded.

Condition 2. $\varphi(t)$, $t \in (0, \infty)$, is bounded from below: there is $\underline{\varphi} \le 0$ such that $\varphi(t) \ge \underline{\varphi}$ for all $t \in (0, \infty)$.

The most known and frequently used logarithmic barrier $\varphi(t) = \log t^{-1} = -\log t$ satisfies Condition 1, but does not satisfy Condition 2, since $\log t \to \infty$ as $t \to \infty$. Therefore, additional barriers have been proposed, for example barrier

$$\varphi(t) \;=\; \log(t^{-1} + 1), \qquad t \in (0, \infty),$$

which is positive ($\underline{\varphi} = 0$), or

$$\begin{aligned} \varphi(t) &= -\log t, && 0 < t \le 1, \\ \varphi(t) &= -(t^{-1} - 4\,t^{-1/2} + 3), && t > 1, \end{aligned}$$

which is bounded from below ($\underline{\varphi} = -3$). Both these barriers satisfy Condition 1 and Condition 2.

Iterative determination of the minimax vector

The necessary conditions for (x,z) to be the minimizer of the barrier function have the form

$$\nabla_x B_\mu(x,z) = -\sum_{i=1}^{m}\sum_{j=1}^{n_i} \nabla f_{ij}(x)\varphi'(z_i - f_{ij}(x)) = 0$$

and

$$\frac{\partial B_\mu(x,z)}{\partial z_i} = h_i(z) + \mu \sum_{j=1}^{n_i} \varphi'(z_i - f_{ij}(x)) = 0, \quad 1 \le i \le m,$$

where $h_i(z) = \partial h(z)/\partial z_i$, $1 \le i \le m$. For solving this system of $n+m$ nonlinear equations, we use the Newton method whose iteration step can be written in the form

$$\begin{bmatrix} W(x,z) & -A_1(x)v_1(x,z) & \dots & -A_m(x)v_m(x,z) \\ -v_1^T(x,z)A_1^T(x) & h_{11}(z) + e_1^T v_1(x,z) & \dots & h_{1m}(z) \\ \dots & \dots & \dots & \dots \\ -v_m^T(x,z)A_m^T(x) & h_{m1}(z) & \dots & h_{mm}(z) + e_m^T v_m(x,z) \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta z_1 \\ \dots \\ \Delta z_m \end{bmatrix}$$

$$= - \begin{bmatrix} \sum_{i=1}^m A_i(x)u_i(x,z) \\ h_1(z) - e_1^T u_1(x,z) \\ \dots \\ h_m(z) - e_m^T u_m(x,z) \end{bmatrix},$$

where

$$W(x,z) = \sum_{i=1}^{m}\sum_{j=1}^{n_i} \nabla^2 f_{ij}(x) u_{ij}(x,z) + \sum_{i=1}^{m}\sum_{j=1}^{n_i} \nabla f_{ij}(x) v_{ij}(x,z) (\nabla f_{ij}(x))^T,$$

$$u_{ij}(x,z) = -\mu\varphi'(z_i - f_{ij}(x)), \quad v_{ij}(x,z) = \mu\varphi''(z_i - f_{ij}(x)),$$

$$h_{ij}(z) = \frac{\partial^2 h(z)}{\partial z_i \partial z_j}, \quad 1 \le i \le m, \quad 1 \le j \le n_i,$$

and where $A_i(x) = [\nabla f_{i1}(x), \dots, \nabla f_{in_i}(x)]$,

$$u_i(x,z) = \begin{bmatrix} u_{i1}(x,z) \\ \vdots \\ u_{in_i}(x,z) \end{bmatrix}, \quad v_i(x,z) = \begin{bmatrix} v_{i1}(x,z) \\ \vdots \\ v_{in_i}(x,z) \end{bmatrix}, \quad e_i = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}.$$

This formula can be easily verified by the differentiation of the KKT conditions by vectors x and z. Setting

$$C(x,z) = [A_1(x)v_1(x,z), \dots, A_m(x)v_m(x,z)],$$

$$g(x,z) = \sum_{i=1}^{m} A_i(x) u_i(x,z),$$

$$\Delta z = \begin{bmatrix} \Delta z_1 \\ \dots \\ \Delta z_m \end{bmatrix}, \quad c(x,z) = \begin{bmatrix} h_1(z) - e_1^T u_1(x,z) \\ \dots \\ h_m(z) - e_m^T u_m(x,z) \end{bmatrix},$$

$$H(z) = \nabla^2 h(z), \quad V(x,z) = \mathrm{diag}(e_1^T v_1(x,z), \dots, e_m^T v_m(x,z)),$$

we can rewrite the Newton system in the form

$$\begin{bmatrix} W(x,z) & -C(x,z) \\ -C^T(x,z) & H(z) + V(x,z) \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta z \end{bmatrix} = - \begin{bmatrix} g(x,z) \\ c(x,z) \end{bmatrix}.$$

Now, let us have a problem, which is large-scale (the number of variables n is large), but partially separable (the functions $f_{ij}(x)$, $1 \le i \le m$, $1 \le j \le n_i$, depend on a small number of variables). Then we can assume that the matrix $W(x,z)$ is sparse and it can be efficiently decomposed. Two cases will be investigated. If m is small (for example in the classic minimax problems, where $m = 1$), we use the fact that

$$\begin{bmatrix} W & -C \\ -C^T & H + V \end{bmatrix}^{-1} =$$

$$\begin{bmatrix} W^{-1} - W^{-1}C(C^T W^{-1} C - H - V)^{-1} C^T W^{-1} & -W^{-1}C(C^T W^{-1} C - H - V)^{-1} \\ -(C^T W^{-1} C - H - V)^{-1} C^T W^{-1} & -(C^T W^{-1} C - H - V)^{-1} \end{bmatrix}.$$

The solution is determined from the formulas

$$\Delta z = (C^T W^{-1} C - H - V)^{-1} (C^T W^{-1} g + c),$$

$$\Delta x = W^{-1}(C \Delta z - g).$$

In this case, we need to decompose the large sparse matrix W of order n and the small dense matrix $C^T W^{-1} C - H - V$ of order m.

In the second case, we assume that the numbers n_i, $1 \le i \le m$, are small and the matrix $H(z)$ is diagonal (as in the sums of absolute values). Denoting $D = H(z) + V(x,z)$, the matrix

$$C(x,z) D^{-1}(x,z) C^T(x,z) = C(x,z)(H(z) + V(x,z))^{-1} C^T(x,z)$$

is sparse and we can use the fact that

$$\begin{bmatrix} W & -C \\ -C^T & D \end{bmatrix}^{-1} =$$

$$\begin{bmatrix} (W - CD^{-1}C^T)^{-1} & (W - CD^{-1}C^T)^{-1} CD^{-1} \\ D^{-1}C^T(W - CD^{-1}C^T)^{-1} & D^{-1} + D^{-1}C^T(W - CD^{-1}C^T)^{-1} CD^{-1} \end{bmatrix}.$$

The solution is determined from the formulas

$$\Delta x = -(W - CD^{-1}C^T)^{-1}(g + CD^{-1}c),$$

$$\Delta z = D^{-1}(C^T \Delta x - c).$$

In this case, we need to decompose the large sparse matrix $W - CD^{-1}C^T$ of order n. The inversion of the diagonal matrix D of order m is trivial.

In every step of the primal interior point method with the iterative determination of the minimax vector, we know the value of the parameter μ and the vectors $x \in R^n$, $z \in R^m$ such that $z_i > F_i(x)$, $1 \le i \le m$. Using the Newton system, we determine direction vectors Δx, Δz and select a step-size α in such a way that

$$B_\mu(x + \alpha\Delta x, z + \alpha\Delta z) < B_\mu(x, z)$$

and $z_i^+ > F_i(x^+)$, $1 \le i \le m$. Finally, we set $x^+ = x + \alpha\Delta x$, $z^+ = z + \alpha\Delta z$ and determine a new value $\mu^+ < \mu$. The above inequality is satisfied for sufficiently small values of the step-size α, if the matrix of the Newton system is positive definite.

Theorem 1. *Let the matrix $G = \sum_{i=1}^m \sum_{j=1}^{n_i} \nabla^2 f_{ij} u_{ij}$ be positive definite. Then the matrix of the Newton system is positive definite.*

Direct determination of the minimax vector

Minimization of the barrier function can be considered as the two-level optimization

$$z(x) = \arg\min_{z \in Z} B_\mu(x, z),$$

$$x = \arg\min_{x \in R^n} B(x; \mu), \quad B(x; \mu) \overset{\Delta}{=} B_\mu(x, z(x)),$$

where Z is the set used in Assumption 2. The first equation serves for the determination of the optimal vector $z(x) \in R^m$ corresponding to a given vector $x \in R^n$. Assuming x fixed, function $B_\mu(x, z)$ is strictly convex (as a function of vector z), since it is a sum of convex function $h(z)$ and strictly convex functions $\mu\varphi(z_i - f_{ij}(x))$, $1 \le i \le m$, $1 \le j \le n_i$. As a stationary point, its minimum is uniquely determined by the KKT conditions. The following theorem holds for the logarithmic barrier.

Theorem 2. *The system of equations*

$$h_i(z) - \sum_{j=1}^{n_i} \frac{\mu}{z_i - f_{ij}(x)} = 0, \quad h_i(z) = \frac{\partial h(z)}{\partial z_i}, \quad 1 \le i \le m,$$

with $x \in R^n$ fixed, has the unique solution $z(x; \mu) \in Z \subset R^m$ such that

$$F_i(x) < \underline{z}_i \le z_i(x; \mu) \le \overline{z}_i, \quad 1 \le i \le m,$$

where

$$\underline{z}_i = F_i(x) + \mu/\overline{h}_i, \quad \overline{z}_i = F_i(x) + n_i\mu/\underline{h}_i,$$

and where $\underline{h}_i > 0$ are bounds used in Assumption 2 and $\overline{h_i} = h_i(\overline{z}_1, \dots, \overline{z}_m)$.

Similar results can be obtained for other barriers as well. Using barrier

$$\varphi(t) = \log(t^{-1} + 1),$$

we get equations

$$h_i(z) - \sum_{j=1}^{n_i} \frac{\mu}{(z_i - f_{ij}(x))(z_i - f_{ij}(x) + 1)} = 0, \quad 1 \le i \le m,$$

and inequalities for $z_i(x;\mu)$ with bound

$$\underline{z}_i = F_i(x) + \frac{2\mu/\overline{h}_i}{1 + \sqrt{1 + 4\mu/\overline{h}_i}}, \quad \overline{z}_i = F_i(x) + \frac{2n_i\mu/\underline{h}_i}{1 + \sqrt{1 + 4n_i\mu/\underline{h}_i}}.$$

The system of nonlinear equations can be solved by the Newton method started, e.g., from the point z such that $z_i = \overline{z}_i$, $1 \le i \le m$. If the Hessian matrix of the function $h(z)$ is diagonal, this system is decomposed on m scalar equations, which can be efficiently solved by robust methods. If we are able to find a solution of the nonlinear system for an arbitrary vector $x \in R^n$, we can restrict our attention to the unconstrained minimization of the function $B(x;\mu) = B_\mu(x, z(x;\mu))$, which has n variables. It is suitable to know the gradient and the Hessian matrix of this function.

Theorem 3. *One has*

$$\nabla B(x;\mu) = \sum_{i=1}^{m} A_i(x) u_i(x),$$

$$\nabla^2 B(x;\mu) = W(x, z(x)) - C(x, z(x)) D(x, z(x))^{-1} C^T(x, z(x)),$$

where $W(x, z(x))$, $C(x, z(x))$, $H(z(x))$, $V(x, z(x))$ are matrices introduced in the previous section and $D(x, z(x)) = H(z(x)) + V(x, z(x))$. If the matrix $H(z(x))$ is diagonal, we can express the Hessian matrix in the form

$$\begin{aligned} \nabla^2 B(x;\mu) \;&=\; G(x, z(x)) + \sum_{i=1}^{m} A_i(x) V_i(x, z(x)) A_i^T(x) \\ &-\; \sum_{i=1}^{m} \frac{A_i(x) V_i(x, z(x)) e_i e_i^T V_i(x, z(x)) A_i^T(x)}{\partial^2 h(z(x))/\partial z_i^2 + e_i^T V_i(x, z(x)) e_i}, \end{aligned}$$

where $A_i(x)$, $V_i(x, z(x))$, $1 \le i \le m$, and $G(x, z(x))$ are matrices introduced in the previous section.

To determine the Hessian matrix inverse, we can use the relation obtained by the decomposition of the Newton system described in the previous section. Using substitution $c(x, z(x)) = 0$ we get

$$
\begin{aligned}
(\nabla^2 B_\mu(x))^{-1} \;\; = \;\; & W(x,z(x))^{-1} - W(x,z(x))^{-1} C(x,z(x)) \\
& \Big(C^T(x,z(x))\, W^{-1}(x,z(x))\, C(x,z(x)) - H(z(x)) - V(x,z(x)) \Big)^{-1} \\
& C^T(x,z(x))\, W(x,z(x))^{-1}.
\end{aligned}
$$

If the nonlinear system is not solved with a sufficient precision, we rather use the Newton system from the previous section, where the actual vector $c(x, z(x;\mu)) \neq 0$ is substituted.

In every step of the primal interior point method with the direct determination of the minimax vector, we know the value of the parameter μ and the vector $x \in R^n$. Solving the nonlinear system we determine the vector $z(x)$. Using the Hessian matrix or its inverse, we determine a direction vector Δx and select a step-size α in such a way that

$$B_\mu(x + \alpha\Delta x, z(x + \alpha\Delta x;\mu)) < B_\mu(x, z(x;\mu))$$

(the vector $z(x+\alpha\Delta x;\mu)$ is obtained as a solution of the nonlinear system, in which x is replaced by $x+\alpha\Delta x$). Finally, we set $x^+ = x + \alpha\Delta x$ and determine a new value $\mu^+ < \mu$. Conditions for the direction vector Δx to be descent are the same as in Theorem 1. It suffices when the matrix $G(x, z(x))$ is positive definite.

Now, we describe an algorithm, in which the direction vector $d = \Delta x$ is determined in such a way that

$$-g^T d \geq \varepsilon_0 \|g\| \|d\|, \quad \underline{c}\|g\| \leq \|d\| \leq \overline{c}\|g\|$$

(uniform descent) where $g = A(x)u(x;\mu)$.

Algorithm 1.

Data: Termination parameter $\underline{\varepsilon} > 0$, precision for the nonlinear equation solver $\underline{\delta} > 0$, bounds for the barrier parameter $0 < \underline{\mu} < \overline{\mu}$, rate of the barrier parameter decrease $0 < \lambda < 1$, restart parameters $0 < \underline{c} < \overline{c}$ and $\varepsilon_0 > 0$, line search parameter $\varepsilon_1 > 0$, rate of the step-size decrease $0 < \beta < 1$, step bound $\overline{\Delta} > 0$, way of direction determination $\mathcal{D}$ ($\mathcal{D} = 1$ or $\mathcal{D} = 2$).

Input: Sparsity pattern of matrix $A(x)$. Initial estimation of vector x.

Step 1: *Initiation.* Set $\mu = \overline{\mu}$. If $\mathcal{D} = 1$, determine the sparsity pattern of matrix $W = W(x;\mu)$ from the sparsity pattern of matrix $A(x)$ and carry out a symbolic decomposition of W. If $\mathcal{D} = 2$, determine the sparsity pattern of matrices $W = W(x;\mu)$ and $C = C(x;\mu)$ from the sparsity pattern of matrix $A(x)$ and carry out a symbolic decomposition of matrix $W - CD^{-1}C^T$. Compute values $f_{ij}(x)$, $1 \leq i \leq m$, $1 \leq j \leq n_i$, $F_i(x) = \max_{1 \leq j \leq n_i} f_{ij}(x)$, $1 \leq i \leq m$, and $F(x) = h(F_1(x), \dots, F_m(x))$. Set $k := 0$ (iteration count) and $r := 0$ (restart indicator).

Step 2: *Termination.* Solving the nonlinear system with precision $\underline{\delta}$, obtain vectors $z(x;\mu)$ and $u(x;\mu)$. Compute matrix $A := A(x)$ and vector $g := g(x;\mu) = A(x)u(x;\mu)$. If $\mu \le \underline{\mu}$ and $\|g\| \le \underline{\varepsilon}$, then terminate the computation. Otherwise set $k := k+1$.

Step 3: *Approximation of the Hessian matrix.* Set $G = G(x;\mu)$ or compute an approximation G of the Hessian matrix $G(x;\mu)$ by using either gradient differences or variable metric updates.

Step 4: *Direction determination.* If $\mathcal{D} = 1$, determine vector $d = \Delta x$ by using the Gill-Murray decomposition of matrix W. If $\mathcal{D} = 2$, determine vector $d = \Delta x$ by using the Gill-Murray decomposition of matrix $W - CD^{-1}C^T$.

Step 5: *Restart.* If $r = 0$ and the direction vector is not uniformly descent, select a positive definite diagonal matrix $\tilde{D}$, set $G = \tilde{D}$, $r := 1$ and go to Step 4. If $r = 1$ and the direction vector is not uniformly descent, set $d := -g$ (the steepest descent direction). Set $r := 0$.

Step 6: *Step-length selection.* Define the maximum step-length $\overline{\alpha} = \min(1, \overline{\Delta}/\|d\|)$. Find a minimum integer $l \ge 0$ such that $B(x + \beta^l \overline{\alpha} d;\mu) \le B(x;\mu) + \varepsilon_1 \beta^l \overline{\alpha} g^T d$ (the nonlinear system has to be solved at all points $x + \beta^j \overline{\alpha} d$, $0 \le j \le l$). Set $x := x + \beta^l \overline{\alpha} d$. Compute values $f_{ij}(x)$, $1 \le i \le m$, $1 \le j \le n_i$, $F_i(x) = \max_{1 \le j \le n_i} f_{ij}(x)$, $1 \le i \le m$, and $F(x) = h(F_1(x), \dots, F_m(x))$.

Step 7: *Barrier parameter update.* Determine a new value of the barrier parameter $\mu \ge \underline{\mu}$ by Procedure A or Procedure B. Go to Step 2.

Procedure A.

Phase 1: If $\|g(x_k;\mu_k)\| \ge \underline{g}$, set $\mu_{k+1} = \mu_k$, i.e., the barrier parameter is not changed.
Phase 2: If $\|g(x_k;\mu_k)\| < \underline{g}$, set

$$\mu_{k+1} = \max\left(\tilde{\mu}_{k+1},\, \underline{\mu}\right),$$

where

$$\tilde{\mu}_{k+1} = \min\left[\max\left(\lambda\mu_k,\, \frac{\mu_k}{\sigma\mu_k + 1}\right), \max(\|g(x_k;\mu_k)\|^2,\, 10^{-2k})\right].$$

The values $\underline{\mu} = 10^{-10}$, $\lambda = 0.85$, and $\sigma = 100$ are chosen as defaults.

Procedure B.

Phase 1: If $\|g(x_k;\mu_k)\|^2 \ge \rho\mu_k$, set $\mu_{k+1} = \mu_k$, (the barrier parameter is not changed).
Phase 2: If $\|g(x_k;\mu_k)\|^2 < \rho\mu_k$, set

$$\mu_{k+1} = \max(\underline{\mu}, \|g_k(x_k;\mu_k)\|^2).$$

The values $\underline{\mu} = 10^{-10}$ and $\rho = 0.1$ are chosen as defaults.

Global convergence for bounded barriers

We first assume that function $\varphi(t)$ is bounded from below, $\underline{\delta} = \underline{\varepsilon} = \underline{\mu} = 0$ and all computations are exact. We will investigate an infinite sequence $\{x_k\}_1^\infty$ generated by Algorithm 1. Proofs of all assertions are given in [4].

Lemma 1. *Let Assumption 1, Assumption 2, Condition 1, Condition 2 be satisfied. Let $\{x_k\}_1^\infty$ and $\{\mu_k\}_1^\infty$ be sequences generated by Algorithm 1. Then sequences $\{B(x_k;\mu_k)\}_1^\infty$, $\{z(x_k;\mu_k)\}_1^\infty$, and $\{F(x_k)\}_1^\infty$ are bounded. Moreover, there is $L \geq 0$ such that*

$$B(x_{k+1};\mu_{k+1}) \leq B(x_{k+1};\mu_k) + L(\mu_k - \mu_{k+1}) \quad \forall k \in N.$$

Lemma 2. *Let assumptions of Lemma 1 and Assumption 3 be satisfied. Then the values $\{\mu_k\}_1^\infty$, generated by Algorithm 1, form a non-increasing sequence such that $\mu_k \to 0$.*

Theorem 4. *Let assumptions of Lemma 1 and Assumption 3 be satisfied. Consider a sequence $\{x_k\}_1^\infty$ generated by Algorithm 1 (with $\underline{\delta} = \underline{\varepsilon} = \underline{\mu} = 0$). Then*

$$\lim_{k\to\infty} \sum_{i=1}^{m} \sum_{j=1}^{n_i} u_{ij}(x_k;\mu_k) \nabla f_{ij}(x_k) = 0, \quad \sum_{j=1}^{n_i} u_{ij}(x_k;\mu_k) = h_i(z(x_k;\mu_k)),$$

$$u_{ij}(x_k;\mu_k) \geq 0, \quad z_i(x_k;\mu_k) - f_{ij}(x_k) \geq 0,$$
$$\lim_{k\to\infty} u_{ij}(x_k;\mu_k)(z_i(x_k;\mu_k) - f_{ij}(x_k)) = 0$$

pro $1 \leq i \leq m$ a $1 \leq j \leq n_i$.

Corollary 1. *Let assumptions of Theorem 4 hold. Then every cluster point $x \in R^n$ of the sequence $\{x_k\}_1^\infty$ satisfies KKT conditions of the original problem, where z and u (with elements z_i and u_{ij}, $1 \leq i \leq m$, $1 \leq j \leq n_i$) are cluster points of sequences $\{z(x_k;\mu_k)\}_1^\infty$ and $\{u(x_k;\mu_k)\}_1^\infty$.*

Theorem 5. *Consider the sequence $\{x_k\}_1^\infty$ generated by Algorithm 1. Let assumptions of Lemma 1 and Assumption 3 hold. Then, choosing $\underline{\delta} > 0$, $\underline{\varepsilon} > 0$, $\underline{\mu} > 0$ arbitrarily, there is an index $k \geq 1$ such that*

$$\|g(x_k;\mu_k)\| \leq \underline{\varepsilon}, \quad |h_i(z(x_k;\mu_k)) - \sum_{j=1}^{n_i} u_{ij}(x_k;\mu_k)| \leq \underline{\delta},$$

$$u_{ij}(x_k;\mu_k) \geq 0, \quad z_i(x_k;\mu_k) - f_{ij}(x_k) \geq 0,$$
$$u_{ij}(x_k;\mu_k)(z_i(x_k;\mu_k) - f_{ij}(x_k)) \leq \overline{\mu}$$

for $1 \leq i \leq m$ and $1 \leq j \leq n_i$.

Global convergence for the logarithmic barrier

We first assume that $\varphi(t) = -\log t$, $\underline{\delta} = \underline{\varepsilon} = \underline{\mu} = 0$ and all computations are exact. We will investigate an infinite sequence $\{x_k\}_1^\infty$ generated by Algorithm 1. Proofs of all assertions are given in [4].

Lemma 3. *Let Assumption 2, Assumption 4 be satisfied, $\varphi(t) = -\log t$ and the Hessian matrix $H(z(x;\mu))$ be diagonal. Let $\{x_k\}_1^\infty$ and $\{\mu_k\}_1^\infty$ be sequences generated by Algorithm 1. Then sequences $\{B(x_k;\mu_k)\}_1^\infty$, $\{z(x_k;\mu_k)\}_1^\infty$, and $\{F(x_k)\}_1^\infty$ are bounded. Moreover, there is $L \ge 0$ such that*

$$B(x_{k+1};\mu_{k+1}) \le B(x_{k+1};\mu_k) + L(\mu_k - \mu_{k+1}) \quad \forall k \in N.$$

Lemma 4. *Let assumptions of Lemma 3 and Assumption 3 be satisfied. Then the values $\{\mu_k\}_1^\infty$, generated by Algorithm 1, form a non-increasing sequence such that $\mu_k \to 0$.*

Theorem 6. *Let assumptions of Lemma 3 and Assumption 3 be satisfied. Consider a sequence $\{x_k\}_1^\infty$ generated by Algorithm 1 (with $\underline{\delta} = \underline{\varepsilon} = \underline{\mu} = 0$). Then*

$$\lim_{k\to\infty} \sum_{i=1}^{m} \sum_{j=1}^{n_i} u_{ij}(x_k;\mu_k) \nabla f_{ij}(x_k) = 0, \quad \sum_{j=1}^{n_i} u_{ij}(x_k;\mu_k) = h_i(z(x_k;\mu_k)),$$

$$u_{ij}(x_k;\mu_k) \ge 0, \quad z_i(x_k;\mu_k) - f_{ij}(x_k) \ge 0,$$
$$\lim_{k\to\infty} u_{ij}(x_k;\mu_k)(z_i(x_k;\mu_k) - f_{ij}(x_k)) = 0$$

for $1 \le i \le m$ a $1 \le j \le n_i$.

Corollary 2. *Let assumptions of Theorem 6 hold. Then every cluster point $x \in R^n$ of the sequence $\{x_k\}_1^\infty$ satisfies KKT conditions of the original problem, where z and u (with elements z_i and u_{ij}, $1 \le i \le m$, $1 \le j \le n_i$) are cluster points of sequences $\{z(x_k;\mu_k)\}_1^\infty$ and $\{u(x_k;\mu_k)\}_1^\infty$.*

Theorem 7. *Consider the sequence $\{x_k\}_1^\infty$ generated by Algorithm 1. Let assumptions of Lemma 3 and Assumption 3 hold. Then, choosing $\underline{\delta} > 0$, $\underline{\varepsilon} > 0$, $\underline{\mu} > 0$ arbitrarily, there is an index $k \ge 1$ such that*

$$\|g(x_k;\mu_k)\| \le \underline{\varepsilon}, \quad |h_i(z(x_k;\mu_k)) - \sum_{j=1}^{n_i} u_{ij}(x_k;\mu_k)| \le \underline{\delta},$$

$$u_{ij}(x_k;\mu_k) \ge 0, \quad z_i(x_k;\mu_k) - f_{ij}(x_k) \ge 0,$$
$$u_{ij}(x_k;\mu_k)(z_i(x_k;\mu_k) - f_{ij}(x_k)) \le \overline{\mu}$$

for $1 \le i \le m$ a $1 \le j \le n_i$.

Special cases and numerical results

The simplest generalized minimax function is the sum

$$F(x) = \sum_{i=1}^{m} F_i(x) = \sum_{i=1}^{m} \max_{1 \le j \le n_i} f_{ij}(x).$$

In this case, $\partial h(z)/\partial z_i = 1$, $1 \le i \le m$, for an arbitrary vector z and the matrix $H(z)$ is diagonal. The nonlinear system decomposes on m scalar equations

$$1 - \sum_{j=1}^{n_i} \frac{\mu}{z_i - f_{ij}(x)} = 0, \qquad 1 \le i \le m,$$

whose solutions lie in the intervals

$$F_i(x) + \mu \le z_i(x) \le F_i(x) + n_i\mu, \quad 1 \le i \le m.$$

If $m = 1$, we obtain the classic minimax problems. Numerical experiments for minimax functions were carried out using a collection of 22 test problems (Test 14) described in [5]. The source texts can be downloaded from `http://www.cs.cas.cz/luksan/test.html`. Compared methods: P1–the logarithmic barrier, P2–positive barrier, P3–bounded barrier, SM–smoothing method, DI–primal-dual method. The results:

Method	NIT	NFV	NFG	NR	NL	NF	NT	Time
P1-NM	1675	3735	11109	327	-	-	4	1.92
P2-NM	2018	6221	12674	605	-	-	7	2.09
P3-NM	1777	3989	11596	379	1	-	7	2.11
SM-NM	4123	12405	32451	823	-	-	7	9.64
DI-NM	1771	3732	17952	90	1	-	10	6.34
P1-VM	1615	2429	1637	-	-	-	1	1.05
P2-VM	2116	3549	2138	2	-	-	3	1.47
P3-VM	1985	3208	2007	1	-	-	3	1.27
SM-VM	7244	21008	7266	-	1	-	8	9.09
DI-VM	1790	3925	1790	5	1	-	9	4.59

If $n_i = 2$, $1 \le i \le m$, the nonlinear equations are quadratic and their solution has the form

$$z_i(x) = \mu + \frac{f_{i1}(x) + f_{i2}(x)}{2} + \sqrt{\mu^2 + \left(\frac{f_{i1}(x) - f_{i2}(x)}{2}\right)^2}, \quad 1 \le i \le m.$$

This formula can be used in the case when function $h : R^m \to R$ contains absolute values $F_i(x) = |f_i(x)| = \max(f_i(x), -f_i(x))$. Then $f_{i1}(x) = f_i(x)$ a $f_{i2}(x) = -f_i(x)$, so that

$$z_i(x) = \mu + \sqrt{\mu^2 + f_i^2(x)}, \quad 1 \le i \le m.$$

Numerical experiments for sums of absolute values were carried out using a collection of 22 test problems (Test 14) described in [5]. The source texts can be downloaded from `http://www.cs.cas.cz/luksan/test.html`. Compared methods: PT–logarithmic barrier and a trust-region realization, PL–logarithmic barrier and a line-search realization, DI–primal-dual method, BM–bundle variable metric method. The results:

Method	NIT	NFV	NFG	NR	NL	NF	NT	Time
PT-NM	3014	3518	27404	1	-	-	4	4.66
PL-NM	2651	12819	22932	3	1	-	6	5.24
DI-NM	5002	7229	42462	328	1	-	13	33.52
PT-VM	3030	3234	3051	-	-	1	1	1.44
PL-VM	2699	3850	2721	-	-	1	2	1.42
DI-VM	7138	14719	14719	9	2	-	9	86.18
BM-VM	34079	34111	34111	22	1	1	11	25.72

The above tables demonstrate the high efficiency of Algorithm 1. The use of a minimax structure together with the two-level optimization give much better results than the use of standard nonlinear programming methods applied to the equivalent nonlinear programming problem.

References

[1] L. Lukšan, C. Matonoha, J. Vlček: *Interior point method for nonlinear nonconvex optimization.* Numer. Linear Algebra Appl. **11** (2004), 431–453.

[2] L. Lukšan, C. Matonoha, J. Vlček: *Primal interior-point method for large sparse minimax optimization.* Technical Report V-941, Inst. Computer Science, Acad. Sci., Czech Rep., 2005.

[3] L. Lukšan, C. Matonoha, J. Vlček: *Trust-region interior-point method for large sparse l_1 optimization.* Optim. Methods Softw. **22** (2007), 737–753.

[4] L. Lukšan, C. Matonoha, J. Vlček: *Primal interior point method for minimization of generalized minimax functions.* Technical Report V-1017, Inst. Computer Science, Acad. Sci., Czech Rep., 2007.

[5] L. Lukšan, J. Vlček: *Sparse and partially separable test problems for unconstrained and equality constrained optimization*, Report V-767, Prague, Inst. Computer Science, Acad. Sci., Czech Rep., 1998.

THE BOX METHOD AND SOME ERROR ESTIMATION[*]

Jaroslav Mlýnek

Abstract

This article focuses its attention on practical use of the box method for solving certain type of partial differential equations. The heat conduction problem of the oil transformer under stationary load is described by this equation. The knowledge of the transformer operating temperature is important for ensuring correct functionality and lifespan of transformer. We consider an elliptic partial differential equation of second order with the Newton boundary condition on a rectangular domain. The paper contains description of a numerical solution procedure of the heat problem and an estimation of local discretization error. The box method is often called the finite volume method, too. The solution of practical examples are presented as well.

1. Introduction

This paper deals with the stationary heat conduction problem. Our objective is to solve the problem of the relative transformer screening warming with respect to cooling oil of the transformer. The classical formulation of the problem is

$$-\frac{\partial}{\partial x_1}\left(a_{11}\frac{\partial u}{\partial x_1}\right)-\frac{\partial}{\partial x_2}\left(a_{22}\frac{\partial u}{\partial x_2}\right)+cu=f \tag{1}$$

in a rectangular domain $\Omega \subset R^2$ with the Newton boundary condition

$$\alpha u+\frac{\partial u}{\partial n_A}=g. \tag{2}$$

The derivative with respect to conormal in (2) is defined by the relation

$$\frac{\partial u}{\partial n_A}=a_{11}\frac{\partial u}{\partial x_1}n_1+a_{22}\frac{\partial u}{\partial x_2}n_2 \tag{3}$$

and $n=(n_1,n_2)$ denotes the unit outward normal to $\partial\Omega$. The unknown function u denotes the relative warming of the transformer screening with respect to cooling oil of the transformer, i.e. the difference of temperatures of these two media. We suppose $u \in C^2\left(\bar{\Omega}\right)$, functions a_{11}, a_{22}, c, $f \in C^1\left(\bar{\Omega}\right)$ and $\alpha, g \in\ C\left(\partial\Omega,\right)$, $\alpha\left(s\right)\geq 0$ on $\partial\Omega$. The coefficients a_{11} and a_{22} describe the heat conduction character of the screening in the x_1-axis and x_2-axis directions, respectively.

We will describe a numerical solution procedure of the heat conduction problem model by the box method, local error estimation of the error of this method and practical examples in the following paragraphs.

[*]This work was supported by Project 1M4674788502 of Ministry of Education, Youth and Sports of the Czech Republic.

2. Use of the box method

We construct the triangulation τ on the closure of rectangle Ω ($X_1 \leq x_1 \leq X_2, Y_1 \leq x_2 \leq Y_2$) in a similar way as if we used the finite element method. We construct a regular rectangular mesh with increments $h_1 = (X_2 - X_1)/p$ and $h_2 = (Y_2 - Y_1)/q$ in the x_1-axis and x_2-axis direction, respectively, where p and q denote the number of segments, to which the region is divided in the x_1-axis and x_2-axis direction, respectively. A general node has coordinates $V_{rs} = [X_1 + rh_1, Y_1 + sh_2]$, where $r \in \{0, 1, \ldots, p\}$, $s \in \{0, 1, \ldots, q\}$. The rectangles with vertices defined at points of mesh create elements of the triangulation τ. We construct a special case of mesh dual to the mesh τ published in [4, p. 215]. Points T_i, $1 \leq i \leq 4$, are midpoints of abscissas defined by the mesh point V_{rs} and adjacent mesh points. Then points S_i, $1 \leq i \leq 4$, are intersection points of axes of the abscissas mentioned. The rectangle corresponds to node V_{rs} and is given by vertices S_1, S_2, S_3 and S_4 thus creating element b_{rs} of the mesh dual to τ (see Fig. 1). If the node V_{rs} lies on the boundary of Ω, the element b_{rs} is modified in the corresponding way. In particular, the case when the node V_{rs} lies at "corner" of Ω is in Fig. 2. The elements b_{rs} are characterized by two conditions: $\bar{\Omega} = \bigcup b_{rs}$, where $0 \leq r \leq p$, $0 \leq s \leq q$, and int $b_{rs} \cap$ int $b_{kl} = \emptyset$ for $V_{rs} \neq V_{kl}$.

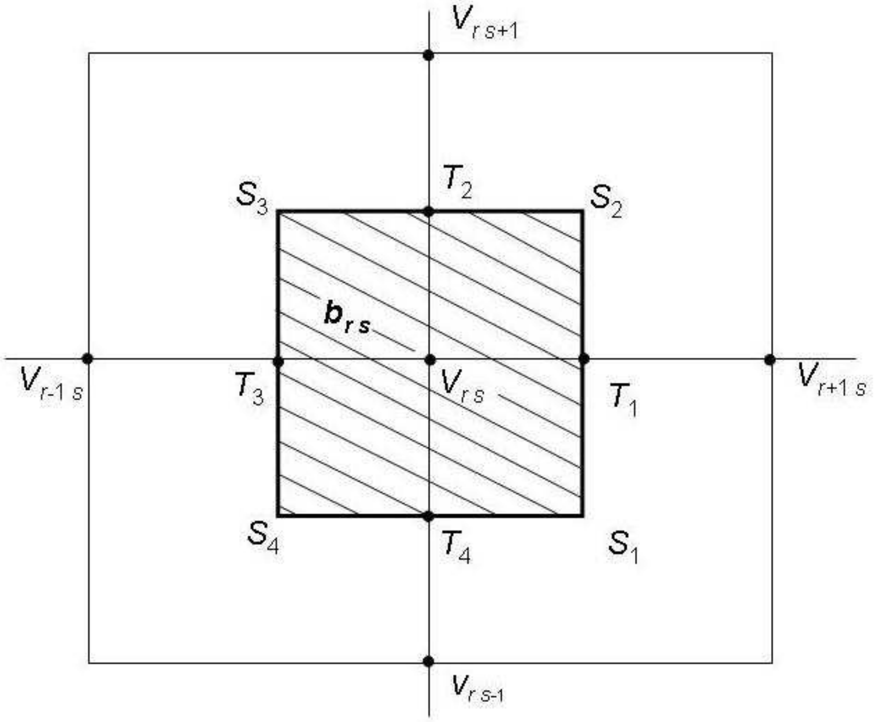


Fig. 1: *Element b_{rs} of the dual mesh corresponding to node V_{rs}.*

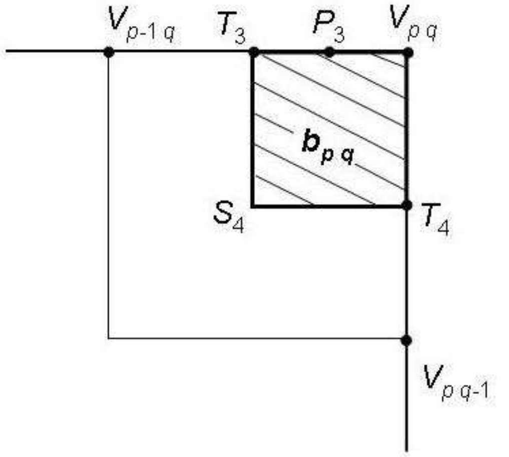


Fig. 2: *Element b_{pq} of the dual mesh corresponding to "corner" node V_{pq}.*

We can transfer the term cu to the right hand side of the equation (1) and integrate the left and right hand sides over the element b_{rs}. Then we get

$$\int\limits_{b_{rs}} \left[-\frac{\partial}{\partial x_1}\left(a_{11}\frac{\partial u}{\partial x_1}\right) - \frac{\partial}{\partial x_2}\left(a_{22}\frac{\partial u}{\partial x_2}\right)\right] \mathrm{d}x = \int\limits_{b_{rs}} (f - cu)\,\mathrm{d}x. \tag{4}$$

Using now Green's formula on the left hand side of the relation (4), we find that

$$\int\limits_{b_{rs}} \left[-\frac{\partial}{\partial x_1}\left(a_{11}\frac{\partial u}{\partial x_1}\right) - \frac{\partial}{\partial x_2}\left(a_{22}\frac{\partial u}{\partial x_2}\right)\right] \mathrm{d}x = -\int\limits_{b_{rs}} \frac{\partial}{\partial x_1}\left(a_{11}\frac{\partial u}{\partial x_1}\right) \mathrm{d}x -$$

$$-\int\limits_{b_{rs}} \frac{\partial}{\partial x_2}\left(a_{22}\frac{\partial u}{\partial x_2}\right) \mathrm{d}x = -\int\limits_{\partial b_{rs}} a_{11}\frac{\partial u}{\partial x_1} n_1\,\mathrm{d}s - \int\limits_{\partial b_{rs}} a_{22}\frac{\partial u}{\partial x_2} n_2\,\mathrm{d}s.$$

Then the relation (4) can be modified to read

$$-\int\limits_{\partial b_{rs}} a_{11}\frac{\partial u}{\partial x_1} n_1\,\mathrm{d}s - \int\limits_{\partial b_{rs}} a_{22}\frac{\partial u}{\partial x_2} n_2\,\mathrm{d}s = \int\limits_{b_{rs}} (f - cu)\,\mathrm{d}x. \tag{5}$$

The left hand side of the equation (5) describes the quantity of heat supplied from or delivered to the boundary of the element b_{rs}, the right hand side expresses the waste heat arising in the element b_{rs}. In case of the boundary mesh point V_{rs}, the equation of type (5) is modified. Using equations of type (5) at all mesh points V_{rs} and applying suitable approximations of derivatives and integrals, we obtain a system of linear algebraic equations with a band matrix. The solution of this system gives us the approximation of warming at nodes V_{rs} of the mesh. Now we will concentrate on the approximation of equations of type (5) and the local approximation error at node V_{rs}.

Because the element b_{rs} is a rectangle, in case of internal element we can use the approximation (see Fig. 1)

$$a_{11}\left(T_1\right)\frac{\partial u\left(T_1\right)}{\partial x_1}n_1 \approx a_{11}\left(T_1\right)\frac{u(V_{r+1s})-u(V_{rs})}{h_1}. \tag{6}$$

With respect to the supposed smoothness of function u, we reach the $O(h_1^2)$-order error in the approximation (6). Similar approximations can be carried out for points T_2, T_3 and T_4 in Fig. 1.

We focus now on the boundary element, for example element b_{rq}, where $1 \leq r \leq p-1$. Using relations (2) and (3), we obtain the expression

$$a_{22}(V_{rq})\frac{\partial u}{\partial x_2}(V_{rq})n_2 = g(V_{rq}) - \alpha(V_{rq})u(V_{rq}). \tag{7}$$

In case of the boundary "corner" element (see Fig. 2), we can form an approximation of the value $u(P_3)$ (where the auxiliary point P_3 is midpoint of abscissa T_3V_{pq}) from the values $u(V_{p-1q})$ and $u(V_{pq})$ using Lagrange's interpolation polynomial of the first degree. As we suppose $u \in C^2(\bar{\Omega})$, the error order of approximation is $O(h_1^2)$ (see [3, p. 64]) and the value

$$a_{22}(P_3)\frac{\partial u}{\partial x_2}(P_3)n_2$$

can be approximated through the use of the relation (2).

Now we target at the approximation of integrals in equations of type (5). We apply the midpoint rule to the approximation. If function $v \in C^2_{[a,\,b]}$ then the midpoint rule gives

$$\left|\int\limits_a^b v(x)\,\mathrm{d}x - v\left(\frac{a+b}{2}\right)(b-a)\right| \leq M\left(b-a\right)^3,$$

where $M \in \mathbf{R}$ (see, for example, [1, p. 178]). In case of the internal element b_{rs}, we use points T_i, $1 \leq i \leq 4$, as midpoints for the integration over the boundary of the element b_{rs} on the left hand side of the equation (5) . The right hand side of the equation (5) is approximated in the form

$$\int\limits_{b_{rs}} (f - cu)\,\mathrm{d}x \approx (f(V_{rs}) - c\,(V_{rs})u(V_{rs}))h_1h_2.$$

In case of the boundary element we use the expressions of type (7) for the approximation of integrals, too. At the boundary "corner" element it is possible to use auxiliary points of type P_3.

Let us set h =max(h_1, h_2). Applying the above mentioned procedure, we find that the local approximation error of the equation of type (5) for every element b_{rs} is $O(h^2)$, where $r \in \{0, 1, \ldots, p\}$, $s \in \{0, 1, \ldots, q\}$. General questions of the box method error estimation are solved in [2].

3. Practical numerical examples

Now we will solve a real-life technical problem of finding the screening warming with respect to cooling oil of the screening (cooling oil flows around the screening) by using the above mentioned box method. Transformer screening is warmed in consequence of existing eddy currents and it is considered in the form of a thin-walled cylinder. The temperature field is supposed to be rotationally symmetric (see Fig. 3). Hence, the warming problem can be solved in the screening cross section on two dimensional untypical closed rectangular domain Ω. Then the equation (1) with the boundary condition (2) can be written in the form

$$\frac{\partial}{\partial x_1}\left(a_{11}\frac{\partial u}{\partial x_1}\right)+\frac{\partial}{\partial x_2}\left(a_{22}\frac{\partial u}{\partial x_2}\right)=-q(x_1,x_2) \tag{8}$$

with the Newton boundary condition

$$a_{11}\frac{\partial u}{\partial x_1}n_1+a_{22}\frac{\partial u}{\partial x_2}n_2+\alpha u=\alpha k(x_2-Y_1) \tag{9}$$

on the rectangle Ω. As mentioned above, the solution u represents the warming of the screening with respect to cooling oil, a_{11} and a_{22} are real values in this case; $q(x_1,x_2)$ is the volume density of losses. In the boundary condition (9), the function α means the heat transfer coefficient on the boundary of the domain, a real constant k allows to express the variable temperature of oil in the vicinity of the screening in the x_2-axis direction. The equation (8) with the boundary condition (9) is suitable to solve as a 2D problem. The solution u depends on the functions $q(x_1,x_2)$ and $\alpha(x_1,x_2)$, too. If q depends only on the variable x_2 and α is constant function on $\partial\Omega$, then this problem can be solved as a 1D problem.

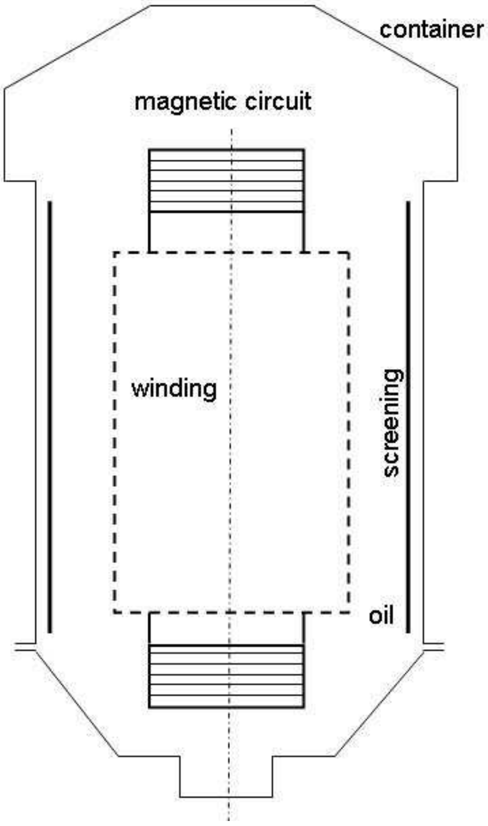


Fig. 3: *Crosscut – the position of the screening in the transformer container.*

Example 1

The function q is given by the relation $q(x_1,x_2)=\rho\,\delta^2(x_1,x_2)$, where ρ is the specific resistance of the screening material, δ denotes the density of eddy currents. Input parameters: $X_1=1.273$m, $X_2=1.280$m, $Y_1=0.000$m, $Y_2=1.200$m, $a_{11}=3$W/mK, $a_{22}=20$W/mK, $\rho=0.143\times10^{-6}\Omega$m, $\alpha(x_1,x_2)=50$W/m^{2}K for $x_2\neq Y_1=0$m and

$\alpha(x_1, x_2) = 0\text{W/m}^2\text{K}$ for $x_2 = Y_1 = 0\text{m}$, $k = 0\text{K/m}$, the current density $\delta(x_1,\ x_2)$ is given by means of 45 values between $0.1158 \times 10^6\text{Am}^{-2}$ and $0.1993 \times 10^7\text{Am}^{-2}$, the current density at the nodes is computed by means of linear interpolation.

Table 1 lists approximate values of warming at chosen nodes computed by using the box method. The values of warming u first of all depend on the input values of function δ. The given values $\delta(x_1,\ x_2)$ are decreasing in the x_1-direction for the constant value of variable x_2. Hence, the computed values of warming u are decreasing in the x_1-axis direction (computed warming u at the nodes with $x_1 = 1.27416\text{m}$ is slightly higher than at the nodes with $x_1 = X_1 = 1.273\text{m}$ in consequence of cooling oil).

x_2[m]	$X_1 = 1.273\text{m}$	$x_1 = 1.27416\text{m}$	$x_1 = 1.27533\text{m}$	$X_2 = 1.280\text{m}$
$Y_2 = 1.200$	3.993	4.021	3.965	3.731
1.104	6.333	6.399	6.267	5.807
1.008	7.940	8.014	7.866	7.299
0.912	10.100	10.197	10.003	9.278
0.816	12.311	12.429	12.193	11.311
0.720	12.394	12.483	12.305	11.455
0.624	13.413	13.516	13.310	12.187
0.528	13.445	13.545	13.345	12.418
0.432	13.466	13.570	13.362	12.430
0.336	13.211	13.313	13.109	12.195
0.240	12.957	13.054	12.860	11.967
0.144	12.640	12.725	12.555	11.692
0.048	12.491	12.581	12.401	11.541
$Y_1 = 0.000$	12.289	12.356	12.213	11.378

Tab. 1: *The values of the screening warming in K for selected nodes, $h_1 = 0.0011667m$ and $h_2 = 0.04800m$.*

Example 2

The volume density of losses q depends on x_2 only. Input parameters: $X_1 = 0.860\text{m}$, $X_2 = 0.868\text{m}$, $Y_1 = 0.033\text{m}$, $Y_2 = 1.900\text{m}$, $a_{11} = 3\text{W/mK}$, $a_{22} = 20\text{W/mK}$, the volume density of losses $q(x_2)$ is given by means of 36 values between $0.4904 \times 10^2\text{W/m}^3$ and $0.9348 \times 10^6\text{W/m}^3$, values of the function q at the nodes are computed by means of linear interpolation, $\alpha(x_1, x_2) = 50\text{W/m}^2\text{K}$ for $x_1 \neq X_2 = 0.868\text{m}$ and $\alpha(x_1, x_2) = 15\text{W/m}^2\text{K}$ for $x_1 = X_2 = 0.868\text{m}$, $k = 0\text{K/m}$.

The volume density of losses q depends only on x_2-axis in this example. The computed values of warming u first of all depend on the impute values of function q. The value of the heat transfer coefficient $\alpha(x_1, x_2)$ is lower for $x_1 = X_2$ than for the other parts of $\partial\Omega$. Hence, the screening is more cooled on the part of $\partial\Omega$ with $x_1 = X_1$ than on the part of $\partial\Omega$ with $x_1 = X_2$. Therefore, computed values of warming u are a little lower near the part of $\partial\Omega$ with $x_1 = X_1$ than near the part of $\partial\Omega$ with $x_1 = X_2$ in the x_1-axis direction. Table 2 lists approximate values of the warming at chosen nodes computed by using the box method.

x_2[m]	$X_1 = 0.860$m	$x_1 = 0.864$m	$x_1 = 0.866$m	$X_2 = 0.868$m
$Y_2 = 1.900$	71.369	74.530	74.640	74.585
1.783	26.997	28.214	28.270	28.242
1.666	6.753	7.056	7.068	7.062
1.550	2.247	2.348	2.352	2.350
1.433	1.736	1.814	1.818	1.816
1.316	3.105	3.244	3.250	3.247
1.200	3.601	3.762	3.768	3.765
1.083	3.144	3.285	3.289	3.287
0.966	2.855	2.983	2.989	2.986
0.850	2.962	3.095	3.101	3.098
0.733	3.457	3.612	3.618	3.615
0.616	3.449	3.602	3.608	3.605
0.500	1.503	1.571	1.575	1.573
0.383	2.319	2.422	2.426	2.424
0.266	2.449	3.604	3.612	3.608
0.150	17.614	18.413	18.451	18.432
$Y_1 = 0.033$	60.035	62.690	62.782	62.736

Tab. 2: *The values of the screening warming in K for selected nodes, $h_1 = 0.002m$ and $h_2 = 0.029167m$.*

References

[1] H.M. Antia: *Numerical methods for scientists and engineers.* Birkhäuser Verlag, Berlin, 2000.

[2] R.E. Bank, D.J. Rose: Some error estimates for the box method. SIAM J. Numer. Anal. **24** (1987), 777–787.

[3] A. Ralston: *A first course in numerical analysis.* Academia, Prague, 1978 (in Czech).

[4] R.S. Varga: *Matrix iterative analysis.* Springer Verlag, Berlin, 2000.

NUMERICAL SOLUTION OF BOUNDARY VALUE PROBLEMS BY MEANS OF B-SPLINES

Vratislava Mošová

Galerkin method is often used for solving boundary value problems. The most favorite method for solving problems from engineering practice, the finite element method (FEM), corresponds to the Galerkin method, where in particular continuous functions with small support form a basis. By Céa's lemma (see [5]), the error of the Galerkin approximation is bounded by means of the minimal error in the space of test functions. It means that success of the Galerkin method depends on the choice of basis functions. If we focus our attention on approximation theory, then B-splines represent a successful tool for approximation of functions. B-splines are piecewise polynomial functions with compact support that can be computed by means of simple schemes. Their differentiation and integration can be algorithmized. They are closely connected to computational geometry (see [5], [4]).

In this article, we deal with solution of boundary value problems using the Galerkin method, where weighted B-splines form the basis. These splines and their properties are described in the first section. Examples of solutions of 1D boundary value problems using B-spline basis are given in the second section.

1. B-splines and their properties

Definition 1 Let $b^0(x)$ be the characteristic function of the interval $[0,1]$ and

$$b^n(x) = \int_{x-1}^{x} b^{n-1}(\xi)\,\mathrm{d}\xi,\ \ n = 1, 2, \dots. \tag{1}$$

For integer $h > 0$ and number $k \in \mathbb{Z}$, the function

$$b^n_{k,h}(x) = b^n(x/h - k) \tag{2}$$

is the B-spline of order n on the grid of width h.

Remark 1 B-splines $b^n_{k,h}(x)$ have the following useful properties:

- B-spline $b^n_{k,h}(x)$ is positive on the interval $(kh, (k+n+1)h)$ and vanishes outside this interval.

- B-splines $b^n_{k,h}(x)$ are polynomials of order n on each interval $(kh, (k+1)h)$, $k = 0, \dots, n$.

- Recursive formulas enable to compute the derivatives

$$\frac{\mathrm{d}b^n_{k,h}(x)}{\mathrm{d}x} = \frac{1}{h}\left(b^{n-1}_{k,h}(x) - b^{n-1}_{k+1,h}(x)\right) \tag{3}$$

  and the scalar products

$$s^n_{k-l} = \int_{\mathbb{R}} b^n_{k,h}(x) b^n_{l,h}(x)\,\mathrm{d}x = hb^{2n+1}(n+1+k-l), \tag{4}$$

$$\int_{\mathbb{R}} (b^n_{k,h}(x))'(b^n_{l,h}(x))'\,\mathrm{d}x = \frac{1}{h}(2s^{n-1}_{k-l} - s^{n-1}_{k-l-1} - s^{n-1}_{k-l+1}), \tag{5}$$

  which we may encounter in the weak formulation of certain boundary value problems.

- The identity

$$b^n_{k,h}(x) = 2^{-n}\sum_{l=0}^{n+1}\binom{n+1}{l} b^n_{2k+l,\frac{h}{2}}(x) \tag{6}$$

  is useful for mesh refinement.

- It is possible, thanks to Marsden's equality

$$(x-t)^n = \sum_{k\in\mathbb{Z}} h^n(k+1-\frac{t}{h})\dots(k+n-\frac{t}{h})b^n_{k,h}(x), \quad x,t\in\mathbb{R}, \tag{7}$$

  to express any polynomial as a linear combination of the B-splines. The relation (7) plays an important role in the stabilization of bases and error estimates.

We receive multivariate B-splines as tensor products of the univariate ones.
Definition 2 For $x \in \mathbb{R}^m$, $k \in \mathbb{Z}^m$, $n \in \mathbb{N}$ and $h > 0$ the function

$$b^n_{k,h}(x) = \prod_{i=1}^{m} b^n_{k_i,h}(x_i) \tag{8}$$

is the m-variate B-spline of degree n on the grid of width h.

The nonzero restrictions of B-splines $b^n_{k,h}$ to Ω can be taken as a basis for the solution of Neumann boundary value problem on a bounded domain $\Omega \subset \mathbb{R}^m$. But these B-splines are not suitable for solving any Dirichlet boundary value problem, because their linear combination

$$\sum_{k\in K} b^n_{k,h}(x)u_k,\ K = \{k|\,\mathrm{supp}\, b^n_{k,h} \cap \Omega \neq 0\}$$

generally does not satisfy essential boundary conditions. It is possible to remove this disproportion if we work with weighted B-splines.

Definition 3 Let a weight function[1] w and a B-spline $b^n_{k,h}$ be given, then

$$b_k(x) = w(x) b^n_{k,h}(x) \tag{9}$$

is called the weighted B-spline.

2. Boundary value problems and B-splines

Example 1 Consider the 1D Neumann boundary value problem

$$u''(x) + 16^2 u(x) = x, \quad x \in (0,1), \tag{10}$$

$$u'(0) = u'(1) = 0. \tag{11}$$

Find an approximation of the weak solution using B-splines defined above.
Solution: We find $u \in W^{1,2}(0,1)$ such that

$$-\int_0^1 u'v' \,\mathrm{d}x + 16^2 \int_0^1 uv \,\mathrm{d}x = \int_0^1 xv \,\mathrm{d}x, \ \forall v \in W^{1,2}(0,1). \tag{12}$$

The unknown function $u(x)$ is approximated by

$$\tilde{u}(x) = \sum_{k=1}^{N} b^3_{k-3,h}(x) u_k$$

over N uniformly distributed nodes ($h = \frac{1}{N-1}$). This approximation, in conjunction with the Galerkin method, provides a mesh-free computational formulation of the boundary value problem. The system of linear equations has the form

$$A\tilde{u} = f, \tag{13}$$

$$\tilde{u} = (u_1, \dots, u_N)^T, \quad f = (f_1, \dots f_N)^T, \quad A = \begin{pmatrix} a_{11} & \dots & a_{1N} \\ \dots & \dots & \dots \\ a_{N1} & \dots & a_{NN} \end{pmatrix},$$

$$f_j = \int_\Omega f(x) b^3_{j-3,h}(x) \,\mathrm{d}x,$$

$$a_{i,j} = \int_\Omega \left[-(b^3_{i-3,h}(x))'(b^3_{j-3,h}(x))' + 16^2\, b^3_{i-3,h}(x) b^3_{j-3,h}(x) \right] \mathrm{d}x.$$

Results for $N = 11$ nodes are given in Figure 1 and in Table 1.

[1] Weight function is a nonnegative continuous function on $\overline{\Omega}$ that vanishes on the boundary $\partial\Omega$. For $r > 0$ we can put $w(x) = \mathrm{dist}(x, \partial\Omega)^r$, $x \in \Omega$. If $r = 1$ then w is called the standard weight function.

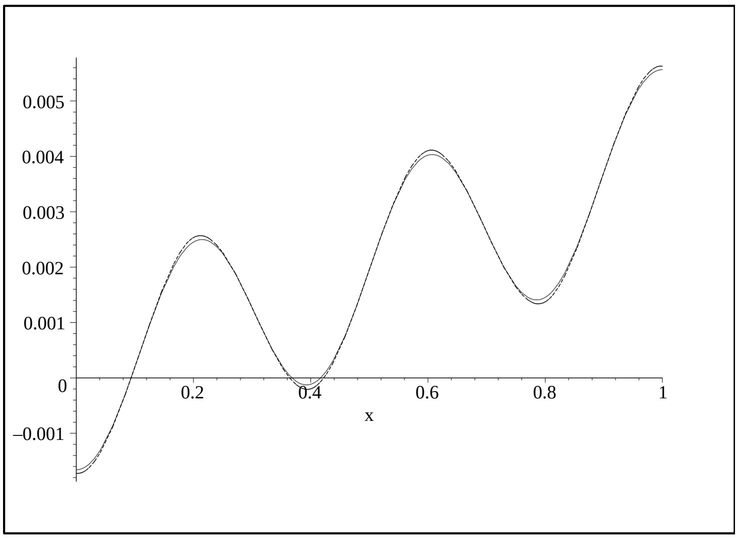


Fig. 1: *The exact solution of the Neumann BVP and its approximation for $N = 11$.*

N	8	11	14	21	31
$\max\lvert u - \tilde{u}\rvert$	1.8×10^{-3}	8×10^{-5}	14×10^{-6}	16×10^{-7}	24×10^{-8}

Tab. 1: *Dependence of the error of the approximation on the number of nodes.*

Example 2 Solve the 1D Dirichlet boundary value problem

$$u''(x) + 16^2 u(x) = x, \quad x \in (0,1), \tag{14}$$

$$u(0) = u(1) = 0 \tag{15}$$

using B-splines.
Solution: We find $u \in W_0^{1,2}(0,1)$ such that

$$-\int_0^1 u'v' \,\mathrm{d}x + 16^2 \int_0^1 uv \,\mathrm{d}x = \int_0^1 xv \,\mathrm{d}x, \ \forall v \in W_0^{1,2}(0,1). \tag{16}$$

We suppose that nodes $x_1, \ldots, x_N$, at which the approximate values are computed, are uniformly distributed.

i) We replace the original set $\{b^3_{k-3,h}(x)\}_{k=1}^N$ by the set of weighted B-splines. We consider the Galerkin approximation in the form

$$\tilde{u}(x) = \sum_{k=1}^{N} w_1(x) b^3_{k-3,h}(x) u_k, \text{ where } w_1(x) = \begin{cases} x/s, & \text{if } 0 < x < s \\ 1, & \text{if } s \le x \le 1-s \\ (1-x)/s, & \text{if } 1-s < x < 1 \end{cases}$$

and the parameter s represents the width of the strip inside $[0,1]$, where the function $w_1 \neq 1$. Results for $N = 11$ and $s = 0.2$ are given in Figure 2. The dependence of

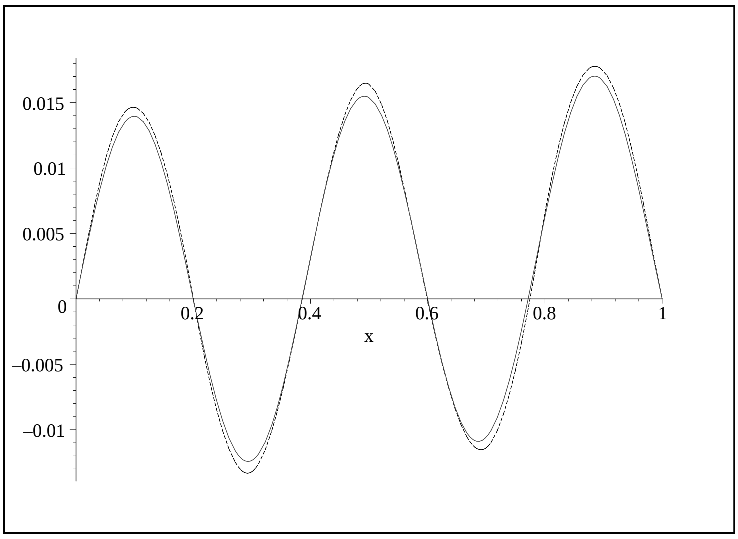


Fig. 2: *The exact solution of the Dirichlet BVP and its approximation for $N = 11$, linear weight function and $s = 0.2$.*

N	8	11	14	21	31
$s = 0.2$	1.5×10^{-2}	1.1×10^{-3}	4×10^{-4}	2×10^{-4}	10^{-4}
$s = 0.3$	1.2×10^{-3}	1.7×10^{-3}	1.2×10^{-3}	8×10^{-4}	4.8×10^{-4}
$s = 0.4$	2.4×10^{-3}	4×10^{-4}	3×10^{-4}	2×10^{-4}	1.5×10^{-4}

Tab. 2: *The error of the approximation for different number of nodes and for different widths of the strip.*

the error $\max|u - \tilde{u}|$ on the number N of nodes and on the width of the strip s can be seen in Table 2.

Not only the width s of the strip affects the quality of the approximate solution, but also the choice of the proper weight function is important. The errors of approximation for the linear weight function w_1 and the quadratic weight function

$$w_2(x) = \begin{cases} \left(2 - \dfrac{x}{s}\right)\dfrac{x}{s}, & \text{if } 0 < x < s \\ 1, & \text{if } s \leq x \leq 1 - s \\ \left(2 - \dfrac{1-x}{s}\right)\dfrac{1-x}{s}, & \text{if } 1 - s < x < 1 \end{cases}$$

for $N = 11$ and $s = 0.2$ are compared in Table 3. The quadratic weight function produces more accurate results than the linear weight function.

The errors for the linear weight function w_1 and for the quadratic weight function w_2 for $N = 11$ and for different values of the parameter s are given in Table 4.

Note that the weighted B-splines $w_1 b^3_{k,h}$ have not the first derivative and $w_2 b^3_{k,h}$ have not the second derivative in some points of the interval $[0, 1]$. Considering that

N	8	11	14	21	31
e_{w_1}	1.5×10^{-2}	1.1×10^{-3}	4×10^{-4}	2×10^{-4}	10^{-4}
e_{w_2}	8×10^{-3}	5.3×10^{-4}	2×10^{-4}	8.8×10^{-5}	6.4×10^{-5}

Tab. 3: *The error* e_{w_i} *=max*$|u - \tilde{u}_{w_i}|$ *of the approximation for different form of weight functions and different values of* s*.*

s	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50
$e_{w_1} \times 10^4$	22	11	16	17	7.4	4	4.3	11
$e_{w_2} \times 10^4$	13	5.3	5.1	6.1	5.4	4	3.1	3.4

Tab. 4: *The error* e_{w_i} *=max*$|u - \tilde{u}_{w_i}|$ *of the approximation for different forms of weight functions and different values of* s*.*

these functions are elements of the space $W_0^{1,2}$, they can be advantageously used in our problem, in spite of the fact that the smoothness of these basis functions is lower than the smoothness of original cubic splines. It can be seen from Figure 2 and Table 3 that this fact has only small influence on approximation properties of the used weighted basis. The amplitude and frequency of the approximation received for $N = 11$ and w_1 are in accordance with the analytic solution. The errors received in the case of piecewise linear and quadratic weight functions are similar. The quadratic weight function gives a bit better results, very similar to the case when we use e.g. the perfectly smooth weight function $w(x) = \sin(\pi x)$. (For more information about weight functions see [4].)

ii) If the basis contains B-splines that only have a small part of their support in the considered interval, then the system of linear equations (13) is ill-conditioned and convergence of the iterative process can be slow. This can be improved if we modify the B-splines whose supports intersect the boundary.

Consider again the uniformly distributed nodes $0 = x_1 < \cdots < x_N = 1$, cubic splines $b^3_{k,h}$, and the weight function w_1. Let

$$\begin{aligned}\tilde{u}(x) = &\sum_{k=5}^{N-4} w_1(x) b^3_{k-3,h}(x) u_k \\ &+ 4w_1(x)\left(b^3_{-2,h}(x)u_1 + b^3_{0,h}(x)u_3 + b^3_{N-3,h}(x)u_N + b^3_{N-5,h}(x)u_{N-2}\right) \\ &- 6w_1(x)\left(b^3_{-1,h}(x)u_2 + b^3_{N-4,h}(x)u_{N-1}\right) - w_1(x)\left(b^3_{1,h}(x)u_4 + b^3_{N-6,h}(x)u_{N-3}\right).\end{aligned}$$

The errors for $s = 0.2$ and for different values of N are provided in Table 5.

N	8	11	14	21	31
$\max_{w_1}\lvert u - \tilde{u}\rvert$	1.5×10^{-2}	1.4×10^{-3}	3×10^{-4}	2×10^{-4}	8×10^{-5}

Tab. 5: *Dependence of the error of approximation on the number of nodes.*

3. Conclusion

In this contribution, we presented methods of solving boundary value problems using the Galerkin method with B-spline basis. This method belongs to the meshless methods, because no explicitly given mesh is required for its realization. (For more information about the meshless methods see [1], [2], [3]).

The weighted B-splines are a simple and comfortable tool from the computational point of view (recursive formulas enable to compute derivatives and scalar products of B-splines easily, see Remark 1). The size of support and smoothness of B-splines depend on the parameters n and h, which we choose at the beginning of the computation. In case of the Neumann boundary value problem it suffices to work with B-splines only, whereas for the Dirichlet boundary value problem it is necessary to use the weighted B-splines.

The error of any approximation depends not only on the number of nodes, but on another factors, too. Example 2 showed that in the case of the Dirichlet problem the error of the resulting approximation can become smaller if a proper weight function is chosen. The influence of the choice of the weigh function and of the width s of the the strip on the approximate solution can be the subject of a further study.

References

[1] I. Babuška, U. Banerjee, J.E. Osborn: *Survey of meshless and generalized finite element mehods: An unified approach*, Acta Numer. **12** (2003), 1–125.

[2] T. Belytschko, Y. Guo, W.K. Liu: *A unified stability analysis of meshless particle methods*, Internat. J. Numer. Methods Engrg. **48** (2000), 1359–1400.

[3] E. Cueto, M. Doblaré, L. Gracia: *Imposing essential boundary conditions in the natural element method by means of density-scaled α-shapes.* Internat. J. Numer. Methods Engrg. **49** (2000), 519–546.

[4] K. Hollig: *Finite element methods with B-Splines*, SIAM, Philadelphia, 2003.

[5] R. Kress: *Numerical analysis*, Springer-Verlag, New York, 1998.

ON UNIQUENESS OF A WEAK SOLUTION TO THE STEADY NAVIER-STOKES PROBLEM IN A PROFILE CASCADE WITH A NONLINEAR BOUNDARY CONDITION ON THE OUTFLOW*

Tomáš Neustupa

1. Introduction

The paper deals with theoretical analysis of the mathematical model of viscous incompressible stationary flow through a plane cascade of profiles. The considered fluid moves around an infinite row of profiles which periodically repeat in one spatial direction. This property enables us to reduce the problem to a bounded domain which represents just one spatial period. We assume that the velocity satisfies the Dirichlet boundary condition on the inflow and on the profile, a certain "natural" nonlinear boundary condition of the "do nothing-type" on the outflow and periodic boundary conditions on the artificial boundaries which separate the chosen spatial period from other periods. We present the weak formulation of the problem and recall the theorem on the existence of a weak solution. Afterwards, we study the question of uniqueness of the weak solution. We arrive at a theorem stating that the solution is unique if the data prescribed on the boundary and the external specific body force are in certain norms "sufficiently small" with respect to the viscosity. This result is in agreement with known theorems on uniqueness in the case of the Dirichlet boundary condition on the whole boundary, see e.g. the books on the Navier-Stokes equations by R. Temam and G. P. Galdi.

2. The geometry of the problem

The considered 2D domain Ω, representing one spatial period of the flow field around the infinite and unbounded series of profiles, is sketched on Fig. 1. Its boundary consists of the curves Γ_i (the inflow), Γ_w (the surface of the profile), Γ_- and Γ_+ (the lower and the upper artificial boundaries), and Γ_o (the outflow). The reduction of the original problem for an infinite profile cascade to just one spatial period is standard and the main idea standing in the background is that the weak solution constructed in one spatial period Ω, periodically extended in the direction of the x_2-axis, becomes a solution for the whole profile cascade, see [2] for details.

*The research was supported by the research plan of the Ministry of Education of the Czech Republic No. MSM 6840770010.

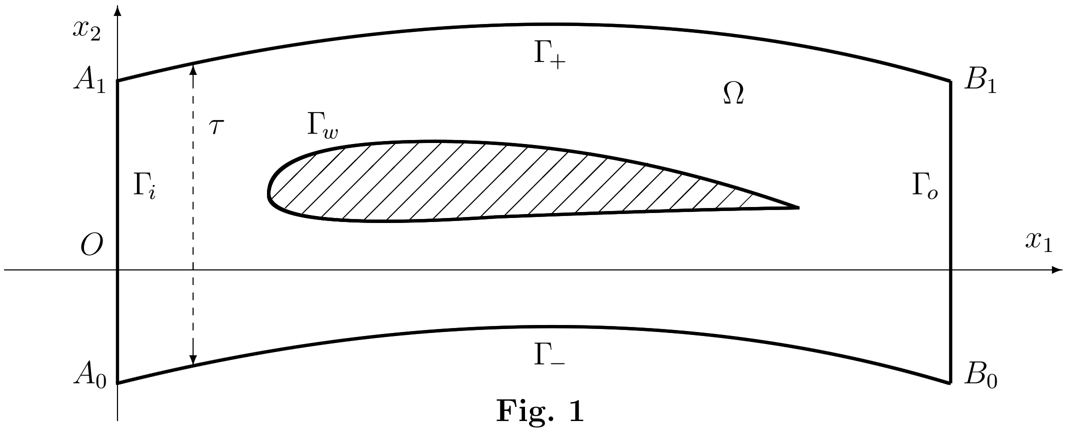


Fig. 1

3. The classical formulation of the boundary-value problem

We denote by $\boldsymbol{u} = (u_1, u_2)$ the (fluid) velocity, by p the kinematic pressure and by $\boldsymbol{n}$ the outer normal vector on the boundary. We study the flow described by 2D steady Navier-Stokes equation

$$(\boldsymbol{u} \cdot \nabla)\boldsymbol{u} = \boldsymbol{f} - \nabla p + \nu \Delta \boldsymbol{u}. \tag{1}$$

The condition of incompressibility is

$$\operatorname{div} \boldsymbol{u} = 0. \tag{2}$$

We prescribe the inhomogeneous Dirichlet boundary condition on the inlet and the homogeneous no-slip Dirichlet boundary condition on the profile

$$\boldsymbol{u} = \boldsymbol{g} \qquad \text{on } \Gamma_i, \tag{3}$$

$$\boldsymbol{u} = \boldsymbol{0} \qquad \text{on } \Gamma_w. \tag{4}$$

We suppose that the conditions of periodicity are fulfilled on the artificial boundaries Γ_- and Γ_+

$$\boldsymbol{u}(x_1, x_2 + \tau) = \boldsymbol{u}(x_1, x_2) \qquad \text{for } (x_1, x_2) \in \Gamma_-, \tag{5}$$

$$\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}}(x_1, x_2 + \tau) = -\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}}(x_1, x_2) \qquad \text{for } (x_1, x_2) \in \Gamma_-, \tag{6}$$

$$p(x_1, x_2 + \tau) = p(x_1, x_2) \qquad \text{for } (x_1, x_2) \in \Gamma_-. \tag{7}$$

We use the nonlinear do-nothing-type boundary condition (proposed by Bruneau and Fabri in [1]) on the outflow Γ_o

$$-\nu \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}} + p\,\boldsymbol{n} - \frac{1}{2}(\boldsymbol{u} \cdot \boldsymbol{n})^- \boldsymbol{u} = \boldsymbol{h}. \tag{8}$$

4. The weak formulation of the boundary-value problem and the theorem on existence

We denote by $H^1(\Omega)$ the usual Sobolev space and by $H^1(\Omega)^2 := \big[H^1(\Omega)\big]^2$ the space of two component vector functions with both components in $H^1(\Omega)$. We define

$$\begin{aligned} V \;&:=\; \big\{\boldsymbol{v} \in H^1(\Omega)^2;\;\; \boldsymbol{v} = \boldsymbol{0} \;\text{ a.e. in } \Gamma_i \cup \Gamma_w,\;\; \boldsymbol{v}(x_1, x_2+\tau) = \boldsymbol{v}(x_1, x_2) \\ &\qquad \text{for a.a. } \;(x_1,x_2) \in \Gamma_-, \;\text{ and }\; \operatorname{div}\boldsymbol{v} = 0 \;\text{ a.e. in } \Omega\big\}. \end{aligned}$$

V is equipped with the norm $|\!|\!|\boldsymbol{v}|\!|\!| := \|\nabla\boldsymbol{v}\|_{L^2(\Omega)^4}$, which is equivalent to the norm in $H^1(\Omega)^2$.

In order to realize the inhomogeneous boundary condition (3), we introduce an auxiliary function $\boldsymbol{g}^*$:

Lemma 4.1 *Let $s \in (\frac{1}{2}, 1]$, let function $\boldsymbol{g}$ belong to the Sobolev-Slobodetskiĭ space $H^s(\Gamma_i)^2$, and let $\boldsymbol{g}(A_1) = \boldsymbol{g}(A_0)$ (where A_0 and A_1 are the end points of Γ_i). Then there exists a constant $c_g > 0$ independent of $\boldsymbol{g}$ and a divergence-free extension $\boldsymbol{g}^* \in H^1(\Omega)^2$ of function $\boldsymbol{g}$ from Γ_i onto Ω such that $\boldsymbol{g}^* = \boldsymbol{0}$ on Γ_w, $\boldsymbol{g}^*$ satisfies the condition of periodicity*

$$\boldsymbol{g}^*(x_1, x_2+\tau) \;=\; \boldsymbol{g}^*(x_1,x_2) \qquad \textit{for } (x_1,x_2) \in \Gamma_- \tag{9}$$

and the estimate

$$\|\boldsymbol{g}^*\|_{H^1(\Omega)^2} \;\le\; c_g\, \|\boldsymbol{g}\|_{H^s(\Gamma_i)^2}. \tag{10}$$

The lemma is proved in [2].

The weak solution $\boldsymbol{u}$ of the problem (1)–(8) can be constructed in the form $\boldsymbol{u} = \boldsymbol{g}^* + \boldsymbol{z}$ where $\boldsymbol{z} \in V$ is a new unknown function. This form of $\boldsymbol{u}$ guarantees that $\boldsymbol{u}$ satisfies the boundary condition (3) on the part Γ_i of $\partial\Omega$.

In order to arrive formally at the weak formulation of the problem (1)–(8), we multiply equation (1) by an arbitrary test function $\boldsymbol{v} = (v_1, v_2) \in V$, integrate over Ω, apply Green's theorem, and use the condition of incompressibility (2), the boundary condition (4), the conditions of periodicity (5)–(7), and the nonlinear condition (8). We obtain an equation, which can be written down in the form

$$a(\boldsymbol{u}, \boldsymbol{v}) \;=\; (\boldsymbol{f}, \boldsymbol{v}) \,+\, b(\boldsymbol{h}, \boldsymbol{v}), \tag{11}$$

with $a(\boldsymbol{u},\boldsymbol{v}) := a_1(\boldsymbol{u},\boldsymbol{v}) + a_2(\boldsymbol{u},\boldsymbol{u},\boldsymbol{v}) + a_3(\boldsymbol{u},\boldsymbol{u},\boldsymbol{v})$ and $b(\boldsymbol{h},\boldsymbol{v}) := -\int_{\Gamma_o} \boldsymbol{h}\cdot\boldsymbol{v}\; \mathrm{d}S$, where

$$a_1(\boldsymbol{u},\boldsymbol{v}) := \nu \int_\Omega \nabla\boldsymbol{u} : \nabla\boldsymbol{v}\; \mathrm{d}\boldsymbol{x}, \qquad a_2(\boldsymbol{u},\boldsymbol{v},\boldsymbol{w}) := \int_\Omega \boldsymbol{u}\cdot\nabla\boldsymbol{v}\cdot\boldsymbol{w}\; \mathrm{d}\boldsymbol{x},$$

$$a_3(\boldsymbol{u},\boldsymbol{v},\boldsymbol{w}) := \int_{\Gamma_o} \frac{1}{2}\,(\boldsymbol{u}\cdot\boldsymbol{n})^-\, \boldsymbol{v}\cdot\boldsymbol{w}\; \mathrm{d}S, \qquad (\boldsymbol{f},\boldsymbol{v}) := \int_\Omega \boldsymbol{f}\cdot\boldsymbol{v}\; \mathrm{d}\boldsymbol{x}.$$

All these forms are defined for $\boldsymbol{u}$, $\boldsymbol{v}$, $\boldsymbol{w} \in V$, $\boldsymbol{f} \in L^2(\Omega)^2$ and $\boldsymbol{h} \in L^2(\Gamma_o)^2$. Thus, we arrive at the definition:

Definition 4.2 *Let $\boldsymbol{g} \in H^s(\Gamma_i)^2$ (for some $s \in (\frac{1}{2}, 1]$) satisfy the condition $\boldsymbol{g}(A_1) = \boldsymbol{g}(A_0)$. Let $\boldsymbol{f} \in L^2(\Omega)^2$ and $\boldsymbol{h} \in L^2(\Gamma_o)^2$. The* **weak solution** *of the problem (1)–(8) is the vector function $\boldsymbol{u} := \boldsymbol{g}^* + \boldsymbol{z}$, where $\boldsymbol{z} \in V$ and $\boldsymbol{u}$ satisfies the identity (11) for all test functions $\boldsymbol{v} \in V$.*

The next theorem brings the information on the existence of a weak solution $\boldsymbol{u}$.

Theorem 4.3 *There exists $\varepsilon > 0$ such that if $\|\boldsymbol{g}\|_{H^s(\Gamma_i)^2} < \varepsilon$ then there exists a solution $\boldsymbol{u} = \boldsymbol{g}^* + \boldsymbol{z}$ of the problem defined in Definition 4.2. Moreover, $\boldsymbol{z}$ satisfies the estimate*

$$|\!|\!|\boldsymbol{z}|\!|\!| \ \leq \ R_1, \tag{12}$$

where $R_1 = R_1\big(\nu, \|\boldsymbol{g}\|_{H^s(\Gamma_i)^2}, \|\boldsymbol{f}\|_{L^2(\Omega)^2}, \|\boldsymbol{h}\|_{L^2(\Gamma_o)^2}\big)$. Consequently, $\boldsymbol{u}$ satisfies

$$\|\nabla \boldsymbol{u}\|_{L^2(\Omega)^4} \ \leq \ R_1 + \|\nabla \boldsymbol{g}^*\|_{L^2(\Omega)^4} \ \leq \ R_1 + c_g \, \|\boldsymbol{g}\|_{H^s(\Gamma_i)^2} \ =: R_2. \tag{13}$$

The proof can be found in [2]. The function $\boldsymbol{u}$ has the form $\boldsymbol{g}^* + \boldsymbol{z}$, where $\boldsymbol{z} \in V$ satisfies

$$a(\boldsymbol{g}^* + \boldsymbol{z}, \boldsymbol{v}) \ = \ (\boldsymbol{f}, \boldsymbol{v}) \ + \ b(\boldsymbol{h}, \boldsymbol{v})$$

for all $\boldsymbol{v} \in V$. The function $\boldsymbol{z}$ was constructed as a limit of an appropriate sequence of Galerkin approximations. We were able to find an explicit form of the dependence of R_1 on ν, $\|\boldsymbol{g}\|_{H^s(\Gamma_i)^2}$, $\|\boldsymbol{f}\|_{L^2(\Omega)^2}$ and $\|\boldsymbol{h}\|_{L^2(\Gamma_o)^2}$ in [2]. The restriction $\|\boldsymbol{g}\|_{H^s(\Gamma_i)^2} < \varepsilon$ follows from the requirement that the form a is coercive.

5. Uniqueness of the weak solution of the problem (1)–(8)

The next theorem presents the main result of this paper. It says that the weak solution $\boldsymbol{u}$ of the problem (1)–(8) is unique in a certain sufficiently small ball.

Theorem 5.1 (on uniqueness of the weak solution) *There exists $R > 0$ such that if $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ are two weak solutions of the problem (1)–(8) defined in Definition 4.2 such that $\|\nabla \boldsymbol{u}_1\|_{L^2(\Omega)^4} \leq R$ and $\|\nabla \boldsymbol{u}_2\|_{L^2(\Omega)^4} \leq R$, then $\boldsymbol{u}_1 \ = \ \boldsymbol{u}_2$.*

Proof. Since $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ are weak solutions of problem (1)–(8), they satisfy the equations

$$\begin{aligned} a(\boldsymbol{u}_1, \boldsymbol{v}) \ &= \ (\boldsymbol{f}, \boldsymbol{v}) \ + \ b(\boldsymbol{h}, \boldsymbol{v}), \\ a(\boldsymbol{u}_2, \boldsymbol{v}) \ &= \ (\boldsymbol{f}, \boldsymbol{v}) \ + \ b(\boldsymbol{h}, \boldsymbol{v}) \end{aligned}$$

for all $\boldsymbol{v} \in V$. Subtracting these equations, we get $a(\boldsymbol{u}_1, \boldsymbol{v}) - a(\boldsymbol{u}_2, \boldsymbol{v}) = 0$.
Expressing the bilinear form a by means of forms a_1, a_2 and a_3 with $\boldsymbol{v} = \boldsymbol{u}_1 - \boldsymbol{u}_2$, we obtain

$$\begin{aligned} a_1(\boldsymbol{u}_1 - \boldsymbol{u}_2, \boldsymbol{u}_1 - \boldsymbol{u}_2) + a_2(\boldsymbol{u}_1, \boldsymbol{u}_1, \boldsymbol{u}_1 - \boldsymbol{u}_2) - a_2(\boldsymbol{u}_2, \boldsymbol{u}_2, \boldsymbol{u}_1 - \boldsymbol{u}_2) \\ + a_3(\boldsymbol{u}_1, \boldsymbol{u}_1, \boldsymbol{u}_1 - \boldsymbol{u}_2) - a_3(\boldsymbol{u}_2, \boldsymbol{u}_2, \boldsymbol{u}_1 - \boldsymbol{u}_2) \ = \ 0. \end{aligned} \tag{14}$$

If we denote

$$\begin{aligned}
I_1 &:= a_1(\boldsymbol{u}_1-\boldsymbol{u}_2,\boldsymbol{u}_1-\boldsymbol{u}_2) \;=\; \nu\int_\Omega |\nabla(\boldsymbol{u}_1-\boldsymbol{u}_2)|^2\,\mathrm{d}\boldsymbol{x} \;=\; \nu\,|||\boldsymbol{u}_1-\boldsymbol{u}_2|||^2,\\
I_2 &:= a_2(\boldsymbol{u}_1,\boldsymbol{u}_1,\boldsymbol{u}_1-\boldsymbol{u}_2)-a_2(\boldsymbol{u}_2,\boldsymbol{u}_2,\boldsymbol{u}_1-\boldsymbol{u}_2),\\
I_3 &:= a_3(\boldsymbol{u}_1,\boldsymbol{u}_1,\boldsymbol{u}_1-\boldsymbol{u}_2)-a_3(\boldsymbol{u}_2,\boldsymbol{u}_2,\boldsymbol{u}_1-\boldsymbol{u}_2),
\end{aligned}$$

then (14) takes the form

$$I_1 \;=\; -I_2 \;-\; I_3. \tag{15}$$

Further, we estimate the terms on the right-hand side of (15).

$$\begin{aligned}
|I_2| \;&=\; \left|\int_\Omega \boldsymbol{u}_1\cdot\nabla\boldsymbol{u}_1\cdot(\boldsymbol{u}_1-\boldsymbol{u}_2)\,\mathrm{d}\boldsymbol{x} \;-\; \int_\Omega \boldsymbol{u}_2\cdot\nabla\boldsymbol{u}_2\cdot(\boldsymbol{u}_1-\boldsymbol{u}_2)\,\mathrm{d}\boldsymbol{x}\right|\\
&\le\; \left|\int_\Omega (\boldsymbol{u}_1-\boldsymbol{u}_2)\cdot\nabla\boldsymbol{u}_1\cdot(\boldsymbol{u}_1-\boldsymbol{u}_2)\,\mathrm{d}\boldsymbol{x}\right| + \left|\int_\Omega \boldsymbol{u}_2\cdot\nabla(\boldsymbol{u}_1-\boldsymbol{u}_2)\cdot(\boldsymbol{u}_1-\boldsymbol{u}_2)\,\mathrm{d}\boldsymbol{x}\right|\\
&\le\; \|\boldsymbol{u}_1-\boldsymbol{u}_2\|^2_{L^4(\Omega)^2}\,\|\nabla\boldsymbol{u}_1\|_{L^2(\Omega)^4} + \|\boldsymbol{u}_2\|_{L^4(\Omega)^2}\,\|\nabla(\boldsymbol{u}_1-\boldsymbol{u}_2)\|_{L^2(\Omega)^4}\,\|\boldsymbol{u}_1-\boldsymbol{u}_2\|_{L^4(\Omega)^2}\\
&\le\; 2c_1^2\,R\,\|\nabla(\boldsymbol{u}_1-\boldsymbol{u}_2)\|^2_{L^2(\Omega)^4} \;=\; 2c_1^2\,R\,|||\boldsymbol{u}_1-\boldsymbol{u}_2|||^2,
\end{aligned}\tag{16}$$

where the constant c_1 comes from the inequality $\|\boldsymbol{u}\|_{L^4(\Omega)^2} \;\le\; c_1\,\|\nabla\boldsymbol{u}\|_{L^2(\Omega)^4}$ (which can be found in [3]) for functions $\boldsymbol{u}$ from $H^1(\Omega)^2$. The term I_3 equals

$$I_3 \;=\; \int_{\Gamma_o}\frac{1}{2}\,(\boldsymbol{u}_1\cdot\boldsymbol{n})^-\,\boldsymbol{u}_1\cdot(\boldsymbol{u}_1-\boldsymbol{u}_2)\,\mathrm{d}S \;-\; \int_{\Gamma_o}\frac{1}{2}\,(\boldsymbol{u}_2\cdot\boldsymbol{n})^-\,\boldsymbol{u}_2\cdot(\boldsymbol{u}_1-\boldsymbol{u}_2)\,\mathrm{d}S.$$

According to the signs of $\boldsymbol{u}_1\cdot\boldsymbol{n}$ and $\boldsymbol{u}_2\cdot\boldsymbol{n}$ on Γ_o, we must split Γ_o into four parts $\Gamma_o = \Gamma_{o1}\cup\Gamma_{o2}\cup\Gamma_{o3}\cup\Gamma_{o4}$, where

(a) $\boldsymbol{u}_1\cdot\boldsymbol{n}\ge 0$, $\boldsymbol{u}_2\cdot\boldsymbol{n}\ge 0$, $(\boldsymbol{u}_1\cdot\boldsymbol{n})^- = 0$, and $(\boldsymbol{u}_2\cdot\boldsymbol{n})^- = 0$ on Γ_{o1};

(b) $\boldsymbol{u}_1\cdot\boldsymbol{n}< 0$, $\boldsymbol{u}_2\cdot\boldsymbol{n}\ge 0$, $(\boldsymbol{u}_1\cdot\boldsymbol{n})^- = \boldsymbol{u}_1\cdot\boldsymbol{n}$, and $(\boldsymbol{u}_2\cdot\boldsymbol{n})^- = 0$ on Γ_{o2};

(c) $\boldsymbol{u}_1\cdot\boldsymbol{n}\ge 0$, $\boldsymbol{u}_2\cdot\boldsymbol{n}< 0$, $(\boldsymbol{u}_1\cdot\boldsymbol{n})^- = 0$, and $(\boldsymbol{u}_2\cdot\boldsymbol{n})^- = \boldsymbol{u}_2\cdot\boldsymbol{n}$, on Γ_{o3};

(d) $\boldsymbol{u}_1\cdot\boldsymbol{n}< 0$, $\boldsymbol{u}_2\cdot\boldsymbol{n}< 0$, $(\boldsymbol{u}_1\cdot\boldsymbol{n})^- = \boldsymbol{u}_1\cdot\boldsymbol{n}$, and $(\boldsymbol{u}_2\cdot\boldsymbol{n})^- = \boldsymbol{u}_2\cdot\boldsymbol{n}$ on Γ_{o4}.

Let us denote by I_3^{o1}, I_3^{o2}, I_3^{o3}, and I_3^{o4} the same integrals as in I_3, but this time considered successively on Γ_{o1}, Γ_{o2}, Γ_{o3}, and Γ_{o4}. Obviously, $I_3^{o1} = 0$ because the integrands are equal to zero on Γ_{o1}. On Γ_{o2} we use the inequality $|\boldsymbol{u}_1\cdot\boldsymbol{n}| \le |\boldsymbol{u}_1\cdot\boldsymbol{n} - \boldsymbol{u}_2\cdot\boldsymbol{n}|$, which holds because $\boldsymbol{u}_1\cdot\boldsymbol{n}<0$ and $\boldsymbol{u}_2\cdot\boldsymbol{n}\ge 0$. We obtain

$$\begin{aligned}
|I_3^{o2}| \;&=\; \left|\int_{\Gamma_{o2}} (\boldsymbol{u}_1\cdot\boldsymbol{n})^-\boldsymbol{u}_1\cdot(\boldsymbol{u}_1-\boldsymbol{u}_2)\,\mathrm{d}S\right| \le \int_{\Gamma_{o2}} |\boldsymbol{u}_1\cdot\boldsymbol{n}-\boldsymbol{u}_2\cdot\boldsymbol{n}|\,|\boldsymbol{u}_1|\,|\boldsymbol{u}_1-\boldsymbol{u}_2|\,\mathrm{d}S\\
&\le\; \int_{\Gamma_{o2}} |\boldsymbol{u}_1-\boldsymbol{u}_2|^2\,|\boldsymbol{u}_1|\,\mathrm{d}S \le \|\boldsymbol{u}_1-\boldsymbol{u}_2\|^2_{L^4(\Gamma_{o2})^2}\,\|\boldsymbol{u}_1\|_{L^2(\Gamma_{o2})^2}\\
&\le\; \|\boldsymbol{u}_1-\boldsymbol{u}_2\|^2_{L^4(\Gamma_o)^2}\,\|\boldsymbol{u}_1\|_{L^2(\Gamma_o)^2} \le c_2\,\|\boldsymbol{u}_1-\boldsymbol{u}_2\|^2_{H^1(\Omega)^2}\,\|\boldsymbol{u}_1\|_{H^1(\Omega)^2}\\
&\le\; c_3\,|||\boldsymbol{u}_1-\boldsymbol{u}_2|||^2\,\|\nabla\boldsymbol{u}_1\|_{L^2(\Omega)^4} \le c_3\,R\,|||\boldsymbol{u}_1-\boldsymbol{u}_2|||^2,
\end{aligned}\tag{17}$$

where the constants c_2 and c_3 come from the inequalities $\|\boldsymbol{u}\|_{L^2(\Gamma_o)^2} \leq c_2 \|\boldsymbol{u}\|_{H^1(\Omega)^2} \leq c_3 \|\nabla\boldsymbol{u}\|_{L^2(\Omega)^4}$ (which can be found in [3]) for functions $\boldsymbol{u}$ from $H^1(\Omega)^2$. The term I_3^{o3} can be estimated in the same way as I_3^{o2}. The term I_3^{o4} can be treated as follows

$$\begin{aligned}
|I_3^{o4}| &= \left| \int_{\Gamma_{o4}} (\boldsymbol{u}_1 \cdot \boldsymbol{n})\, \boldsymbol{u}_1 \cdot (\boldsymbol{u}_1 - \boldsymbol{u}_2)\, \mathrm{d}S - \int_{\Gamma_{o4}} (\boldsymbol{u}_2 \cdot \boldsymbol{n})\, \boldsymbol{u}_2 \cdot (\boldsymbol{u}_1 - \boldsymbol{u}_2)\, \mathrm{d}S \right| \\
&= \left| \int_{\Gamma_{o4}} [(\boldsymbol{u}_1 - \boldsymbol{u}_2) \cdot \boldsymbol{n}]\ \boldsymbol{u}_1 \cdot (\boldsymbol{u}_1 - \boldsymbol{u}_2)\, \mathrm{d}S + \int_{\Gamma_{o4}} (\boldsymbol{u}_2 \cdot \boldsymbol{n})\ (\boldsymbol{u}_1 - \boldsymbol{u}_2) \cdot (\boldsymbol{u}_1 - \boldsymbol{u}_2)\, \mathrm{d}S \right| \\
&\leq \int_{\Gamma_{o4}} |\boldsymbol{u}_1 - \boldsymbol{u}_2|\, |\boldsymbol{u}_1|\, |\boldsymbol{u}_1 - \boldsymbol{u}_2|\, \mathrm{d}S + \int_{\Gamma_{o4}} |\boldsymbol{u}_2|\, |\boldsymbol{u}_1 - \boldsymbol{u}_2|\, |\boldsymbol{u}_1 - \boldsymbol{u}_2|\, \mathrm{d}S \\
&\leq \|\boldsymbol{u}_1 - \boldsymbol{u}_2\|^2_{L^4(\Gamma_{o4})^2} \left(\|\boldsymbol{u}_1\|_{L^2(\Gamma_{o4})^2} + \|\boldsymbol{u}_2\|_{L^2(\Gamma_{o4})^2} \right) \\
&\leq \|\boldsymbol{u}_1 - \boldsymbol{u}_2\|^2_{L^4(\Gamma_o)^2} \left(\|\boldsymbol{u}_1\|_{L^2(\Gamma_o)^2} + \|\boldsymbol{u}_2\|_{L^2(\Gamma_o)^2} \right) \\
&\leq c_4 \|\nabla\boldsymbol{u}_1 - \nabla\boldsymbol{u}_2\|_{L^2(\Omega)^4}\ c_3 \left(\|\nabla\boldsymbol{u}_1\|_{L^2(\Omega)^4} + \|\nabla\boldsymbol{u}_2\|_{L^2(\Omega)^4} \right) \leq 2c_4\, |||\boldsymbol{u}_1 - \boldsymbol{u}_2|||^2\ c_3\, R,
\end{aligned}$$

where the constant c_4 comes from the inequality $\|\boldsymbol{u}\|_{L^4(\Gamma_o)^2} \leq c_4 \|\nabla\boldsymbol{u}\|_{L^2(\Omega)^4}$ (which can be found in [3]) for functions $\boldsymbol{u}$ from $H^1(\Omega)^2$. Substituting from (16), (17) and from the last inequality into (15), we obtain

$$\nu\, |||\boldsymbol{u}_1 - \boldsymbol{u}_2|||^2 \leq (2c_1^2 + 2c_3 + 2c_3c_4)\, R\, |||\boldsymbol{u}_1 - \boldsymbol{u}_2|||^2.$$

Now it is seen that if R is so small that $\nu > (2c_1^2 + 2c_3 + 2c_3c_4)\, R$ then $\boldsymbol{u}_1 = \boldsymbol{u}_2$. This proves the theorem. □

References

[1] C.H. Bruneau, P. Fabrie: *New efficient boundary conditions for incompressible Navier-Stokes equations: A well-posedness result.* M2AN Math. Model. Numer. Anal. **30** (1996), 815–840.

[2] M. Feistauer, T. Neustupa: *On nonstationary viscous incompressible flow through a cascades of profiles.* Math. Methods Appl. Sci. **29** (2006), 1907–1941.

[3] M. Feistauer, T. Neustupa: *On some aspects of analysis of incompressible flow through cascades of profiles.* In: I. Gohberg, A.F. dos Santos, F.-O. Speck, F.S. Teixeira, W.L. Wendland (Eds.), Operator Theoretical Methods and Applications to Mathematical Physics: The Erhard Meister Memorial Volume, Operator Theory: Advances and Applications 147, Birkhauser, Basel, 2004, pp. 257–276.

REMARKS ON THE ECONOMIC CRITERION – THE INTERNAL RATE OF RETURN*

Carmen Simerská

Abstract

The internal rate of return (IRR) together with the present value (PV) is used as a popular measure for financial project. When used appropriately, it can be a valuable aid in project acceptance or selection. The purpose of this article is to survey the facts about this criterion published so far. More, we investigate the cases of multiple or nonexistent IRRs and try to choose the relevant one and explain its economic meaning.

1. Introduction

The internal rate of return (IRR) is frequently used as a valuation for investment transactions and financial securities described by a sequence of cash flows in time. We are interested in the existence, uniqueness or multiplicity of the IRR solutions. The IRR can be unambiguously used in decision making if it is unique and simple.

From time to time, the question of how we should find and interpret the IRRs, that are not unique and simple, appears among economists or even mathematicians. This was the case in the 1990's: The Canadian Institute of Actuaries came up with the problem of IRR uniqueness when trying to clarify a section of the Canadian criminal code which made it offense to lend money at an effective interest rate exceeding 60 % p.a., see [10]. In the Czech legislation, there is currently a law requiring that all providers of consumer loans include the Annual percentage rate of charge (APRC) in the loan conditions. APRC (in Czech: RPSN) is defined to be a solution of the equation

$$\sum_{j=0}^{J} \frac{C_{t_j}}{(1+i)^{t_j}} = 0,$$

for unknown i, where $C_{t_0}, \dots, C_{t_J}$ are the cash flows, positive or negative, of the loan in time terms t_j (including all related costs of the loan to the client: fees, etc.). There are no restrictions on the APRC value but unfortunately, the law includes also loans, the advances and repayments which alternate in time (e.g. when an application fee is considered as a repayment of the loan before the loan is received). In these cases, the loans can have multiple APRC's, APRC being a double root, or not existing at all.

*This work was supported by the project MSM 413/05/0608 of the Ministry of Education, Youth and Sports of the Czech Republic.

2. Basic notions

Let us consider a *project* $\mathbf{B} = (B_0, \dots, B_n)$, $B_0 \neq 0$, $B_n \neq 0$, i.e. a sequence of equally spaced (periodic) *cash flows* $B_0, \dots, B_n$. Given the estimated market rate of interest i per period, at which the money may be borrowed or invested, a usual procedure is to accept the project if its *present value*

$$PV(i) = \; PV(\mathbf{B}, i) = \sum_{k=0}^{n} \frac{B_k}{(1+i)^k}$$

is greater than zero.

An *internal rate of return* (IRR) i^*, $i^* \in (-1, \infty)$, attached to the project $\mathbf{B}$ can be defined by three equivalent definitions:

- i^* is the root of the present value function $PV(\mathbf{B}, i)$, i.e. $PV(\mathbf{B}, i^*) = 0$.
- $i^* = \dfrac{1}{\nu^*} - 1$, where $\nu^* \in (0, \infty)$ is the root of the polynomial $g(\nu) = \sum\limits_{k=0}^{n} B_k \nu^k$.
- $i^* = x^* - 1$, where $x^* \in (0, \infty)$ is the root of the polynomial $h(x) = \sum\limits_{k=0}^{n} B_k x^{n-k}$.

Many applications require only positive IRRs; these correspond to the roots of $g(\nu)$ in the interval $(0, 1)$ and to the roots of $h(x)$ in the interval $(1, \infty)$.
Notes:

- IRR is the rate that equalizes time value of expected earnings and the investment outlays of the project. When unique, IRR defines the marginal value of interest rates (the efficiency of capital or the cost of loan) for which PV is nonnegative. Evidently, multiple or double IRRs can potentially occur.
- The value of $PV(\mathbf{B}, i)$ is an absolute criterion of the project. It is dependent on the size of the cash flows as opposed to the value and number of IRRs, which depend on the cash flows structure.
- The function $PV(i)$ is a continuous (and differentiable) function of the rate i. The IRRs, i.e. the roots of polynomials, are not continuous function of their coefficients $B_0, \dots, B_n$, e.g., when one of the IRRs is double, a small change of a cash flows can cause the double root to disappear. In case of multiple IRRs, any numerical method to calculate them (Newton, Bairstow) can run into difficulties.
- Every finite sequence of cash flows $\mathbf{C} = (C_{t_0}, \dots, C_{t_J})$, e.g. of an arbitrary loan, can always be considered as a periodic (e.g. daily) project $\mathbf{B} = (B_0, \dots, B_n)$ (simply putting: $B_k = 0$ in the days with no flow). Then APRC is the effective annualized IRR of the corresponding $\mathbf{B}$ n-year period, APRC $= (1 + i^*)^{365} - 1$.
- When studying the behaviour of PV and IRR, it may be assumed, without loss of generality, that $B_0 < 0$, i.e the project requires an initial outlay. Sign reverse/identical results may be produced in the case, where the project has an initial income.

If the present value $PV(i)$ of the project is a monotonous function in $(-1,\infty)$ (most loans) and if there exists an IRR, then it is unique. Subsequently, for decision making the IRR can be simply compared to the usual opportunity cost of capital (market interest rate) to accept or reject the project. The application of this IRR criterion becomes problematic if PV is not monotonous and/or the IRR does not exist ($i^* < -1$ or complex-valued) or if there are too many of them. Evidently, the uniqueness of IRR does not imply monotonicity of PV.

3. Conditions for existence and uniqueness of IRR

In the 1970's, great effort was directed to obtain sufficient conditions for determining a unique IRR. Some "new" rules were reproved by means of old mathematical facts dealing with the roots of polynomials. We present a brief survey of the localization rules with the corresponding references.
Assuming $B_0 < 0$, it is easy to verify:

- $B_n > 0 \quad \Rightarrow \quad$ exists IRR in $(-1,\infty)$.
- $PV(0) > 0 \quad \Rightarrow \quad$ exists IRR in $(0,\infty)$.
  ($PV(0) = \sum_{k=0}^{n} B_k > 0$ is the minimum economic convenience of investment.)

The number and uniqueness of IRR in an interval can be guaranteed by means of the number of sign-changes in certain sequences $\mathbf{S}$.

- Descartes theorem, $\mathbf{S} = \{B_0, \ldots, B_n\}$.
  Corollary: Exactly one sign change in $\mathbf{S}$ implies a unique IRR.
- Budan-Fourier theorem, $\mathbf{S} = \{g(\nu), g'(\nu), \ldots, g^{(n)}(\nu)\}$ → Jean's rule [6].
- Soper's theorem [11], $\mathbf{S} = \{B_0,\, B_0 + B_1,\, \ldots,\, \sum_{k=0}^{n} B_k\}$ → Norstrom's rule [8]. Corollary: Exactly one sign change in $\mathbf{S}$ implies a unique positive IRR.
- Vincent's theorem, $\mathbf{S}$ is the diagonal of Vincent's matrix → Bernhard-de Faro condition for non-negative IRRs [2], [1].
- Sturm's theorem, $\mathbf{S}$ results from the Euclid's algorithm applied to the polynomial g → Kaplan's rule, exact number of IRRs ignoring multiplicity [7].

It is worth giving here in more details the Soper-Gronchi (S-G) conditions, the only ones that have a meaningfull economic interpretation. First, for the given project $\mathbf{B}$ and $i \in (-1,\infty)$, we define the project *balance stream* $\mathbf{A}(i) = (a_0(i), \ldots, a_{n-1}(i))$, where the unrecovered financial *balances* are the functions

$$a_m(i) = \sum_{k=0}^{m} B_k (1+i)^{m-k}, \qquad m = 0, \ldots, n-1.$$

This definition comes form [10]. The balances are given equivalently by the relations

$$B_0 = a_0(i)\,, \qquad B_m = a_m(i) - (1+i)a_{m-1}(i), \quad m = 1,\dots,n-1\,.$$

- Assuming any value i^* of the project IRRs was found, Soper [11] and Gronchi [3] stated the sufficient (not necessary) conditions
$$a_m(i^*) \le 0\,, \quad \forall\, m = 0,\dots,n-1\,, \tag{S-G}$$
for the IRR value to be unique.

We recommend the following proposition that is more practical for determining the uniqueness of IRR without IRR computation.

If for a given $r \in (-1,\infty)$ *the conditions:* $a_m(r) \le 0\,, \forall\, m = 0,\dots,n-1\,,$ *are valid and* $PV(r) > 0$*, then there exists a unique IRR* i^* *of the project and* $i^* > r\,.$

Gronchi called the rate i^* for which (S-G) conditions are valid *pure lending* rate. It means that the investor does not borrow from the project at any time during its project life and only recovers its investment at the end, earning the interest i^*. The i^* of the project that has at least one period m with the balance $a_m(i^*) > 0$ should be regarded as a *lending* rate and also as a *borrowing* rate, to be paid (still by the investor) on the balance financed by the project. Then using the IRR of ambiguous meaning for this *mixed* project becomes questionable.

It was also shown that a direct comparison of the IRRs i_1 and i_2 of various projects $\mathbf{B}_1, \mathbf{B}_2$ for the purpose of ranking is not recommendable. When $i_1 > i_2$ and both are lending rates, project $\mathbf{B}_1$ should be preferable. But $\mathbf{B}_2$ is preferable if i_1 and i_2 are regarded as the borrowing ones (the lower borrowing rate is better). Hajdasinski, e.g. in [4], deals with the comparison of mutually exclusive projects by means of the IRR using the method of incremental approach. The comparison of multiple IRRs projects is an unresolved problem so far.

4. Nonuniqueness of IRR

The multiplicity or not real-values of IRR have been regarded as a fatal defect for the IRR criterion. Many objections for using IRR as a criterion of the project have been expressed till now and to use only PV criterion was often proposed. But Oehmke in [9] shows that in exactly those projects that may give either none or multiple IRRs the PV criterion exhibits anomalous behaviour as well. E.g., there are investment projects, where PV can be an increasing function in some interval, see Fig. 1. The use of market interest rate i_m smaller than the lower evaluated IRR may reduce the calculated PV. In order to investigate the cases of IRR nonuniqueness, it is useful to decompose a project by means of the proposition (e.g. [3]):

Given a project $\mathbf{B}$ *and a rate of interest* r*, there exists a set of consecutive financial operations*

$$\{A_m,\, -(1+r)A_m\}\,, \quad m = 0, \dots, n-1\,,$$

where $A_m = a_m(r)$ *are the values of balances attached to* $\mathbf{B}$ *and* $B_n = -(1+r)A_{n-1}$, if and only if r *is an IRR of the project. Moreover, given* $\mathbf{B}$ *and* r *the set is unique.*

I.e., an IRR is an interest rate i^* uniformly applied to the single-one operations $\{B_0,\, -(1+i^*)B_0\}\,, \{A_1,\, -(1+i^*)A_1\}, \dots, \{A_{n-1},\, B_n\}$, into which a project can be uniquely decomposed. Then instead of $\mathbf{B}$ we can deal with any balance stream $\mathbf{A}(i^*)$.

Hazen in [5] generalized the notion of IRR of the project $\mathbf{B}$ to any root i^* of PV (possibly complex-valued). Then $\mathbf{B}$ can always be interpreted as a result of (possibly complex-valued) balance stream $\mathbf{A}(i^*)$ for any i^*. Using the known equation

$$PV(\mathbf{B}, i) = \frac{i - i^*}{1+i}\, PV(\mathbf{A}(i^*), i)\,,$$

for a given interest rate i, he proved the following implications:

(a) *If* $PV(\mathrm{Re}(\mathrm{A}(\mathrm{i}^*)), \mathrm{i}) < 0$ *then* $PV(\mathbf{B}, i) \geq 0 \Leftrightarrow \mathrm{Re}(\mathrm{i}^*) \geq \mathrm{i}\,.$

(b) *If* $PV(\mathrm{Re}(\mathrm{A}(\mathrm{i}^*)), \mathrm{i}) > 0$ *then* $PV(\mathbf{B}, i) \geq 0 \Leftrightarrow \mathrm{Re}(\mathrm{i}^*) \leq \mathrm{i}\,.$

From his point of view, it does not matter which IRR is used to accept or reject the project. Every IRR is meaningful. All what is important is whether IRR exceeds the market rate $i = i_m$. The magnitude of i^* by itself is not significant.

When treating the problem of multiplicity, we take into account that the project (e.g. Fig. 1) can behave differently depending on the market rate i_m used. For some investor the same project can be rejected as an investment and for a borrower at different i_m rejected as a loan. Therefore, the project cannot be defined as an investment or as a loan unless i_m is specified. It is natural to determine the intervals of monotonicity $(-1, r_1\rangle\,, \dots, \langle r_k, r_{k+1}\rangle\,, \dots, \langle r_K, \infty)$, where $K < n$, i.e the intervals between the extremal points r_k of the function PV.

We call such an interval I the *investment (or loan) interval* of the project if

$$\frac{\partial PV}{\partial i}(i) = \sum_{k=1}^{n} \frac{-kB_k}{(1+i)^{k+1}} \leq 0 \quad \left(\text{or } \frac{\partial PV}{\partial i}(i) \geq 0\right) \quad \forall i \in I.$$

When $i_m \in I$, the type of relevant interval is given. If $i_m = r_k$, it does not matter which interval, right or left, we choose. Moreover, in case $i^* \in I$ (at most one IRR) we take it as the relevant IRR with respect to i_m. The significance of this i^* corresponds to the investment (loan) interval I. When investment, we accept (reject) the whole project if $i^* \geq i_m$ ($i^* \leq i_m$). (When loan, then vice versa.)

In the case when no IRRs are in I, we accept the project if PV is positive at the endpoints of the interval I, regardless of the magnitudes of the IRRs of the project, see Fig. 2. The presence of complex roots means the sign of derivative changes twice without crossing the i axis.)

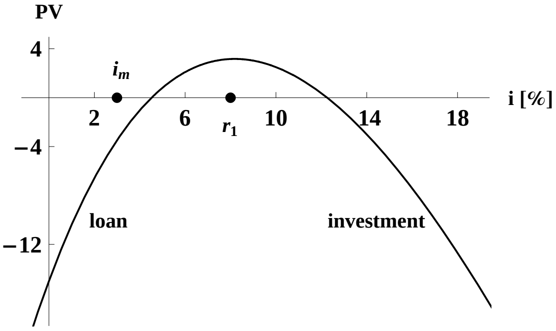


Fig. 1: *Anomalous project* P1.

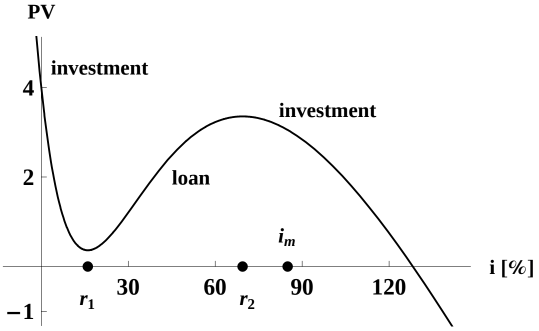


Fig. 2: *Project* P2*, unique IRR.*

	B_0	B_1	B_2	B_3	B_4	B_5	IRR[%]
P1	-815	900	-100	1200	-1200	0	4.5 ; 12.3
P2	-77	340	-470	252	-110	69	-108.5 ± 53.7 i ; 15.1 ± 6.9 i ; 128.2

5. Conclusions

The problem of rating the project has been substantially simplified with the help of computers graphing the function PV and by means of special software tools (Maple, Mathematica), which can provide all existing (even complex-valued) roots. But how the financial manager should deal with a project when the symbolic programs are not available? When the cash flows have multiple sign changes, several spreadsheet calculators tend to give “ERR” (an error message) instead of IRR.

Before any calculation one should ask about the uniqueness of the IRR because if it is unique the decision making is straightforward. We found the (S-G) conditions very strong for determining the uniqueness, though they have reasonable economic meaning. There are a lot of projects with the monotonic PV and unique IRR that do not fulfill the (S-G) conditions.

When multiplicity occurs, the IRRs should not be compared even within a single project. Every real IRRs are significant, some are the measures for the investment return and others have the economic meaning as the cost of loan. If the market rate i_m is specified, the project follows the type of corresponding interval. As an investment decision tools, the IRR and PV are coherent criteria. Together give a better analysis than the former or the latter alone.

References

[1] R.H. Bernhard: *A simplification and an extension of the Bernhard-de Faro sufficient condition for a unique non-negative internal rate of return.* J. Fin. Quant. Anal. **15** (1996), 201–209.

[2] C. De Faro: *A sufficient condition for a unique non-negative internal rate of return: futher comments.* J. Fin. Quant. Anal. **13** (1978), 577–584.

[3] S. Gronchi: *On investment criteria based on the internal rate of return.* Oxford Economic Papers **38** (1986), 174–180.

[4] M.M. Hajdasinski: *Technical note–the internal rate of return (IRR) as a financial indicator.* The Engineering Economist **49** (2004), 185–197.

[5] G.B. Hazen: *A new perspective on multiple internal rates of return.* The Engineering Economist **48** (2003), 31–51.

[6] W.H. Jean: *On multiple rates of return.* The Journal of Finance **23** (1968), 187–191.

[7] S. Kaplan: *A note on a method for precisely determining the uniqueness or nonuniqueness of the internal rate of return for a proposed investment.* J. Industr. Engrg. **16** (1965), 70–71.

[8] C.J. Norstrom: *A sufficient conditions for a unique nonnegative internal rate of return.* J. Fin. Quant. Anal. **7** (1972), 1835–1839.

[9] J.F. Oehmke: *Anomalies in net present value calculations.* Economics Letters **67** (2000), 349–351.

[10] S.D. Promislow, D. Spring: *Uniqueness of yield rates.* Actuarial Reserch Clearing House **1** (1996), 225–236.

[11] C.S. Soper: *The marginal efficiency of capital: A further note.* The Economic Journal **69** (1959), 174–177

ON CONSTRUCTION OF THE COARSE SPACE IN THE BDDC METHOD*

Jakub Šístek, Pavel Burda, Marta Čertíková, Jaroslav Novotný

1. Introduction

Domain Decomposition (DD) methods are getting increasingly popular in various areas of engineering for offering a convenient way to parallelize analysis by the finite element method (FEM). Among the most popular members of this family for symmetric positive definite problems, such as linear elasticity, are the FETI-DP method of Farhat et al. [3] and the BDDC method of Dohrmann [1]. It has been recently proved by Mandel, Dohrmann, and Tezaur [5] that the two methods are spectrally equivalent, which unifies the theory for both methods and allows application of various results already known for FETI-DP to BDDC and vice versa.

Both methods use a coarse space based on a set of selected nodes, called *corners*, in which the continuity of subdomain solutions is required. These nodes assure that the subdomain problems are also positive definite and might be solved by a standard direct method. The set of corners gives rise to an important part of the *coarse space* and the corresponding *coarse problem.*

While corners assure a convenient solvability of subdomain problems, they do not suffice for robust preconditioning with respect to discretization parameter h in three dimensions. This fact was first observed for FETI-DP experimentally in [3], and theoretically in [4]. The theoretical treatment requires adding constraints on equality of averages over subdomain *edges* and *faces* to the coarse problem. This might be done in a uniform way (as was done in [1, 2]), or in a more sophisticated adaptive way, which nearly optimally decreases the condition number of the preconditioned operator (see [6]).

In the present paper, we investigate different and more straightforward approach to the generation of coarse space, that consists of simple addition of more nodes from the interface to the set of corners. Although this is not the optimal case, presented numerical experiments for an industrial application of linear elasticity show that combining both approaches can lead to synergic effect and further reduce the overall computational time.

*This work was supported by grant No. 106/08/0403 of the Czech Science Foundation.

2. The Schur complement method

Consider a boundary value problem with self-adjoint operator defined on domain $\Omega \subset \mathbb{R}^2$ or $\mathbb{R}^3$. If we discretize the problem by means of the standard finite element method (FEM), we arrive at the solution of system of linear equations in the matrix form

$$\mathbf{K}\mathbf{u} = \mathbf{f}, \tag{1}$$

where $\mathbf{K}$ is a large, sparse, symmetric positive definite (SPD) matrix and $\mathbf{f}$ is the vector of right-hand-side.

Let us decompose domain Ω into N non-overlapping subdomains Ω_i, $i = 1, \ldots, N$. Unknowns common to at least two subdomains are called *boundary unknowns* and the union of all boundary unknowns is called the *interface.* Remaining unknowns belong to subdomain *interiors.*

The first step is the reduction of the problem to the interface. Without loss of generality, suppose that unknowns are ordered so that interior unknowns form the first part and the interface unknows form the second part of the solution vector, i.e. $\mathbf{u} = \begin{bmatrix} \mathbf{u}_\mathrm{o} & \widehat{\mathbf{u}} \end{bmatrix}^T$, where $\mathbf{u}_\mathrm{o}$ stands for all interior unknowns and $\widehat{\mathbf{u}}$ for unknowns at interface. System (1) can now be formally rewritten to block form

$$\begin{bmatrix} \mathbf{K}_\mathrm{oo} & \mathbf{K}_\mathrm{or} \\ \mathbf{K}_\mathrm{ro} & \mathbf{K}_\mathrm{rr} \end{bmatrix} \begin{bmatrix} \mathbf{u}_\mathrm{o} \\ \widehat{\mathbf{u}} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_\mathrm{o} \\ \widehat{\mathbf{f}} \end{bmatrix}. \tag{2}$$

The hat symbol ($\widehat{\ }$) is used to denote global interface quantities. If we suppose the interior unknowns ordered subdomain after subdomain, then the submatrix $\mathbf{K}_\mathrm{oo}$ is block-diagonal with each diagonal block corresponding to one subdomain.

After eliminating all the interior unknowns from (2), we arrive to *Schur complement problem* for the interface unknowns

$$\widehat{\mathbf{S}}\, \widehat{\mathbf{u}} = \widehat{\mathbf{g}}, \tag{3}$$

where $\widehat{\mathbf{S}} = \mathbf{K}_\mathrm{rr} - \mathbf{K}_\mathrm{ro}\mathbf{K}_\mathrm{oo}^{-1}\mathbf{K}_\mathrm{or}$ is a *Schur complement* of (2) with respect to interface and $\widehat{\mathbf{g}} = \widehat{\mathbf{f}} - \mathbf{K}_\mathrm{ro}\mathbf{K}_\mathrm{oo}^{-1}\mathbf{f}_\mathrm{o}$ is sometimes called *condensed right hand side.* Interior unknowns $\mathbf{u}_\mathrm{o}$ are determined by interface unknowns $\widehat{\mathbf{u}}$ as

$$\mathbf{K}_\mathrm{oo}\mathbf{u}_\mathrm{o} = \mathbf{f}_\mathrm{o} - \mathbf{K}_\mathrm{or}\widehat{\mathbf{u}}. \tag{4}$$

The solution can now be divided into three steps: (i) construction of problem (3), (ii) solution of problem (3), and (iii) resolution of interior unknowns by (4). Because problem (4) represents N independent subdomain problems with Dirichlet boundary condition prescribed on the interface, steps (i) and (iii) are performed in parallel and are very fast. Thus, the main concern represents the solution of problem (3) in step (ii). This problem is solved by the preconditioned conjugate gradient method (PCG). Since only matrix-vector multiplications are necessary in PCG, the Schur complement matrix $\widehat{\mathbf{S}}$ need not be constructed explicitly. Instead, we only need to factorize the block $\mathbf{K}_{\mathrm{oo}}$ and perform the multiplications with $\widehat{\mathbf{S}}$ as $\widehat{\mathbf{S}}\widehat{\mathbf{v}} = \mathbf{K}_{\mathrm{rr}}\widehat{\mathbf{v}} - \mathbf{K}_{\mathrm{ro}}\mathbf{w}$, where $\mathbf{K}_{\mathrm{oo}}\mathbf{w} = \mathbf{K}_{\mathrm{or}}\widehat{\mathbf{v}}$. Also this process may be performed subdomain by subdomain in parallel, using blocks of local subdomain matrices $\mathbf{K}_i$ defined in the next section.

3. The BDDC method

The BDDC method may be viewed as a preconditioner for problem (3) when it is solved by the PCG method. The main idea of the preconditioner is to split the problem into independent subdomain problems and the global coarse problem. This process is described in this section.

Let $\mathbf{K}_i$ be the local subdomain matrix obtained by the sub-assembling of element matrices of elements contained in subdomain Ω_i. The global stiffness matrix $\mathbf{K}$ might be obtained by further assembling of these matrices on the interface. We introduce the *coarse space basis functions* on each subdomain Ω_i represented by columns of matrix $\boldsymbol{\Psi}_i$, which is the solution to the saddle point problem with multiple right hand sides

$$\begin{bmatrix} \mathbf{K}_i & \mathbf{C}_i^T \\ \mathbf{C}_i & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Psi}_i \\ \boldsymbol{\Lambda}_i \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix}. \tag{5}$$

This is a key problem in the BDDC method and deserves a careful explanation. Matrix $\boldsymbol{\Lambda}_\mathbf{i}$ is a block of Lagrange multipliers, $\mathbf{I}$ is an identity block.

Matrix $\mathbf{C}_i$ represents constraints on functions $\boldsymbol{\Psi}_i$, one row per each. These constraints enforce prescribed values of the *coarse degrees of freedom* on subdomain. They guarantee continuity of solution in selected points (*corners*) or equality of averages over some subsets of interface (*edges* or *faces*) of adjacent subdomains. While the former type of constraints corresponds to exactly one nonzero entry in a row of $\mathbf{C}_i$, the latter leads to several nonzeros in a row. Each column of matrix $\boldsymbol{\Psi}_i$ defined by (5) represents one coarse space basis function on subdomain Ω_i and corresponds to one local coarse degree of freedom.

Using the coarse basis functions $\boldsymbol{\Psi}_i$, we define the *local coarse matrix* $\mathbf{K}_{Ci} = \boldsymbol{\Psi}_i^T \mathbf{K}_i \boldsymbol{\Psi}_i$ on each subdomain. This matrix has the dimension equal to the number of constraints for each subdomain.

Let $\mathbf{R}_{Ci}$ realize the restriction of global coarse degrees of freedom to local coarse degrees of freedom on subdomain Ω_i. Using this matrix, we can construct the global *coarse matrix* by the assembling procedure, formally written as

$$\mathbf{K}_C = \sum_{i=1}^{N} \mathbf{R}_{Ci}^T \mathbf{K}_{Ci} \mathbf{R}_{Ci}. \tag{6}$$

We are ready to describe the algorithm of the BDDC method. Suppose $\widehat{\mathbf{r}} = \widehat{\mathbf{g}} - \widehat{\mathbf{S}}\,\widehat{\mathbf{u}}$ is a residual within the PCG method. Let us define matrices $\mathbf{E}_i^T$ for distribution of $\widehat{\mathbf{r}}$ to subdomains. Each matrix selects the interface unknowns of subdomain Ω_i and weights them so that the decomposition of unity applies to the residual across subdomains. It puts zeros to unknowns interior to subdomains. This corresponds to computing with Schur complement (see [7] for details). The residual assigned to subdomain Ω_i is computed as $\mathbf{r}_i = \mathbf{E}_i^T \widehat{\mathbf{r}}$. The subdomain correction from Ω_i is now defined as the solution to system

$$\begin{bmatrix} \mathbf{K}_i & \mathbf{C}_i^T \\ \mathbf{C}_i & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{z}_i \\ \lambda_i \end{bmatrix} = \begin{bmatrix} \mathbf{r}_i \\ \mathbf{0} \end{bmatrix}. \tag{7}$$

The residual for the coarse problem is constructed using the coarse basis functions subdomain by subdomain and assembling the contribution as

$$\mathbf{r}_C = \sum_{i=1}^{N} \mathbf{R}_{Ci}^T \mathbf{\Psi}_i^T \mathbf{E}_i^T \widehat{\mathbf{r}}. \tag{8}$$

The coarse correction is defined as the solution to problem

$$\mathbf{K}_C \, \mathbf{z}_C = \mathbf{r}_C. \tag{9}$$

Both corrections are then added together and averaged on the interface by matrices $\mathbf{E}_i$ to produce the preconditioned residual

$$\widehat{\mathbf{z}} = \sum_{i=1}^{N} \mathbf{E}_i \left(\mathbf{\Psi}_i \mathbf{R}_{Ci} \mathbf{z}_C + \mathbf{z}_i \right). \tag{10}$$

Properties of the coarse space are fully determined by constraints in matrices $\mathbf{C}_i$. The more constraints are prescribed, the more efficient preconditioner is constructed, but the larger coarse space is obtained, making factorization of matrix $\mathbf{K}_C$ more expensive.

4. Numerical results

We investigate the two ways of constructing the coarse space on a problem of elasticity analysis of a turbine nozzle, through which the steam enters the turbine blades. The geometry is discretized using 2 696 quadratic elements, which leads to 13 418 nodes and 40 254 unknowns. The mesh was divided into 16 and 32 subdomains, respectively. The division into 16 subdomains is depicted in Figure 1. Presented calculations were performed on 16 processors of SGI Altix 4700 computer of Supercomputing Centre of Czech Technical University in Prague.

The first experiment consists in adding more interface nodes to the set of corners. It should be noted that the initial set of corners is already sufficient for all subdomain problems to be nonsingular, as well as the coarse problem. In Figure 2, we present the plot of number of PCG iterations with respect to the number of corners. The effect on condition number is presented in Figure 3.

We can observe from these plots that some initial amount of corners is necessary for fast decrease of these values, while from some amount, these values behave quite linearly in dependence on number of corners. These points of break are quite important, because they correspond to the optimal values of wall clock times presented in Figure 4. This is caused by the fact that the number of PCG iterations decreases rapidly at this point, while for adding more corners, the factorization of the growing coarse problems starts to dominate the time. Thus, it is desirable to set-up the preconditioner in such a way, that it works around this point or slightly to the right. The problem is that this point is unknown a priori. For the turbine nozzle, it corresponds to approximately 20 percent of all interface nodes for 16 subdomains and to 25 percent for the case of 32 subdomains. Similar results were also observed for other industrial problems. Although this value highly depends on problem topology and division into subdomains, from the practical point of view it seems that putting as much as a quarter of interface nodes into the set of corners is a reasonable set-up.

In the second experiment, we vary the size of the coarse problem in a conceptually different way – besides the continuity at corners, we enforce the equality of arithmetic averages of the approximate solution on all edges, on all faces, and on both edges and faces.

The results for the division into 16 subdomains are summarized in Table 1 for the initial set of 30 corner nodes, and in Table 2 for 280 corner nodes, the optimal number determined in the first experiment.

We can observe that adding constraints on averages can significantly improve the preconditioner and decrease the computational time. However, averages on faces might be too expensive considering time of computation. It could also be seen from the tables that using the optimal number of corner nodes can lead to improvement of computational times after addition of averages on edges.

5. Conclusion

We have presented a comparison of two ways for generating the coarse space in the BDDC method. We can conclude that while adding averages over edges and faces

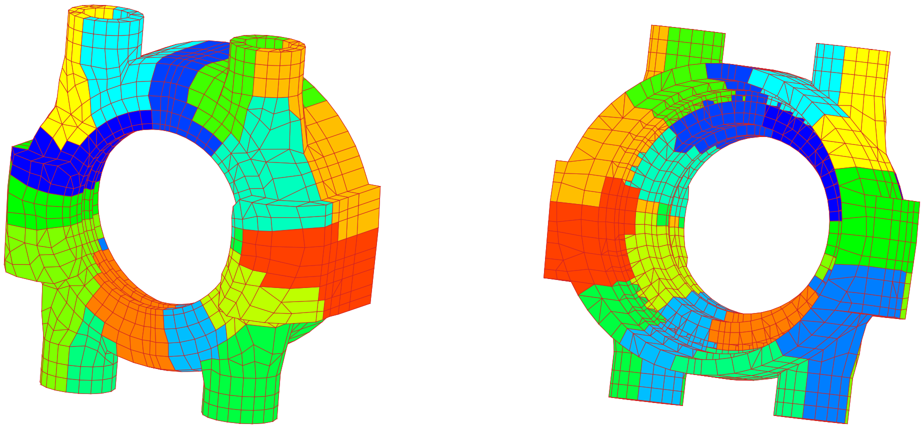


Fig. 1: *Turbine nozzle, division into 16 subdomains.*

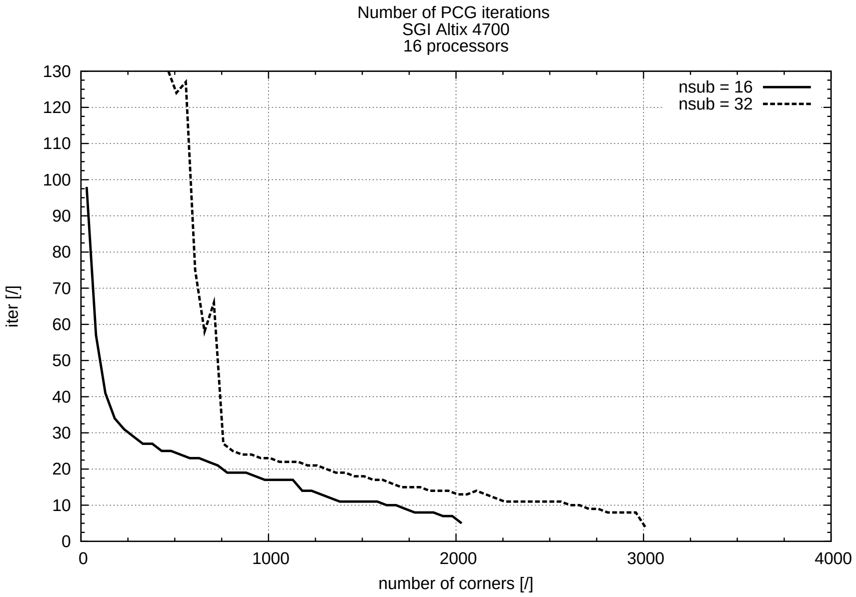


Fig. 2: *Turbine nozzle, number of iterations in dependence on number of corners, 16 and 32 subdomains.*

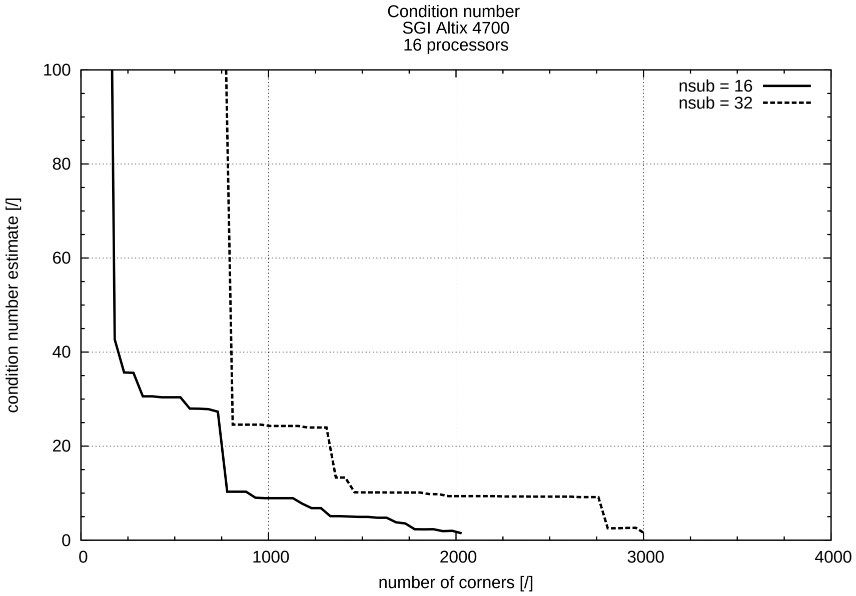


Fig. 3: *Turbine nozzle, condition number in dependence on number of corners, 16 and 32 subdomains.*

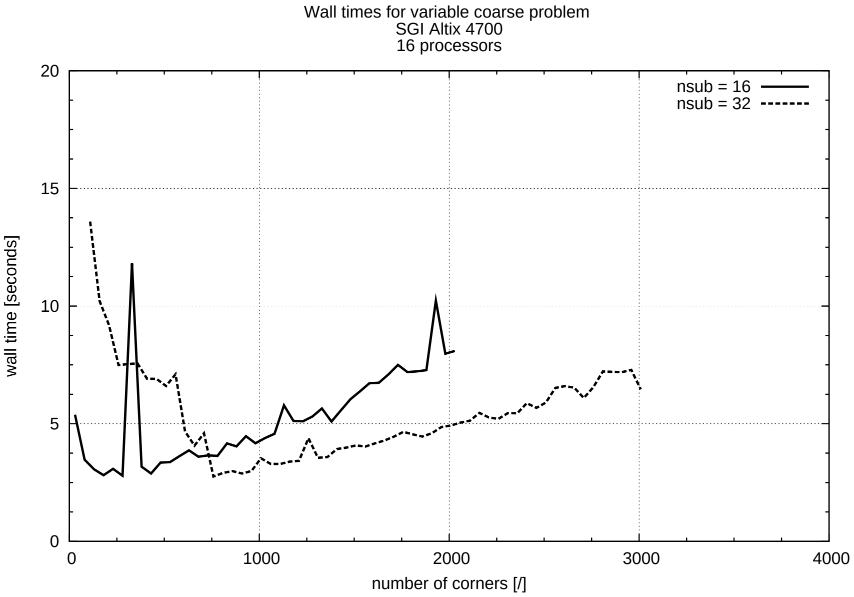


Fig. 4: *Turbine nozzle, wall clock time in dependence on number of corners, 16 and 32 subdomains.*

coarse problem	corners	corners+edges	corners+faces	corners+edges+faces
iterations	98	78	41	36
cond. number est.	2 933	1 546	164	142
factorization (sec)	0.5	0.6	0.6	0.8
pcg iter (sec)	4.6	3.7	1.9	1.9
total (sec)	5.3	4.5	2.8	2.9

Tab. 1: *Turbine nozzle, 16 subdomains, 30 corners, adding averages.*

coarse problem	corners	corners+edges	corners+faces	corners+edges+faces
iterations	29	26	26	23
cond. number est.	36	23	25	13
factorization (sec)	1.2	1.1	1.5	1.8
pcg iter (sec)	1.9	1.6	1.7	1.8
total (sec)	3.6	2.9	3.5	3.8

Tab. 2: *Turbine nozzle, 16 subdomains, 280 corners, adding averages.*

is often applied in literature, addition of more corners might be also contributive and the best results are likely to be obtained by combination of both approaches. According to our observations, there is an optimal number of corner nodes, with a steep decrease of number of iterations followed by the lowest computational time. Constraints on averages over faces might be too expensive compared to averages over edges. These observations advocate the adaptive approach for selecting constraints described in [6].

References

[1] C.R. Dohrmann: *A preconditioner for substructuring based on constrained energy minimization.* SIAM J. Sci. Comput. **25** (2003), 246–258.

[2] C. Farhat, M. Lesoinne, P. Le Tallec, K. Pierson, D. Rixen: *FETI-DP: a dual-primal unified FETI method. I. A faster alternative to the two-level FETI method.* Internat. J. Numer. Methods Engrg. **50** (2001), 1523–1544.

[3] C. Farhat, M. Lesoinne, K. Pierson: *A scalable dual-primal domain decomposition method.* Numer. Linear Algebra Appl. **7** (2000), 687–714.

[4] A. Klawonn, O. B. Widlund, M. Dryja: *Dual-primal FETI methods for three-dimensional elliptic problems with heterogeneous coefficients.* SIAM J. Numer. Anal. **40** (2002), 159–179.

[5] J. Mandel, C.R. Dohrmann, R. Tezaur: *An algebraic theory for primal and dual substructuring methods by constraints.* Appl. Numer. Math. **54** (2005), 167–193.

[6] J. Mandel, B. Sousedík: *Adaptive selection of face coarse degrees of freedom in the BDDC and the FETI-DP iterative substructuring methods.* Comput. Methods Appl. Mech. Engrg. **196** (2007), 1389–1399.

[7] J. Šístek, J. Novotný, J. Mandel, M. Čertíková, P. Burda: *BDDC by a frontal solver and stress computation in a hip joint replacement.* Submitted to Math. Comp. Simul., 2008.

SPACE-TIME ADAPTIVE *hp*-FEM: METHODOLOGY OVERVIEW*

Pavel Šolín, Karel Segeth, Ivo Doležel

Abstract

We present a new class of self-adaptive higher-order finite element methods (*hp*-FEM) which are free of analytical error estimates and thus work equally well for virtually all PDE problems ranging from simple linear elliptic equations to complex time-dependent nonlinear multiphysics coupled problems. The methods do not contain any tuning parameters and work reliably with both low- and high-order finite elements. The methodology was used to solve various types of problems including thermoelasticity, microwave heating, flow of thermally conductive liquids etc. In this paper we use a combustion problem described by a system of two coupled nonlinear parabolic equations for illustration. The algorithms presented in this paper are available under the GPL license in the form of a modular C++ library HERMES[1].

1. Introduction

Partial differential equations (PDEs) describe many physical processes whose prediction and control are important to people. The most frequently used technique for the numerical solution of PDEs is the finite element method (FEM). The origins of this method are often associated with R. Courant [2] who solved numerically torsion problems in cylinders, drawing on a large body of earlier results for PDEs developed by Rayleigh, Ritz, and Galerkin. Since the 1940s, the method achieved a high degree of maturity and also the computational standards have changed. Nowadays, the significance of error control is greater than ever before, and the number of computations where adaptive mesh refinement algorithms are employed is rising very quickly.

On the other hand, self-adaptive finite element methods for PDEs have been studied by mathematicians for decades but so far, practitioners have been rather reluctant to use them. To understand why, note that every self-adaptive finite element method is guided by an *error estimator*. With a suitable error estimator in hand, the rest of automatic adaptivity (such as mesh refinement) is a purely technical matter. The current standard in computational PDEs are *analytical error estimators* – mathematical formulae "on paper". However, there is a very large number of such formulae, often they contain simplifying assumptions, are restricted to numerical

*This work was supported by grant 102/07/0496 of the Czech Science Foundation, grants IAA100190803 and IAA100760702 of the Grant Agency of the Academy of Sciences of the Czech Republic, Research Plan of the Academy of Sciences of the Czech Republic AV0Z10190503, and by Research Center 1M4674788502 of the Ministry of Education, Youth, and Sports.

[1] See the home page of the HERMES project http://spilka.math.unr.edu/hermes/.

methods of low order of accuracy, involve constants of unknown size, and/or include problem-dependent parameters that need to be tuned. In general, analytical error estimators are neither simple to use nor universal enough to cover a wide spectrum of problems of interest to practitioners. One cannot use them efficiently without a deep understanding of the underlying mathematics. From the point of view of a practitioner whose expertise is elsewhere and who would like to solve PDEs routinely, in order to obtain information that he or she needs for his research or application, the cost of dealing with burdens associated with self-adaptive methods often is not acceptable.

In this paper, we propose a way to circumvent this problem and make self-adaptive computational methods easily accessible to the broad computational community. The main idea of our approach is to use universal, computational error estimators that are motivated by modern embedded self-adaptive methods for ordinary differential equations (ODEs). These ODE methods are highly popular among engineers and practitioners due to their simplicity and universality: In every step, the algorithm computes two approximations with different orders of accuracy, and the error is estimated by their difference. Note that such error estimator is virtually independent of the underlying equation. The key requirement for practical applicability, however, is that the two approximations are computed efficiently. For example, in embedded Runge-Kutta RK2(3) methods, one evaluates two stages to obtain a second-order accurate approximation, and adds one more stage to obtain an approximation that is third-order accurate. Hence, one only pays the cost of a third-order method but also obtains a second-order error estimator.

The outline of the paper is as follows: In Sections 2 and 3 we present two techniques which are essential for the space-time adaptive algorithms: conforming higher-order approximation with arbitrary-level hanging nodes and the multimesh *hp*-FEM. In Section 4 we present a universal adaptivity algorithm for higher-order finite element methods. In Section 5 we extend this technique to time-dependent problems by combining the multi-mesh FEM with the classical Rothe's method. Example application to a flame propagation problem is shown in Section 7.

2. Approximation with arbitrary-level hanging nodes

The efficiency and algorithmic simplicity of our adaptive higher-order finite element algorithms is largely due to the technique of arbitrary-level hanging nodes [7]. When working with regular meshes (where two elements either share a common vertex, common edge, or their intersection is empty), adaptivity often is done using the *red-green refinement strategy* [1]. This technique first subdivides desired elements into geometrically convenient subelements with hanging nodes and then it eliminates the hanging nodes by forcing refinement of additional elements, as illustrated in Fig. 1.

This approach preserves the regularity of the mesh at the price of producing additional (forced) refinements and new degrees of freedom. Often, it creates elements

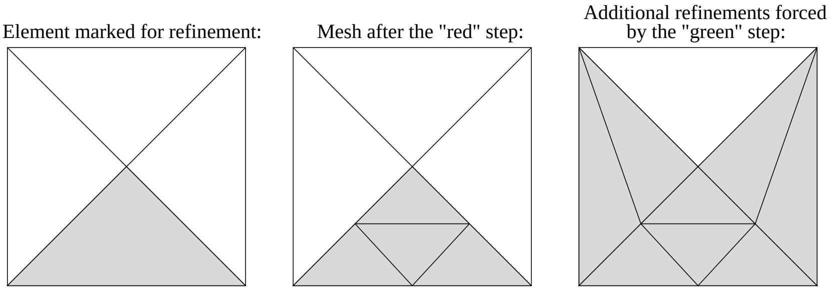

Element marked for refinement:
Mesh after the "red" step:
Additional refinements forced by the "green" step:

Fig. 1: *Red-green refinement.*

with sharp angles which, in general, are not desirable in finite element analysis.

The "green" refinements can be avoided by introducing *hanging nodes*, i.e., by allowing *irregular meshes* where element vertices can lie in the interior of edges of other elements. To ease the computer implementation, most finite element codes working with hanging nodes limit the maximum difference of refinement levels of adjacent elements to one (1-*irregularity rule*) – see, e.g., [3, 9]. In the following, by k-*irregularity rule* (or k-*level hanging nodes*) we mean this type of restriction where the maximum difference of refinement levels of adjacent elements is k. In this context, $k = 0$ corresponds to adaptivity with regular meshes and $k = \infty$ to adaptivity with arbitrary-level hanging nodes. It is illustrated in Fig. 2 that even the 1-irregularity rule does not avoid all forced refinements:

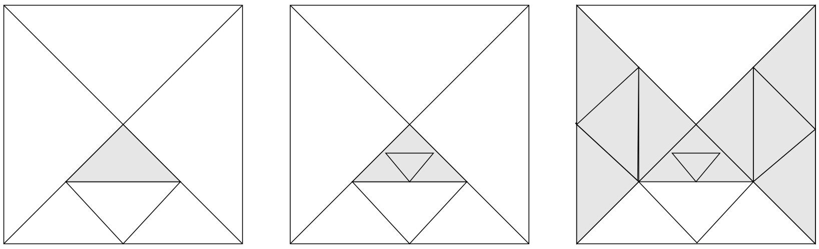


Fig. 2: *Refinement with 1-irregularity rule.*

The amount of forced refinements in the mesh depends on the level of hanging nodes allowed. Let us introduce a simple model problem which shows how the level of hanging nodes influences the number of degrees of freedom and condition number of the stiffness matrices: Consider a square domain $\Omega = (-1, 1)^2$ covered with a mesh consisting of four cubic elements, as shown in Fig. 3.

We solve the Poisson equation $-\Delta u = f$ in Ω with $u = 0$ on the boundary. Assume a right-hand side f such that the corresponding exact solution u is zero everywhere in Ω with the exception of a significant local perturbation contained inside of a small triangle T_n with the vertices $[-2^{-n}, -2^{-n}]$, $[0, 0]$, $[-2^{-n}, 2^{-n}]$. Fig. 4 shows, for $n = 5$, meshes obtained under various irregularity rules:

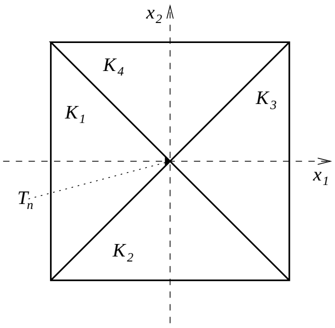


Fig. 3: *Domain* Ω *and initial coarse mesh.*

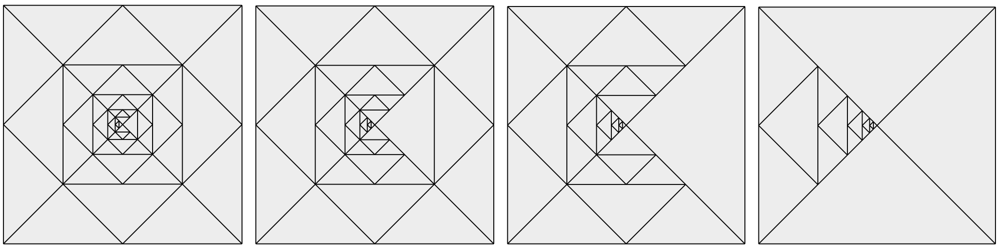


Fig. 4: *Meshes obtained with* k*-irregularity rules,* $k = 1, 2, 3, \infty$.

Notice that the amount of forced refinements within the elements K_2, K_3, and K_4 decreases as the parameter k grows. Next we run the adaptive procedure with cubic elements for $n = 1, 2, \ldots, 15$. Fig. 5 shows the number of degrees of freedom corresponding to the final meshes. The horizontal axis represents the spatial scale 2^{-n}. Fig. 6 shows the condition number of the corresponding stiffness matrices.

These results demonstrate that the performance of automatic adaptivity with arbitrary-level hanging nodes is superior to adaptivity on regular meshes, and even to adaptivity with one-, two-, or three-irregular meshes. Obviously, quantitative gains in the number of degrees of freedom and condition number of the stiffness matrix depend on specific features of the solved problem. In our experience, the advantages of the technique of arbitrary-level hanging nodes are most apparent in problems containing curvilinear material interfaces or boundary/internal layers.

3. Multi-mesh *hp*-FEM

The basic ingredient for the self-adaptive finite element methods presented in this paper is an algorithmic framework that allows us to work efficiently with various physical fields or various approximations of the same physical field defined on geometrically and polynomially different meshes. The higher-order multi-mesh FEM was first introduced in the context of linear thermoelasticity in [8], where the displacement components u_1, u_2 and the temperature T were approximated on different meshes equipped with independent adaptivity mechanisms.

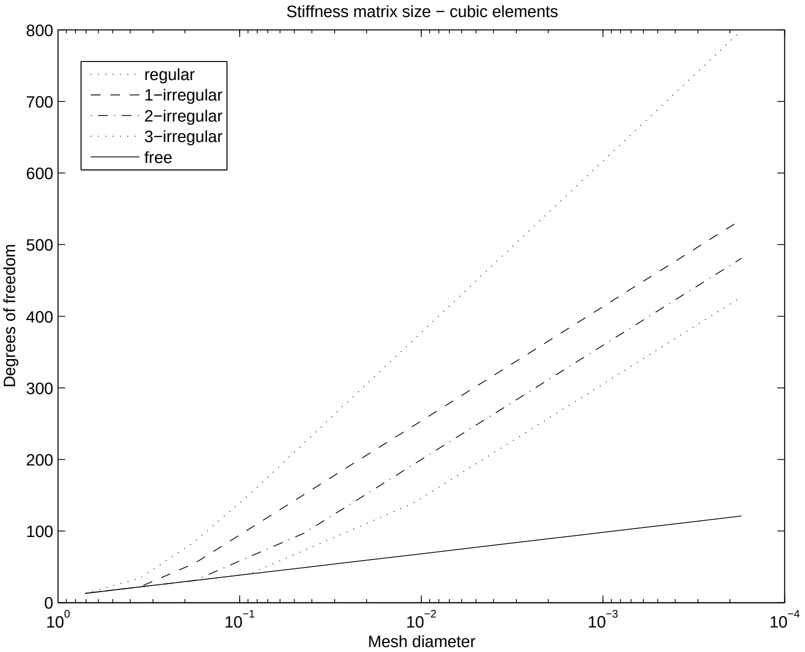


Fig. 5: *Relation between the size of the stiffness matrix and the level k of hanging nodes ($k = 0, 1, 2, 3, \infty$).*

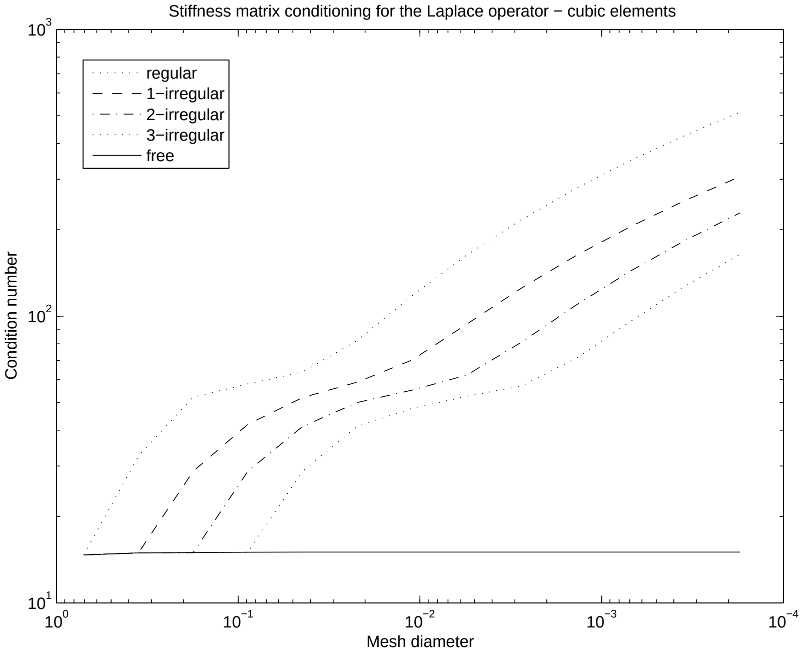


Fig. 6: *Relation between the condition number of the stiffness matrix and the level k of hanging nodes ($k = 0, 1, 2, 3, \infty$).*

The main ideas of the multi-mesh hp-FEM are as follows: For the sake of programming feasibility, we restrict ourselves to meshes derived from a common coarse *master mesh* τ_m via sequences of mutually independent local refinements. The master mesh τ_m is very coarse and often it is not even used for discretization purposes – it serves as the top of a tree-like data structure which is utilized by the multi-mesh assembling procedure. The situation is illustrated in parts A – D of Fig. 7. In part E of Fig. 7 we also show the geometrical union of all meshes in the system that we call *union mesh* and denote by τ_u.

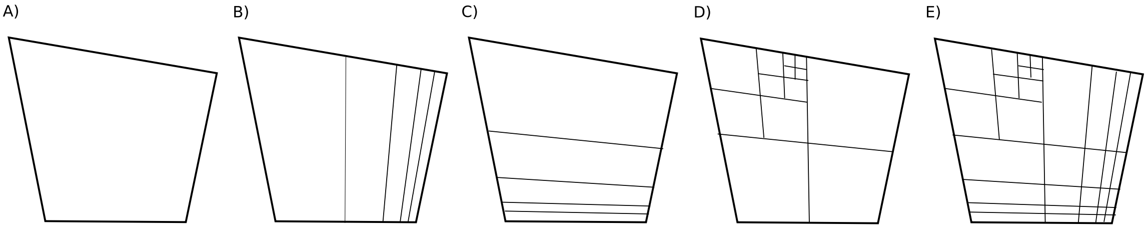


Fig. 7: *Example of a master mesh τ_m (left), meshes τ_1, τ_2, τ_3 obtained by its refinements, and the corresponding union mesh τ_u (right).*

It is worth mentioning that the union mesh is never created physically in the computer memory but its virtual elements are parsed by the element-by-element assembling procedure as usual in higher-order finite element methods [10], and that the multi-mesh FEM uses hanging nodes of arbitrary level (see [7], also available as a preprint on-line[2]). This technique eliminates forced refinements and thus contributes greatly to the modularity and efficiency of automatic adaptivity algorithms. The multi-mesh FEM would work (less efficiently) also with one-level hanging nodes, but not on regular meshes (due to possibly conflicting green refinements).

4. Adaptive hp-FEM with arbitrary-level hanging nodes

In contrast to standard adaptive FEM (h-FEM), automatic adaptivity in the hp-FEM requires more information about the behavior of the error in element interiors (see, e.g., [3, 4, 10] and the references therein). Some authors investigate numerically the analyticity of the solution in every element in order to decide between p- and h-refinement [4]. Such approach uses two refinement candidates per element, as illustrated in Fig. 8 (the numbers in elements stand for their polynomial degrees). According to our experience, at least for elliptic problems this strategy yields exponential convergence.

We prefer a different approach where more refinement candidates are considered, as shown in Fig. 9.

Typically, we vary the polynomial degrees in the subelements by two, which for a triangular element yields $3^4 = 81$ h-refinement candidates. The strategy was described in detail in [7]. Since in the latter case every refinement candidate can be

[2] http://www.math.utep.edu/preprints/2006/2006-07.pdf

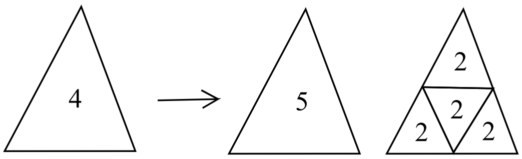


Fig. 8: *hp-adaptivity with two refinement candidates (p and h refinement).*

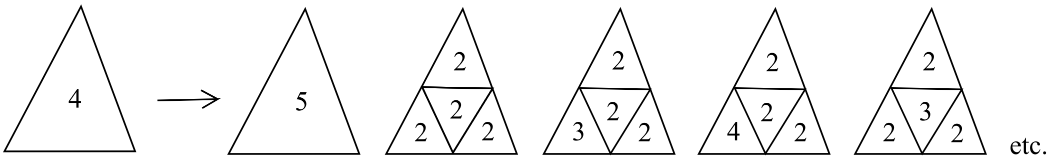


Fig. 9: *hp-adaptivity with multiple refinement candidates.*

reproduced using several steps with the pair of candidates of the former strategy, it is not surprising that usually the convergence curves are almost identical when error is plotted as a function of the number of degrees of freedom. However, according to our experience, computations with the latter approach usually take less CPU time since fewer adaptivity steps are needed and thus the discrete problem is solved less frequently. This is illustrated in Fig. 10.

Obviously, the latter strategy requires even more information about the error than the level of its analyticity. In order to select an optimal refinement candidate, we need to know the approximate *shape* of the error function $\epsilon_{h,p} = u - u_{h,p}$. In principle, this information could be recovered from suitable estimates of higher derivatives of the solution, but such approach is not very practical and it has not been used by anyone to our best knowledge. In practice, we employ the technique of *reference solutions* [3]. The reference solution u_{ref} is sought in an enriched finite element space V_{ref}, and the error function is approximated as $\epsilon_{h,p} \approx u_{ref} - u_{h,p}$. The reference space V_{ref} is constructed in such a way that all elements in the mesh are subdivided uniformly and their polynomial degree is increased, i.e., $V_{ref} = V_{h/2,p+1}$. The method for selecting the optimal refinement candidate will be described in the following.

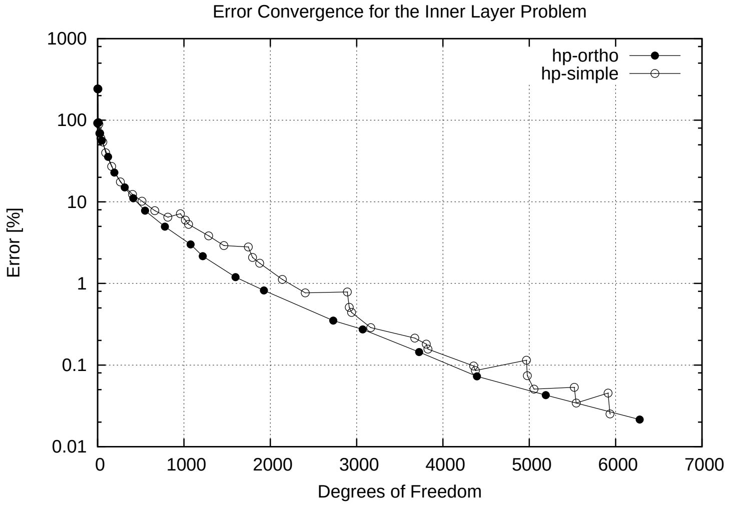
4.1. Element-by-element adaptivity algorithm

With an a-posteriori error estimate of the form

$$\epsilon_{h,p} \approx u_{ref} - u_{h,p}, \tag{1}$$

the outline of our *hp*-adaptivity algorithm is as follows:

1. Assume an initial coarse mesh $\tau_{h,p}$ consisting of (usually) quadratic elements. Besides other technical data, user input includes a prescribed tolerance $TOL > 0$ for the H^1-norm of the approximate error function (1) and the number D_{DOF} of degrees of freedom to be added in every *hp*-adaptivity step.



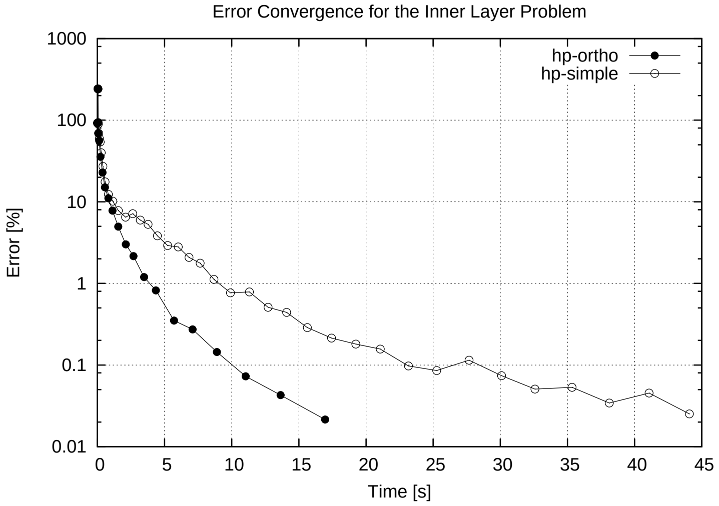


Fig. 10: *Illustration of performance of adaptivity schemes with two refinement candidates per element (simple) and multiple candidates per element (ortho). The horizontal axis shows the number of DOF (top) and CPU time (bottom).*

2. Compute coarse mesh approximation $u_{h,p} \in V_{h,p}$ on $\tau_{h,p}$.

3. Find reference solution $u_{ref} \in V_{ref}$, where V_{ref} is obtained by dividing all elements and increasing the polynomial degrees by one.

4. Construct the approximate error function (1), calculate its norm

$$ERR_i^2 = \|\epsilon_{h,p}\|_{1,2}^2$$

on every element K_i in the mesh, $i = 1, 2, \ldots, M$. Calculate the global error,

$$ERR^2 = \sum_{i=1}^{M} ERR_i^2.$$

5. If $ERR \le TOL$, stop computation and proceed to postprocessing.

6. Sort all elements into a list L according to their ERR_i values in decreasing order.

7. While the number of newly added degrees of freedom in this step is less than D_{DOF} do:

   (a) Take the next element K from the list L.

   (b) Perform *hp*-refinement of K (to be described in more detail in Paragraph 4.2). Note that the refinement of K may introduce new hanging nodes on its edges, but the surrounding mesh elements are not affected.

8. Adjust polynomial degrees on unconstrained edges (edges without hanging nodes, cf. [10]) using the *minimum rule* (every unconstrained edge is assigned the minimum of the polynomial degrees on the pair of adjacent elements).

9. Continue with step 2.

4.2. Selection of optimal *hp*-refinement of an element

Let $K \in \tau_{h,p}$ be an element of polynomial degree p_K that was marked for refinement. Without loss of generality, assume that K is a triangle, the procedure for refinement of quadrilateral elements is analogous. We consider the following $N_{ref} = k + (k+1)^4$ refinement options, where $k \ge 0$ is a user input parameter:

1. Increase the polynomial degree of K by $1, 2, \ldots, k$ without spatial subdivision. This yields k refinement candidates.

2. Split K into four similar triangles K_1, K_2, K_3, K_4. Define p_0 to be the integer part of $p_K/2$. For each K_i, $1 \le i \le 4$ consider $k+1$ polynomial degrees $p_0 \le p_i \le p_0 + k$. This yields additional $(k+1)^4$ refinement candidates. In this case, edges lying on the boundary of K inherit the polynomial degree p_j of the adjacent interior element K_j. Polynomial degrees on interior edges are determined using the minimum rule.

For each of these N_{ref} options, we perform a standard H^1-projection of the reference solution u_{ref} onto the corresponding vector-valued piecewise-polynomial space on the refinement candidate. The candidate with minimum projection error relative to the number of added degrees of freedom is selected.

Note that if the technique of arbitrary-level hanging nodes is not in effect, *hp*-refinements involving spatial subdivision can be more costly than *p*-refinement candidates, since the latter never cause forced refinements. If the selection of the optimal element refinement is done locally, i.e., without taking the forced refinements into account, the *hp*-adaptive algorithm may make wrong decisions.

5. Space-time adaptive *hp*-FEM

Our space-time adaptive algorithm is based on a combination of the multi-mesh *hp*-FEM presented in Section 3 with *Rothe's method.* Rothe's method is a natural counterpart of the more widely used Method of Lines (MOL). While MOL preserves the continuity in time and discretizes the spatial variable, yielding a system of ODEs in time, Rothe's method preserves the continuity of the spatial variable while discretizing time, which leads to one or more PDEs in space per time step. The actual number of these PDEs is proportional to the order of accuracy of the time discretization method. For example, the implicit Euler method yields one PDE in space per time step. Note that Rothe's method is fully equivalent to the MOL if no adaptivity takes place. However, in contrast to MOL, Rothe's method provides an excellent opportunity to exploit spatially adaptive algorithms in the context of time-dependent problems.

Let us illustrate the space-time adaptive algorithms on a simple example of a parabolic heat transfer equation of the form

$$\frac{\partial u}{\partial t} - \Delta u = f. \tag{2}$$

Note that the methodology is not restricted to linear parabolic problems as will be shown in Section 7. When applying the backward Euler method to (2), we obtain

$$\frac{\partial u}{\partial t} \approx \frac{u^{n+1} - u^n}{\Delta t} \Rightarrow -\Delta t \Delta u^{n+1} + u^{n+1} = u^n + \Delta t f^{n+1}. \tag{3}$$

If we choose to use a second-order accurate backward differentiation formula instead, we obtain

$$\frac{\partial u}{\partial t} \approx \frac{3u^{n+2} - 4u^{n+1} + u^n}{2\Delta t} \Rightarrow -2\Delta t \Delta u^{n+2} + 3u^{n+2} = 4u^{n+1} - u^n + 2\Delta t f^{n+2}. \tag{4}$$

Note that equations (3) and (4) contain spatial derivatives only, and can therefore be solved using the spatially-adaptive algorithm which was described in Section 4.

Application of the multi-mesh hp-FEM

The approximation u^n on the right-hand side of (3) is defined on a locally refined mesh that was constructed using an adaptive process during the previous time step. The unknown approximation u^{n+1} corresponding to the end of the current time step is obtained using a new adaptive process that starts from some coarse mesh. Thus in every step of the adaptive algorithm, assembling is done over two different meshes. While the mesh for u^n remains the same during the adaptivity process, the mesh for u^{n+1} changes after each refinement step. Assembling over different meshes is done using the multi-mesh technology which was described in Section 3. At the end of the current time step, u^{n+1} is defined on a new locally refined mesh that is different from the mesh for u^n – it is finer in some regions and coarser in others. This effect can be seen as simultaneous mesh refinement and coarsening between time steps. In every time step, the adaptivity process stops when a prescribed accuracy in space is reached. The stopping criterion is related to the norm of the difference between the reference and coarse mesh solutions, as described in Section 4. Adaptive selection of the time step is a simple matter and one can use standard ODE methods to do that. The process will be illustrated on a flame propagation problem in Section 7.

In practice, the initial mesh for u^{n+1} is either chosen to be the master mesh (coarsest mesh in the multi-mesh hierarchy, see Section 3), or we take off the last one or two refinement layers from the final mesh for u^n, and start from there. The former approach yields a sequence of meshes which is more optimal from the point of view of the number of DOF used but the computation is longer. On the other hand, the latter one is faster but the final mesh for u^{n+1} may contain some local refinements which are due to u^n and not needed for u^{n+1}. As a consequence, one may need more DOF to obtain u^{n+1} on the desired level of accuracy. In our opinion, the latter approach is preferable for practical computations where CPU time matters. The former one can be used for presentation purposes, such as producing videos of meshes evolving in time, etc. The situation is similar for the second-order backward differentiation formula (4), where we need to assemble simultaneously more than three different meshes.

6. Adaptive control of time step

A simple strategy for adaptive time step control was introduced by Valli et al. [11]. Their *PID controller* is based on the relative changes of a suitable indicator variable (temperature, concentration, turbulent kinetic energy, eddy viscosity etc.) and can be summarized as follows:

1. Compute the relative changes of the chosen indicator variable u

$$e_n = \frac{||u^{n+1} - u^n||}{||u^{n+1}||}.$$

2. If they are too large ($e_n > \delta$), reject u^{n+1} and recompute it using

$$\Delta t_* = \frac{\delta}{e_n} \Delta t_n.$$

3. Adjust the time step smoothly to

$$\Delta t_{n+1} = \left(\frac{e_{n-1}}{e_n}\right)^{k_P} \left(\frac{TOL}{e_n}\right)^{k_I} \left(\frac{e_{n-1}^2}{e_n e_{n-2}}\right)^{k_D} \Delta t_n.$$

4. Limit the growth and reduction of the time step so that

$$\Delta t_{\min} \leq \Delta t_{n+1} \leq \Delta t_{\max}, \qquad m \leq \frac{\Delta t_{n+1}}{\Delta t_n} \leq M.$$

The default values of the PID parameters as proposed by Valli et al. [11] are

$$k_P = 0.075, \quad k_I = 0.175, \quad k_D = 0.01.$$

Unlike in the case of adaptive time-stepping based on the local truncation error, there is no need to compute an extra solution with a different time step. Hence, the cost of the feedback mechanism is negligible. The method has been used with favorable results by various researchers including [6].

7. Example: A flame propagation problem

We consider a freely propagating laminar flame and its response to a heat-absorbing obstacle represented by a set of cooled parallel rods with a rectangular cross-section. The domain Ω is shown in Fig. 11.

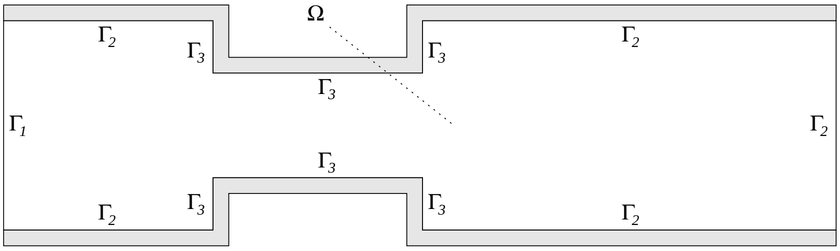


Fig. 11: *Computational domain for the flame propagation problem.*

The domain has dimensionless length $l = 60$ and width $w = 16$. The narrow part has width $w/2$, length $l/4$, and starts at $l/4$ from the left end point.

We use a low Mach number laminar flame propagation model taken from [5] which assumes that the motion of the fluid is independent from the temperature and species concentration. The flow velocity in the burner is considered to be zero. The model consists of a system of two coupled nonlinear parabolic equations

$$\begin{aligned} \frac{\partial \theta}{\partial t} - \Delta\theta &= \omega(\theta, Y) \quad \text{in } \Omega \times (0, T_0), \\ \frac{\partial Y}{\partial t} - \frac{1}{\text{Le}}\Delta Y &= -\omega(\theta, Y) \quad \text{in } \Omega \times (0, T_0) \end{aligned}$$

for the dimensionless temperature θ, $0 \leq \theta \leq 1$, and dimensionless concentration Y, $0 \leq Y \leq 1$. The dimensionless time $T_0 = 60$. The goal of the computation is accurate resolution of the nonstationary reaction rate (flame intensity) $\omega(\theta, Y)$ which is defined by the Arrhenius law

$$\omega(\theta, Y) = \frac{\beta^2}{2\text{Le}} Y \exp \frac{\beta(\theta - 1)}{1 + \alpha(\theta - 1)}. \tag{5}$$

Here, Le $= 1$ is the Lewis number (ratio of diffusivity of heat and diffusivity of mass), $\alpha = 0.8$ the gas expansion coefficient in a flow with nonconstant density, and $\beta = 10$ the dimensionless activation energy.

On the left boundary edge Γ_1, Dirichlet boundary conditions corresponding to the burnt state are prescribed, i.e.,

$$\begin{aligned} \theta &= 1 \quad \text{in } \Gamma_1 \times (0, T_0), \\ Y &= 0 \quad \text{in } \Gamma_1 \times (0, T_0). \end{aligned}$$

The remaining part Γ_2 of the boundary is assumed adiabatic with the homogeneous Neumann conditions

$$\begin{aligned} \frac{\partial \theta}{\partial \nu} &= 0 \quad \text{in } \Gamma_2 \times (0, T_0), \\ \frac{\partial Y}{\partial \nu} &= 0 \quad \text{in } \Gamma_2 \times (0, T_0). \end{aligned}$$

The absorption of heat along Γ_3 is modeled via Newton boundary conditions with a heat loss parameter $k = 0.1$,

$$\begin{aligned} \frac{\partial \theta}{\partial \nu} &= -k\theta \quad \text{in } \Gamma_3 \times (0, T_0), \\ \frac{\partial Y}{\partial \nu} &= 0 \quad \text{in } \Gamma_3 \times (0, T_0). \end{aligned}$$

As the initial condition, we prescribe the analytical solution of a one-dimensional model [5]:

$$\theta(0, x_1) = \begin{cases} 1 & \text{for } x_1 < x^*, \\ \exp(x^* - x_1) & \text{for } x_1 \geq x^* \end{cases} \tag{6}$$

and

$$Y(0, x_1) = \begin{cases} 0 & \text{for } x_1 < x^*, \\ 1 - \exp(\text{Le}(x^* - x_1)) & \text{for } x_1 \geq x^* \end{cases} \tag{7}$$

with $x^* = 9$.

Figs. 12–14 show the reaction rate $\omega(Y, \theta)$ and the underlying *hp*-FEM meshes for three different time instants t_1, t_2, t_3. The numbers inside elements indicate their polynomial degrees. Notice that very small elements on the flame front are adjacent to very large elements. This is possible due to the technique of arbitrary-level hanging nodes [7]. For problems with sharp fronts or curvilinear material interfaces, this technique saves large amount of degrees of freedom which otherwise would be needed to keep the mesh regular. Movies showing the dynamical evolution of $\omega(Y, \theta)$ along with the corresponding *hp*-FEM meshes for this problem can be found at `http://spilka.math.unr.edu/gallery/`.

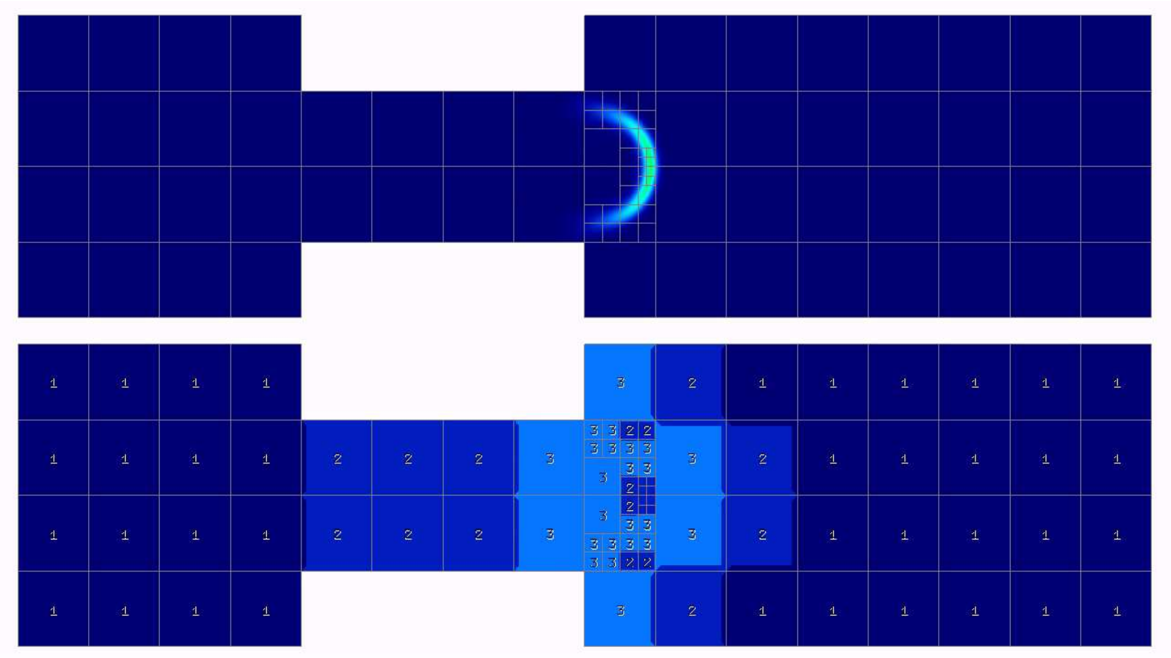


Fig. 12: *Reaction rate and higher-order finite element mesh at time t_1.*

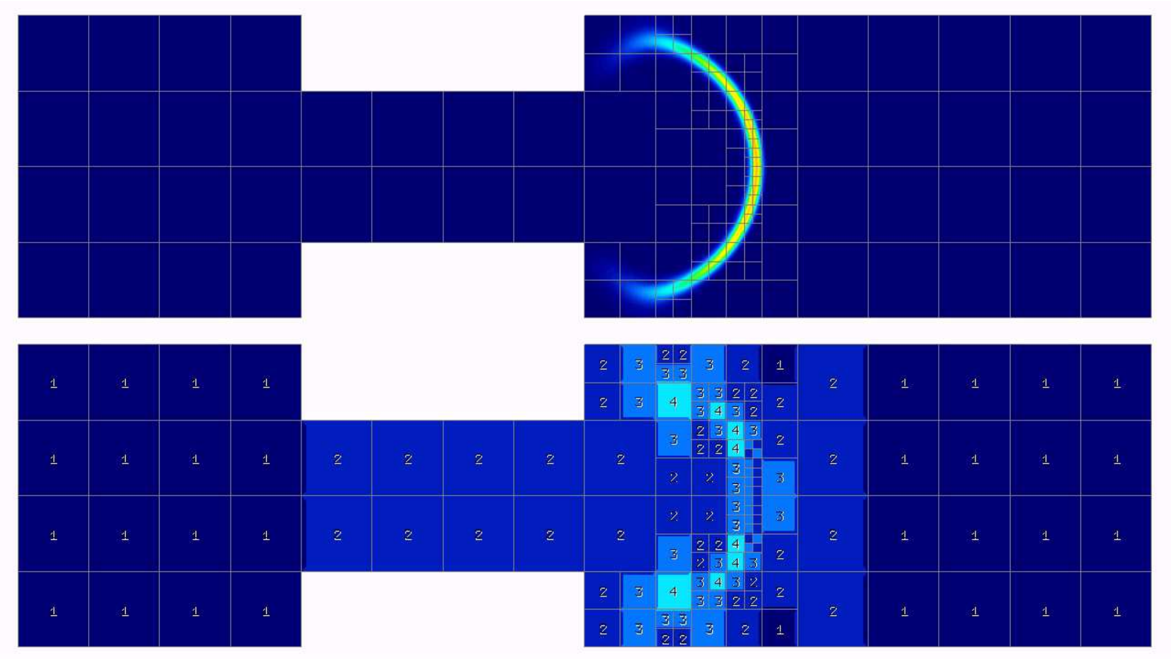


Fig. 13: *Reaction rate and higher-order finite element mesh at time t_2.*

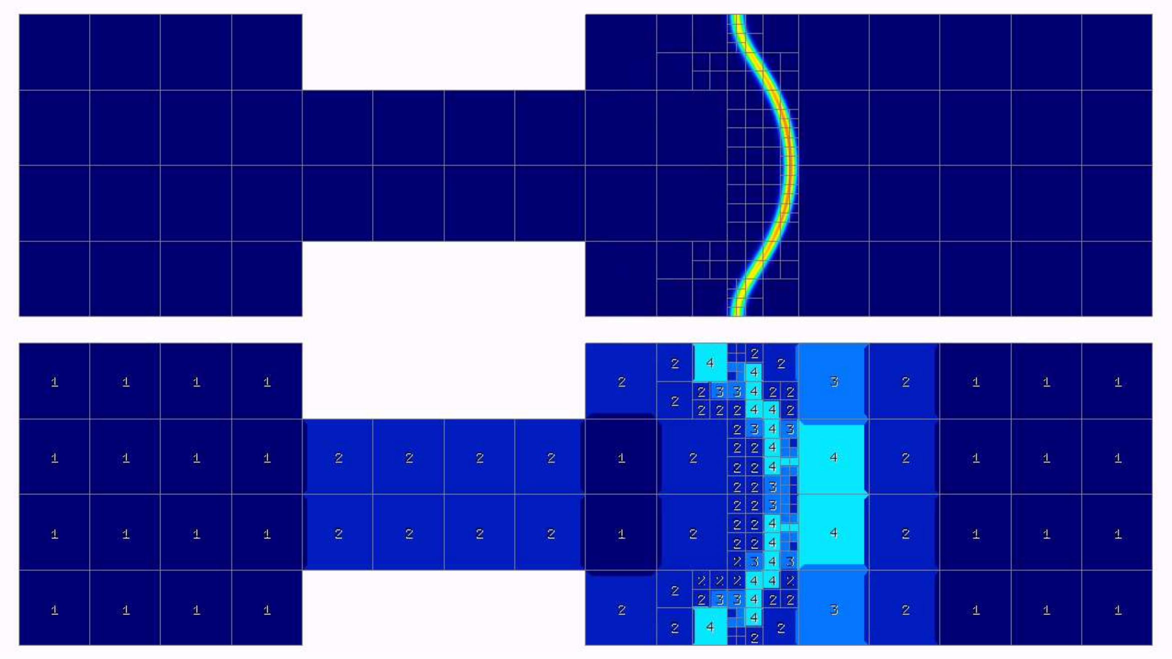


Fig. 14: *Reaction rate and higher-order finite element mesh at time t_3.*

References

[1] B. Aksoylu, S. Bond, M. Holst: *An odyssey into local refinement and multilevel preconditioning III: Implementation and numerical experiments*. SIAM J. Sci. Comput. **25** (2003), 478–498.

[2] R. Courant: *Variational methods for the solution of problems of equilibrium and vibrations*. Bull. Amer. Math. Soc. **49** (1943), 1–23.

[3] M. Paszynski, J. Kurtz, L. Demkowicz: *Parallel, fully automatic hp-adaptive 2D finite element package*. TICAM Report 04-07. The University of Texas at Austin 2004.

[4] T. Eibner, J.M. Melenk: *An adaptive strategy for hp-FEM based on testing for analyticity*. Comput. Mech. **39** (2007), 575–595.

[5] J. Lang: *Adaptive multilevel solution of nonlinear parabolic PDE systems: Theory, algorithm and applications*. Lecture Notes Comput. Sci. Engrg. 16. Berlin, Springer-Verlag, 2001.

[6] D. Kuzmin, S. Turek: *Numerical simulation of turbulent bubbly flows*. In: G.P. Celata, P. Di Marco, A. Mariani, R.K. Shah (eds.): Two-Phase Flow Modeling and Experimentation. Pisa, Edizioni ETS 2004, Vol. I, pp. 179–188.

[7] P. Solin, J. Cerveny, I. Dolezel: *Arbitrary-level hanging nodes and automatic adaptivity in the hp-FEM*. Math. Comput. Simulation **77** (2008), 117–132.

[8] P. Solin, J. Cerveny, L. Dubcova: *Adaptive multi-mesh hp-FEM for linear thermoelasticity*. Research Report No. 2007-08. Department of Mathematical Sciences, University of Texas at El Paso, to be downloaded at `http://www.math.utep.edu/preprints/2007/2007-08.pdf`. Submitted to Math. Comput. Simulation.

[9] P. Solin, L. Demkowicz: *Goal-oriented hp-adaptivity for elliptic problems*. Comput. Methods Appl. Mech. Engrg. **193** (2004), 449–468.

[10] P. Šolín, K. Segeth, I. Doležel: *Higher-order finite element methods*. Boca Raton, FL, Chapman & Hall/CRC Press, 2004.

[11] A.M.P. Valli, G.F. Carey, A.L.G.A. Coutinho: *Control strategies for timestep selection in simulation of coupled viscous flow and heat transfer*. Commun. Numer. Methods Engrg. **18** (2002), 131–139.

NUMERICAL APPROXIMATION OF THE NON-LINEAR FOURTH-ORDER BOUNDARY-VALUE PROBLEM[*]

Ivona Svobodová

Abstract

We consider functionals of a potential energy $\mathcal{P}_\psi(u)$ corresponding to *an axisymmetric boundary-value problem.* We are dealing with *a deflection of a thin annular plate* with *Neumann boundary conditions.* Various types of the subsoil of the plate are described by various types of the *nondifferentiable* nonlinear term $\psi(u)$. The aim of the paper is to find a suitable computational algorithm.

1. Introduction

Let us consider an axisymmetric annular elastic thin plate. In addition, the body is in the contact with an elastic unilateral subsoil. Consequently, the reaction of the (a-priori unknown) active part of the subsoil has to be taken into account in the mathematical model.

The fourth-order model is based on the well-known Kirchhoff theory for thin plates, for derivation see [1], [2]. This model can be formulated in terms of a variational equation. The potential $\mathcal{P}$ of this problem is quadratic.

The nonlinear term ψ, which represents the subsoil, is added to the identity in the classical formulation. The form ψ is a combination of the positive and negative parts u^+ and u^- of the deflection $u = u(r)$. The complete potential $\mathcal{P}_\psi$ is the sum of the quadratic potential $\mathcal{P}$ and the potential of the subsoil response.

The aim of the paper are to introduce and to discuss the difficulties with the discrete version of the problem obtained through the finite element method. Finally, we propose one possible way of the numerical realization of the mathematical problem.

2. Setting of the problem

The set $\big\{(r,\varphi,z)\ ;\ a \le r \le b\,, -\pi < \varphi \le \pi\,,\ -\mathsf{h}/2 \le z \le \mathsf{h}/2\big\}$ describes the plate in three dimensions *in cylindrical coordinates* (r,φ,z).

Let the elastic axisymmetric annular thin plate be represented by the domain $(a,b) \subset \mathbb{R}^+$, where $\mathbb{R}^+$ is the set of positive real numbers. The body thickness h will be involved in the equilibrium equation as a constant (see [1] for details).

In general, the operator ψ is defined for the deflection function u and it is of the form

$$\psi(u) = \sum_{i=1}^{m} k_{Ni} u^+ \chi_{A_i} - \sum_{j=1}^{n} k_{Pj} u^- \chi_{B_j} \qquad m,n \in \mathbb{N}\,, \tag{1}$$

[*] I would like to thank here the advisor of my PhD. studies doc. RNDr. Ing. J.V. Horák, CSc. for inspiring advisement.

where k_{Ni} and k_{Pj} are non-negative functions defined on (a,b) and χ_{A_i}, χ_{B_j} are characteristic function of the closed subintervals $A_i, B_j \subset (a,b)$. The functions u^+ and u^- are positive and negative part of the function u, respectively, i.e. $u^+ := \frac{1}{2}(|u|+u)$ and $u^- := \frac{1}{2}(|u|-u)$.

Classical formulation. According to the definition, the *classical solution* $u = u(r)$ satisfies the *equilibrium equation*

$$\mathsf{h}^2 D\,[r[\tfrac{1}{r}[r\cdot u']']']' + r\,\psi(u) = r\,\hat{f}, \qquad r \in (a,b) \tag{2}$$

with classical *boundary conditions* (for $r \in \{a,b\}$) of the following types

$$\text{Dirichlet conditions} \qquad u(r) = \hat{u}_r^{(0)} \qquad \text{and} \qquad u'(r) = \hat{u}_r^{(1)} \tag{3a}$$

or

$$\text{Neumann conditions} \qquad \mathcal{M}u(r) = \hat{m}_r \qquad \text{and} \qquad \mathcal{T}u(r) = \hat{t}_r \tag{3b}$$

or any reasonable combination of them. The symbol $[\,\cdot\,]'$ means $\frac{\mathrm{d}}{\mathrm{d}r}(\,\cdot\,)$ and the constant D is the combination of the elastic material coefficients namely the Young's modulus E and the Poisson's ratio μ. The function $\hat{f} = \hat{f}(r)$ describes given volume forces. The operators $\mathcal{T}$ and $\mathcal{M}$ represent *shear forces* and *bending moments* on the boundary, respectively. They are defined through the identities $\mathcal{T}(u) := -\mathsf{h}^3 D\,(r\,u''' + u'' - \frac{1}{r}u')$ and $\mathcal{M}(u) := -\mathsf{h}^3 D\,(r\,u'' + \mu\,u')$. The values $\hat{u}_r^{(0)}$, $\hat{u}_r^{(1)}$, $\hat{m}_r$, and $\hat{t}_r$ are given.

This is the linear elasticity problem. Accordingly, the components of the *small strain tensor* $\boldsymbol{\varepsilon}$ for the homogenous izotropic plate are in the forms $\varepsilon_{rr} = -z\,u''(r)$, $\varepsilon_{\varphi\varphi} = -z\,\frac{1}{r}\,u'(r)$, and $\varepsilon_{\alpha\beta} = 0$ for α, β otherwise, where $\alpha, \beta \in \{r, \varphi, z\}$ and $z \in (-\frac{\mathsf{h}}{2}, \frac{\mathsf{h}}{2})$.

The unstable *Neumann boudary conditions* with $\hat{m}_a = 0$, $\hat{m}_b = 0$, and $\hat{t}_a = 0$, $\hat{t}_b = 0$ will be considered in the following paragraphs. These conditions correspond to the so-called "free" plate.

Weak formulation. In order to get the weak form of the problem, we introduce the *finite energy function space.* This is the weighted Sobolev space which is defined as

$$H^2\big((a,b);[r,\tfrac{1}{r},r]\big) := \{v = v(r) \mid v, v'' \in L^2_r(a,b) \text{ and } v' \in L^2_{\frac{1}{r}}(a,b)\}\,. \tag{4}$$

Weighted Lebesgue spaces $L^2_{\varrho(r)}(a,b)$ are Hilbert spaces with the norm $|\cdot|_{\varrho(r)}$ induced by the inner product $(u,v)_{\varrho(r)} := \int_a^b u(r)\,v(r)\,\varrho(r)\,\mathrm{d}r$. The norm $\|\cdot\|_{[r,\frac{1}{r},r]}$ of the Hilbert space $H^2\big((a,b);[r,\frac{1}{r},r]\big)$ is related to the inner product $(u,v)_{[r,\frac{1}{r},r]} := (u,v)_r + (u',v')_{\frac{1}{r}} + (u'',v'')_r$. See [3] for the details concerning weighted spaces.

The linear space $H^2\big((a,b);[r,\frac{1}{r},r]\big)$ is not only finite energy function space but also virtual displacement space for the problem. Indeed, the Neumann boundary conditions were choosen in the previous text.

We introduce the following forms for $w, v \in V = H^2\big((a,b); [r, \frac{1}{r}, r]\big)$:

$$a_0(w,v) := D\mathsf{h}^2 \Big((w', v')_{\frac{1}{r}} + (w'', v'')_r + \mu(w'', v')_1 + \mu(w', v'')_1 \Big) \,, \tag{5a}$$

$$a_\psi(w,v) := a_0(w,v) + (\psi(w), v)_r \,, \tag{5b}$$

$$\mathcal{F}(v) := (\hat{f}, v)_r \,. \tag{5c}$$

We say that a function u in V is a *weak solution* of the problem (2), (3) whenever

$$a_\psi(u,v) = \mathcal{F}(v) \qquad \forall v \in V. \tag{6}$$

For the sake of brevity, we consider the operator ψ from (1) in the special case: $m = 1$, $k_{N_1} \equiv k_N$, and $k_{P_j} \equiv 0$ for $\forall j$, i.e.

$$\psi(u) = k_N u^+ \,, \tag{7}$$

where $k_N \in L^\infty\big((a,b)\big)$, $k_N(r) \geq \widehat{k}_N$, $\widehat{k}_N \in \mathbb{R}^+$. This form of ψ describes *the nonlinear unilateral upper subsoil* of the Winkler's type. A sufficient solvability condition of this problem is the inequality $\mathcal{F}(1) > 0$. See [2].

3. Discretization and numerical realization

We assume the discretization of the closed domain $\langle a, b\rangle$ as follows

$$a = r_1 < r_2 < \cdots < r_N < r_{N+1} = b$$

for $N \in \mathbb{N}$. The discretization parameter h is defined as $h := \max\limits_{i=1,\ldots N}(r_{i+1} - r_i)$.

Finite element method has been used for the discrete formulation of the problem. For any h, we introduce the finite dimensional space V_h which consists of piecewise cubic smooth functions

$$V_h := \{v_h \in C^1\big((a,b)\big) \ : \ v_h|_{\langle r_i, r_{i+1}\rangle} \in P_3 \ \ \forall i = 1, \ldots N\} \,. \tag{8}$$

The standard basis in V_h will be denoted by $\{\varphi_k\}_{k=1}^{2N+2}$, for details see [4]. Therefore, every function $v_h \in V_h$ is represented by

$$v_h(r) = \sum_{k=1}^{N+1} \Big(v_{2k-1}\varphi_{2k-1}(r) + v_{2k}\varphi_{2k}(r) \Big) \,, \tag{9}$$

where $v_{2k-1} = v_h(r_k)$ and $v_{2k} = v_h'(r_k)$.

We look for the *discrete solution* which is defined as the function $u_h \in V_h$ satisfying the identity

$$a^h_\psi(u_h, v_h) = \mathcal{F}^h(v_h) \qquad \forall v_h \in V_h \,, \tag{10}$$

where forms a_ψ and $\mathcal{F}$ are defined in (5). The superscript ^{h} means the usage of the numerical quadrature procedure on every subinterval $\langle r_i, r_{i+1}\rangle$ instead of the exact integration. The two points Gaussian quadrature rule was used.

The computional problem. The detailed analysis of the form a^h_ψ enables us to see, where the main problem is. The second term $([u_h]^+, v_h)_r$ has the following form

$$\Big(k_N(r)\,\Big[\sum_{k=1}^{2N+2} u_k\varphi_k(r)\Big]^+ ,\, v_h(r)\Big)^h_r =$$

$$= \Big(k_N(r)\,\frac{1}{2}\sum_{k=1}^{2N+2} u_k\varphi_k(r) + k_N(r)\,\frac{1}{2}\Big|\sum_{k=1}^{2N+2} u_k\varphi_k(r)\Big| ,\, v_h(r)\Big)^h_r .$$

The necessity of the "C^0–function" u^+ expression in terms of the base $\{\varphi_k\}_k$ is the origin of numerical difficulties in the computational algorithm. We propose a possible way how to overcome this difficulty. It is based on the following equivalent formulation of (6):

$$\begin{aligned}
a^h_\psi(u_h, v_h) &= \mathcal{F}^h(v_h) \qquad \forall v_h \in V_h \,,\\
a^h_0(u_h, v_h) + (k_N\, u^+_h, v_h)^h_r &= \mathcal{F}^h(v_h)\,,\\
a^h_0(u_h, v_h) + (k_N\, u_h, v_h)^h_r &= \mathcal{F}^h(v_h) - (k_N\, u^-_h, v_h)^h_r\,,\\
a^h_0(u_h, \varphi_k) + (k_N\, u_h, \varphi_k)^h_r &= (\widehat{f}, \varphi_k)^h_r - (k_N\, u^-_h, \varphi_k)^h_r, \qquad k = 1, \dots 2N+2\,,\\
\sum_{j=1}^{2N+2} \big(a^h_0(\varphi_j, \varphi_k) + (k_N\, \varphi_j, \varphi_k)^h_r\big) u_j &= (\widehat{f}, \varphi_k)^h_r - (k_N\, u^-_h, \varphi_k)^h_r,\\
(K + k_N M)\vec{u} &= \vec{f} - (k_N\, u^-_h, \varphi_k)^h_r\,.
\end{aligned}$$

The relation $u = u^+ - u^-$ has been used. Note that the matrix $K + k_N M$ is positive definite. More general types of the form ψ can be used in the previous procedure. Other boundary conditions do not affect the last identity. Hereafter, we could formulate the computational algorithm.

The suggested computational algorithm

1. Setting of the system stiffness matrix $K_N = K + k_N M$; standard modifying of K_N with respect to the given boundary conditions (see [4]),
2. setting of the vector $\vec{f}$ without any boundary conditions adjustments,
3. the initial choice of $\vec{u}_0$,
4. procedure in the n^{th} iteration:
   (a) setting of the vector $(k_N(u_{n-1})^-, \vec{\varphi})^h_r$, which represents the reaction of the subsoil active parts,
   (b) setting of the right side vector $\vec{f_n} = \vec{f} - (k_N(u_{n-1})^-, \vec{\varphi})_r$; standard modifying of the vector with the respect to the given boundary conditions (see [4]),

(c) solving of the system $K_N \vec{u}_n = \vec{f}_n$,

(d) the termination criterion $\dfrac{\|\vec{u}_n - \vec{u}_{n-1}\|}{\|\vec{u}_n\|} \leq tol.$

3.1. Numerical examples

We illustrate the efficiency of the suggested computational algorithm by two numerical examples. We suppose the axisymmetric annular elastic thin plate with the following characteristics. The length of inner radius is $a = 1\text{m}$ and of outer one is $b = 5\text{m}$. The plate thickness is $\mathsf{h} = 0.01\text{m}$ and the elastic constants are $E = 10^7\text{N/m}^2$ and $\mu = 0.5$. The magnitudes E and μ are choosen to get a small deformation representation of very ellastic material in order to get a better visual verification.

Example 1: The zero Dirichlet boundary conditions (3a) for all $r \in \{a, b\}$ have been prescribed. The data in equation (6) are the following: $\mathcal{F}(v) = (-10^3 \frac{1}{r}, v(r))_r$, the operator $\psi(u) = -10^7 u^- \chi_B$, where $B = \langle 2.25\text{m}, 3.75\text{m}\rangle$. Note that the choosen boundary conditions are stable, so we do not require anything in order to get the solution existence. The table of iterations and the solution diagram follow.

Example 2: The zero Neumann boundary conditions (3b) for all $r \in \{a, b\}$ have been prescribed. The given data in equation (6) are the following:

$$\mathcal{F}(v) = 5 \cdot 10^3 \big(-r^{(1)} v(r^{(1)}) - r^{(2)} v(r^{(2)}) + r^{(3)} v(r^{(3)}) \big)$$

iteration	rel. error	residual
1	5.98826×10^{-4}	1.49466×10^2
2	5.93289×10^{-4}	1.48116×10^2
⋮	⋮	⋮
449	1.00296×10^{-5}	2.56704
450	9.93897×10^{-6}	2.54386

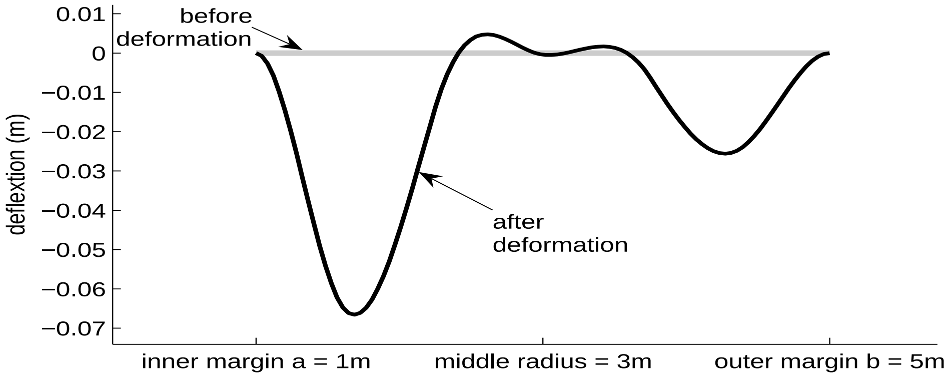


Fig. 1: *The character of resulting bending for the first problem.*

for $[r^{(i)}]_{i=1}^3 = [1.5, 2.7, 3.7]$. This $\mathcal{F}$ describes the forces concerned on rings with radii $[r^{(i)}]_i$ and with the origin on the axis of the plate symmetry. Finally, the form ψ is the following:

$$\psi(u) = 10^7 u^+ \chi_{A_1} + 10^5 u^+ \chi_{A_2} - 10^5 u^- \chi_{B_1} - 10^7 u^- \chi_{B_2},$$

where $A_1 = \langle 2.0\text{m}, 3.25\text{m}\rangle$, $A_2 = \langle 4.0\text{m}, 5.0\text{m}\rangle$, $B_1 = \langle 1.0\text{m}, 2.25\text{m}\rangle$, and the last $B_2 = \langle 3.0\text{m}, 4.0\text{m}\rangle$. It is prescribed upper and lower subsoil. Hence, the weak solution of the problem exists. The table of iterations and the solution diagram follow.

iteration	rel. error	residual
1	5.11890×10^{-2}	1.60807×10^{8}
2	4.47150×10^{-2}	1.52994×10^{8}
$\vdots$	$\vdots$	$\vdots$
1929	9.99668×10^{-6}	3.36741×10^{5}

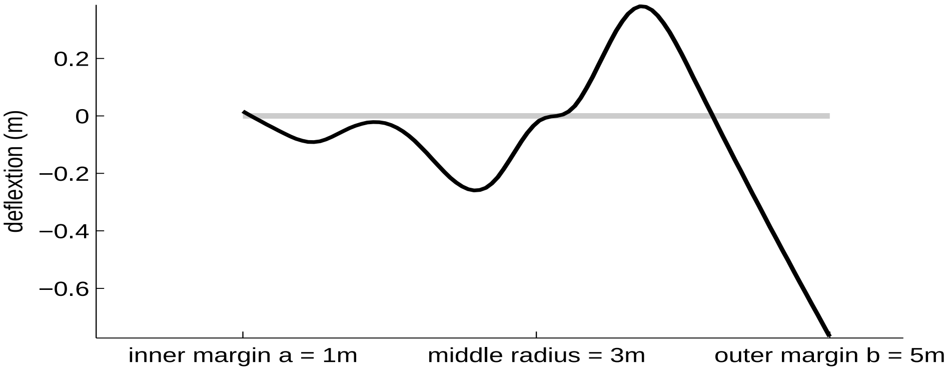


Fig. 2: *The character of resulting bending for the second problem.*

References

[1] P.G. Ciarlet: *Mathematical elasticity, Volume II: Theory of Plates.* Amsterdam, Elsevier 1997.

[2] J.V. Horák, I. Svobodová: *Modelování mezikruhové desky s podložím*, vol. of 13th Confer. Modern Mathematical Methods in Engineering, Dolní Lomná 2004.

[3] A. Kufner: *Weighted Sobolev spaces.* Leipzig, Teubner 1980.

[4] R.H. Gallagher: *Finite element analysis: fundamentals.* Englewood Cliffs, New Jersey, Prentice-Hall 1975.

[5] P. Salač: *Optimal design of an elastic circular plate on a unilateral elastic foundation.* II: Approximate Problems, Z. Angew. Math. Mech. **82** (2002) 1, 33–42.

MODELLING OF MULTICOMPONENT DIFFUSIVE PHASE TRANSFORMATION IN SOLIDS[*]

Jiří Vala

Abstract

Physical analysis of phase transformation of materials consisting from several (both substitutional and interstitial) components, coming from the Onsager extremal thermodynamic principle, leads, from the mathematical point of view, to a system of partial differential equations of evolution type, including certain integral term, with substantial differences in particular phases (α, γ) and in moving interface of finite thickness (β), in whose center the ideal liquid material behaviour can be detected. The numerical simulation of this process in MATLAB is able to explain some phenomena (e.g. the interface velocity as a function of temperature) better than known simplified models assuming the sharp interface and additional boundary and transfer conditions.

1. Introduction

The simulation of diffusional phase transformation requires to solve the coupled problem of bulk diffusion and interface migration. Most models pay attention especially to binary (two-component) alloys with substitutional components – cf. [1] and [6]. Usually the interface is assumed to be sharp (in other word: its thickness is supposed to be negligible), thus some artificial boundary and transfer conditions have to be applied at the interface, as e.g. the ortho- or para-equilibrium contact conditions for a multi-component model in [10]. However, a real migrating interface of finite thickness h may drag segregated impurity atoms forming concentration profiles across the interface. Such a local diffusion process reduces the migration velocity v due to the Gibbs energy dissipated by this process; this decelerating effect is known as solute drag. In this paper we shall consider alloys with a finite number (at least two) of components. In [11], following some ideas of [8], coming from the Onsager extremal thermodynamic principle (derived originally in [7], for more details and various generalizations see [5]), the steady-state diffusion of solute across the interface is driven by the difference of chemical potentials $\mu(c^\star)$, corresponding to the vector $c^\star$ of mole fractions (as concentrations characteristics) related to all q substitutional components, e.g. $c^\star = (c_1, \dots, c_q)$; since $c_1 + \dots + c_q = 1$, it is useful to introduce $c = (c_1, \dots, c_{q-1})$, too. As discussed in [2], at least for the steady-state case this approach gives identical results with the solute drag formula proposed in [4]. In [15] the same approach is generalized to admit the evolution of molar fractions in time

[*] This work was supported by grant No. IAA200410601 of the Academy of Sciences of the Czech Republic.

and the presence of r interstitial components; consequently $c^\star = (c_1, \ldots, c_q, \ldots, c_{q+r})$ and $c = (c_1, \ldots, c_{q-1}, c_{q+1}, \ldots, c_{q+r})$. Nevertheless, [15] shows only one practical example of such evolution near the initial time; the algorithm suggested in this paper handles also slow long-time redistributions.

All material characteristics (chemical potentials, diffusion factors, interface mobility) should correspond to a material structure where usual lengths are in micrometers; moreover, the usual interface thickness can be 10^{-10} m. Consequently, it is not easy to identify such characteristics in the laboratory. This is the first reason for the one-dimensional formulation of the problem in this paper, the second one is the requirement of simple, transparent and reader-friendly notations; some useful generalizations will be sketched in concluding remarks. For certain material sample of length H, in addition to $c^\star$ and c it is useful to introduce vectors of diffusive fluxes $j^\star = (j_1, \ldots, j_q, \ldots, j_{q+r})$ where $j_1 + \ldots + j_q = 0$. We shall study the redistribution of c and j in an arbitrary positive time t in a closed system with $j(.) = 0$ at the boundary (consisting of two points, whose distance is H). Such system requires no additional boundary conditions; we need only to know all initial values c for $t = 0$. If x refers to the standard Cartesian coordinate system and v is positive for the interface motion from the left to the right we can localize the interface (for x) into the interval $\langle 0, h\rangle$ and the exterior boundary of a sample into two points

$$x_L(t) = x_L(0) - \int_0^t v(\varsigma)\,\mathrm{d}\varsigma \qquad x_R(t) = x_R(0) - \int_0^t v(\varsigma)\,\mathrm{d}\varsigma\,;$$

clearly $x_R(t) - x_L(t) = H$ and $j(x_L) = j(x_R) = 0$ for any t. Finally we have the first phase, denoted in all following considerations by α, for $x < 0$, separated from the second phase, denoted by γ, for $x > h$, by the phase interface, denoted formally by β, for $0 \le x \le h$.

If the dot symbol denotes the partial derivative with respect to t and the prime symbol the partial derivative with respect to x then we are able to calculate the total time derivative of a variable u as $\mathrm{d}u/\mathrm{d}t = \dot{u} - vu'$. Namely the mass conservation law for the constant molar volume Ω reads

$$\mathrm{d}c^\star/\mathrm{d}t + \Omega j^{\star\prime} = \dot{c}^\star - vc^{\star\prime} + \Omega j^{\star\prime} = 0\,; \tag{1}$$

the integration of (1) from x_L to x_R, making use of the new notation

$$C^\star(x,t) = \int_0^x c^\star(\xi,t)\mathrm{d}\xi\,,$$

then yields

$$\dot{C}^\star(x_R) - \dot{C}^\star(x_L) - v(c^\star(x_R) - c^\star(x_L)) = 0\,. \tag{2}$$

In the second section of this paper we shall sketch the physical background of the diffusive and massive phase transformation, applying the Onsager arguments. In the third section the derived system of equations for an unknown field c (because both j and v can be identified with certain functions of c) will be analyzed with the aim to construct an effective algorithm searching for its approximate solution. The fourth section will demonstrate new numerical results for a special Fe-rich 3-component Fe-Cr-Ni alloy. The last section is reserved for concluding remarks and possible generalizations in several directions.

2. Physical background

Let us consider a closed system with the simple geometry, introduced above. Every index in all following relations can be understood as a sum index in sense of the Einstein rule; only an underlined index prohibits summation. Let i be an arbitrary index from $1, \ldots, q+r$ and f an arbitrary index from $\{\alpha, \beta, \gamma\}$ The chemical potential $\mu_i(x, c^\star)$ can be evaluated at every point of the sample as

$$\mu_i(x, c^\star) = w^f(x)\mu_i^f(c^\star) \,, \tag{3}$$

making use of some reasonably continuous weight functions $w^f(x)$, having the properties

$$\begin{array}{lll} w^\alpha(x) = 1\,, \quad w^\gamma(x) = 0 & \text{if} & x_L < x < h/2\,, \\ w^\alpha(x) = 0\,, \quad w^\gamma(x) = 1 & \text{if} & h/2 < x < x_R\,, \\ w^\beta(x) = 1 - w^\alpha(x) - w^\gamma(x) & \text{if} & x_L < x < x_R\,. \end{array}$$

The well-known Gibbs-Duhem relation, formulated e.g. in [9], yields

$$c_i \frac{\mathrm{d}\mu_i^f}{\mathrm{d}t} = 0\,, \qquad c_i \mu_i^{f\prime} = 0\,. \tag{4}$$

The total Gibbs energy of the system is given by

$$G = \frac{1}{\Omega} \int_{x_L}^{x_R} c_i \mu_i \,\mathrm{d}x\,.$$

Its time derivative can be expressed as

$$\frac{\mathrm{d}G}{\mathrm{d}t} = \frac{1}{\Omega} \int_{x_L}^{x_R} \left(\frac{\mathrm{d}c_i}{\mathrm{d}t} \mu_i + c_i \frac{\mathrm{d}\mu_i}{\mathrm{d}t} \right) \mathrm{d}x\,.$$

Inserting μ_i into the second additive term from (3) and integrating by parts, we obtain

$$\mathrm{d}G/\mathrm{d}t = \frac{1}{\Omega} \int_{x_L}^{x_R} \left(\mathrm{d}c_i/\mathrm{d}t\, \mu_i + c_i w^f \mathrm{d}\mu_i^f/\mathrm{d}t - v c_i (w^f \mu_i^f)' + v c_i w^f \mu_i^{f\prime} \right) \mathrm{d}x\,.$$

By (4) the second and fourth additive terms vanish, moreover $w^f = 1$ for any $f \in \{\alpha, \gamma\}$ if $x \le 0$ or $x \ge h$, thus we come to the result

$$\mathrm{d}G/\mathrm{d}t = \frac{1}{\Omega} \int_{x_L}^{x_R} \mathrm{d}c_i/\mathrm{d}t\, \mu_i \,\mathrm{d}x - \frac{v}{\Omega} \int_0^h c_i \mu_i' \,\mathrm{d}x\,.$$

By (1), integrating by parts for a closed system again, we have finally

$$\mathrm{d}G/\mathrm{d}t = \int_{x_L}^{x_R} j_i \mu_i' \,\mathrm{d}x - \frac{v}{\Omega}\int_0^h c_i \mu_i' \,\mathrm{d}x\,. \tag{5}$$

The rate of dissipation Q of the total Gibbs energy can be evaluated by [13] in the form

$$Q = \int_{x_L}^{x_R} \frac{j_i^2}{A_i}\,\mathrm{d}x + \frac{v^2}{M} \tag{6}$$

where

$$A_i = \frac{c_i D_i}{\Omega R T}\,, \tag{7}$$

D_i is the tracer diffusion coefficient, R is the gas constant, T is the absolute temperature and M is the interface mobility.

The kinetics of our system corresponds to the variation

$$\delta\left(\mathrm{d}G/\mathrm{d}t + \frac{Q}{2}\right)(j^\star, v) = 0$$

with respect to the above mentioned constraint for substitutional components: if k is an index similar to i, but from $\{1,\ldots,q\}$ only, then we can write

$$\delta\left(\mathrm{d}G/\mathrm{d}t + \frac{Q}{2} + \lambda\delta_{kk}\right)(j^\star, v, \lambda) = 0$$

with certain Lagrange multiplier λ, but without any additional constraints; all δ with a couple of indices, here and everywhere later, refer to Kronecker symbols. To support the brief notation, let us introduce the component type factor a_i, equal to 1 for $i \le q$, zero otherwise. Performing the variation, step by step, for $j_1,\ldots,j_q,\ldots,j_{q+r}$, v and λ, we obtain

$$\int_{x_L}^{x_R}\left(\tilde{j}_i \mu_i' + \frac{\tilde{j}_i j_i}{A_i} + \tilde{j}_i a_i \lambda\right)\mathrm{d}x = 0$$

for every $\tilde{j}_i$,

$$-\frac{\tilde{v}}{\Omega}\int_0^h c_i \mu_i' \,\mathrm{d}x + \frac{\tilde{v}\,v}{M} = 0$$

for every $\tilde{v}$ and formally also $\tilde{\lambda} j_i a_i = 0$ for every $\tilde{\lambda}$. In this way we come to $q + r$ differential equations

$$\mu_i' + \frac{j_i}{A_i} + \lambda a_i = 0 \tag{8}$$

with a parameter λ for all both substitutional and interstitial components and to one integral equation

$$v = \frac{\Omega}{M}\int_0^h c_i \mu_i' \,\mathrm{d}x \tag{9}$$

for the interface velocity. It is not difficult to remove λ from (8): multiplying (8) by A_i and summing results with non-zero a_i, we have

$$\delta_{kk} A_k \lambda = -A_k \mu'_k - \delta_{kk} j_k = -A_k \mu'_k$$

and consequently

$$\lambda = -\frac{A_k \mu'_k}{\delta_{ll} A_l}$$

where l is a sum index with the same properties as k. This enables us to evaluate all fluxes as

$$j_i = -A_i \left(\mu'_i - a_i \frac{A_k \mu'_k}{\delta_{ll} A_l} \right) ; \tag{10}$$

let us notice that we have

$$\delta_{kk} j_k = \delta_{ii} a_i j_i = 0$$

and consequently the system of $q + r$ equations (8) can be reduced to the system of $q - 1 + r$ equations.

For practical calculations we need to express j_i by means of (10), (7) and (3); it is useful to introduce the decomposition

$$\mu^f_i(c^\star) = \mu^f_{0i} + RT \ln c_i + \varphi^f_i(c^\star) \tag{11}$$

where μ^f_{0i} are constants for a given temperature T and φ^f_i are certain functions of $c^\star$ (usually not dominant, but non-negligible and formally complicated). Inserting this decomposition together with (10) into (8), after rather long calculations, performed in [15], p. 75, we are able to evaluate

$$N \Omega j = -B c' - K c \tag{12}$$

where B, K (functions of c) and N (dependent on x only) are square matrices of order $q-1+r$, B full one, K and N diagonal ones, of the following material characteristics:

$$B_{mn} = \delta_{mn} + \frac{c_m\,(\zeta_q - \zeta_n)}{\eta} + \frac{\overline{\varphi}_{mn} - \overline{\varphi}_{mn}}{RT} , \qquad K_{\underline{mm}} = \frac{\overline{\mu}_m}{RT} , \qquad N_{\underline{mm}} = \frac{1}{\zeta_m D} ;$$

here m or n refer (instead of i) to a sum index from $\{1, \ldots, r-1, q+1, \ldots, q+r\}$ and moreover $\zeta_i = D_i/D$ for some (non-zero) reference value D of the tracer diffusion coefficient, $\eta = \zeta_i c_i$ and

$$\overline{\varphi}_{mn} = \widehat{\varphi}_{mn} - a_m a_n \frac{\zeta_l c_l}{\eta} \widehat{\varphi}_{ln} , \qquad \overline{\mu}_m = \widehat{\mu}_m - a_m \frac{\zeta_l c_l}{\eta} \widehat{\mu}_l$$

with $\widehat{\mu}_m = w^{f\prime} \left(\mu^f_{0m} + \varphi^f_m \right)$ and $\widehat{\varphi}_{mn} = w^f \partial \varphi^f_m / \partial c_n$. For any variable u let us introduce the simplified notation $u^\diamond = u(0)$, $u^L = u(x_L)$ and $u^R = u(x_R)$. Then, integrating (1) from 0 to x, omitting c_q and j_q, we receive

$$\dot{C} - v(c - c^\diamond) + \Omega(j - j^\diamond) = 0 . \tag{13}$$

In particular, subtracting (13) with $x = x_R$ and $x = x_L$,

$$\dot{C}^R - v(c^R - c^\diamond) - \Omega j^\diamond = 0\,, \qquad \dot{C}^L - v(c^L - c^\diamond) - \Omega j^\diamond = 0\,, \tag{14}$$

we have only a formal modification of (2)

$$\dot{C}^R - \dot{C}^L - v(c^R - c^L) = 0\,, \tag{15}$$

but inserting (12) into (13), we obtain a new result

$$-N\dot{C} + Bc' + (K + vN)c = vNc^\diamond - N\Omega j^\diamond\,. \tag{16}$$

3. Mathematical formulation and computational algorithms

We suppose that all values of molar fractions c are prescribed for $t = 0$. For their initial time derivatives we usually have no better information than $\dot{c} = 0$, thus also $\dot{C} = 0$ and $j^\diamond = 0$ from (14). Let us also notice that C can be computed as integrals of $c - c^a$ instead of c, using arbitrary reference constant admissible molar fractions c^a. Our problem is to find c from (16) with v inserted from (9). For a priori known B, K and v and also x_L, to solve (16) numerically (to construct a sequence of approximate solutions, whose limit could be expected to coincide with the solution of (16)) means to discretize (16) in time; this can be done by means of the Euler implicit scheme

$$Bc' + (K + vN)c - N\frac{C}{\tau} = vNc^\diamond - N\Omega j^\diamond - N\frac{C^\times}{\tau} \tag{17}$$

where τ denotes the time step and all variables are evaluated in time t, except $C^\times = C(t - \tau)$. (The application of more advanced schemes of discretization in time instead of (17) is possible, but leads to rather complicated forms of following equations.) To obtain a system of linear algebraic equations, we have to apply the discretization in $\langle x_L, x_R \rangle$, too. In practice only some estimates of all material characteristics B and K, of the interface velocity v and of the boundary position x_L (then clearly $x_R = x_L + H$) are available, usually those from the previous time step, thus (17) forms a basis for an iteration procedure where v can be recalculated from (9) using the Simpson rule; also the evaluation of C needs some numerical integration.

Let us consider a sufficiently large fixed interval $\mathcal{I}$, containing $\langle x_L, x_R \rangle$, decomposed to a finite number σ of subintervals $\langle x_{s-1}, x_s \rangle$, using $\sigma + 1$ nodes $x_0, x_1, \ldots, x_\sigma$. Then we can write (17) in the form

$$\overline{B}^s \frac{c^s}{\Delta_s} + \left(\overline{K}^s + v\overline{N}^s\right) \frac{c^s}{2} - \overline{N}^s \frac{\Delta_s c^s}{2\tau} \tag{18}$$

$$= \overline{B}^s \frac{c^{s-1}}{\Delta_s} - \left(\overline{K}^s + v\overline{N}^s\right) \frac{c^{s-1}}{2} + v\overline{N}^s c^\diamond - \overline{N}^s \Omega j^\diamond - \overline{N}^s \frac{2(C^{\times s} - C^{s-1}) - \Delta_s c^{s-1}}{2\tau}$$

where an integer s refers to the s-th node in $\mathcal{I}$ (values at x_L and x_R, in general not identical with any x_s, are interpolated), $\Delta_s = x_s - x_{s-1}$ and overlined s-indexed symbols refer to averaged values on $\langle x_{s-1}, x_s \rangle$. Let us notice that $c^\diamond$ coincides always with some element of the set $\{c^0, c^1, \dots, c^\sigma\}$. Our aim is to study the long-time behaviour of a system, thus it is useful to take sufficiently small Δ_s in comparison with τ. We would like to solve $c^0, c^1, \dots, c^s, \dots$ effectively, step by step, but this is impossible because of unknown values $c^\diamond$ and $j^\diamond$ (the system of linear algebraic equation is not triangular). However, we shall show that this difficulty can be overcome: the main idea will be demonstrated on (17), its formal implementation into (18) will be left to the reader. Let $c^{\diamond e}$ be some estimate of $c^\diamond$ (from the preceding iteration, if not available yet then from the previous time step). Let us consider $c^\diamond_m = \xi^I_{\underline{m}} c^{\diamond e}_m$ and $j^\diamond_m = \xi^{II}_{\underline{m}} v c^{\diamond e}_m$ for some positive real $2(q-1+r)$ factors ξ^I_m and ξ^{II}_m. We are allowed to seek for molar fractions c in the form $c = c^\diamond + \tilde{c}$ where $\tilde{c}_m = \tilde{c}^O_m + \xi^I_{\underline{m}} \tilde{c}^I_m + \xi^{II}_{\underline{m}} \tilde{c}^{II}_m$. Then (17) degenerates to

$$B\tilde{c}' + K\tilde{c} + vN\tilde{c} - N\frac{\tilde{C}}{\tau} = F^O + \xi_I F^I + \xi_{II} F^{II}$$

with $\tilde{C}$ integrated from $\tilde{c}$ (for comparison: C is integrated from $c - c^a$) and with

$$F^O = N\frac{C^\times - c^a x}{\tau}, \qquad F^I = \left(N\frac{x}{\tau} - K\right) c^{\diamond e}, \qquad F^{II} = -N\Omega v c^{\diamond e}.$$

Thus we are able to solve all $\tilde{c}^O$, $\tilde{c}^I$ and $\tilde{c}^{II}$ separately (which is very simple) and just at the end to calculate ξ^I and ξ^{II} $(q-1+r)$-times from the system of two linear algebraic equations

$$\begin{bmatrix} \tilde{C}^{LI}_m/\tau - v\tilde{c}^{LI}_m + c^{\diamond e}_m x_L/\tau & \tilde{C}^{LII}_m/\tau - v\tilde{c}^{LII}_m \\ \tilde{C}^{RI}_m/\tau - v\tilde{c}^{RI}_m + c^{\diamond e}_m x_R/\tau & \tilde{C}^{RII}_m/\tau - v\tilde{c}^{RII}_m \end{bmatrix} \cdot \begin{bmatrix} \xi^I_m \\ \xi^{II}_m \end{bmatrix} =$$

$$\begin{bmatrix} -\tilde{C}^{LO}_m/\tau + v\tilde{c}^{LO}_m + C^{L\times}_m/\tau + c^a_m x_L/\tau \\ -\tilde{C}^{RO}_m/\tau + v\tilde{c}^{RO}_m + C^{R\times}_m/\tau + c^a_m x_R/\tau \end{bmatrix}.$$

The above sketched algorithm have been tested with the support of standard MATLAB environment. No special packages were needed, except the toolbox *symbolic*, referring to the core of MAPLE. Typically the material description for one calculation, generated for a fixed temperature of phase transformation, contain thousands of instructions; the most complicated are the expressions for chemical potentials in particular phases, especially their nonlinear parts, occurring as the last additive terms in the decomposition (11). Open questions are both in the theory of existence of solutions c and v and in the convergence of all algorithms. Some modifications of such algorithms, namely the application of higher-order Hermite splines (in the approximation of both unknown molar fractions and material characteristics) and of the spectral analysis in the phases α and γ, far from the interface, where nearly exponential distributions of molar fractions can be expected, are discussed in [15], p. 79.

The rather complicated non-local integro-differential character of the problem does not admit the application of some transparent homogenization technique. Moreover, the identification of material characteristics, included in B, K and N, is very complicated; consequently it is not clear how to formulate and study the inverse problems (formulation of all chemical potentials and diffusion factors, generating B, K and N, and setting the interface mobility M, the interface thickness h, etc., from experimental results for $c(x,t)$ at some set of fixed guaranteed temperatures) correctly. Most material characteristics can be classified as a semi-empirical ones, based both on some physical considerations and on the extensive experimental study, unfortunately not covering all mole fractions of particular components between 0 and 1. Moreover no physical barrier is incorporated into our system of equations to prevent negative or other non-realistic mole fractions; very different quantitative values of some characteristics, namely of the interface mobility M, can be found in the literature, too.

4. Numerical example

The numerical example, presented in this paper, makes use of the same source of quantitative material data as [14] from the Montanuniversität Leoben (Austria) and from the Institute of Physics of Materials of the Czech Academy of Sciences in Brno. We have the purely substitutional three-component Fe-Cr-Ni system; in our notation $q = 3$ and $r = 0$, moreover Fe will be dominant.

The tracer diffusion coefficients can be interpolated using the formula

$$\ln D_k = w^f \ln D_k^f \ ,$$

thus it is sufficient to set nine values D_k^f. In general we have

$$D_k^f = D_{k0}^f \exp\left(-\frac{E^f}{RT}\right) \ , \qquad M = M_0 \exp\left(-\frac{E^\star}{RT}\right) \ .$$

The applied constants are for Cr (corresponding to $k = 1$) $D_{10}^\alpha = 0.00032\ \mathrm{m}^2\,\mathrm{s}^{-2}$, $D_{10}^\beta = 0.00022\ \mathrm{m}^2\,\mathrm{s}^{-2}$, $D_{10}^\gamma = 0.00035\ \mathrm{m}^2\,\mathrm{s}^{-2}$, for Ni ($k = 2$) $D_{20}^\alpha = 0.000048\ \mathrm{m}^2\,\mathrm{s}^{-2}$, $D_{20}^\beta = 0.000022\ \mathrm{m}^2\,\mathrm{s}^{-2}$, $D_{20}^\gamma = 0.000035\ \mathrm{m}^2\,\mathrm{s}^{-2}$, for Fe ($k = 3$) $D_{30}^\alpha = 0.00016\ \mathrm{m}^2\,\mathrm{s}^{-2}$, $D_{30}^\beta = 0.00011\ \mathrm{m}^2\,\mathrm{s}^{-2}$, $D_{30}^\gamma = 0.00007\ \mathrm{m}^2\,\mathrm{s}^{-2}$, and for all components $E^\alpha = 240000\ \mathrm{J\,mol}^{-1}$, $E^\beta = 155000\ \mathrm{J\,mol}^{-1}$, $E^\gamma = 286000\ \mathrm{J\,mol}^{-1}$, $E^\star = 140000\ \mathrm{J\,mol}^{-1}$; it remains to set only $M_0 = 0.00041\ \mathrm{m}^2\ \mathrm{s\ kg}^{-1}$.

Three figures show the time-variable distributions of c_1 and c_2. The interface thickness is $h = 5 \cdot 10^{-10}$ m, the sample length $H = 10^{-4}$ m. From the originally constant mole fractions $c_1 = 0.001$ and $c_2 = 0.019$ (consequently $c_3 = 0.980$) in all phases due to the phase transformation driven by changes in chemical potentials, the time development from $t = 0$ to $t = 70000$ s leads to qualitative new distributions. All figures make use of the same computational results at various scales: Fig. 1 shows Cr and Ni (strongly nonlinear) mole fractions inside the interface, Fig. 2 documents different behaviour of Cr and Ni components near the interface, Fig. 3 demonstrates quasi-constant distributions with seemingly sharp interface, whose physically transparent macroscopic description is not available. The numbers of particular curves from $\{1, \ldots, 7\}$ (quite omitted in Fig. 1, somewhere hardly recognizable even for larger scales) refer to $t \in \{10000, \ldots, 70000\}$ s.

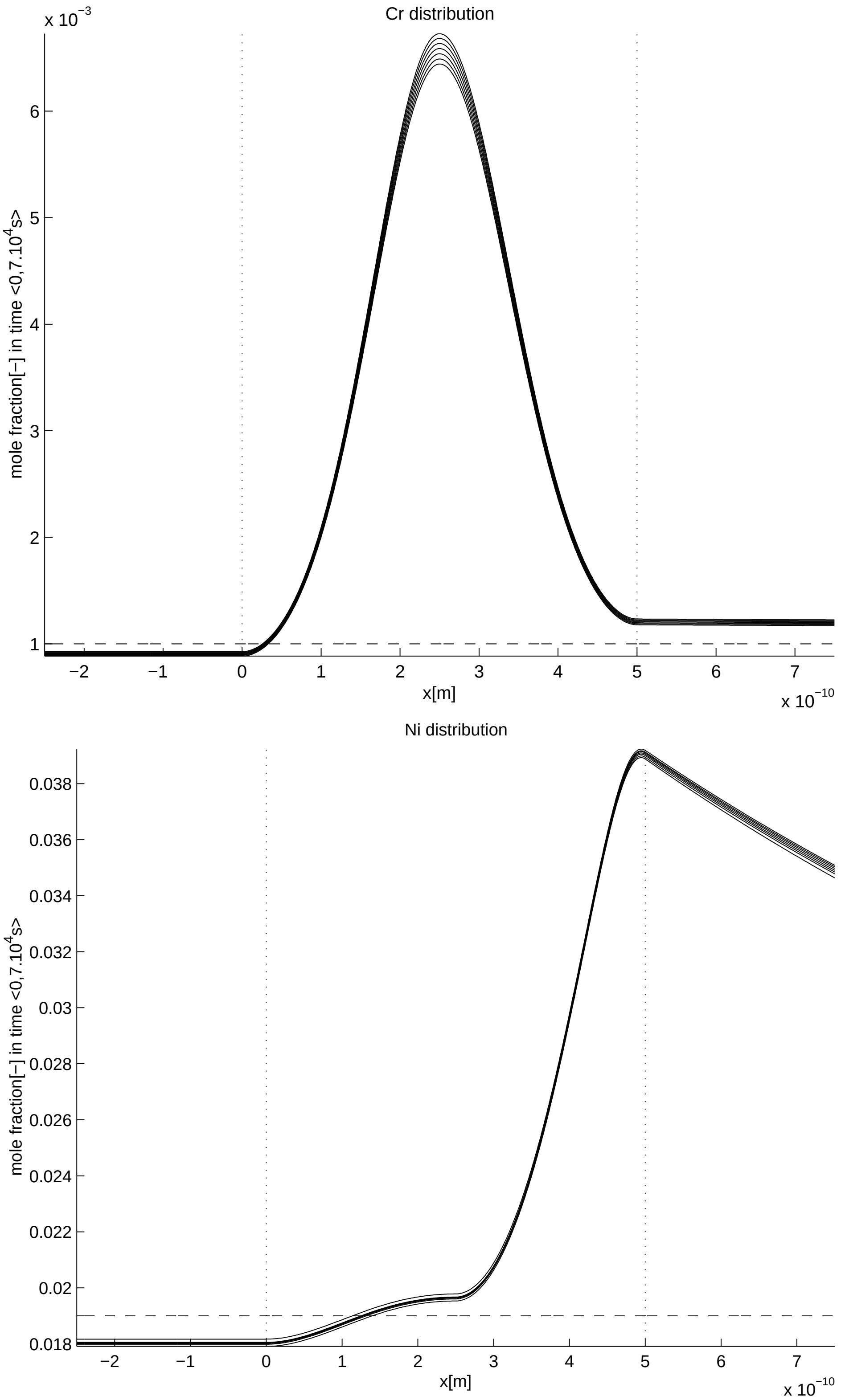


Fig. 1: *Nano-scale distribution of Cr and Ni inside the interface.*

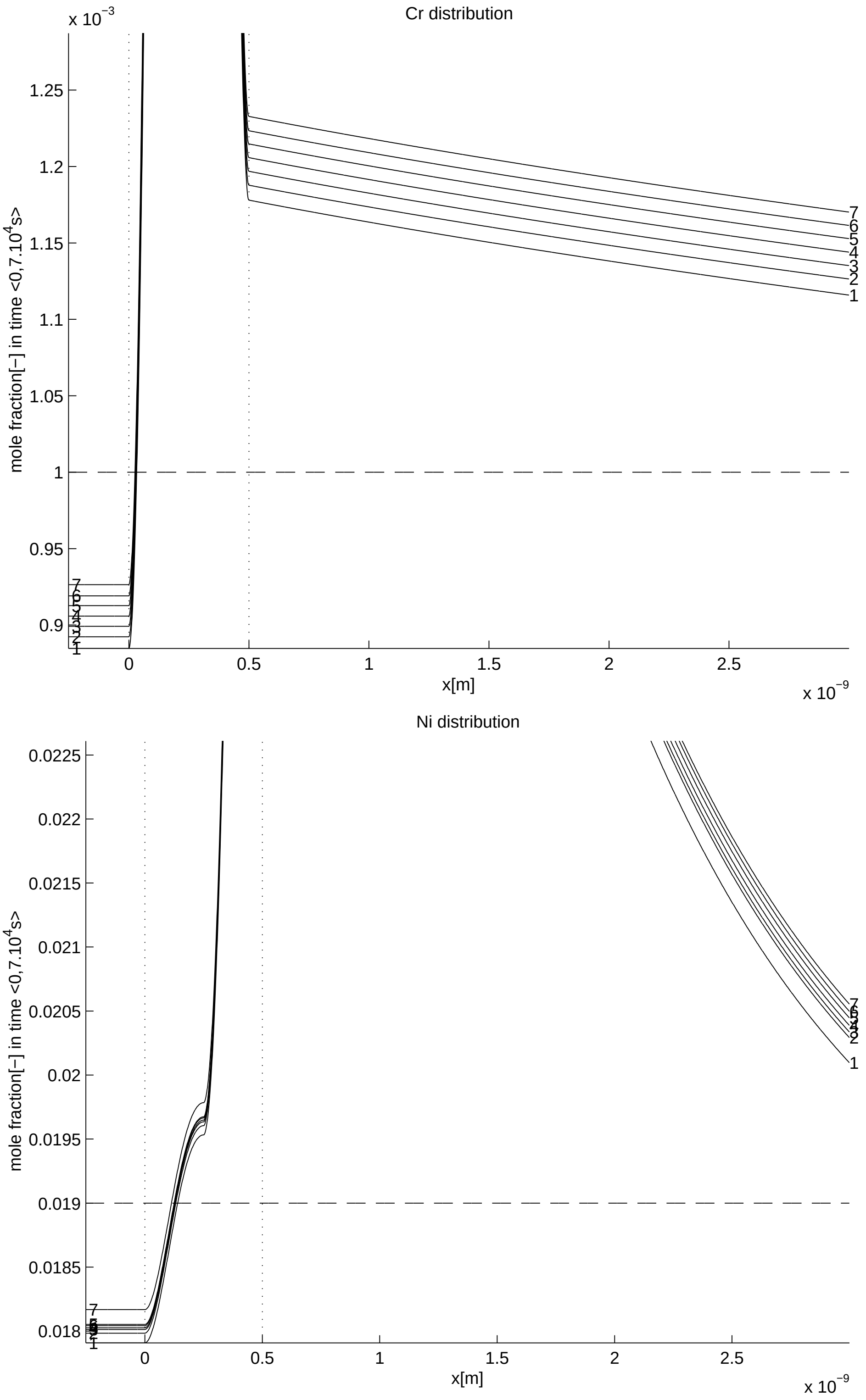


Fig. 2: *Meso-scale distribution of Cr and Ni near the interface.*

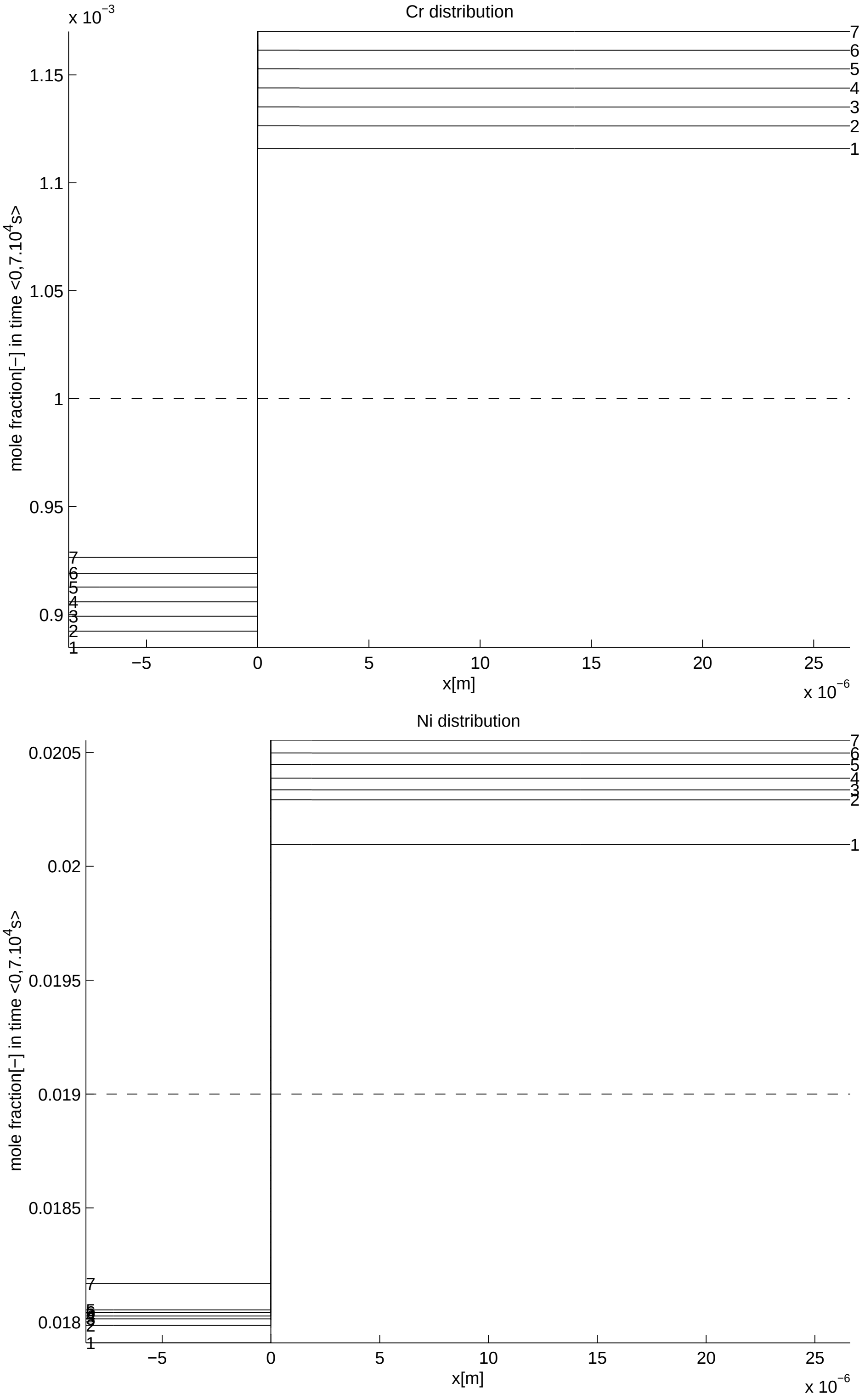


Fig. 3: *Larger-scale observable distribution of Cr and Ni.*

5. Conclusions and generalizations

In our numerical example we have seen the typical non-stationary behaviour of one special Fe-rich Fe-Cr-Ni substitutional system, described by (16) and (9), with respect to the physical limitations (a finite closed system, interface of constant thickness, substitutional components). However, the created software has been tested for more classes of problems of practical importance. In [14] the stationary solver was applied to the Fe-rich Fe-Cr-Ni substitutional system with various types of chemical potentials and values of material characteristics, which may be rather uncertain in practice, namely in case of the interface mobility and thickness; further numerical simulations has been done also for the similar system with the interstitial C-component and for the binary Al-Mg system yet. For every fixed interface thickness h the numerical simulations show that the interface velocity v decreases with the increasing temperature T; finally the phase transformation stops at certain critical temperature. This critical temperature increases with the increasing interface thickness h; the limit case $h \to 0$ returns the (less realistic) results for an idealized sharp interface. The simulation of the massive $\gamma \to \alpha$ transformation shows that the existence of the solute drag in the interface influences the contact conditions at the interface allowing the massive transformation to occur also in the two-phase region. By choosing α and γ as identical phases and by imposing fluxes to the interface (grain boundary), diffusion induced grain boundary motion was simulated. The interface and grain boundary Gibbs energy were calculated; their realistic values support the responsibility of the model.

Both theoretical and experimental works yield that the diffusion in multi-component alloys can be characterized by three attributes: a) the vacancy mechanism for "slowly" diffusing substitutional components, b) the existence of certain sources or sinks of vacancies, c) the "quick" motion of atoms of interstitial components. In our description only the attributes a) and c) have been incorporated properly; the attribute b) should be involved using the detailed analysis [11], referring to [3]. Another important research direction is to admit more complicated thermal processes. This forces (from the point of view of the Onsager relation) coupling of various fluxes, namely the particle flux due to a temperature gradient (Soret effect) and the transport of heat due to a concentration gradient (Dufour effect); more information is contained in [12]. Still another direction of possible generalizations leads to two- or three-dimensional simulations. Up to now, such computations suffer from the lack of reasonable material data; nevertheless, an introductory discussion is included in [15], p. 85.

References

[1] D. Fan, S.P. Chen, L.-Q. Chen: *Computer simulation of grain growth kinetics with solute drag.* Journal of Material Research **14** (1999), 1113–1123.

[2] E. Gamsjäger, J. Svoboda, F.-D. Fischer: *Solute drag or diffusion process in a migrating thick interface.* Philosophical Magazine Letters **88** (2008), 415-420.

[3] M.A. Grinfeld: *Thermodynamic methods in the theory of heterogenous systems.* Longman, New York, 1991.

[4] M. Hillert: *Solute drag in grain boundary migration and phase transformation.* Acta Materialia **52** (2004), 5289–5293.

[5] U.R. Kattner: *Thermodynamic modelling of multicomponent phase equilibria.* JOM **49** (1997) 14–19.

[6] U.F. Mayer, G. Simonett: *Classical solutions for diffusion-induced grain boundary motion.* Journal of Mathematical Analysis and Applications **234** (1999), 660–674.

[7] L. Onsager: *Reciprocal relations in irreversible processes.* Physical Review **37** (1931), 405–426.

[8] J. Odqvist, B. Sundman, J. Ågren: *A general method for calculating deviation from local equilibrium at phase interfaces.* Acta Materialia **51** (2003), 1035–1043.

[9] M. Sacchetti: *The general form of the Gibbs-Duhem equation for multiphase/multicomponent systems and its application to solid-state activity measurements.* Journal of Chemical Education **78** (2001), 260–262.

[10] A. Schneider, G. Inden: *Fundamentals and basic methods for microstructure simulation above the atomic scale.* In: D. Raabe, F. Roters, F. Barlat, L.-Q. Chen (Eds.), Continuum Scale Simulation of Engineering Materials: Fundamentals – Microstructures – Process Applications. Wiley-VCH, Hoboken (New Jersey), 2004, pp. 1–36.

[11] J. Svoboda, F.-D. Fischer, P. Fratzl: *Diffusion and creep in multi-component alloys with non-ideal sources and sinks for vacancies.* Acta Materialia **54** (2006), 3043–3053.

[12] J. Svoboda, F.-D. Fischer, J. Vala: *Thermodynamic extremal principle and its application to Dufour and Sorret effects and plasticity.* Atti della Accademia Peloritana dei Pericolanti, to appear.

[13] J. Svoboda, E. Gamsjäger E, F.-D. Fischer, P. Fratzl: *Application of the thermodynamic extremal principle to the diffusional phase transformation.* Acta Materialia **52** (2004), 959–967.

[14] J. Svoboda, J. Vala, E. Gamsjäger, F.-D. Fischer: *A thick-interface model for diffusive and massive phase transformation in substitutional alloys.* Acta Materialia **54** (2006), 3953–3960.

[15] J. Vala: *Modelling of diffusive and massive phase transformation.* In: V. Kompiš (Ed.), Composites with Micro- and Nano-Structure. Computational Methods in Applied Sciences 9, Springer, Berlin, 2008, pp. 67–86.

DETERMINISTIC AND STOCHASTIC MODELS OF DYNAMICS OF CHEMICAL SYSTEMS[*]

Tomáš Vejchodský, Radek Erban

1. Introduction

The deterministic and stochastic models are two principal approaches for modeling of the dynamics of the chemical reactions. The deterministic models are usually based on differential equations for concentrations (or amounts of molecules) of particular chemical species whereas the *stochastic simulation algorithms* (SSA) use the pseudorandom number generators. Of course, different realizations of the SSA differ from each other, but a mean value over many realizations is a well reproducible quantity which describes the average behavior of the system.

In this paper, we examine an example motivated by chemical processes in living cells. In this example, we observe qualitatively different behaviors of the deterministic and stochastic models. Namely, the solution of the deterministic model converges to a stationary state while the stochastic solution exhibits an oscillatory character. This discrepancy is caused by the fact that the deterministic model is inexact if the number of molecules of a chemical species is too small. In this case, the more accurate stochastic model should be used. However, the disadvantage of the stochastic approach lies in its high computational cost. We show that certain quantities obtained from the SSA can be computed as solutions of deterministic partial differential equations which is much less computationally intensive.

2. Chemical system with SNIPER bifurcation

The chemical processes in cells often exhibit *saddle-node infinite period* (SNIPER) bifurcation, see for example the model of the cell cycle regulation [3]. The following simple system of seven chemical reactions exhibits the same behavior. We consider two chemical species X and Y in a well-mixed reactor of volume V which are subject to the following reactions

$$\emptyset \xrightarrow{k_{1d}} Y \xrightarrow{k_{2d}} X \xrightarrow{k_{3d}} \emptyset \qquad\qquad 2X \underset{k_{5d}}{\overset{k_{4d}}{\rightleftarrows}} 3X \tag{1}$$

$$X + Y \xrightarrow{k_{6d}} X + 2Y \qquad\qquad 2X + Y \xrightarrow{k_{7d}} 2X. \tag{2}$$

[*]This work has been supported by grants No. 102/07/0496 of the Czech Science Foundation, No. IAA100760702 of the Grant Agency of the Academy of Sciences and by the institutional research plan No. AV0Z10190503 of the Academy of Sciences of the Czech Republic.

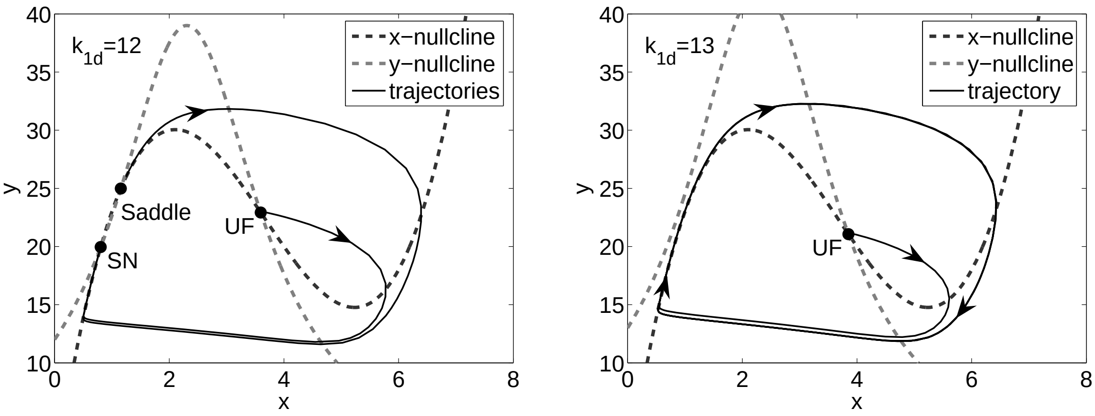


Fig. 1: *Nullclines of the ODE system* (3) *for* $k_{1d} = 12$ *(left) and* $k_{1d} = 13$ *(right). The other parameter values are given by* (4)*. The steady states are denoted by dots. Illustrative trajectories which start close to the steady states are plotted as thin black lines. The stable direction of the saddle is indicated by the dashed line.*

Here k_{1d}, ..., k_{7d} are the so-called rate constants which describe the speed of the reactions. The symbol $\emptyset$ denotes the chemical species of no interest. Hence, for example the first reaction is the production of Y from the source with the rate constant k_{1d}.

Let $X = X(t)$ and $Y = Y(t)$ stand for the number of molecules of the two chemical species. If the numbers X and Y are sufficiently high then the dynamics of the system (1)–(2) can be described by the mean-field ODE model

$$\frac{\mathrm{d}\widetilde{x}}{\mathrm{d}t} = k_{2d}\,\widetilde{y} - k_{5d}\,\widetilde{x}^3 + k_{4d}\,\widetilde{x}^2 - k_{3d}\,\widetilde{x}, \qquad \frac{\mathrm{d}\widetilde{y}}{\mathrm{d}t} = -k_{7d}\,\widetilde{x}^2\,\widetilde{y} + k_{6d}\,\widetilde{x}\,\widetilde{y} - k_{2d}\,\widetilde{y} + k_{1d}, \quad (3)$$

where $\widetilde{x} = X/V$ and $\widetilde{y} = Y/V$ stand for the concentrations of X and Y, respectively. We choose the values of the rate constants as

$$k_{1d} = 12\ [\mathrm{sec}^{-1}\mathrm{mm}^{-3}], \quad k_{2d} = 1\ [\mathrm{sec}^{-1}], \quad k_{3d} = 33\ [\mathrm{sec}^{-1}], \quad k_{4d} = 11\ [\mathrm{sec}^{-1}\mathrm{mm}^{3}],$$
$$k_{5d} = 1\ [\mathrm{sec}^{-1}\mathrm{mm}^{6}], \quad k_{6d} = 0.6\ [\mathrm{sec}^{-1}\mathrm{mm}^{3}], \quad k_{7d} = 0.13\ [\mathrm{sec}^{-1}\mathrm{mm}^{6}]. \quad (4)$$

In this case the nullclines $\mathrm{d}\widetilde{x}/\mathrm{d}t = 0$ and $\mathrm{d}\widetilde{y}/\mathrm{d}t = 0$ intersect at three steady states denoted by SN (stable node), Saddle, and UF (unstable focus), see Figure 1 (left), where also illustrative trajectories are shown. We observe that the system converges to the steady state at the SN. However, if we change the bifurcation parameter k_{1d} to have the value 13, we observe the behavior shown in Figure 1 (right). The system possesses a periodic solution.

The SN and the Saddle lie on the invariant cycle, which is a union of these two steady states and two heteroclinic trajectories connecting these steady states. While increasing the value of k_{1d}, the SN and the Saddle collaps into a single steady state which disappears if we increase the value of k_{1d} over a critical value $K_d \doteq 12.2$. This is known as the SNIPER bifurcation.

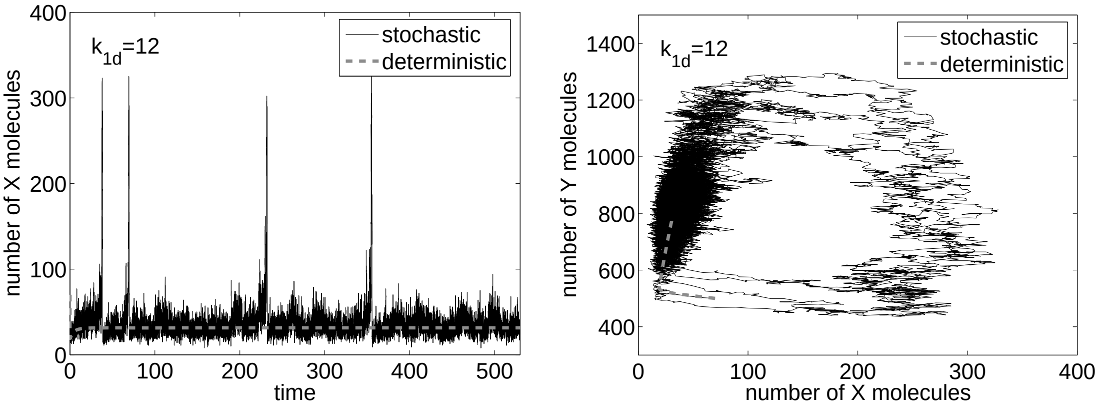


Fig. 2: *Trajectories of the deterministic and stochastic models for* $k_{1d} = 12$.

In Figure 2 we may compare the trajectories of the deterministic model (3) and the stochastic model for the parameter values given by (4). Notice the convergence of the deterministic solution to the steady sate and the oscillatory behavior of the stochastic solution. On the other hand if we change the bifurcation parameter to $k_{1d} = 13$ then both deterministic and stochastic models oscillate.

3. Gillespie stochastic simulation algorithm

The stochastic trajectories in Figures 2 were obtained by the Gillespie SSA [2]. To describe this algorithm we define the following propensity functions

$$\alpha_1(x,y) = k_1, \quad \alpha_2(x,y) = k_2 y, \quad \alpha_3(x,y) = k_3 x, \quad \alpha_4(x,y) = k_4 x(x-1),$$
$$\alpha_5(x,y) = k_5 x(x-1)(x-2), \quad \alpha_6(x,y) = k_6 xy, \quad \alpha_7(x,y) = k_7 x(x-1)y, \quad (5)$$

where $k_1 = k_{1d}V,\ k_2 = k_{2d},\ k_3 = k_{3d},\ k_4 = k_{4d}/V,\ k_5 = k_{5d}/V^2,\ k_6 = k_{6d}/V,\ k_7 = k_{7d}/V^2$ are the scaled rate constants. The Gillespie SSA follows these steps.

1. Generate two random numbers r_1, r_2 uniformly distributed in $(0,1)$.
2. Compute the cumulative propensity function $\alpha_0(t) = \sum_{i=1}^{7} \alpha_i(X(t), Y(t))$.
3. Compute the time interval to the next reaction $\tau = \frac{1}{\alpha_0(t)} \ln\left(\frac{1}{r_1}\right)$.
4. At time $t + \tau$ the j-th reaction takes place. Determine j by
$$\frac{1}{\alpha_0(t)} \sum_{i=1}^{j-1} \alpha_i(X(t), Y(t)) \le r_2 < \frac{1}{\alpha_0(t)} \sum_{i=1}^{j} \alpha_i(X(t), Y(t)).$$
5. Update the numbers of reactants and products according to the j-th reaction.
6. Put $t := t + \tau$ and go to 1.

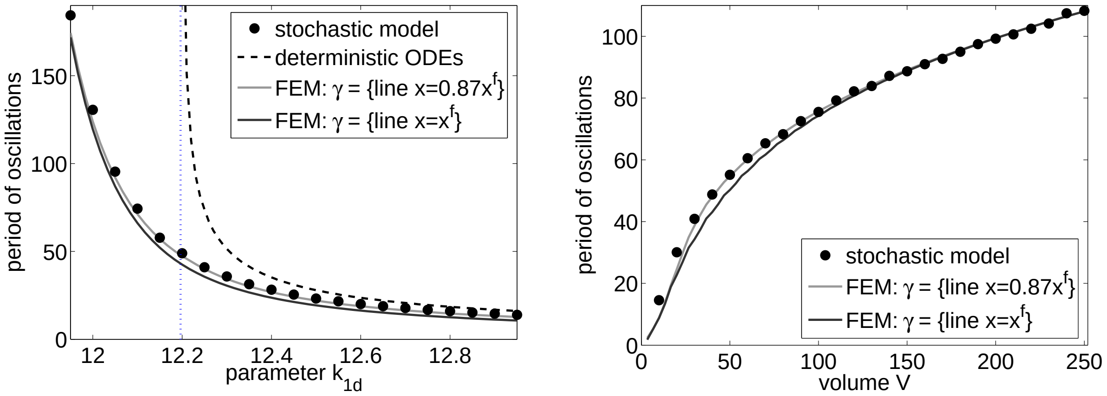


Fig. 3: *The mean period of oscillations of the deterministic (dashed line, see [1] for details) and stochastic (points) models as a function of k_{1d} for $V = 40$ (left) and as a function of V for the bifurcation value $k_{1d} = K_d \doteq 12.2$ (right). Two estimates computed by formula* (12) *with different γ are indicated by gray lines.*

4. Period of oscillations

An important characteristic of the chemical system (1)–(2) is the mean period of its oscillations. This period is shown in Figure 3 for both deterministic and stochastic models. The periods for the stochastic model are computed as an average over 10 000 realizations which is computationally intensive. Here, we show that these values can be obtained by solving and analyzing the chemical Fokker-Planck equation.

The stationary chemical Fokker-Planck equation for the stationary distribution P_s, see [1] for more details, can be formulated in the following way

$$-\operatorname{div}(\mathcal{A}\nabla P_s + P_s\mathbf{b}) = 0, \tag{6}$$

where $\mathcal{A} = -\begin{pmatrix} d_x & d_{xy}/2 \\ d_{xy}/2 & d_y \end{pmatrix}$, $\mathbf{b} = \left(v_x - \dfrac{\partial d_x}{\partial x} - \dfrac{1}{2}\dfrac{\partial d_{xy}}{\partial y}, v_y - \dfrac{\partial d_y}{\partial y} - \dfrac{1}{2}\dfrac{\partial d_{xy}}{\partial x} \right)$, and $v_x = \alpha_2 - \alpha_3 + \alpha_4 - \alpha_5$, $v_y = \alpha_1 - \alpha_2 + \alpha_6 - \alpha_7$, $d_x = [\alpha_2 + \alpha_3 + \alpha_4 + \alpha_5]/2$, $d_y = [\alpha_1 + \alpha_2 + \alpha_6 + \alpha_7]/2$, $d_{xy} = -\alpha_2$.

The stationary distribution $P_s = P_s(x, y)$ is normalized to satisfy

$$\int_0^\infty \int_0^\infty P_s(x,y)\,\mathrm{d}x\mathrm{d}y = 1, \qquad P_s(x,y) \geq 0, \qquad (x,y) \in [0,\infty) \times [0,\infty). \tag{7}$$

Here, $P_s(x,y)$ is the probability that $X(t) \to x$ and $Y(t) \to y$ as $t \to \infty$. To find numerically the stationary distribution P_s for system (1)–(2) with parameter values (4) we truncate the infinite domain $(0,\infty) \times (0,\infty)$ to $S = (0,500) \times (0,2000)$, denote by $\mathbf{n}$ is the unit outward normal vector to ∂S and prescribe the boundary conditions

$$(\mathcal{A}\nabla P_s + P_s\mathbf{b}) \cdot \mathbf{n} = 0 \quad \text{on } \partial S. \tag{8}$$

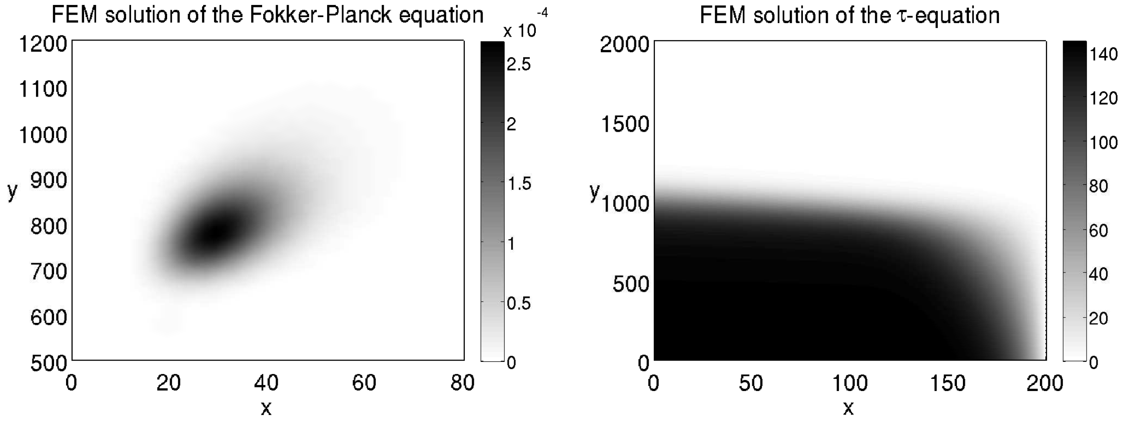


Fig. 4: *The stationary distribution $P_{s,h}$ (left) and the mean exit time τ_h (right).*

The finite element solution $P_{s,h}$ to problem (6)–(8) is derived in a standard way and it is determined by the requirements $P_{s,h} \in W_h$ and

$$\int_S \big(\mathcal{A}\nabla P_{s,h} + P_{s,h}\mathbf{b}\big) \cdot \nabla\varphi_h \,\mathrm{d}x\mathrm{d}y = 0 \quad \forall\varphi_h \in W_h,$$

where W_h is a suitable finite dimensional subspace of the Sobolev space $H^1(S)$. In our case W_h consists of continuous and piecewise linear functions over a triangulation of S. The finite element solution $P_{s,h}$ is provided in Figure 4 (left). Alternatively, the stationary distribution can be obtained by computationally very intensive long time stochastic simulations. However, the numerical solution of (6)–(8) is much faster and equally accurate as the stochastic simulations.

Let us point out that equation (6) with boundary condition (8) possesses a trivial solution $P_s = 0$ but we are interested in a nontrivial one. We obtain the nontrivial approximation $P_{s,h}$ by a standard software for computation of eigenvectors corresponding to the zero eigenvalue of a sparse matrix. The resulting solution is then normalized to satisfy (7) with the integrals taken over S.

The stationary distribution, see Figure 4 (left), shows that the system spends most of the time in the strip $X < 200$. Observing Figure 2 we may say that an oscillation occurs if $X > 200$. The stationary distribution P_s is almost zero for $x > 200$ and therefore, it is very unlikely to find the system in a state with $X > 200$. Thus, we neglect the time the system spends in the halfplane $X > 200$ and approximate the mean period of oscillations as the average time to leave the domain $X < 200$ provided the system just entered it.

To this end, we define the subdomain $\widetilde{S} = (0, 200) \times (0, 2000)$ of S and formulate the adjoint equation to (6) with suitable boundary conditions

$$-\operatorname{div}(\mathcal{A}\nabla\tau) + \mathbf{b} \cdot \nabla\tau = -1 \quad \text{in } \widetilde{S}, \tag{9}$$

$$\tau = 0 \quad \text{on the line } x = 200, \tag{10}$$

$$(\mathcal{A}\nabla\tau) \cdot \mathbf{n} = 0 \quad \text{on lines } y = 0,\ y = 2000,\ x = 0. \tag{11}$$

The quantity $\tau = \tau(x, y)$ is known [1] to model the average time to leave $\widetilde{S}$ provided the system is in the state $X(t) = x$ and $Y(t) = y$. The finite element approximation $\tau_h \in \widetilde{W}_h$ of τ is uniquely determined by the identity

$$\int_{\widetilde{S}} \big(\mathcal{A}\nabla\tau_h\big) \cdot \nabla\varphi_h \,\mathrm{d}x\mathrm{d}y + \int_{\widetilde{S}} \mathbf{b} \cdot \nabla\tau_h\, \varphi_h \,\mathrm{d}x\mathrm{d}y = \int_{\widetilde{S}} -1 \cdot \varphi_h \,\mathrm{d}x\mathrm{d}y \quad \forall\varphi_h \in \widetilde{W}_h,$$

where $\widetilde{W}_h$ is a space of continuous and piecewise linear functions over a triangulation of $\widetilde{S}$. The finite element solution τ_h of (9)–(11) is presented in Figure 4 (right).

The actual period of oscillations T is estimated as a weighted average of the exit times over a suitably chosen set (line segment) γ

$$T(\gamma) = \int_\gamma \tau(x, y)\, P_s(x, y)\,\mathrm{d}\gamma \bigg/ \int_\gamma P_s(x, y)\,\mathrm{d}\gamma. \tag{12}$$

A natural choice of the line segment is $\gamma = \{(x, y) : x = x^f,\ 0 \le y \le y^f\}$, where (x^f, y^f) is the unstable focus of the system $\mathrm{d}x/\mathrm{d}t = v_x$ and $\mathrm{d}y/\mathrm{d}t = v_y$, cf. (3). However, if the line segment γ is slightly shifted to $x = 0.87x^f$ then formula (12) provides more accurate results, as we can observe in Figure 3. The left panel shows the period of oscillations as a function of k_{1d} and the right one as a function of V.

5. Conclusions

To conclude, we stress that the period of oscillation can be computed by (12) using the solution of the Fokker-Planck equation (6) and of the τ-equation (9) with no need of long time stochastic simulations. Figure 3 shows the accuracy of this approach.

The presented technique is not limited to simple system (1)–(2). It can be applied for example to systems with more than two chemical species which are of great interest especially in the cell cycle modeling. This however requires numerical solution of high dimensional Fokker-Planck and τ-equations.

References

[1] R. Erban, J. Chapman, I. Kevrekidis, T. Vejchodský: *Analysis of a stochastic chemical system close to a SNIPER bifurcation of its mean-field model*, submitted to SIAM J. Appl. Math., 2008.

[2] D. Gillespie: *Exact stochastic simulation of coupled chemical reactions.* J. Phys. Chem. **81** (1977), 2340–2361.

[3] J. Tyson, A. Csikasz-Nagy, B. Novak: *The dynamics of cell cycle regulation.* BioEssays **24** (2002), 1095–1109.

LIMITED-MEMORY VARIABLE METRIC METHODS BASED ON INVARIANT MATRICES[*]

Jan Vlček, Ladislav Lukšan

We present a new family of limited-memory variationally-derived variable metric (VM) line search methods with quadratic termination property (see [4]) for unconstrained minimization. Starting with $x_0 \in \mathcal{R}^N$, VM line search methods (see [4]) generate iterations $x_{k+1} \in \mathcal{R}^N$ by the process $x_{k+1} = x_k + s_k$, $s_k = t_k d_k$, where the direction vectors $d_k \in \mathcal{R}^N$ are descent, i.e. $g_k^T d_k < 0$, and the stepsizes $t_k > 0$ satisfy

$$f(x_{k+1}) - f(x_k) \le \varepsilon_1 t_k g_k^T d_k, \qquad g_{k+1}^T d_k \ge \varepsilon_2 g_k^T d_k, \tag{1}$$

$k \ge 0$, with $0 < \varepsilon_1 < 1/2$ and $\varepsilon_1 < \varepsilon_2 < 1$, where f is an objective function, $g_k = \nabla f(x_k)$. We denote $y_k = g_{k+1} - g_k$, $k \ge 0$ and by $\|.\|_F$ the Frobenius matrix norm.

1. A new family of limited-memory methods

Our methods are based on approximations $\bar{H}_k = U_k U_k^T$, $k > 0$, $\bar{H}_0 = 0$, of the inverse Hessian matrix, which are **invariant** under linear transformations (it is significant in case of ill-conditioned problems), where $N \times \min(k, m)$ matrices U_k, $1 \le m \ll N$, are obtained by limited-memory updates that satisfy the quasi-Newton condition

$$\bar{H}_{k+1} y_k = s_k \qquad \text{or equivalently} \qquad U_+^T y = z, \quad U_+ z = s, \quad z^T z = b. \tag{2}$$

We frequently omit index k, replace index $k+1$ by symbol $+$, index $k-1$ by symbol $-$ and denote $V_r = I - r y^T / r^T y$ for $r \in \mathcal{R}^N$, $r^T y \ne 0$ (projection matrix),

$$B = H^{-1}, \quad b = s^T y > 0, \quad \bar{a} = y^T \bar{H} y, \quad \bar{b} = s^T B \bar{H} y, \quad \bar{c} = s^T B \bar{H} B s, \quad \bar{\delta} = \bar{a}\bar{c} - \bar{b}^2 \ge 0.$$

Standard VM updates can be derived as updates with the **minimum change** of VM matrix (see [4]), which we extend to limited-memory methods (see [6], [7]).

Theorem 1.1. *Let T be a symmetric positive definite matrix, $z \in \mathcal{R}^m$, $1 \le m \le N$, $p = Ty$, and $\mathcal{U}$ the set of $N \times m$ matrices. Then the unique solution to* $\min\{\varphi(U_+) : U_+ \in \mathcal{U}\}$ *s.t. (2), where $\varphi(U_+) = y^T T y \, \|T^{-1/2}(U_+ - U)\|_F^2$, is*

$$U_+ = s z^T / b + V_p U \Big(I - z z^T / z^T z\Big), \quad \bar{H}_+ = s s^T / b + V_p U \Big(I - z z^T / z^T z\Big) U^T V_p^T. \tag{3}$$

[*] This work was supported by the Grant Agency of the Czech Academy of Sciences, project No. IAA1030405, and the Institutional research plan No. AV0Z10300504

Updates (3) can be **invariant** under linear transformations, i.e. can preserve the same transformation property of $\bar{H} = UU^T$ as inverse Hessian (see [7]).

Theorem 1.2. *Consider a change of variables $\tilde{x} = Rx + r$, where R is $N \times N$ nonsingular matrix, $r \in \mathcal{R}^N$. Let $p \in \operatorname{span}\{s, \bar{H}y, Uz\}$ and suppose that z and coefficients in the linear combination of vectors s, $\bar{H}y$ and Uz forming p are invariant under the transformation $x \to \tilde{x}$, i.e. they are not influenced by this transformation. Then for $\tilde{U} = RU$ matrix U_+ given by (3) also transforms to $\tilde{U}_+ = RU_+$.*

In the special case (this choice satisfies the assumptions of Theorem 1.2)

$$p = (\lambda/b)s + [(1-\lambda)/\bar{a}]\bar{H}y \ \text{ if } \ \bar{a} \neq 0, \qquad p = (1/b)s, \ \lambda = 1 \ \text{ otherwise} \tag{4}$$

we can easily compare (3) with the Broyden class update of $\bar{H}$ with parameter $\eta = \lambda^2$, to obtain $\bar{H}_+ = \bar{H}_+^{BC} - V_p Uz(V_p Uz)^T/z^T z$, where $\bar{H}_+^{BC} = ss^T/b + V_p \bar{H} V_p^T$ (see [6]). The last update is useful for **starting** iterations. Setting $U_+ = [\sqrt{1/b}\, s]$ in the first iteration, every such update modifies U and adds one column $\sqrt{1/b}\, s$ to U_+. Except for the starting iterations, we will assume that matrix U has $m \geq 1$ columns.

To choose parameter z, we utilize analogy with standard VM methods (see [7]).

Lemma 1.3. *Let $H = SS^T$ with $N \times N$ matrix S and let $z = \alpha(S^T Bs + \theta S^T y)$, α, $\theta \in \mathcal{R}$, with $z^T z = b$. Then every update (3) with S, S_+, $S_+ S_+^T$ instead of U, U_+, $\bar{H}_+$ and with p given by (4) belongs to the scaled Broyden class with*

$$\eta = \lambda^2 - b\,\alpha^2\, y^T H y \left(\theta\lambda/b - (1-\lambda)/y^T H y\right)^2. \tag{5}$$

Thus we concentrate here on the choice $z = \alpha(S^T Bs + \theta S^T y)$, $z^T z = b$. Lemma 1.4 (see [7]) gives simple conditions for this z to be invariant under linear transformations.

Lemma 1.4. *Let number θ/t be invariant under transformation $\tilde{x} = Rx + r$, where t is the stepsize, R is $N \times N$ nonsingular matrix and $r \in \mathcal{R}^N$, and suppose that $\tilde{U} = RU$. Then vector z used in Lemma 1.3 is invariant under this transformation.*

We use the choice $\theta = -\bar{b}/\bar{a}$. Then θ/t is invariant and $z^T z = b$ gives (if $\bar{a}\bar{\delta} = 0$, we do not update) $z = \pm\sqrt{b/(\bar{a}\bar{\delta})}\,(\bar{a}\, U^T Bs - \bar{b}\, U^T y)$, $y^T Uz = 0$, and $V_p Uz = Uz$.

2. Variationally-derived simple correction

To have matrices $\bar{H}_k$ invariant, we use such updates that $-\bar{H}_k g_k$ cannot be used as the direction vectors d_k. Thus we find the minimum correction (in the sense of Frobenius matrix norm, see [7]) of matrix $\bar{H}_+ + \zeta I$, $\zeta > 0$, in order that the resultant matrix H_+ may satisfy the quasi-Newton condition $H_+ y = s$. First we give the **projection variant** of the well-known Greenstadt's theorem, see [3].

For $M = \bar{H}_+ + \zeta I$, the resulting correction (8) together with update (3) give our family of limited-memory VM methods.

Theorem 2.1. *Let M, W be symmetric matrices, W positive definite, $q = Wy$ and denote $\mathcal{M}$ the set of $N \times N$ symmetric matrices. Then the unique solution to*

$$\min\{\|W^{-1/2}(M_+ - M)W^{-1/2}\|_F : M_+ \in \mathcal{M}\} \quad \text{s.t.} \quad M_+ y = s \tag{6}$$

is given by $V_q\,(M_+ - M)\,V_q^T = 0$ and can be written (the usual form is on the right)

$$M_+ = E + V_q\,(M - E)V_q^T \equiv M + (wq^T + qw^T)/q^T y - w^T y \cdot qq^T/(q^T y)^2\,, \tag{7}$$

where E is any symmetric matrix satisfying $Ey = s$, e.g. $E = ss^T/b$, $w = s - My$.

Theorem 2.2. *Let W be a symmetric positive definite matrix, $\zeta > 0$, $q = Wy$ and denote $\mathcal{M}$ the set of $N \times N$ symmetric matrices. Suppose that matrix $\bar{H}_+$ satisfies the quasi-Newton condition (2). Then the unique solution to*

$$\min\{\|W^{-1/2}(H_+ - \bar{H}_+ - \zeta I)W^{-1/2}\|_F : H_+ \in \mathcal{M}\} \quad \text{s.t.} \quad H_+ y = \varrho s$$

is

$$H_+ = \bar{H}_+ + \zeta V_q V_q^T. \tag{8}$$

As regards parameter ζ, the widely used choice is $\zeta = b/y^T y$ which minimizes $|(\bar{H}_+ - \zeta I)y|$. We can obtain slightly better results, e.g. by the choice

$$\zeta = \varrho\, b/(y^T y + 4\,\bar{a}). \tag{9}$$

The following lemmas (see [7]) help us to obtain vector q in such a way that corrections (7), (8) represent the **Broyden class** updates of $\bar{H}_+ + \zeta I$ with parameter η.

Lemma 2.3. *Let A be a symmetric matrix and denote $a = y^T Ay$. Then every update (7) with $M = A$, $M_+ = A_+$, $q = s - \alpha Ay$, $a \neq 0$, and $\alpha a \neq b$ represents the Broyden class update with parameter $\eta = (b^2 - \alpha^2 ab\,)/(b - \alpha a)^2$.*

Lemma 2.4. *Let $\zeta > 0$, $\kappa = \zeta\, y^T y/b$, $\eta > -1/(1+\kappa)$ and let matrix $\bar{H}_+$ satisfy the quasi-Newton condition (2). Then correction (8) with $q = s - \sigma y$, where $\sigma = b(1 \pm \sqrt{(1+\kappa)/(1+\eta\kappa)}\,)/y^T y$ represents the Broyden class update of matrix $\bar{H}_+ + \zeta I$ with parameter η.*

For $q = s$, i.e. $\eta = 1$, we get the BFGS update. Better results were obtained with the formula, based on analogy with the shifted VM methods (see [6], [7]):

$$\eta = \min\left[1, \max\left[0\,, 1 + (1/\kappa)(1 + 1/\kappa)\,(1.2\,\zeta_-/(\zeta_- + \zeta) - 1)\right]\right]. \tag{10}$$

3. Correction formula

Corrections in Section 2 respect only the latest vectors s_k, y_k. Thus we can again correct (without scaling) matrices $\check{H}_{k+1} = \bar{H}_{k+1} + \zeta_k V_{q_k} V_{q_k}^T$, $k > 0$, obtained from (8), using **previous vectors** s_i, y_i, $i = k-j, \ldots, k-1$, $j \le k$. Our experiments indicate that the choice $j = 1$ brings the maximum improvement. This leads to the formula $H_+ = ss^T/b + V_s\left[s_- s_-^T/b_- + V_s^-\left(\bar{H}_+ + \zeta V_q V_q^T\right)(V_s^-)^T\right]V_s^T$, where $V_s^- = I - s_- y_-^T/b_-$, which is less sensitive to the choice of ζ than (8).

4. Quadratic termination property

We give conditions for our family of limited-memory VM methods with exact line searches to terminate on a quadratic function in at most N iterations (see [7]).

Theorem 4.1. *Let $m \in \mathcal{N}$ be given and let $Q(x) = \frac{1}{2}(x-x^*)^T G(x-x^*)$, where G is an $N \times N$ symmetric positive definite matrix. Suppose that $\zeta_k > 0$, $t_k > 0$, $k \ge 0$, and that for $x_0 \in \mathcal{R}^N$ iterations $x_{k+1} = x_k + s_k$ are generated by the method $s_k = -t_k H_k g_k$, $g_k = \nabla Q(x_k)$, $k \ge 0$, with exact line searches, i.e. $g_{k+1}^T s_k = 0$, where*

$$H_0 = I, \quad H_{k+1} = U_{k+1}U_{k+1}^T + \zeta_k V_{q_k} V_{q_k}^T,\ k \ge 0, \tag{11}$$

$N \times \min(k,m)$ matrices U_k, $k > 0$, satisfy

$$U_1 = \left[s_0/\sqrt{b_0}\,\right], \quad U_{k+1}U_{k+1}^T = s_k s_k^T/b_k + V_{p_k} U_k U_k^T V_{p_k}^T,\ 0 < k < m, \tag{12}$$

$$U_{k+1}U_{k+1}^T = s_k s_k^T/b_k + V_{p_k} U_k \Big(I - z_k z_k^T / z_k^T z_k\Big) U_k^T V_{p_k}^T,\ z_k \in \mathcal{R}^m,\ k \ge m, \tag{13}$$

vectors p_k, q_k, $k > 0$, lie in $\mathrm{range}([U_k, s_k])$ and satisfy $p_k^T y_k \ne 0$, $q_k^T y_k \ne 0$, and $q_0 = s_0$. Then there exists a number $\bar{k} \le N$ with $g_{\bar{k}} = 0$ and $x_{\bar{k}} = x^*$.

5. Computational experiments

Our VM methods were tested, using the collection [5] of sparse, usually ill-conditioned problems for large-scale nonlinear least squares (Test 15, 21 problems) with $N = 500$ and 1000, $m = 10$, ζ given by (9) and the final precision $\|g(x^\star)\|_\infty \le 10^{-5}$.

η_p	$N = 500$				$N = 1000$			
	Corr-0	Corr-1	Corr-2	Corr-q	Corr-0	Corr-1	Corr-2	Corr-q
0.0	2-76916	32504	22626	24016	3-99957	1-58904	44608	1-47204
0.1	3-99032	36058	21839	35756	3-98270	1-54494	42649	1-47483
0.2	2-97170	29488	23732	29310	3-89898	1-52368	36178	1-44115
0.3	1-79978	28232	18388	18913	3-80087	47524	33076	38030
0.4	1-70460	24686	18098	17673	3-78498	44069	32403	34437
0.5	60947	22532	17440	17181	3-88918	41558	32808	31874
0.6	56612	21240	17800	17164	2-76264	38805	31854	30784
0.7	52465	20289	17421	17021	2-72626	39860	32345	30802
0.8	51613	20623	17682	17076	1-69807	37501	32292	32499
0.9	50877	20548	18102	17424	2-69802	38641	32926	31385
1.0	49672	20500	18109	17913	1-68603	38510	33539	32456
1.1	52395	20994	18694	18470	1-65676	41284	35103	33053
1.2	51270	21444	19230	18372	1-68711	41332	35649	34028
1.5	1-51094	22808	20487	20060	2-66220	42906	36775	36323
2.0	1-50776	24318	21710	21639	2-66594	46139	40279	39199
BNS	18444				33131			

Tab. 1: *Comparison of various correction methods.*

Results of these experiments are given in two tables, where $\eta_p = \lambda^2$ is the value of parameter η of the Broyden class used to determine parameter p by (4) and η_q is the value of this parameter used in Lemma 2.4 to determine parameter $q = s - \sigma y$.

In Table 1 we compare the method described in [2] (BNS) with our new family, using various values of η_p and the following **correction methods:** Corr-0 – the adding of matrix ζI to $\bar{H}_+$, Corr-1 – correction (8), Corr-2 – correction from Section 3. We use $\eta_q = 1$ (i.e. $q = s$) in columns Corr-0, Corr-1 and Corr-2 and η_q given by (10) in columns Corr-q together with correction from Section 3. We present the total numbers of function and also gradient evaluations (over all problems), preceded by the minus-sign with the number of problems (if any occurred) which were not solved successfully (the number of evaluations reached its limit 19000 evaluations).

	$\eta_p\,,\quad N = 500$							$\eta_p\,,\quad N = 1000$						
η_q	0.4	0.5	0.6	0.7	0.8	0.9	1.0	0.4	0.5	0.6	0.7	0.8	0.9	1.0
0.0	-343	-394	-967	-813	-538	32	141	1916	-912	-681	-876	-119	-744	116
0.1	211	-1154	-1028	-1100	-880	-585	-188	1052	-732	-974	-1647	-1043	-1215	320
0.2	2424	1902	1759	2088	1869	2268	2746	903	-187	-1669	-1708	-1219	-28	-567
0.3	-492	-1064	-1136	-992	-1036	-901	-939	793	-363	-975	-1731	-289	360	-484
0.4	-599	-1069	-718	-1160	-668	-934	-512	925	-1398	-1708	-1554	-1184	-498	-482
0.5	-493	-722	-727	-665	-487	-516	-399	-757	-644	-965	-1729	-1380	-926	-207
0.6	-251	-648	-798	-965	-750	-176	-371	1	-1396	-1291	-835	-1044	-767	190
0.7	-342	-764	-441	-320	-474	-749	-284	-195	-901	-356	-1019	-1482	-398	-454
0.8	-481	-706	-857	-579	-449	-497	-606	-770	-690	-1763	-886	-1009	-256	-977
0.9	-872	-759	-370	-559	-820	275	-135	8	-821	-939	-674	-696	-764	657
1.0	-346	-1004	-644	-1023	-762	-342	-335	-728	-323	-1277	-786	-839	-205	408
1.1	1939	1265	2326	791	2444	1958	1910	-773	115	183	48	-411	-619	736
1.2	1024	700	719	1452	967	1479	1982	269	155	-670	295	-649	-113	647
1.5	-596	-436	-912	-937	-770	-285	307	377	-181	-29	908	1323	441	1310
2.0	150	-396	85	259	336	222	684	2164	767	994	2035	2577	2869	3036
(10)	-771	-1263	-1280	-1423	-1368	-1020	-531	1306	-1257	-2347	-2329	-632	-1746	-675

Tab. 2: *Comparison with BNS for various η_p, η_q.*

In Table 2 we give the **differences** $n_{p,q} - n_{BNS}$, where $n_{p,q}$ is the total number of function and also gradient evaluations (over all problems) for selected values of η_p and η_q with correction from Section 3 and n_{BNS} is the number of evaluations for method BNS (negative values indicate that our method is better than BNS). In the last row, we present this difference for η_q given by (10).

For a better comparison with method BNS, we performed additional tests with problems from the widely used **CUTE** collection [1] with various dimensions N and the final precision $\|g(x^\star)\|_\infty \le 10^{-6}$. In Table 3 we give the values of $(n_{p,q} - n_{BNS})/(n_{p,q} + n_{BNS}) * 100$ for $\eta_p = \eta_q = 0.5$ (all the others are the same as above).

Our limited numerical experiments indicate that methods from our new family can compete with the well-known BNS method.

Problem	*N*	%	Problem	*N*	%	Problem	*N*	%
ARWHEAD	5000	=0	BDQRTIC	5000	16	BROWNAL	500	=0
BROYDN7D	2000	-3	BRYBND	5000	-2	CHAINWOO	1000	-0
COSINE	5000	22	CRAGGLVY	5000	=0	CURLY10	1000	-5
CURLY20	1000	-9	CURLY30	1000	-3	DIXMAANA	3000	4
DIXMAANB	3000	21	DIXMAANC	3000	10	DIXMAAND	3000	12
DIXMAANE	3000	23	DIXMAANF	3000	28	DIXMAANG	3000	30
DIXMAANH	3000	22	DIXMAANI	3000	50	DIXMAANJ	3000	38
DIXMAANK	3000	27	DIXMAANL	3000	41	DQRTIC	5000	59
EDENSCH	5000	-2	EG2	1000	=0	ENGVAL1	5000	7
EXTROSNB	5000	-3	FLETCBV2	1000	3	FLETCHCR	1000	11
FMINSRF2	1024	9	FMINSURF	1024	4	FREUROTH	5000	19
GENHUMPS	1000	9	GENROSE	1000	-4	LIARWHD	1000	2
MOREBV	5000	=0	MSQRTALS	529	3	NCB20	510	20
NCB20B	1010	12	NONCVXU2	1000	-19	NONCVXUN	1000	<-35
NONDIA	5000	-22	NONDQUAR	5000	64	PENALTY1	1000	-2
PENALTY3	100	-1	POWELLSG	5000	11	POWER	1000	64
QUARTC	5000	61	SCHMVETT	5000	-6	SINQUAD	5000	7
SPARSINE	1000	-2	SPARSQUR	1000	-2	SPMSRTLS	4999	-4
SROSENBR	5000	-10	TOINTGSS	5000	-8	TQUARTIC	5000	-9
VARDIM	1000	1	VAREIGVL	1000	-8	WOODS	4000	11

Tab. 3: *Comparison with BNS for CUTE.*

References

[1] I. Bongartz, A.R. Conn, N. Gould, P.L. Toint: *CUTE: constrained and unconstrained testing environment*, ACM Trans. Math. Software **21** (1995), 123–160.

[2] R.H. Byrd, J. Nocedal, R.B. Schnabel: *Representation of quasi-Newton matrices and their use in limited memory methods*, Math. Programming **63** (1994), 129–156.

[3] J. Greenstadt: *Variations on variable metric methods*, Math. Comput. **24** (1970), 1–22.

[4] L. Lukšan, E. Spedicato: *Variable metric methods for unconstrained optimization and nonlinear least squares*, J. Comput. Appl. Math. **124** (2000), 61–95.

[5] L. Lukšan, J. Vlček: *Sparse and partially separable test problems for unconstrained and equality constrained optimization*, Report V-767, Inst. of Computer Science, Acad. Sci., Czech Rep., 1998.

[6] J. Vlček, L. Lukšan: *Shifted variable metric methods for large-scale unconstrained optimization*, J. Comput. Appl. Math. **186** (2006) 365–390.

[7] J. Vlček, L. Lukšan: *New class of limited-memory variationally-derived variable metric methods*, Report V-973, Inst. of Computer Science, Acad. Sci., Czech Rep., 2006.

List of Participants

Stanislav Bartoň, doc. RNDr., CSc.
Ústav základů techniky
Agronomická fakulta MZLU v Brně
Zemědělská 1, 613 00 Brno
e-mail: barton@mendelu.cz

Tomáš Berka, Ing.
Katedra matematiky
Fakulta aplikovaných věd ZČU v Plzni
Univerzitní 22, 306 14 Plzeň
e-mail: berkat@kma.zcu.cz

Marek Brandner, Ing., Ph.D.
Katedra matematiky
Fakulta aplikovaných věd ZČU v Plzni
Univerzitní 22, 306 14 Plzeň
e-mail: brandner@kma.zcu.cz

Pavel Burda, prof. RNDr., CSc.
Ústav technické matematiky
Fakulta strojní ČVUT v Praze
Karlovo náměstí 13, 121 35 Praha 2
e-mail: pavel.burda@fs.cvut.cz

Lubor Buřič, Ing.
Ústav matematiky, Fakulta
chemicko-inženýrská VŠCHT v Praze
Technická 5, 166 28 Praha 6 Dejvice
e-mail: lubor.buric@vscht.cz

Libor Čermák, doc. RNDr., CSc.
Ústav matematiky
Fak. strojního inženýrství VUT v Brně
Technická 2896/2, 616 69 Brno
e-mail: cermak@fme.vutbr.cz

Dana Černá, Mgr.
Katedra mat. a didaktiky matematiky
Fakulta přírodovědně-humanitní
a pedagogická TU v Liberci
Studentská 2, 461 17 Liberec 1
e-mail: dana.cerna@tul.cz

Marta Čertíková, RNDr.
Ústav technické matematiky
Fakulta strojní ČVUT v Praze
Karlovo náměstí 13, 121 35 Praha 2
e-mail: marta.certikova@fs.cvut.cz

Jan Chleboun, RNDr., CSc.
Katedra matematiky
Fakulta stavební ČVUT v Praze
Thákurova 7, 166 29 Praha 6
e-mail: chleboun@mat.fsv.cvut.cz

Pavol Chocholatý, doc. RNDr., CSc.
Katedra numerických a optimal. metód
Fakulta matematiky, fyziky a informat.
UK v Bratislave
Mlynská dolina, 842 48 Bratislava
Slovensko
e-mail: chocholaty@fmph.uniba.sk

Robert Cimrman, Ing., Ph.D.
Katedra matematiky
Fakulta aplikovaných věd ZČU v Plzni
Univerzitní 22, 306 14 Plzeň
e-mail: cimrman3@ntc.zcu.cz

Josef Dalík, doc. RNDr., CSc.
Ústav mat. a deskriptivní geometrie
Fakulta stavební VUT v Brně
Žižkova 17, 602 00 Brno
e-mail: dalik.j@fce.vutbr.cz

Alexandr Damašek, Ing., Ph.D.
Ústav termomechaniky AV ČR
Dolejškova 1402/5, 182 00 Praha 8
e-mail: damasek@it.cas.cz

Cyril Fischer, RNDr., Ph.D.
Ústav teoretické a aplikované
mechaniky AV ČR
Prosecká 76, 190 00 Praha 9
e-mail: fisherc@itam.cas.cz

Milan Hanuš, Bc.
Katedra matematiky
Fakulta aplikovaných věd ZČU v Plzni
Univerzitní 22, 306 14 Plzeň
e-mail: mhanus@students.zcu.cz

Martin Holík, Mgr.
Katedra numerické matematiky
Matematicko-fyzikální fak. UK v Praze
Sokolovská 83, 186 75 Praha 8
e-mail: martelh@seznam.cz

Jiří Hozman, Mgr.
Katedra numerické matematiky
Matematicko-fyzikální fak. UK v Praze
Sokolovská 83, 186 75 Praha 8
e-mail: jhozmi@volny.cz

Petr Koňas, Ing., Ph.D.
Ústav nauky o dřevě
MZLU v Brně
Zemědělská 3, 613 00 Brno
e-mail: petr.konas@gmail.com

Vladimír Kracík, doc. Ing., CSc.
Katedra aplikované matematiky
Fakulta přírodovědně-humanitní
a pedagogická TU v Liberci
Studentská 2, 461 17 Liberec 1
e-mail: vladimir.kracik@tul.cz

Václav Kučera, RNDr., Ph.D.
Katedra numerické matematiky
Matematicko-fyzikální fak. UK v Praze
Sokolovská 83, 186 75 Praha 8
e-mail: vaclav.kucera@email.cz

Pavel Kůs, Mgr.
Matematický ústav AV ČR
Žitná 25, 115 67 Praha 1
e-mail: pavel.kus@gmail.com

Ladislav Lukšan, prof. Ing., DrSc.
Ústav informatiky AV ČR
Pod Vodárenskou věží 2, 182 07 Praha 8
e-mail: luksan@cs.cas.cz

Jitka Machalová, RNDr., Ph.D.
Katedra mat. analýzy a aplikací mat.
Přírodovědecká fakulta UP v Olomouci
Tomkova 40, 779 00 Olomouc-Hejčín
e-mail: machalova@inf.upol.cz

Ctirad Matonoha, RNDr., Ph.D.
Ústav informatiky AV ČR
Pod Vodárenskou věží 2, 182 07 Praha 8
e-mail: matonoha@cs.cas.cz

Petr Mayer, doc. RNDr., Dr.
Katedra matematiky
Fakulta stavební ČVUT v Praze
Thákurova 7, 166 29 Praha 6
e-mail: pmayer@karlin.mff.cuni.cz

Stanislav Míka, prof. RNDr., CSc.
Katedra matematiky
Fakulta aplikovaných věd ZČU v Plzni
Univerzitní 22, 306 14 Plzeň
e-mail: mika@kma.zcu.cz

Jaroslav Mlýnek, doc. RNDr., CSc.
Katedra mat. a didaktiky matematiky
Fakulta přírodovědně-humanitní
a pedagogická TU v Liberci
Studentská 2, 461 17 Liberec 1
e-mail: jaroslav.mlynek@tul.cz

Vratislava Mošová, RNDr., CSc.
Ústav exaktních věd
Moravská vysoká škola Olomouc, o.p.s.
Jeremenkova 1142/42, 772 00 Olomouc
e-mail: vratislava.mosova@mvso.cz

Eva Neumanová, RNDr., Ph.D.
Ústav technické matematiky
Fakulta strojní ČVUT v Praze
Karlovo nám. 13, 121 35 Praha 2
e-mail: neumanov@marian.fsik.cvut.cz

Tomáš Neustupa, RNDr., PhD.
Ústav technické matematiky
Fakulta strojní ČVUT v Praze
Karlovo nám. 13, 121 35 Praha 2
e-mail: tneu@centrum.cz

Petr Přikryl, prof. RNDr., CSc.
Matematický ústav AV ČR
Žitná 25, 115 67 Praha 1
e-mail: prikryl@math.cas.cz

Eduard Rohan, doc. Dr. Ing.
Katedra matematiky
Fakulta aplikovaných věd ZČU v Plzni
Univerzitní 22, 306 14 Plzeň
e-mail: rohan@kme.zcu.cz

Karel Segeth, prof. RNDr., CSc.
Matematický ústav AV ČR
Žitná 25, 115 67 Praha 1
e-mail: segeth@math.cas.cz

Carmen Simerská, doc. RNDr., CSc.
Ústav matematiky VŠCHT v Praze
Technická 5, 166 28 Praha 6
e-mail: simerskc@vscht.cz

Jakub Šístek, Ing.
Ústav technické matematiky
Fakulta strojní ČVUT v Praze
Karlovo náměstí 13, 121 35 Praha 2
e-mail: jakub.sistek@fs.cvut.cz

Pavel Šolín, RNDr., Ph.D.
Ústav termomechaniky AV ČR
Dolejškova 5, 182 00 Praha 8
e-mail: solin@utep.edu

Ivona Svobodová, Mgr.
Katedra mat. a deskriptivní geometrie
VŠB–TU v Ostravě
17. listopadu 15, 708 33 Ostrava-Poruba
e-mail: ivona.svobodova@vsb.cz

Jiří Vala, doc. Ing., CSc.
Ústav mat. a deskriptivní geometrie
Fakulta stavební VUT v Brně
Veveří 95, 602 00 Brno
e-mail: Vala.J@fce.vutbr.cz

Tomáš Vejchodský, RNDr., Ph.D.
Matematický ústav AV ČR
Žitná 25, 115 67 Praha 1
e-mail: vejchod@math.cas.cz

Jan Vlček, prom. mat.
Ústav informatiky AV ČR
Pod Vodárenskou věží 2, 182 07 Praha 8
e-mail: vlcek@cs.cas.cz

Michal Zajac, Bc.
Katedra numerické matematiky
Matematicko-fyzikální fak. UK v Praze
Sokolovská 83, 186 75 Praha 8
e-mail: zajac.michal@gmail.com

Pavel Ženčák, RNDr., Ph.D.
Katedra mat. analýzy a aplikací mat.
Přírodovědecká fakulta UP v Olomouci
Tomkova 40, 779 00 Olomouc-Hejčín
e-mail: zencak@inf.upol.cz

Jan Zítko, doc. RNDr., CSc.
Katedra numerické matematiky
Matematicko-fyzikální fak. UK v Praze
Sokolovská 83, 186 75 Praha 8
e-mail: zitko@karlin.mff.cuni.cz