Technical article

Latest trends in solving multiphysic problems

Abstract — **New trends in solving multiphysics problems are presented and discussed. The principal attention is paid to modern ways of their numerical solutions. The algorithms exhibiting various advanced features (fully adaptive higher-order finite element method, hanging nodes of any level, multimesh techniques, curvilinear elements, time adaptive methods, etc.) are illustrated by several typical examples.**

I Introduction

Operation of complex physical systems is often affected by a great number of accompanying phenomena of both internal and external origins. For electromagnetic systems, for example, the first group includes the thermal and force effects of electromagnetic field generated by this system or ageing of materials leading to slow deterioration of physical properties of the structural elements. External influences are caused by the properties of environment such as temperature, pressure, humidity, chemical composition, or by external physical fields. Many of these phenomena can be considered deterministic, but some of them may exhibit random features.

The development and design of such systems must be based on deep knowledge of these phenomena in order to get

- the complete idea about their properties, performance and behaviour in all admissible operation conditions,
- operation parameters and characteristics of their individual parts,
- and every information necessary for their optimization, selection of control strategies, etc.

Presently, the most powerful and cheapest tool for reaching this goal is computer modelling. Although good hardware and software for this purpose are not cheap, building of sophisticated physical models or industrial prototypes and their experimental testing in many variants is usually much more expensive.

Practically every complex system is characterized by an interaction of several physical fields and circuits. We can mention electromagnetic field, temperature field, field of mechanical forces and torques, field of mechanical or thermal strains and stresses, field of displacements or deformations, field of flow or acoustic field. As for circuits, in many problems we can meet with electric and magnetic circuits, mechanical (including hydraulic or pneumatic) circuits or thermal circuits.

The mathematical models of the listed physical fields are mostly given by generally nonlinear and nonstationary second-order partial differential equations (PDEs). Their coefficients containing the physical properties of materials and media are, moreover, nonlinear functions of various state variables, such as temperature or pressure.

The mathematical models of particular circuits are usually given by the first-order or second-order nonlinear and nonstationary ordinary differential equations (ODEs). For their coefficients there holds the same as above.

The computer modelling of any system starts from the numerical solution of its mathematical model, usually consisting of several PDEs and ODEs of the mentioned types. The temporal and spatial distributions of particular physical fields and time behaviour of the circuits represent the basis for finding the required (both local and integral) quantities fully describing the system and its properties.

II Computer Solution - State of the Art

Any complete evaluation of state of the art in solving coupled problems is presently a practically infeasible business. Nevertheless, in the following paragraphs we will try to summarize general ideas of what is already feasible and what will still require further research.

A Modelling of single physical fields

As for single physical fields, it is estimated that about 50–90 % (pending on particular field) of typical tasks can successfully be solved using the existing commercial or Academia codes, mostly based on the finite element method (FEM).

In the domain of electrical engineering we can mention, for example, well-known specialised high-level commercial codes OPERA (Cobham), Flux (Cedrat), infolytica or CST Studio Suite (all of them also being able to solve specific coupled problems) or more versatile codes COMSOL Multiphysics and ANSYS that allow solving a substantially wider spectrum of the multi-physics tasks, but in a limited number of formulations. A number of other codes are intended for solving problems in the domains of structural mechanics, hydrodynamics, aerodynamics and other technical disciplines.

Commercial codes (that are very expensive) usually offer user-friendly pre-processors and post-processors, often very powerful processors and various formulations of the model allowing the most efficient solution of the given task. On the other hand, they do not support advanced elements of adaptivity, hanging nodes, multi-mesh technologies and exhibit only limited possibilities of combinations of different types of elements.

Academia codes are mostly open sources. We can mention, for example, deal.II [1], FEniCS Project [2], FEMM [3], FreeFem++ [4], MOOSE [5] and also Agros2D [6], which is an application developed for years by the authors of this paper. These codes usually offer more advanced features (sophisticated adaptive techniques, etc.), but their user-friendliness is usually low. Moreover, most of the above codes are just libraries. An exception are FEMM and Agros2D that implement a graphical interface and allow solving complex technical problems. Consequently, both these codes are usable in technical practice.

B Modelling of coupled problems

This is a much more demanding business. In many cases, when particular fields are supposed to affect one another only slightly, one still often uses a weakly coupled formulation, i.e., the fields are solved independently of one another, as a system of single fields.

But if mutual influence of the fields involved is strong and nonlinear, the task should be solved in the hard-coupled (monolithic) formulation. Unfortunately, such a way of solution often takes extremely long time or, due to the claims on the capacity of computers available, cannot be realized at all.

That is why in the last years much effort has been devoted to the development of various tools contributing to the acceleration of computations and reduction of the memory needed.

C Still incompletely solved problems

Despite all possible efforts, however, numerous single-field and coupled problems can still be solved only partially using specific methods that cannot be generalized, their solution requires an unacceptably long time or high computer memory, convergence of results or stability of computations is poor, etc. We can mention, for example:

- Strongly non-linear fields.
- Fields containing geometrically incommensurable domains of different physical parameters and multi-scale problems.
- Extremely fast transient fields (shock-turbulence interaction).
- Fields containing materials and media of complex physical properties (anisotropic, hysteresis or intelligent materials).
- Turbulent flows, non-Newtonian flows.
- Problems with uneasily determinable boundary conditions (temperature field, flow field, acoustic field).
- Problems with material parameters and their temperature dependences that are often known only approximately.

III New trends

Generally, new trends in solving complex multiphysics problems can be divided into four categories described below.

Reduction of DOFs

Reduction of the degrees of freedom (DOFs) of the problem at the same or even better accuracy of results.

- Finite elements of higher orders of accuracy [7] (powerful, the convergence of results is exponential [8], but their implementation is extremely complicated).
- Fully adaptive algorithms [9], [10] (very efficient but complicated, their application often prolongs the time of computation).
- Multi-mesh technologies [11], [12] (different fields are calculated on different meshes best corresponding to their specific features).
- Dynamic meshes for transient problems [13], [14] (efficient and low cost quadrature in each time step).
- Hanging nodes of any order [15], [16] (their appropriate application leads to significant savings of DOFs without any loss of accuracy).
- Combination of elements of several types (for example, curvilinear elements that excellently fit curved interfaces and boundaries).

Acceleration of the computational algorithms

Acceleration of the computational algorithms (mainly assembling of the stiffness matrix and solving the corresponding systems of algebraic equations). The domains of interest are:

- Parallel computing (computation of the coefficients of the stiffness matrix, selected parts of the solution of the system).
- Efficient methods of the domain decomposition.
- Proposal of more efficient computational algorithms and solvers, particularly for nonlinear problems (such as Krylov Jacobian-free methods).
- Time adaptive methods for transient problems.
- Handling time as another dimension with all elements of adaptivity.

Optimization and model reduction

Model reduction is a technology based on real-time computing of simplified models with still acceptable accuracy of the results. Optimization is one of the crucial tasks of the designer and its purpose is to improve the performance of the system and reduce its costs. The principal tasks are:

- Proposal of model reduction strategies [17], [18].
- Development of fast and reliable methods of optimization (shape optimization, optimization of parameters, selection of materials, etc.).

Dynamic behaviour of the system

Incorporation of the computer model of a particular device into the whole dynamic system generally represented by a system of PDEs, ODEs and algebraic equations. Of great importance is:

- Proposal of fast and robust methods for analysis of dynamic behaviour of the system in the given conditions.
- Development of tools for control of the system and its simulation.

IV Selected topics

We will illustrate some of these advanced features by typical examples calculated using the application Agros2D [19] and library Hermes developed in our Department. The code is based on a fully adaptive higher-order finite element method and its application is intended for solving 2D multiphysics problems (3D version being prepared for testing).

A Automatic space adaptivity

The automatic space adaptivity serves for reduction of errors brought about by the numerical solution of the given problem. Its algorithms are applied at the moment when the local error of solution is higher than the acceptable tolerance. This error defined as the difference between the current numerical solution and exact solution may be caused by locally rougher mesh, presence of one or more singular points, curvilinear boundaries or interfaces approximated by polygonal lines, etc. Such errors must be identified in the course of computation and appropriate measures have to be taken for their reduction.

Unfortunately, the exact solution f is only known in very simple analytically solvable cases. Moreover, even when for various (mostly linear) classes of PDEs there exist methods of estimation of the error of solution, we have no tools for estimating it in case of a general nonlinear PDE. That is why we work with the reference solution f_{ref} instead, that is obtained on a globally refined mesh. By evaluation of the difference of both solutions we get the candidates for adaptivity even without knowledge of the exact solution f. Agros2D works with very sophisticated and subtle tools based on the above considerations.

If the error of solutions f and f_{ref} expressed by an appropriate norm is higher than a given threshold, the adaptation process is started. The calculated local error in the candidate is first weighted with respect to the way of adaptivity that should be used. This weight w is selected in the following way:

- $w = 2$ for the h-adaptivity,
- $w = 1$ for the p-adaptivity,
- $w = \sqrt{2}$ for the hp-adaptivity.

The value w is determined from the score s of the candidate. This core is given by the formula

$$s = \frac{\log_{10}\left(\frac{\varepsilon}{\varepsilon_0}\right)}{(d - d_0)^{\xi}}\,, \tag{1}$$

where ε is the estimated error in the candidate, d denotes its number of DOFs, ε_0 and d_0 are selected parameters and ξ stands for the convergence exponent.

The adaptive technique should be used mainly when very accurate results are required. On the other hand, it may substantially prolong the time of calculations.

Bidirectional comb drive microactuator

The device is intended for precise position control that is required in numerous scientific disciplines (optics, microscope techniques or microsurgery). This control may be realized on the basis of several physical principles (mechanical, hydraulic, pneumatic, thermoelastic, piezoelectric, electrostatic, etc.). In very small applications working with low energies and forces, the microactuators based on the electrostatic principles may represent an efficient and reliable solution. They are generally very fast and their operation requires only a relatively low voltage. Comb drives are, in fact, capacitive actuators working on the basis of electrostatic forces acting between two electrically conductive combs.

Electric field in the domain of such a microactuator (see Fig. 1) is described by the equation for the electric potential φ in the form

$$\mathrm{div}\,(\varepsilon\ \mathrm{grad}\ \varphi) = 0, \tag{2}$$

where symbol ε stands for the dielectric permittivity. The boundary conditions are given by the known values of the electric potential on the electrodes and the Neumann condition along the artificial boundary placed at a sufficiently distance from the device.

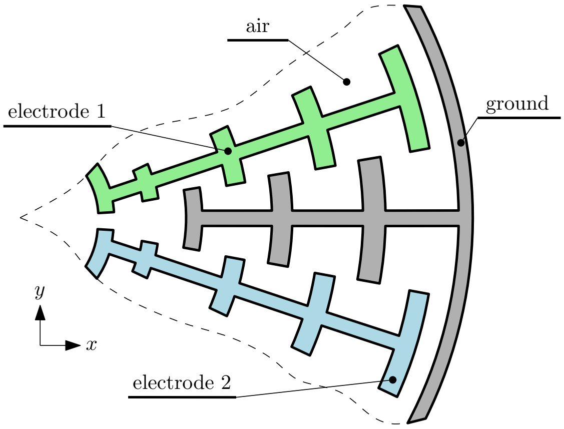

Figure 1: Scheme of the comb actuator

Figure 2 shows the discretization mesh after the adaptive process and Fig. 3 depicts the distribution of electric scalar potential. The evolution of the error of the potential distribution in the dependence on the number of DOFs is depicted in Fig. 4.

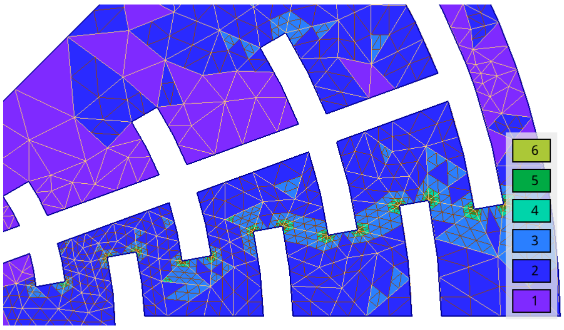

Figure 2: Mesh after adaptive process (numbers in rectangles showing orders of particular elements)

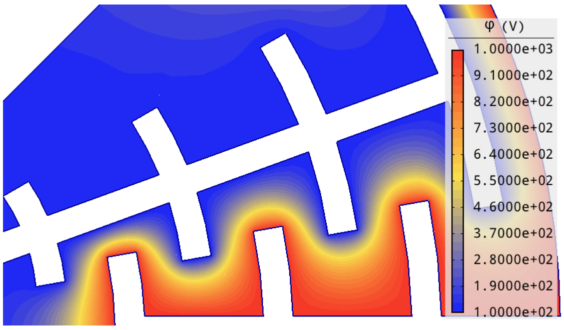

Figure 3: Distribution of scalar potential obtained on adapted mesh

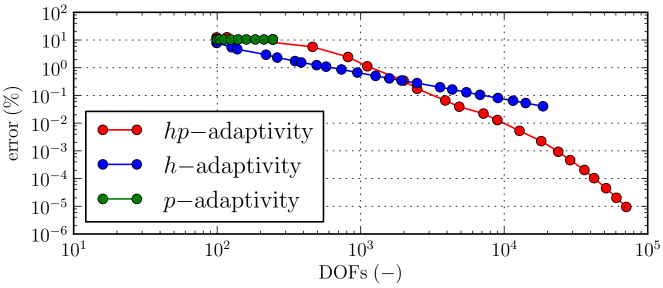

Figure 4: Convergence of results for h-, p- and hp-adaptivity (the adaptive process starting in all cases with $p = 1$)

Figure 5 shows the structure of the stiffness matrix for the computation without adaptivity and Fig. 6 depicts this structure after the third step of the adaptive process.

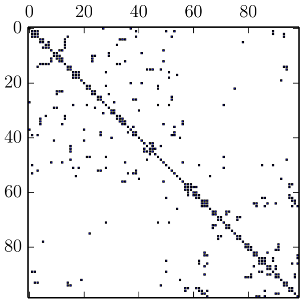

Figure 5: Structure of stiffness matrix without adaptivity

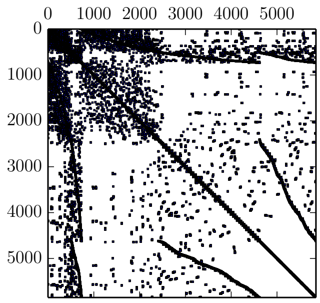

Figure 6: Structure of stiffness matrix after 3 adaptivity steps

B Dynamic meshes and multimesh

Dynamic meshes represent a very useful advanced tool for mapping time-dependent processes. The mesh (or meshes in case of several fields) change in each time step in the dependence on the actual solution in order to obtain the most accurate results in the next step.

Traditionally, many coupled problems have been solved using non-adaptive low-order numerical methods that approximated all fields on the same mesh. More recently, operator splitting (OS) schemes have been proposed to solve particular fields more efficiently using various meshes for individual field quantities. Unfortunately, OS schemes typically are non-adaptive, and, moreover, they are known to suffer from the loss of accuracy and/or stability caused by the transfer of data between meshes and/or by incomplete fixed point iteration. About five years ago, a novel technique was proposed that makes it possible to solve coupled problems monolithically (without operator splitting and all associated problems), using an adaptive higher-order finite element discretization based on individual meshes for the meshes involved.

The technology can also be modified for the time-dependent problems. In these cases we combine the classic Rothe method with the novel multimesh *hp*-FEM. From the viewpoint of applying spatially adaptive numerical schemes, the Rothe method is substantially better than the method of lines (MOL). The solved time-dependent PDE is in every time step approximated by one or several PDEs only with space variables. Now, these equations can be solved numerically using suitable adaptive techniques for suppressing eventual discretization errors. Moreover, no problem is even with selection of appropriate time steps that can be determined using common techniques for ordinary differential equations (ODEs). Agros2D performs the adaptive time integration by a couple of first-order backward-difference formulas whose combination provides a second-order scheme. The difference between both results then serves as a basis for estimation of the local error that is then used for adaptation of the corresponding time step.

Laminar flame in the channel

Consider a very simple flame propagation model (laminar flame, no fluid mechanics involved). The basic arrangement is shown in Fig. 7. The computational domain contains in the middle a narrow portion (cooling rods) whose purpose is to slow down the chemical reaction.

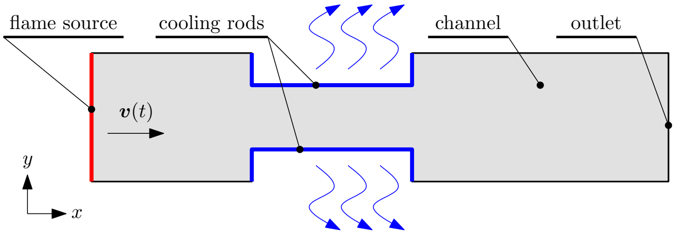

Figure 7: Scheme of the system for flame propagation

The equations for the temperature T and species concentration Y can be written in the form

$$\frac{\partial T}{\partial t} - \Delta T = \omega(T, Y), \tag{3}$$

and

$$\frac{\partial Y}{\partial t} - \frac{1}{\mathrm{Le}}\Delta Y = -\omega(T, Y). \tag{4}$$

The boundary conditions are of the Dirichlet type on the inlet, Newton type $\partial T/\partial n \;=\; -\kappa T$ on the cooling rods, and Neumann type $\partial T/\partial n \;=\; 0, \partial Y/\partial n \;=\; 0$ elsewhere. The objective of the computation is to obtain the reaction rate defined by the Arrhenius law in the form

$$\omega(T, Y) = \frac{\beta^2}{2\mathrm{Le}} Y e^{\frac{\beta(T-1)}{1+\alpha(T-1)}}. \tag{5}$$

Here, α is the gas expansion coefficient in a flow with nonconstant density, β denotes the activation energy, and Le stands for the Lewis number (ratio of diffusivity of heat and diffusivity of mass).

At the very beginning of the process, a very coarse master mesh τ_{m} is generated to cover the definition area of the problem. Solution of the $(n+1)$st time step starts from the input data represented by the values $T^n(x)$ and $Y^n(x)$ obtained from the solution of the previous step. These data were calculated on a refined mesh τ_n created automatically from τ in the nth time step. The unknowns $T^{n+1}(x)$ and $Y^{n+1}(x)$ to be found are solved adaptively starting from the same coarser mesh, but the resultant mesh τ_{n+1} generally differs from τ_n. Therefore, the meshes obtained at each time level are generally different, i.e., the mesh changes dynamically in time.

As the quantities T and Y exhibit various features, the solution pairs $T^n(x)$, $Y^n(x)$ and $T^{n+1}(x)$, $Y^{n+1}(x)$ must be calculated on different meshes. This is realized using an appropriate *hp*-technology. Here, the stiffness matrix is assembled on a virtual union mesh τ_{u} which is represented by a geometrical union of the meshes τ_n and τ_{n+1}. This is indicated in Fig. 8. Due to this fact, any transfer of information between the meshes is not necessary, which results in no additional error of calculation.

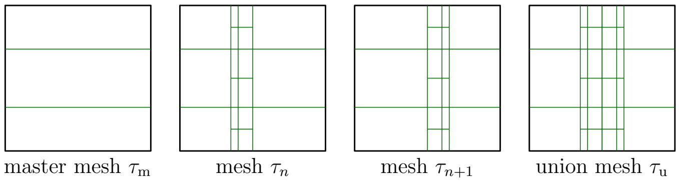
master mesh τ_{m}
mesh τ_n
mesh τ_{n+1}
union mesh τ_{u}

Figure 8: Example of master mesh τ_{m}, meshes τ_n, τ_{n+1} and union mesh τ_{u}

Agros2D uses another tool for speeding up the computations consisting in implementation of specific time step adaptation based on a PID controller.This tool is not able to control the time discretization error, but it accelerates the adaptation of the time step Δt so that a suitable indicator, for example

$$\epsilon_n = \frac{\|\omega(T^{n+1}, Y^{n+1}) - \omega(T^n, Y^n)\|_{H_1}}{\|\omega(T^{n+1}, Y^{n+1})\|_{H_1}} \tag{6}$$

is kept within prescribed bounds. If ϵ_n is too large ($\epsilon_n >$ tol.), the solution T_{n+1}; Y_{n+1} is discarded and recomputed using the relation

$$\Delta t_* = \frac{\mathrm{tol.}}{\epsilon_n}\Delta t_n. \tag{7}$$

Otherwise the time step is determined smoothly using the PID formula

$$\Delta t_{n+1} = \left(\frac{\epsilon_{n-1}}{\epsilon_n}\right)^{k_{\mathrm{P}}} \left(\frac{\mathrm{tol.}}{\epsilon_n}\right)^{k_{\mathrm{I}}} \left(\frac{\epsilon_{n-1}^2}{\epsilon_n \epsilon_{n-2}}\right)^{k_{\mathrm{D}}} \Delta t_n\,. \tag{8}$$

Here, the recommended values of the exponents are $k_{\mathrm{P}} = 0.075$, $k_{\mathrm{I}} = 0.175$ and $k_{\mathrm{D}} = 0.01$. The algorithm is very fast because the solutions are discarded only rarely. Moreover, no extra solutions are required for each time step as in other approaches.

As was proved by careful testing, the PID controller accelerated the total CPU time by about 40 %, due to the increased time step sizes when the solution only changes slowly (about $10\,\mathrm{s} < t < 45\,\mathrm{s}$).

The upper part of Fig. 9 shows the position of the flame after $t = 15.6\,\mathrm{s}$; its bottom part depicts the corresponding mesh. Figure 10 contains similar results for $t = 46.8\,\mathrm{s}$.

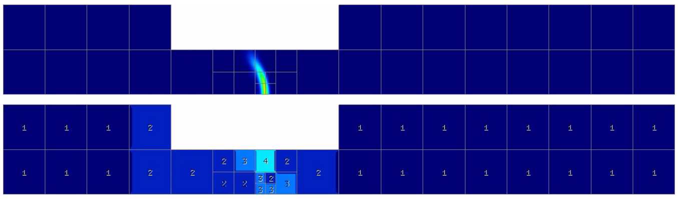

Figure 9: Flame propagation at time 15.6 s

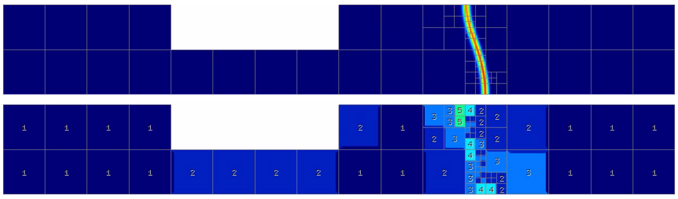

Figure 10: Flame propagation at time 46.8 s

Figure 11 shows the dependence of DOFs necessary for reaching the prescribed accuracy on time.

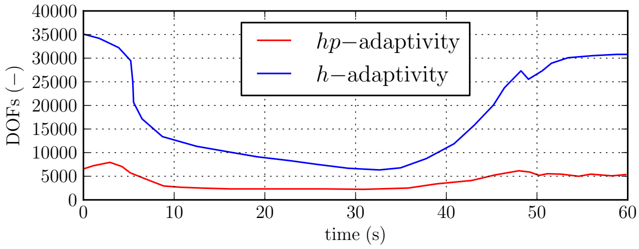

Figure 11: Time evolution of number of DOFs

Thermoelastic acturator

Thermoelastic deformations of metal bodies produced by induction heating are exploited in numerous industrial technologies (hot pressing, production of high forces etc.). But thermoelasticity may also be advantageous for producing small, but well controllable shifts. A schematic arrangement of the corresponding device is depicted in Fig. 12. A dilatation element made of appropriate metal is inserted into a current-carrying coil in a fixing ferromagnetic frame. The whole system is placed in the insulating shell. The device is clamped by its bottom part (insulating front) in the basement (ideally stiff wall). The time-variable magnetic field generated by the field coil induces in the dilatation element eddy currents that produce its heating and consequent geometrical changes of the thermoelastic origin.

Distribution of electromagnetic field in the system in the period of heating is described by the equation for magnetic vector potential $\boldsymbol{A}$

$$\mathrm{curl}\,\frac{1}{\mu}\,\mathrm{curl}\,\boldsymbol{A} + \gamma\cdot\frac{\partial \boldsymbol{A}}{\partial t} = \boldsymbol{J}_{\mathrm{ext}}\,, \qquad (9)$$

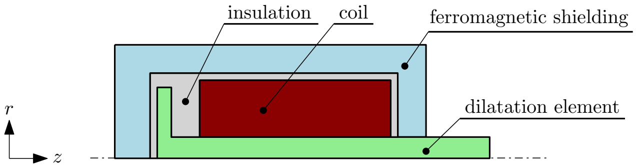

Figure 12: Basic arrangement of the device

where μ denotes the magnetic permeability, γ is the electric conductivity and $\boldsymbol{J}_{\mathrm{ext}}$ stands for the vector of external harmonic current density in the inductor.

The temperature field in the system is described by the heat transfer equation

$$\mathrm{div}\,\lambda(T)\,\mathrm{grad}\,T - \rho(T)\,c(T)\cdot\frac{\partial T}{\partial t} = -p_{\mathrm{J}}\,, \qquad (10)$$

where $\lambda(T)$ is the thermal conductivity, $\rho(T)$ denotes the mass density and symbol $c(T)$ stands for the specific heat (all of these parameters are generally temperature-dependent functions). Finally, p_{J} denotes the time average volume Joule losses due to eddy currents in electrically conductive parts.

The solution of the thermoelastic problem starts from the vector Lamé equation that reads

$$(\lambda+\mu)\,\mathrm{grad}\,\mathrm{div}\,\boldsymbol{u}+\mu\triangle\boldsymbol{u}-(3\lambda+2\mu)\,\alpha_{\mathrm{T}}\,\mathrm{grad}\,T = -\boldsymbol{f}\,, \qquad (11)$$

where λ and μ are coefficients associated with material parameters by relations

$$\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)}\,, \qquad \mu = \frac{E}{2(1+\nu)}\,.$$

Here E denotes the modulus of elasticity and ν the Poisson coefficient of the transverse contraction, α_{T} is the coefficient of the linear thermal dilatability of material and $\boldsymbol{f}$ the vector of the internal volume forces. Finally $\boldsymbol{u}$ represents the vector of resultant displacements.

Figure 13 shows the three independent meshes (at the initial stage) for computation of magnetic field, temperature field and field of thermoelastic displacements.

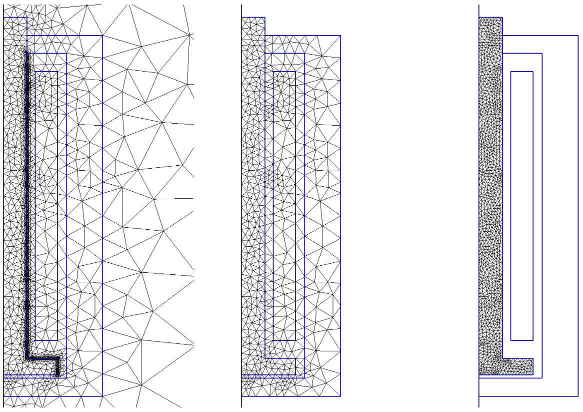

Figure 13: Used meshes (from left to right) for computation of magnetic field, heat transfer and thermoelasticity (multimesh technology)

Figure 14 shows (from left to right) the distribution of magnetic field, temperature field and field of thermoelastic displacements after 30 s of heating.

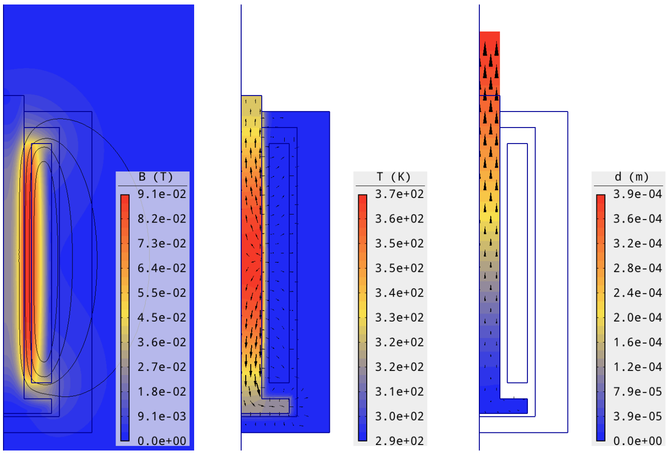

Figure 14: Solution at time $t = 30$ s (magnetic field, heat transfer and thermoelasticity)

C Hanging nodes

Hanging nodes are specific nodes lying on the abscissas of selected mesh elements that are characteristic by missing connections to one or more neighbouring vertices of the mesh. They are produced, for example, during the refinement of its selected part in the frame of an adaptation process. Usually, the hanging nodes bring about a considerable increase of the number of the degrees of freedom (DOFs). The code Agros2D, however, contains higher-order algorithms for respecting these nodes without any need of an additional refinement of the external parts neighbouring with the refined subdomain.

We can illustrate the generation of hanging nodes in the course of an adaptive process. For example, one of the most efficient algorithms for h-adaptivity on regular meshes is the red-green refinement strategy. In the course of this process, the corresponding elements are first divided into several subelements with hanging nodes. These hanging nodes are then removed by a forced refinement of relevant neighbour elements, see Fig. 15.

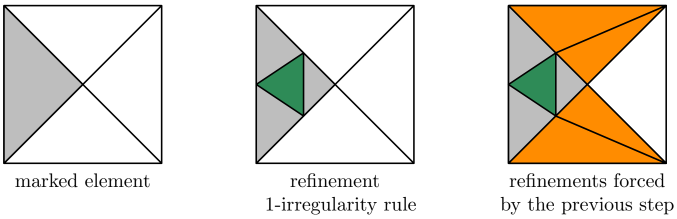

Figure 15: Red-green refinement

In this way, the regularity of the mesh is preserved, but new smaller elements produced during the process often have sharp angles which are not desirable in finite element analysis. The situation becomes worse and worse when repeated refinement is necessary, such as in the vicinity of singular points or a boundary layer.

Thus, the second step of the process — forced refinement — may bring about difficulties. These can be avoided by introducing hanging nodes, i.e., nodes lying in the interior of edges of other elements. But as the computer implementation of the relevant procedures is far from being easy, 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., [20] and [21].

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 $k = \inf$ to adaptivity with arbitrary-level hanging nodes. It is illustrated in Fig. 16 that even the 1-irregularity rule does not avoid all forced refinements.

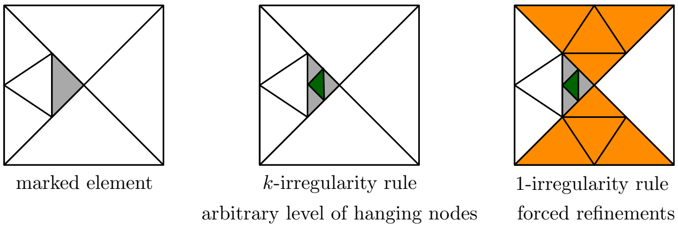

Figure 16: Refinement with k-irregularity rule

The amount of forced refinements in the mesh depends on the level of hanging nodes.

Deflection of a bracket

The use of hanging nodes is illustrated by the solution of a planar elastic problem—deflection of a bracket. The arrangement is depicted in Fig. 17 and the distribution of the elastic displacements obeys the vector equation

$$(\lambda + \mu) \text{ grad div } \boldsymbol{u} + \mu \triangle \boldsymbol{u} = -\boldsymbol{f} \, , \tag{12}$$

where λ and μ are coefficients connected with the material parameters by relations

$$\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)}, \quad \mu = \frac{E}{2(1+\nu)} \, . \tag{13}$$

Here E denotes the modulus of elasticity, ν is the Poisson coefficient of the transverse contraction, and $\boldsymbol{f}$ is the vector of the internal volume forces. Finally, $\boldsymbol{u}$ represents the vector of the displacements.

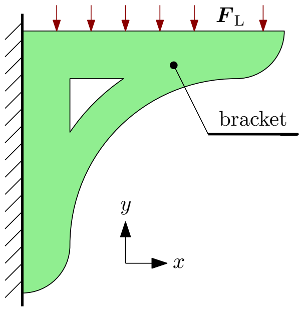

Figure 17: Arrangement of investigated beam

Figure 18 shows the discretization meshes for the computation of the x and y components of the displacements after the adaptive process. Both meshes contain the hanging nodes. Figure 19 shows the final distribution of the modules of displacement in the whole beam.

The numbers of elements necessary for reaching the error of both components of displacement (in directions x and y) $\eta = 0.0025\,\%$ are listed in Tab. 1.

D Curvilinear elements

Curvilinear elements are important when discretizing regions with curvilinear boundaries and interfaces. Use of such elements

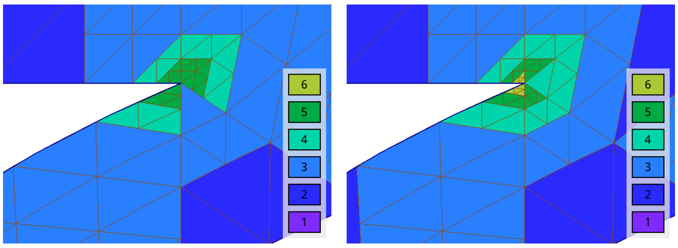

Figure 18: Meshes for x (left) and y (right) components of displacement

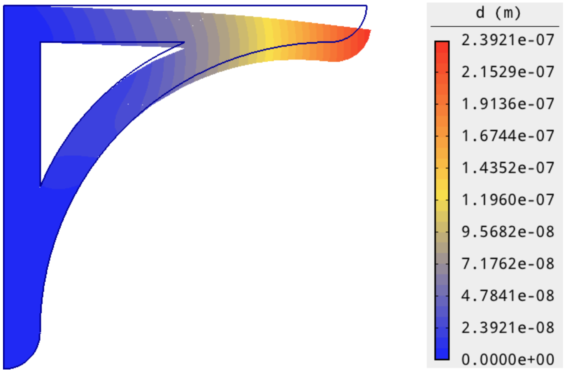

Figure 19: Distribution of module of displacement

	x component	y component
initial mesh	49	49
h-adaptivity	385	400
hp-adaptivity	10501	9802

Table 1: Number of elements after adaptivity process (expected error $\eta = 0.0025$ %)

can significantly save the number of DOFs, accelerate the computation of fields in the vicinity of curved edges and make the maps smoother.

As the technique is not widely known, it will be described in more detail. Agros2D discretises 2D domains on the base of SW Triangle that provides a high-quality triangular mesh. The corresponding input data for modelling curvilinear boundaries or interfaces in Triangle are given by a series of points lying on them (together with the markers carrying information that these points belong to such a line) while the output is represented by a set of triangular elements. In the second step Agros2D repeats analysing the curved lines and when any of the newly generated nodes approximating the curve does not lie on it, it is automatically projected on the original arc (circular, elliptic or more sophisticated, described, for example, by a NURBS curve). At the same time a special procedure determines the corresponding angles.

In the course of numerical processing of the task the curvilinear elements are mapped back on normal triangles where all remaining operations (such as Gaussian numerical integration) are carried out and only in the final step — post-processing — they are mapped again to the curvilinear elements.

Electric field near a high-voltage insulator for outdoor use

The insulator (see Fig. 20) is made of a ceramic material whose relative permittivity $\varepsilon_r = 6$. Knowledge of the electric field distribution along its surface is crucial for its design. Generally used threshold value for its electric field is 450 kV/m. The arrangement is considered axi-symmetric (in fact, it is 3D due to the source electrode, which is a long conductor) and the environment of the insulator is air.

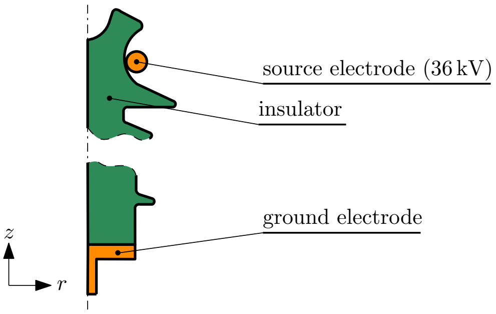

Figure 20: Arrangement of the insulator

Electric field in the domain is described by the equation for the electric potential φ in the form

$$\mathrm{div}\,(\varepsilon\ \mathrm{grad}\ \varphi) = 0, \tag{14}$$

where symbol ε stands for the dielectric permittivity. The boundary conditions are given by the known values of the electric potential on the electrodes and the Neumann condition along the artificial boundary placed at a sufficiently distance from the insulator.

To obtain sufficiently accurate results, several adaptive steps had to be realized for getting an appropriate final mesh.

The accuracy of the results was tested on the value of the total electric field energy W_e in the definition area and also relative error ε. Figure 21 shows the convergence curves obtained using several codes and ways of adaptation.

Fig. 22 depicts the dependence of the relative error on the number of DOFs for different adaptive techniques and Fig. 23 shows an analogous dependence for the normal and curvilinear elements.

Induction heating in rotation

Rotational heating of nonmagnetic (mostly aluminium) billets in uniform magnetic field is often used for their softening before subsequent hot forming. Its efficiency is substantially higher than in the case of the classical heating by a static inductor. A cylindrical billet rotates in a uniform magnetic field produced by static, direct currents carrying field coils. The basic arrangement of the system is depicted in Fig. 24.

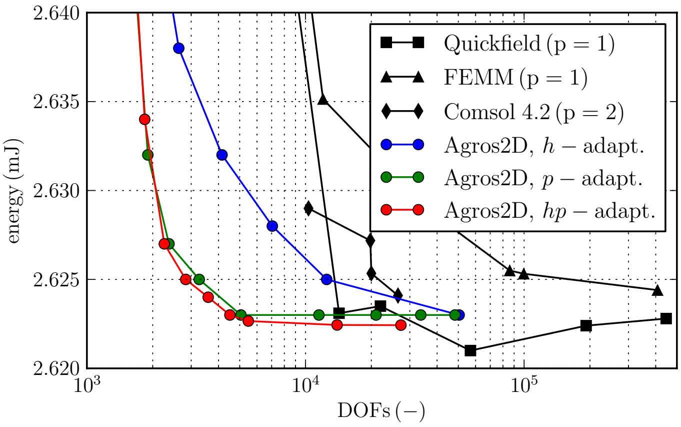

Figure 21: Convergence curves of total electrostatic energy W_e

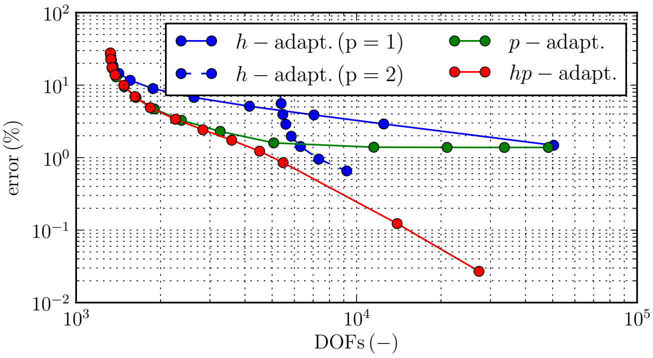

Figure 22: Error of computation dependent on number of DOFs for different adaptations

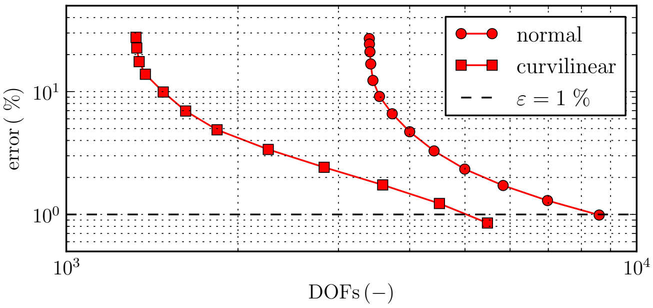

Figure 23: Error of computation dependent on number of DOFs for different types of elements (hp-adaptivity)

The continuous mathematical model of heating consists of two nonlinear partial differential equations whose solution provides the distribution of static magnetic field and evolutionary temperature field in the system. The distribution of magnetic field is described in terms of magnetic vector potential $\boldsymbol{A}$ by the equation

$$\mathrm{curl}\,\left(\frac{1}{\mu}\,\mathrm{curl}\,\boldsymbol{A}\right) - \gamma\,(\boldsymbol{v} \times \mathrm{curl}\,\boldsymbol{A}) = \boldsymbol{J}_{\mathrm{ext}}\,, \qquad (15)$$

where μ stands for the magnetic permeability, γ is the electrical conductivity, $\boldsymbol{v}$ is the local velocity of movement at a point and $\boldsymbol{J}_{\mathrm{ext}}$ is external current density. A sufficiently distant artificial boundary is characterized by the Dirichlet condition $\boldsymbol{A} = \boldsymbol{0}$. In this example we do not need the temperature equation.

Distribution of the module of magnetic flux density in the system is shown in Figures 25 and 26. Its upper part depicts the distribution for classic triangular elements (3305 DOFs), white the bottom part shows the analogous distribution obtained using a combination of classic triangular and curvilinear elements (2849 DOFs). It is obvious that the latter distribution is smoother, even when the number of DOFs is here substantially smaller (by about 15 %).

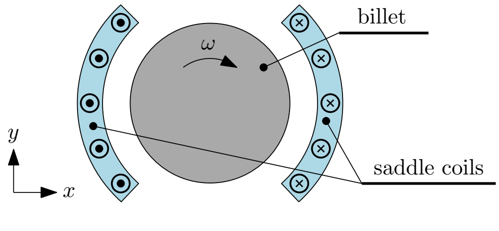

Figure 24: Induction heating of billet rotating in static magnetic field

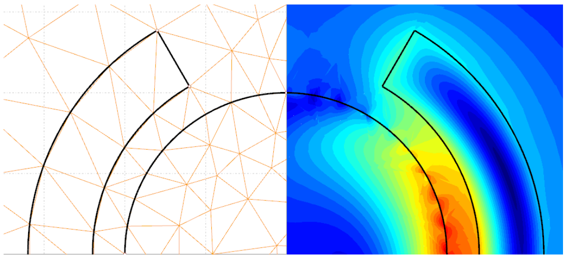

Figure 25: Magnetic flux density for classic triangular mesh (up, 3305 DOFs)

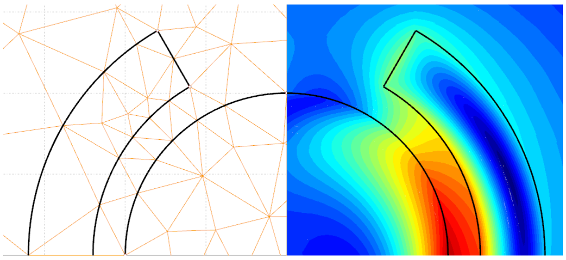

Figure 26: Magnetic flux density for a combination of classic and curvilinear triangular elements (bottom, 2849 DOFs)

E Time-adaptive methods for transient problems

These efficient and accurate methods are highly useful in solution of evolutionary partial differential equation. A lot of interest is usually paid to the spatial discretization and to minimizing its error at still reasonable computational cost. It is equally important to minimize the number of time steps, since the total computational time is proportional to it. Using the basic implicit or explicit Euler method is here simply insufficient.

The usual strategy of solving transient problems is to discretize the equation in space, solve it and, hence, obtain a system of ordinary differential equations in time, which can then be handled separately using standard tools for their solution.

If the mesh is different in each time step due to possible space adaptivity (dynamic meshes), time discretization must be performed first. In this case, the Rothe method can be used. One way how to discretize the time derivative is to use the backward differential formula (BDF). For ordinary differential equations

$$\frac{\partial y}{\partial t} = F(y,t), \quad y(t_0) = y_0\,, \qquad (16)$$

the n-step BDF is given by the expression

$$\sum_{l=0}^{n} \alpha_{n,l}\, y_{k-l} = \tau_k F(y_k)\,, \qquad (17)$$

where t_k is the kth time level, $y_k = y(t_k)$ denotes the corresponding solutions, $\tau_k = t_k - t_{k-1}$ is the length of the kth time step length and coefficients $\alpha_{n,l}$ depend on the lengths of previous steps and, thus, allow the time step to be changed. After the time discretization is done using this formula, the space discretization can be performed in the same way as it would be done for a steady state problem. The only difference is the necessity of providing the values of solutions from n previous time steps projected to the current mesh.

Using higher-order time discretization itself can bring a significant acceleration of computations, but it can further be used for developing an adaptive time-step method. Solving the problem using two different orders can bring an estimate of the error. It can then be used for determining the length of the next time step (that can be enlarged or shortened) in order to keep the error as close to the prescribed tolerance as possible and thus obtain solution with desired accuracy using as few time steps as possible. Obviously, the question of choosing the right value of the tolerance and finding the relation for the errors caused by spatial and time discretizations is very complicated and is far beyond the scope of this article.

Tin melting

The time-adaptive method was also used for modelling of induction melting of tin. Tin in a ceramic crucible is heated by a flat inductor below it (see Fig. 27).

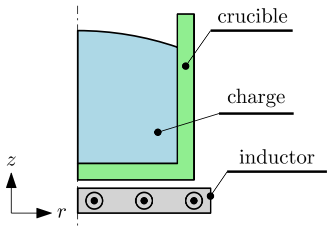

Figure 27: Basic arrangement of tin melting

The use of a time-adaptive method will be illustrated by the heat-transfer equation

$$\mathrm{div}\ (\lambda\ \mathrm{grad}\ T) - \rho c_\mathrm{p}\frac{\partial T}{\partial t} = -Q\,, \tag{18}$$

in case of melting of tin by induction heating. Here, λ denotes the thermal conductivity, ρc_p is the specific heat capacity and Q stands for the volumetric Joule losses. All material parameters exhibit strong nonlinearities near the temperature of melting (505 K). For the subsequent analysis, it is crucially important to reach a sufficiently high accuracy of results in the period of melting the material. With respect to a relatively long time of melting, the use of standard procedures requires a very short time step. But this does not hold for the adaptive methods.

Figure 29 shows the time evolution of the average temperature of molten material. It is obvious that the use of an adaptive solution of the transient leads to an automatic decrease of the time step just in the period of the phase change. But except for this period (and several starting steps) the computations are realized with substantially longer time steps, which reduces the total computing time. Figure 28 depicts the lengths of particular time steps in the course of computation.

F Computation of forces in strongly non-linear magnetic fields

The evaluation of forces and torques acting on ferromagnetic elements in non-linear magnetic fields is still a challenge. The algorithms based on the virtual work and Maxwell stress tensor are often slow, the convergence of computations is rather poor and accuracy of the results is far from being high. Lower-order Eggshell methods [22] are generally better, but sometimes they suffer from the same reasons. Agros2D contain a higher-order modification of this method that seems to be more efficient.

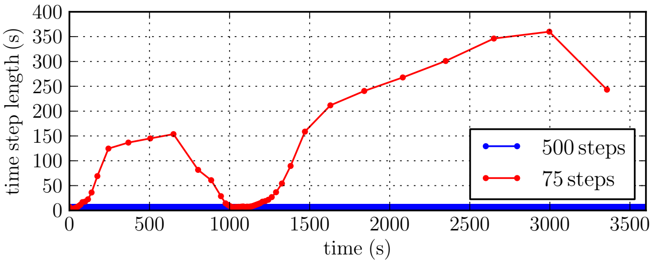

Figure 28: Time evolution of time step length

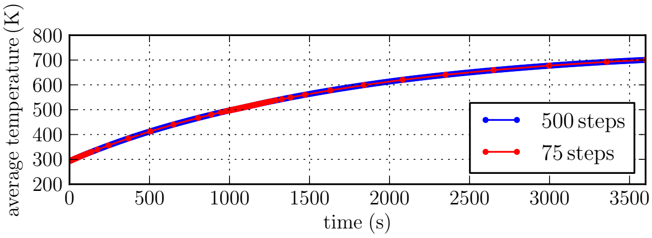

Figure 29: Average temperature of heated tin

The Maxwell stress tensor whose knowledge is crucial for using the above method, requires mapping the field vectors $\boldsymbol{B}$ and $\boldsymbol{H}$ along the surface of the investigated body. But computation of magnetic field in highly nonlinear systems using the classic Newton method is often slow and accelerating algorithms have to be developed for this purpose.

Static characteristic of an actuator

The basic structural arrangement of the solved actuator is depicted in Fig. 30. The actuator consists of a DC field coil (consisting of two parts connected in series, the total number of turns being 520) placed in magnetic circuit. The hollow pipe-like plunger moves along a sliding cylindrical plastic rod in its longitudinal axis.

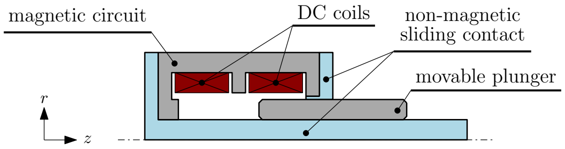

Figure 30: Basic arrangement of actuator (position of plunger is given by distance δ from front of the actuator)

For the analysis of behaviour of the actuator it is important to accurately know its static characteristic. The classic procedure used for its computation is based on the Newton method representing an iterative process when every following step follows from a correction of the previous step according to the formula

$$\boldsymbol{Y}^{n+1} = \boldsymbol{Y}^{n} + \boldsymbol{D}^{n+1}\,, \tag{19}$$

where $\boldsymbol{Y}^{n+1}$ denotes the result in the $n+1$th step and $\boldsymbol{Y}^{n}$ is the same value in nth step. Symbol $\boldsymbol{D}^{n+1}$ is the correction that is obtained from the solution of a system of nonlinear algebraic equations with the Jacobi matrix. In this way, after satisfying the convergence condition given by the relative difference ε between two following steps we obtain an acceptable approximation of the quantity $\boldsymbol{Y}$.

Application of this method requires high computational demands that can be crucial in case of solving practical problems incorporating parametric analyses, sensitivity analyses or optimizations, and even may make their solutions infeasible. For this reason, it is more advantageous to use various modifications of this method that allow performing the computations more safely with respect to the convergence of the solution, and also substantially faster. Out of these modifications we can mention utilization of an automatic selection of the damping factor and reuse of the Jacobian.

Mainly the automatic selection of the damping factor represents the modification securing a high level of the convergence of solution and its acceleration. This modification is realized by an artificial reduction of the correction term $\boldsymbol{D}^{n+1}$ in every step of the process.

This method is demonstrated in Fig. 31, containing the convergence curves of solution for the case of field current $I_{\mathrm{ext}} = 8\,\mathrm{A}$ and full insertion of the plunger into the magnetic circuit. The curves represent the evolutions of relative changes ε of the solution in the dependence on the step of the Newton method for the selected damping factor F. The graph only contains the curves leading to the solution of the task for the relative change of solution $\varepsilon < 10^{-2}$ and it is obvious that (in the given implementation) the problem can be solved only with the damping factor $F < 0.6$. Finally, Fig. 32 shows the corresponding damping factor in the particular steps of the solver.

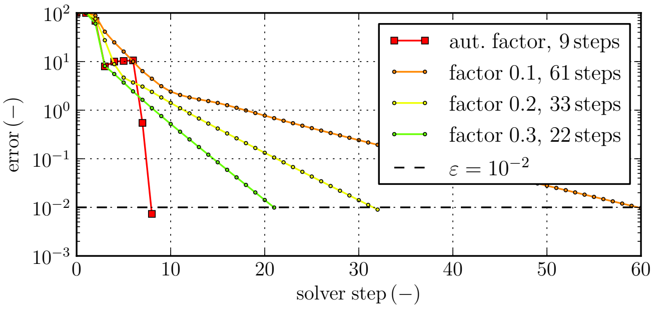

Figure 31: Convergence of the nonlinear solver

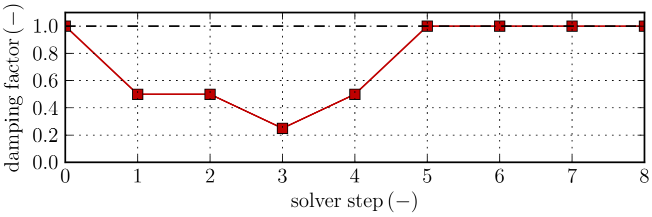

Figure 32: Value of damping factor in particular steps of solver

The automatic selection of the damping factor is of a very high importance for the computation of the static characteristic of the actuator requiring to repeat a series of nonlinear calculations. In the presented case, the total time of computations with the automatic selection of the damping factor is 203 s, while with its constant value it reaches 492 s.

An appropriate algorithm must be selected even for the computation of the force acting on the plunger. Calculation of the force acting on the body using the Eggshell method starts from the formula

$$\boldsymbol{F}_{\mathrm{Te}} = \int_{\mathrm{V}} \sigma_{\mathrm{M}} \cdot \operatorname{grad} \gamma \,\mathrm{d}V\,, \tag{20}$$

where V is the volume of a thin shell around the body and γ is a function satisfying the conditions $\gamma = 0$ along the external boundary of the shell and $\gamma = 1$ along its internal boundary.

Comparison of the third-order Eggshell method and method based on virtual work is, together with the measured values, depicted in Fig. 33. Calculation based on the method of virtual work exhibits considerable inaccuracies. To present a complete information, Fig. 34 depicts the dynamic characteristics of the actuator calculated from the equation of motion of the plunger.

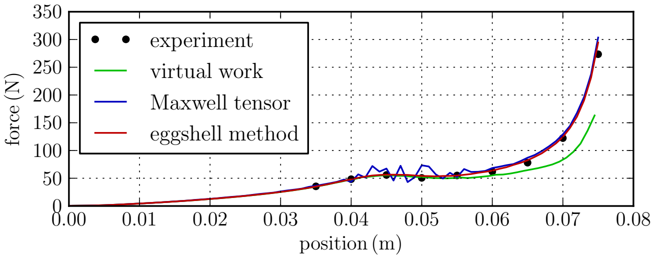

Figure 33: Comparison of calculated and measured distribution of force

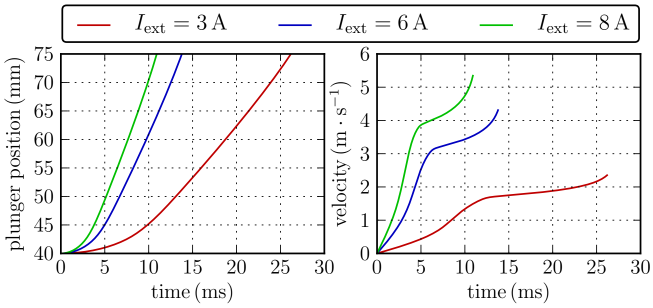

Figure 34: Time dependence of the position (left) and velocity (right) of the plunger

G Multicriteria shape optimization

For basic design of the actuator we performed the sensitivity analysis of selected geometrical dimensions. It showed that a very important parameter is the air gap between the plunger and central ring of the magnetic circuit. This knowledge was used for subsequent optimization of the actuator dimensions.

The goal of the optimization was to obtain as flat static characteristic of the device as possible and also the maximum average value of the force. This should be reached by changing the thickness of the air gap. The requirement of the flat characteristic can be described by the maximization of the value F and the minimization of the functional R by the expressions

$$F = \frac{1}{n}\sum_{i}^{n} F_i\,; \quad R = \sqrt{\sum_{i}^{n} (F_i - F)^2}\,. \tag{21}$$

First, the external edge of the hollow movable plunger was divided by a number of control points, whose shift with respect to the position representing the initial shape of the plunger represents the vector of the optimized parameters. Such a task leads to a multicriteria optimization.

The optimization itself was realized using a specific implementation of the genetic algorithms. The process of optimization is depicted in Fig. 35.

The results of the optimization are shown in Fig. 36, containing the static characteristics for all variants located along the Pareto front of the final population. The curves are grouped into three categories with respect to the values of the functional R. In the first graph we can see the variants satisfying the condition $R < 8$, while in the following graphs $8 < R < 12$ and $R > 12$. From the graphs also and from their comparison with the original static characteristic in Fig. 33 it is obvious that the optimization led to a considerable reduction of ripple of force acting on the plunger.

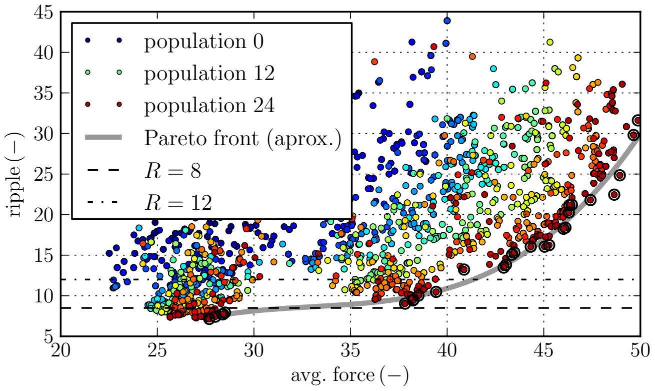

Figure 35: Process of optimization depicted by all results in the selected populations

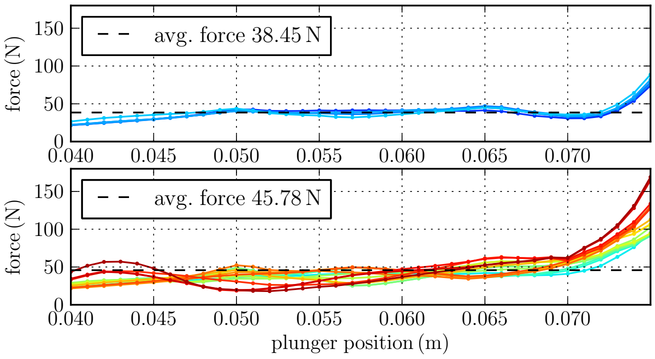

Figure 36: Static characteristics for resultant variants lying on the Pareto front

H Combination of PDE and ODEs

Nowadays, the distribution of numerous physical fields cannot be solved independently. Very often the corresponding PDEs must be solved together with ODEs describing the behaviour of various circuits.

The last example shows the cooperation between electrostatic field and moving charged particles in a triboelectric separator.

Electrostatic separator

This separator is intended for highly efficient separation of particular kinds of plastics. Its function is based on the triboelectric effect: when electrically non-conductive plastic particles of two different levels come into contact with electric charge, one of them becomes more positive (or negative) with respect to the other. Consequently, their trajectories in electric field (affected by the charge they carry) may be quite different. This is clear from Fig. 37 showing a scheme of the analysed device.

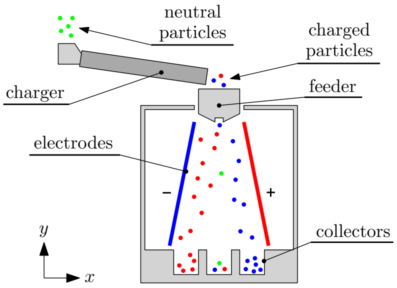

Figure 37: Basic arrangement of triboelectric separator

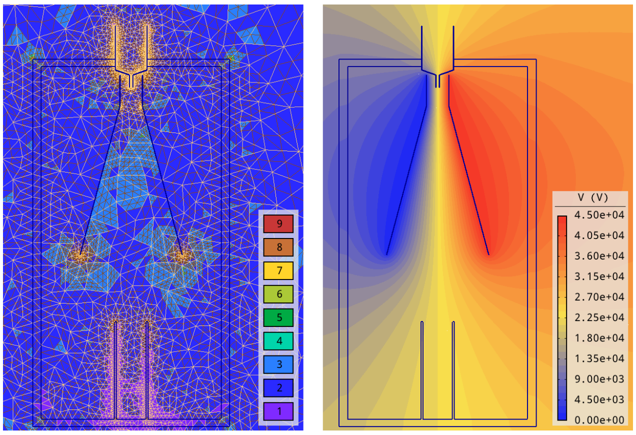

Figure 38: Mesh after adaptive process (left) and distribution of potential in the chamber (right)

Each of two aluminium electrodes (one of them being grounded) is divided to several segments in order to have a possibility of optimizing the distribution of electric field. The voltage between the electrodes is also variable, within the range of 10–50 kV. The charger consists of a high-speed induction motor equipped with a frequency converter and a polypropylene pipe. The induction motor drives the pipe by means of a plastic belt. At the bottom of the device there are several recycle bins used for collecting particular kinds of plastics.

The goal was to map the trajectories of particular plastic particles and evaluate the efficiency of the process of separation. The operation of the separator was modelled as a coupled evolutionary PDE-ODE problem using numerous advanced techniques described above. The distribution of electric field between the electrodes was determined using a fully adaptive higher-order finite element method while the movement of the particles was modelled by an adaptive Runge-Kutta-Fehlberg method with the adaptive time step.

The distribution of electric field $\boldsymbol{E}$ within the separator (due to the voltage applied to the electrodes) obeys the equation

$$\operatorname{div}\,(\varepsilon\,\operatorname{grad}\,\varphi) = 0\ , \tag{22}$$

where ε denotes the relative permittivity and φ is the electric potential. The trajectory of a particle with index i is then described by the equations

$$m_i \frac{\mathrm{d}\boldsymbol{v}_i}{\mathrm{d}t} = \boldsymbol{F}_{\mathrm{e}i} + \boldsymbol{F}_{\mathrm{t}ij} + \boldsymbol{F}_{\mathrm{g}i} + \boldsymbol{F}_{\mathrm{a}i}\ ,\ \boldsymbol{v}_i = \frac{\mathrm{d}\boldsymbol{s}_i}{\mathrm{d}t}\ , \tag{23}$$

where

$$\boldsymbol{F}_{\mathrm{e}i} = Q\boldsymbol{E} = -Q\cdot\nabla\varphi\ ,\ \boldsymbol{F}_{\mathrm{t}ij} = \Sigma_{j=1,j\neq i}^{j=n}\boldsymbol{F}_{ij}\ ,$$

$$\boldsymbol{F}_{\mathrm{g}} = m\boldsymbol{g}\ ,\ \boldsymbol{F}_{\mathrm{a}} = -\boldsymbol{v}\frac{1}{2}\rho c S v\ . \tag{24}$$

Here, m stands for the mass of the particle, Q is its charge, $\boldsymbol{v}$ denotes its velocity, $\boldsymbol{s}$ is the trajectory, ρ represents the mass density of air, c is the coefficient of friction, S denotes the cross section of the particle, n is the number of the particles and, finally, $\boldsymbol{F}_{\mathrm{e}i}$ is the force exerted on the particle by the external electric field, $\boldsymbol{F}_{\mathrm{t}ij}$ denotes the forces exerted on the ith particle by other moving particles, $\boldsymbol{F}_{\mathrm{g}i}$ stands for the gravitational force and $\boldsymbol{F}_{\mathrm{a}i}$ denotes the drag aerodynamic force.

The adapted mesh and distribution of electric field between the electrodes is in Fig. 38. Figure 39 shows trajectories of selected particles.

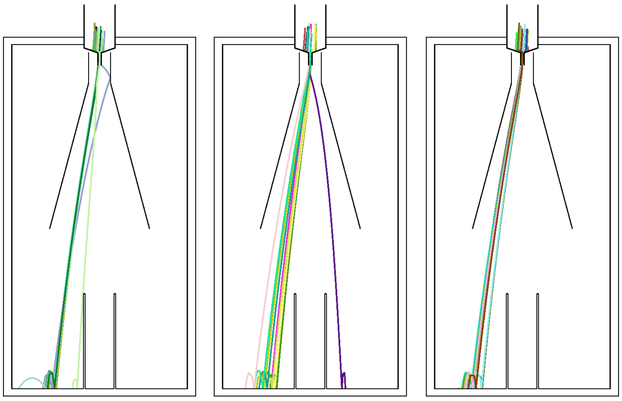

Figure 39: Trajectories of selected charged particles

The computation process is illustrated by several following figures. First, we mapped the particles of particular trajectories for non-adaptive and fully adaptive solutions. The trajectories of particles for the non-adaptive solution and *hp*-adaptive solution are depicted in Fig. 40. These trajectories were calculated for different charges Q of particles. Generally, the more negative is the charge, the more distinct is the deflection of the trajectory. Further decrease of charge Q would result in the impact of the particle with the electrode.

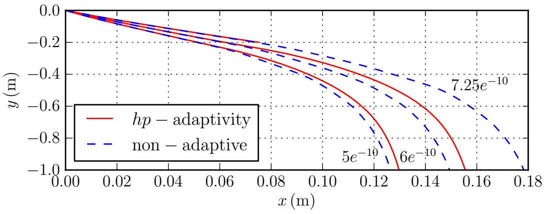

Figure 40: Trajectories of particles for different charges for non-adaptive (dashed line) and adaptive (full line) solutions

V Acknowledgment

This work was financially supported by the project GACR P102/11/0498 (Grant Agency of the Czech Republic).

VI References

[1] deal II: http://www.dealii.org/.

[2] Fenics: http://fenicsproject.org/.

[3] FEMM: http://www.femm.info/.

[4] FreeFem++: http://www.freefem.org/ff++/.

[5] MOOSE: http://moose.ncbs.res.in/.

[6] Agros2D: http://www.agros2d.org/.

[7] Solin, P., Segeth, K., Dolezel, I.: Higher-order finite element methods. CRC Press, Boca Raton, FL (2003).

[8] Babuska, I., Suri, M.: The *p*- and *hp*-versions of the finite element method, basic principles and properties. SIAM Rev. 36 (4), pp. 578–632 (1994).

[9] Castillo, L.E.G., Zubiaur, D.P., Demkowicz, L.F.: Fully automatic *hp*-adaptivity for electromagnetics, application to the analysis of H-plane and E-plane rectangular waveguide discontinuities. Proc. Microwave Symposium 2007. IEEE/MTT-S International, Honolulu, June 2007, pp. 285–288.

[10] Ainsworth, M., Senior, B.: An adaptive refinement strategy for *hp*-finite element computations. Appl. Numer. Math. 26 (1–2), pp. 165–178 (1997). (1997), no. 1-2, 165–178.

[11] Li, R.: On multi-mesh *h*-adaptive methods. J. Sci. Comput 24 (3), pp. 321–341 (2005).

[12] Solin, P., Cerveny, J., Dubcova, L., Andrs, D.: Monolithic discretization of linear thermoelasticity problems via adaptive multimesh *hp*-FEM. J. Comput. Appl. Math. 234, pp. 2350–2357 (2010).

[13] Schmich, M., Vexler, B.: Adaptivity with dynamic meshes for space-time finite element discretizations of parabolic equations. SIAM J. Sci. Comput. 30 (1), pp. 369–393 (2008).

[14] Yang, G.W., Zheng, G.N., Nie, X.Y.: Fast dynamic mesh method and its application on aeroelasticity. Procedia Engineering 67, pp. 438–446 (2013).

[15] Solin, P., Cerveny, J., Dolezel, I.: Arbitrary-level hanging nodes and automatic adaptivity in the *hp*-FEM. Math. Comput. Simul. 77, pp 117–132 (2008).

[16] Schröder, A.: Constrained approximation in *hp*-FEM: unsymmetric subdivisions and multi-level hanging nodes. Springer, Lecture Notes in Computational Science and Engineering 76, pp 317–325 (2010).

[17] Chinesta, F., Ladeveze, P., Cueto, E.: A short review on model order reduction based on proper generalized decomposition. Archives of Computational Methods in Engineering 18 (4), pp 395–404 (2011).

[18] Henneron, T.,Clénet, S.: Model order reduction applied to the numerical study of electrical motor based on POD method taking into account rotation movement. International Journal of Numerical Modelling: Electronic Networks, Devices and Fields 27 (3), pp 485–494 (2014).

[19] Karban, P., Mach, F., Kus, P., Panek, D., Dolezel, I.: Numerical solution of coupled problems using code Agros2D. Computing 95 (1), Supplement, pp. 381–408 (2013).

[20] Paszynski, L., Kurtz, J., Demkowicz, L.: Parallel, fully automatic *hp*-adaptive 2D finite element package. TICAM Report 04–07, The University of Texas at Austin (2004).

[21] Rachowicz, W., Demkowicz, L.: An *hp*-adaptive finite element method for electromagnetics. Part II: A 3D implementation. Int. J. Numer. Methods Engrg. 53, pp. 147–180 (2002).

[22] Henrotte, F., Felden, M., van der Giet, M., Hameyer, K.: The eggshell approach for the computation of electromagnetic forces in 2D and 3D. COMPEL 23 (4), pp. 996–1005 (2004).

Authors name and Affiliation

Pavel Karban, Frantisek Mach, Ivo Dolezel, University of West Bohemia, Pilsen, Faculty of Electrical Engineering, Department of Theory of Electrical Engineering, Univerzitni 26, 306 14 Pilsen, Czech Republic.

`karban, fmach, idolezel@kte.zcu.cz`