RUSSIAN ACADEMY OF SCIENCE

KELDYSH INSTITUTE OF APPLIED MATHEMATICS

I. L. Sofronov, N. A. Zaitsev

INFLUENCE OF DEFECTS ON ELECTROMAGNETIC FIELDS IN THE PROBLEM OF DIAGNOSTIC OF FERROMAGNETIC PIPES

Moscow, 2006

N. Zaitsev and I. Sofronov. *Influence of defects on electromagnetic fields in the problem of diagnostic of ferromagnetic pipes*

Abstract. We describe further modifications of the multigrid solver for 3D frequency domain Maxwell equations in cylindrical geometry proposed in previous preprint of the Keldysh Inst.Appl. Maths., No. 128, 2005. Here we incorporate an approach of embedded grids and introduce real arithmetic in operations with the main matrix. Both modifications enhance about 3-5 times the capability of the solver (1.5x due to real arithmetic and 2x-4x due to embedded grids). Smaller details of geometry can be also resolved now on PC owing to safe of memory.

Н.А. Зайцев, И.Л. Софронов. *Расчет влияния дефектов сложной структуры на электромагнитное поле в задаче диагностики ферромагнитных труб*

Аннотация. Представлена дальнейшая модификация многосеточного алгоритма решения трехмерных стационарных уравнений Максвелла в цилиндрической системе координат, предложенного в препринте ИПМ им. М.В.Келдыша №128, 2005. Мы описываем подход вложенных сеток для локального уточнения решения. Кроме того, вычисления для основной матрицы сведены к реальной арифметике. Обе модификации ускоряют решение задачи примерно в 3-5 раз (2-4 раза благодаря вложенным сеткам и 1.5 раза из-за реальной арифметики). Также, благодаря экономии памяти, появилась возможность моделировать на персональном компьютере более мелкие детали геометрии задачи.

This work was supported by RFBR grant № 04-01-00567

Introduction: short description of the 3D algorithm

In [1] we have proposed a numerical method of calculating harmonic three-dimensional electromagnetic field in the metallic pipes with local defects. The goal of the present paper is to describe the modifications of the algorithm which allows computing fields around relatively small or complex defects.

In this section we give short description of the initial algorithm.

We consider one or several coaxial cylindrical pipes that can have defects like pits or holes. For the simplicity we will consider the case of a single pipe, although the algorithm permits to compute the case of several coaxial pipes. The cylindrical system of coordinates with the same axes z as the pipe(s) is used. It is supposed that the defects can be approximated by a set of hexahedral cells.

In order to avoid solution of a full external problem we subdivide it onto two stages. First the unperturbed two-dimensional field in the domain

$$z_{-\infty} < z < z_{+\infty},$$
$$0 < r < R_{\infty}$$

is calculated for the clean pipes (without defects) or for the pipes with 2D defects (which do not depend on the azimuth angle) with a given geometry and excitation conditions. Then a smaller three-dimensional subdomain (box)

$$z_{-\infty} < z_{\min} < z < z_{\max} < z_{+\infty},$$
$$R_{\min} < r < R_{\max} < R_{\infty}$$

with the given defect is considered, and the three-dimensional problem for Maxwell equations with Dirichlet boundary conditions of the electrical field on the external boundaries is solved, see Figure 1 and Figure 2.

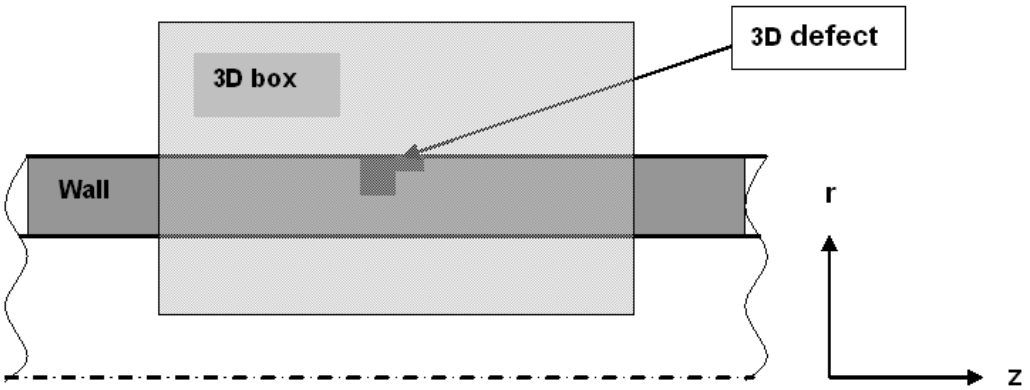


Figure 1. Computational domain, cross-section at θ=const

The two-dimensional unperturbed problem is solved by the algorithm described in [2]; so we have the solution for the azimuth component A_θ^{2D} of the vector potential

of an axis-symmetric background solution. For the three-dimensional problem the electrical field $\mathbf{E}$ is represented in the box by sum of vector and scalar potentials

$$\mathbf{E} = \mathbf{A} + \operatorname{grad}\varphi, \quad \mathbf{A} \equiv (A_r, A_\theta, A_z)$$

with the boundary values

$$A_r = A_z = 0, \;\; A_\theta = A_\theta^{2D}, \;\; \varphi = 0 .$$

It is supposed that the box is big enough to believe that the Dirichlet conditions from unperturbed problem can provide a required accuracy of modeling.

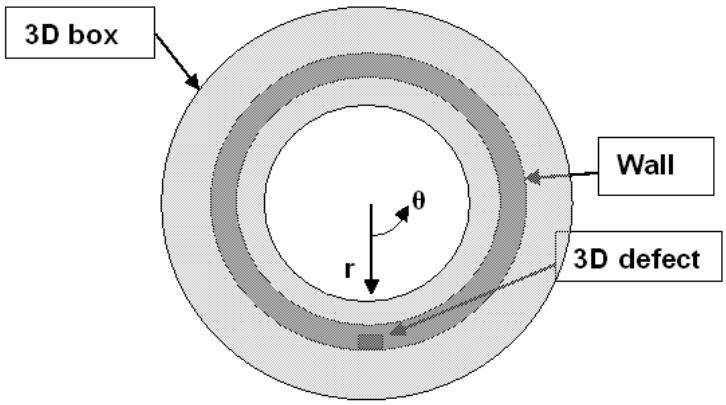


Figure 2. Computational domain, cross-section at z=const

The three-dimensional problem in the box is solved by the multigrid iteration method starting from the two-dimensional solution as an initial guess.

In order to overcome the known difficulties with kernel of the Maxwell equations, see, for example, [3], we use the following modified Maxwell equations written for the fixed frequency ($\tilde{\mathbf{E}}(\mathbf{r},t) = \mathbf{E}(\mathbf{r})e^{-i\omega t}$, $\tilde{\mathbf{H}}(\mathbf{r},t) = \mathbf{H}(\mathbf{r})e^{-i\omega t}$) in media with a non-constant magnetic permeability μ , conductivity σ , and electrical permittivity ε :

$$\operatorname{rot}\mathbf{H} - \operatorname{grad}\psi - \hat{\mathbf{J}} = \mathbf{J}^S; \quad \hat{\sigma}\left(\mathbf{A} + \operatorname{grad}\varphi\right) - \hat{\mathbf{J}} = 0; \quad \operatorname{rot}\mathbf{A} - i\omega\mu\mathbf{H} = 0;$$

$$\mathbf{E} = \mathbf{A} + \operatorname{grad}\varphi; \qquad \operatorname{div}\hat{\mathbf{J}} = -\operatorname{div}\mathbf{J}^S; \qquad \operatorname{div}\mathbf{A} - i\Omega\mu\psi = 0,$$

where $\mathbf{J}^S$ is a given source current density, ψ is an auxiliary function, and Ω is an auxiliary constant, see details in the cited paper.

This system is not the final one. It is possible to reduce the system to the second-order equations with 4 unknown scalar functions: three components of the vector potential $\mathbf{A}$ and the scalar potential φ:

$$\operatorname{rot}\frac{1}{i\omega\mu}\operatorname{rot}\mathbf{A}-\operatorname{grad}\frac{1}{i\Omega\mu}\operatorname{div}\mathbf{A}-\hat{\sigma}\left(\mathbf{A}+\operatorname{grad}\varphi\right)=\mathbf{J}^{s},$$
$$\operatorname{div}\hat{\sigma}\left(\mathbf{A}+\operatorname{grad}\varphi\right)=-\operatorname{div}\mathbf{J}^{s}. \qquad (1)$$

In the cylindrical system of coordinates

$$x=r\cos\theta,\quad y=r\sin\theta,\quad z=z,\quad r=\sqrt{x^2+y^2},\quad \operatorname{tg}\theta=y/x,$$

the system of equations (1) for components of the vector potential $\mathbf{A}$,

$$\mathbf{A}\equiv(A_r,A_\theta,A_z),$$

and the scalar potential φ is approximated with the second order of accuracy by the following system of difference equations:

$$\begin{aligned}
&A^r_{i+1/2,j,k}\left[\frac{r_{i+1/2}\sigma_{i+1/2,j,k}}{r_i h^r_i}\right]+A^r_{i-1/2,j,k}\left[\frac{-r_{i-1/2}\sigma_{i-1/2,j,k}}{r_i h^r_i}\right]\\
&+A^\theta_{i,j+1/2,k}\left[\frac{\sigma_{i,j+1/2,k}}{r_i h^\theta_j}\right]+A^\theta_{i,j-1/2,k}\left[\frac{-\sigma_{i,j-1/2,k}}{r_i h^\theta_j}\right]\\
&+A^z_{i,j,k+1/2}\left[\frac{\sigma_{i,j,k+1/2}}{h^z_k}\right]+A^z_{i,j,k-1/2}\left[\frac{-\sigma_{i,j,k-1/2}}{h^z_k}\right]\\
&+\varphi_{i+1,j,k}\left[\frac{r_{i+1/2}\sigma_{i+1/2,j,k}}{h^r_{i+1/2}r_i h^r_i}\right]+\varphi_{i-1,j,k}\left[\frac{r_{i-1/2}\sigma_{i-1/2,j,k}}{h^r_{i-1/2}r_i h^r_i}\right]\\
&+\varphi_{i,j+1,k}\left[\frac{\sigma_{i,j+1/2,k}}{r_i h^\theta_j r_i h^\theta_{j+1/2}}\right]+\varphi_{i,j-1,k}\left[\frac{\sigma_{i,j-1/2,k}}{r_i h^\theta_j r_i h^\theta_{j-1/2}}\right]\\
&+\varphi_{i,j,k+1}\left[\frac{\sigma_{i,j,k+1/2}}{h^z_k h^z_{k+1/2}}\right]+\varphi_{i,j,k-1}\left[\frac{\sigma_{i,j,k-1/2}}{h^z_k h^z_{k-1/2}}\right]\\
&+\varphi_{i,j,k}\left[\frac{-r_{i+1/2}\sigma_{i+1/2,j,k}}{h^r_{i+1/2}r_i h^r_i}+\frac{-\sigma_{i,j+1/2,k}}{r_i h^\theta_j r_i h^\theta_{j+1/2}}+\frac{-\sigma_{i,j,k+1/2}}{h^z_k h^z_{k+1/2}}\right.\\
&\qquad\left.+\frac{-r_{i-1/2}\sigma_{i-1/2,j,k}}{h^r_{i-1/2}r_i h^r_i}+\frac{-\sigma_{i,j-1/2,k}}{r_i h^\theta_j r_i h^\theta_{j-1/2}}+\frac{-\sigma_{i,j,k-1/2}}{h^z_k h^z_{k-1/2}}\right]=0.
\end{aligned} \qquad (2)$$

$$A^r_{i-1/2,j,k}\left[\frac{1}{i\omega\mu_{i-1/2,j+1/2,k}r^2_{i-1/2}h^\theta_{j+1/2}h^\theta_j}+\frac{1}{i\omega\mu_{i-1/2,j-1/2,k}r^2_{i-1/2}h^\theta_{j-1/2}h^\theta_j}+\frac{r_{i-1/2}}{i\Omega\mu_{i,j,k}r_ih^r_ih^r_{i-1/2}}\right.$$
$$\left.+\frac{1}{i\omega\mu_{i-1/2,j,k+1/2}h^z_{k+1/2}h^z_k}+\frac{1}{i\omega\mu_{i-1/2,j,k-1/2}h^z_{k-1/2}h^z_k}+\frac{r_{i-1/2}}{i\Omega\mu_{i-1,j,k}r_{i-1}h^r_{i-1}h^r_{i-1/2}}-\sigma_{i-1/2,j,k}\right]$$
$$+A^r_{i-1/2,j+1,k}\left[\frac{-1}{i\omega\mu_{i-1/2,j+1/2,k}r_{i-1/2}h^\theta_{j+1/2}r_{i-1/2}h^\theta_j}\right]+A^r_{i-1/2,j,k-1}\left[\frac{-1}{i\omega\mu_{i-1/2,j,k-1/2}h^z_{k-1/2}h^z_k}\right]$$
$$+A^r_{i-1/2,j,k+1}\left[\frac{-1}{i\omega\mu_{i-1/2,j,k+1/2}h^z_{k+1/2}h^z_k}\right]+A^r_{i-1/2,j-1,k}\left[\frac{-1}{i\omega\mu_{i-1/2,j-1/2,k}r_{i-1/2}h^\theta_{j-1/2}r_{i-1/2}h^\theta_j}\right]$$
$$+A^r_{i+1/2,j,k}\left[\frac{-r_{i+1/2}}{i\Omega\mu_{i,j,k}r_ih^r_ih^r_{i-1/2}}\right]+A^r_{i-3/2,j,k}\left[\frac{-r_{i-3/2}}{i\Omega\mu_{i-1,j,k}r_{i-1}h^r_{i-1}h^r_{i-1/2}}\right]$$

$$+A^\theta_{i,j+1/2,k}\left[\frac{r_i}{i\omega\mu_{i-1/2,j+1/2,k}r_{i-1/2}h^r_{i-1/2}r_{i-1/2}h^\theta_j}-\frac{1}{i\Omega\mu_{i,j,k}r_ih^\theta_jh^r_{i-1/2}}\right]$$
$$+A^\theta_{i,j-1/2,k}\left[\frac{1}{i\Omega\mu_{i,j,k}r_ih^\theta_jh^r_{i-1/2}}-\frac{r_i}{i\omega\mu_{i-1/2,j-1/2,k}r_{i-1/2}h^r_{i-1/2}r_{i-1/2}h^\theta_j}\right]$$
$$+A^\theta_{i-1,j-1/2,k}\left[\frac{r_{i-1}}{i\omega\mu_{i-1/2,j-1/2,k}r_{i-1/2}h^r_{i-1/2}r_{i-1/2}h^\theta_j}-\frac{1}{i\Omega\mu_{i-1,j,k}r_{i-1}h^\theta_jh^r_{i-1/2}}\right]$$
$$+A^\theta_{i-1,j+1/2,k}\left[\frac{1}{i\Omega\mu_{i-1,j,k}r_{i-1}h^\theta_jh^r_{i-1/2}}-\frac{r_{i-1}}{i\omega\mu_{i-1/2,j+1/2,k}r_{i-1/2}h^r_{i-1/2}r_{i-1/2}h^\theta_j}\right]$$

$$+A^z_{i,j,k+1/2}\left[\frac{1}{i\omega\mu_{i-1/2,j,k+1/2}h^r_{i-1/2}h^z_k}-\frac{1}{i\Omega\mu_{i,j,k}h^z_kh^r_{i-1/2}}\right]+\varphi_{i-1,j,k}\left[\frac{\sigma_{i-1/2,j,k}}{h^r_{i-1/2}}\right]$$
$$+A^z_{i-1,j,k+1/2}\left[\frac{1}{i\Omega\mu_{i-1,j,k}h^z_kh^r_{i-1/2}}-\frac{1}{i\omega\mu_{i-1/2,j,k+1/2}h^r_{i-1/2}h^z_k}\right]$$
$$+A^z_{i-1,j,k-1/2}\left[\frac{1}{i\omega\mu_{i-1/2,j,k-1/2}h^r_{i-1/2}h^z_k}-\frac{1}{i\Omega\mu_{i-1,j,k}h^z_kh^r_{i-1/2}}\right] \qquad (3)$$
$$+A^z_{i,j,k-1/2}\left[\frac{1}{i\Omega\mu_{i,j,k}h^z_kh^r_{i-1/2}}-\frac{1}{i\omega\mu_{i-1/2,j,k-1/2}h^r_{i-1/2}h^z_k}\right]+\varphi_{i,j,k}\left[-\frac{\sigma_{i-1/2,j,k}}{h^r_{i-1/2}}\right]=0;$$

$$
\begin{aligned}
&A^{r}_{i-1/2,j-1,k}\left[\frac{1}{i\omega\mu_{i-1/2,j-1/2,k}r_{i-1/2}h^{\theta}_{j-1/2}h^{r}_{i}}-\frac{r_{i-1/2}}{i\Omega\mu_{i,j-1,k}r_{i}h^{r}_{i}r_{i}h^{\theta}_{j-1/2}}\right]\\
&+A^{r}_{i-1/2,j,k}\left[\frac{r_{i-1/2}}{i\Omega\mu_{i,j,k}r_{i}h^{r}_{i}r_{i}h^{\theta}_{j-1/2}}-\frac{1}{i\omega\mu_{i-1/2,j-1/2,k}r_{i-1/2}h^{\theta}_{j-1/2}h^{r}_{i}}\right]\\
&+A^{r}_{i+1/2,j-1,k}\left[\frac{r_{i+1/2}}{i\Omega\mu_{i,j-1,k}r_{i}h^{r}_{i}r_{i}h^{\theta}_{j-1/2}}-\frac{1}{i\omega\mu_{i+1/2,j-1/2,k}r_{i+1/2}h^{\theta}_{j-1/2}h^{r}_{i}}\right]\\
&+A^{r}_{i+1/2,j,k}\left[\frac{1}{i\omega\mu_{i+1/2,j-1/2,k}r_{i+1/2}h^{\theta}_{j-1/2}h^{r}_{i}}-\frac{r_{i+1/2}}{i\Omega\mu_{i,j,k}r_{i}h^{r}_{i}r_{i}h^{\theta}_{j-1/2}}\right]\\
&+A^{\theta}_{i-1,j-1/2,k}\left[\frac{-r_{i-1}}{i\omega\mu_{i-1/2,j-1/2,k}r_{i-1/2}h^{r}_{i-1/2}h^{r}_{i}}\right]+A^{\theta}_{i,j-3/2,k}\left[\frac{-1}{i\Omega\mu_{i,j-1,k}r_{i}h^{\theta}_{j-1}r_{i}h^{\theta}_{j-1/2}}\right]\\
&+A^{\theta}_{i,j-1/2,k-1}\left[\frac{-1}{i\omega\mu_{i,j-1/2,k-1/2}h^{z}_{k-1/2}h^{z}_{k}}\right]\\
&+A^{\theta}_{i,j-1/2,k}\left[\frac{1}{i\omega\mu_{i,j-1/2,k+1/2}h^{z}_{k+1/2}h^{z}_{k}}+\frac{1}{i\omega\mu_{i,j-1/2,k-1/2}h^{z}_{k-1/2}h^{z}_{k}}+\frac{r_{i}}{i\omega\mu_{i+1/2,j-1/2,k}r_{i+1/2}h^{r}_{i+1/2}h^{r}_{i}}\right.\\
&\left.+\frac{r_{i}}{i\omega\mu_{i-1/2,j-1/2,k}r_{i-1/2}h^{r}_{i-1/2}h^{r}_{i}}+\frac{1}{i\Omega\mu_{i,j,k}r_{i}h^{\theta}_{j}r_{i}h^{\theta}_{j-1/2}}+\frac{1}{i\Omega\mu_{i,j-1,k}r_{i}h^{\theta}_{j-1}r_{i}h^{\theta}_{j-1/2}}-\sigma_{i,j-1/2,k}\right]\\
&+A^{\theta}_{i,j-1/2,k+1}\left[\frac{-1}{i\omega\mu_{i,j-1/2,k+1/2}h^{z}_{k+1/2}h^{z}_{k}}\right]+A^{\theta}_{i,j+1/2,k}\left[\frac{-1}{i\Omega\mu_{i,j,k}r_{i}h^{\theta}_{j}r_{i}h^{\theta}_{j-1/2}}\right]\\
&+A^{\theta}_{i+1,j-1/2,k}\left[\frac{-r_{i+1}}{i\omega\mu_{i+1/2,j-1/2,k}r_{i+1/2}h^{r}_{i+1/2}h^{r}_{i}}\right]\\
&+A^{z}_{i,j-1,k-1/2}\left[\frac{1}{i\omega\mu_{i,j-1/2,k-1/2}r_{i}h^{\theta}_{j-1/2}h^{z}_{k}}-\frac{1}{i\Omega\mu_{i,j-1,k}h^{z}_{k}r_{i}h^{\theta}_{j-1/2}}\right]\\
&+A^{z}_{i,j-1,k+1/2}\left[\frac{1}{i\Omega\mu_{i,j-1,k}h^{z}_{k}r_{i}h^{\theta}_{j-1/2}}-\frac{1}{i\omega\mu_{i,j-1/2,k+1/2}r_{i}h^{\theta}_{j-1/2}h^{z}_{k}}\right]\\
&+A^{z}_{i,j,k-1/2}\left[\frac{1}{i\Omega\mu_{i,j,k}h^{z}_{k}r_{i}h^{\theta}_{j-1/2}}-\frac{1}{i\omega\mu_{i,j-1/2,k-1/2}r_{i}h^{\theta}_{j-1/2}h^{z}_{k}}\right] \qquad (4)\\
&+A^{z}_{i,j,k+1/2}\left[\frac{1}{i\omega\mu_{i,j-1/2,k+1/2}r_{i}h^{\theta}_{j-1/2}h^{z}_{k}}-\frac{1}{i\Omega\mu_{i,j,k}h^{z}_{k}r_{i}h^{\theta}_{j-1/2}}\right]\\
&+\varphi_{i,j-1,k}\left[\frac{\sigma_{i,j-1/2,k}}{r_{i}h^{\theta}_{j-1/2}}\right]+\varphi_{i,j,k}\left[\frac{-\sigma_{i,j-1/2,k}}{r_{i}h^{\theta}_{j-1/2}}\right]=S^{\theta}_{i,k};
\end{aligned}
$$

$$
\begin{aligned}
&A^{r}_{i-1/2,j,k-1}\left[\frac{r_{i-1/2}}{i\omega\mu_{i-1/2,j,k-1/2}h^{z}_{k-1/2}r_i h^{r}_{i}}-\frac{r_{i-1/2}}{i\Omega\mu_{i,j,k-1}r_i h^{r}_{i}h^{z}_{k-1/2}}\right]\\
&+A^{r}_{i-1/2,j,k}\left[\frac{r_{i-1/2}}{i\Omega\mu_{i,j,k}r_i h^{r}_{i}h^{z}_{k-1/2}}-\frac{r_{i-1/2}}{i\omega\mu_{i-1/2,j,k-1/2}h^{z}_{k-1/2}r_i h^{r}_{i}}\right]\\
&+A^{r}_{i+1/2,j,k-1}\left[\frac{r_{i+1/2}}{i\Omega\mu_{i,j,k-1}r_i h^{r}_{i}h^{z}_{k-1/2}}-\frac{r_{i+1/2}}{i\omega\mu_{i+1/2,j,k-1/2}h^{z}_{k-1/2}r_i h^{r}_{i}}\right]\\
&+A^{r}_{i+1/2,j,k}\left[\frac{r_{i+1/2}}{i\omega\mu_{i+1/2,j,k-1/2}h^{z}_{k-1/2}r_i h^{r}_{i}}-\frac{r_{i+1/2}}{i\Omega\mu_{i,j,k}r_i h^{r}_{i}h^{z}_{k-1/2}}\right]\\
&+A^{\theta}_{i,j-1/2,k-1}\left[\frac{1}{i\omega\mu_{i,j-1/2,k-1/2}h^{z}_{k-1/2}r_i h^{\theta}_{j}}-\frac{1}{i\Omega\mu_{i,j,k-1}r_i h^{\theta}_{j}h^{z}_{k-1/2}}\right]\\
&+A^{\theta}_{i,j-1/2,k}\left[\frac{1}{i\Omega\mu_{i,j,k}r_i h^{\theta}_{j}h^{z}_{k-1/2}}-\frac{1}{i\omega\mu_{i,j-1/2,k-1/2}h^{z}_{k-1/2}r_i h^{\theta}_{j}}\right]\\
&+A^{\theta}_{i,j+1/2,k}\left[\frac{1}{i\omega\mu_{i,j+1/2,k-1/2}h^{z}_{k-1/2}r_i h^{\theta}_{j}}-\frac{1}{i\Omega\mu_{i,j,k}r_i h^{\theta}_{j}h^{z}_{k-1/2}}\right]\\
&+A^{\theta}_{i,j+1/2,k-1}\left[\frac{1}{i\Omega\mu_{i,j,k-1}r_i h^{\theta}_{j}h^{z}_{k-1/2}}-\frac{1}{i\omega\mu_{i,j+1/2,k-1/2}h^{z}_{k-1/2}r_i h^{\theta}_{j}}\right]\\
&+A^{z}_{i-1,j,k-1/2}\left[\frac{-r_{i-1/2}}{i\omega\mu_{i-1/2,j,k-1/2}h^{r}_{i-1/2}r_i h^{r}_{i}}\right]+A^{z}_{i+1,j,k-1/2}\left[\frac{-r_{i+1/2}}{i\omega\mu_{i+1/2,j,k-1/2}h^{r}_{i+1/2}r_i h^{r}_{i}}\right]\\
&+A^{z}_{i,j-1,k-1/2}\left[\frac{-1}{i\omega\mu_{i,j-1/2,k-1/2}r_i h^{\theta}_{j-1/2}r_i h^{\theta}_{j}}\right]+A^{z}_{i,j+1,k-1/2}\left[\frac{-1}{i\omega\mu_{i,j+1/2,k-1/2}r_i h^{\theta}_{j+1/2}r_i h^{\theta}_{j}}\right]\\
&+A^{z}_{i,j,k-3/2}\left[\frac{-1}{i\Omega\mu_{i,j,k-1}h^{z}_{k-1}h^{z}_{k-1/2}}\right]+A^{z}_{i,j,k+1/2}\left[\frac{-1}{i\Omega\mu_{i,j,k}h^{z}_{k}h^{z}_{k-1/2}}\right]\\
&+A^{z}_{i,j,k-1/2}\left[\frac{r_{i+1/2}}{i\omega\mu_{i+1/2,j,k-1/2}h^{r}_{i+1/2}r_i h^{r}_{i}}+\frac{r_{i-1/2}}{i\omega\mu_{i-1/2,j,k-1/2}h^{r}_{i-1/2}r_i h^{r}_{i}}\right.\\
&+\frac{1}{i\omega\mu_{i,j+1/2,k-1/2}r_i h^{\theta}_{j+1/2}r_i h^{\theta}_{j}}+\frac{1}{i\omega\mu_{i,j-1/2,k-1/2}r_i h^{\theta}_{j-1/2}r_i h^{\theta}_{j}}\\
&\left.+\frac{1}{i\Omega\mu_{i,j,k}h^{z}_{k}h^{z}_{k-1/2}}+\frac{1}{i\Omega\mu_{i,j,k-1}h^{z}_{k-1}h^{z}_{k-1/2}}-\sigma_{i,j,k-1/2}\right]\\
&+\varphi_{i,j,k-1}\left[\frac{\sigma_{i,j,k-1/2}}{h^{z}_{k-1/2}}\right]+\varphi_{i,j,k}\left[\frac{-\sigma_{i,j,k-1/2}}{h^{z}_{k-1/2}}\right]=0.
\end{aligned}
\tag{5}
$$

We write the system of the algebraic equations in matrix form for the vector of unknowns

$$X(i,j,k,l) = X^{l}_{i,j,k},$$

where

$$X^{1}_{i,j,k} = X^{r}_{i,j,k} = A^{r}_{i-1/2,j,k}$$
$$X^{2}_{i,j,k} = X^{\theta}_{i,j,k} = A^{\theta}_{i,j-1/2,k}$$
$$X^{3}_{i,j,k} = X^{z}_{i,j,k} = A^{z}_{i,j,k-1/2}$$
$$X^{4}_{i,j,k} = \varphi_{i,j,k}$$

as follows:

$$\sum_{i}\sum_{j}\sum_{k}\sum_{l} C_{i_e,j_e,k_e,l_e}(i,j,k,l)X(i,j,k,l) = F_{i_e,j_e,k_e,l_e}$$

or

$$\mathbf{CX} = \mathbf{F}$$

where l is a number of the unknown function (corresponds to the superscript of $X^{l}_{i,j,k}$), (i_e, j_e, k_e, l_e) is the equation vector number (number of the matrix row).

The system has a large conditional number because of a non-uniform grid and discontinuous coefficients on the metal-air boundaries. That is why finding of solution to this system is another difficult problem. We have observed that the multigrid algorithm is a most effective one in both computational time and memory. In [1] we described our implementation of the multigrid method applying to the problem. In particular, we decided to make additional smoother iterations in the "singular" subdomains just after the prolongation operator. The correction function is changed locally in these subdomains (having several points width), and it becomes sharper near the air/metal boundary. Evidently this procedure is sufficiently computationally cheap, and it can be treated as a special kind of the interpolation correspondent to the solution structure. The procedure can be called the local smoother.

The rest of the paper we devote to computing electromagnetic fields in cases of small, multiple, or complex defects. Namely, we discuss ways to safe required computational resources: some "technical" tricks and so called embedded grids approach.

2 Increase of grid volume of solving problems

There exists a limitation to the grid volume for the Windows PC (2 GB application memory). That is why we made additional efforts to increase the possible grid volume which can be processed using PCs.

A variant of modifying matrix **C** by eliminating the unknowns at the boundary points from the vector **X** has been examined. Such elimination is possible because the values $X^l_{i,j,k}$ are explicitly defined at the boundary points (Dirichlet conditions). So the corresponding entries are put to the right-hand-side. One of the advantages of such modification is a noticeable reduction of the number of unknowns, especially for the coarse grids of the multigrid algorithm. For instance, in case of $O(N)$ grid cells in each direction, the number of difference equations is $O(N^3)$, the number of the boundary entries is about $4O(N^2)$ (the condition of periodicity was taken into account earlier).

The following renormalization of matrix **C** rows gives an additional advantage while eliminating the boundary points: we multiply (2) by $r_i^f h_i^{rf} h_j^{\theta c} h_k^{zc} i_{\max} j_{\max} k_{\max}$, (3) by $r_i^c h_i^{rc} h_j^{\theta f} h_k^{zc} i_{\max} j_{\max} k_{\max}$, (4) by $r_i^c h_i^{rc} h_j^{\theta c} h_k^{zf} i_{\max} j_{\max} k_{\max}$, and (5) by $r_i^c h_i^{rc} h_j^{\theta c} h_k^{zc} i_{\max} j_{\max} k_{\max}$. Thus the solution of the system remains the same but the matrix **C** becomes symmetrical (the factor $i_{\max} j_{\max} k_{\max}$ provides independency of the matrix norms on the coarse/fine grid). The matrix symmetry property permits to double reduce the memory for matrix elements keeping.

One of the possible ways of computer's memory saving is also taking into account a structure of the coefficient matrix **C** for our difference equations. From formulas (2) – (5) it is clear, that overwhelming number of coefficients are either purely imaginary, or purely real, and only coefficients for some unknowns have nonzero both imaginary and real parts. At the same time, to store a complex number we need 2 times more main memory, than for a real number. Therefore, from the point of view of memory's saving, it is natural to store coefficients of difference equations not as complex numbers, but as pair of independent real numbers. Using of a standard storage method for sparse matrices allows not storing zero which correspond to nonzero real or imaginary parts of coefficients of a matrix **C**. Although we must form two arrays of indices: for real and imaginary parts separately.

Formally it is expressed as follows: let's expand complex matrix **C** into a sum of real matrices **A** and **B**:

$$\mathbf{C} = \mathbf{A} + i\mathbf{B}.$$

To save each matrix we need 3 arrays which can be described as follows:

```
real*8 A(M_Re), B(M_Im)
integer col_A(M_Re),col_B(M_Im)
integer addr_row_A(N+1), addr_row_B(N+1)
```

where

N is a number of difference equations (rows of matrix **C**),

M_Re is a number of elements with a nonzero real part in a matrix **C** (number of nonzero elements in a matrix **A**),

M_Im is number of elements with a nonzero imaginary part in a matrix **C** (number of nonzero elements in a matrix **B**),

A and **B** are arrays of nonzero elements in matrices **A** and **B** which are stored row-wisely,

addr_row_A is an array of addresses which point out to the beginning of nonzero coefficients of a corresponding line of a matrix **A**,

col_A is an array of matrix **C** columns numbers which correspond to nonzero elements of a matrix **A**.

If we want to apply for a coefficient of the matrix $A_{i,j}$, the number **addr_row_A(i)** specifies the address of the beginning of information about nonzero elements in the i-th row of **A**. This information is stored in arrays **A** and **col_A.** And the number **addr_row_A(i+1)** specifies the address of the beginning of information about nonzero elements in the (i+1)-th row. Nonzero elements of the i-th row of a matrix **A** are stored in a subarray **A(addr_row_A(i):addr_row_A(i+1)-1).** Numbers of columns, they belong to, are stored in a subarray **col_A(addr_row_A(i):addr_row_A(i+1)-1)** on positions with the same index. Therefore, to learn a value of the coefficient $A_{i,j}$, it is necessary to try to find a number $\texttt{addr\_row\_A(i)} \le k < \texttt{addr\_row\_A(i+1)}$ such, that **col_A(k)=j.** If ***k*** exists then $A_{i,j}$ = **A(k).** Otherwise $A_{i,j}$=0.

The matrix **B** is similarly stored and processed.

To reduce number of arrays (and parameters of subroutines) it is possible to unite pairs of arrays: **A** and **B** in **AiB**, **col_A** and **col_B** in **col_AiB.** Also all elements of an array **addr_row_B** must be increased by value **M_Re.**

In process of iterations of the multigrid algorithm it is necessary to calculate the vector $\mathbf{W} = \mathbf{C} \cdot \mathbf{Z}$ many times; every j-th vector's element is computed from the formula

$$w_j = \sum_{k=1}^{N} C_{j,k} z_k$$

We can write $C_{j,k} = A_{j,k} + iB_{j,k}$, $z_k = x_k + iy_k$. Therefore

$$C_{j,k}z_k = A_{j,k}x_k - B_{j,k}y_k + i\ A_{j,k}y_k + B_{j,k}x_k \ .$$

But in most cases either $A_{j,k} = 0$ or $B_{j,k} = 0$. Therefore either

$$C_{j,k}z_k = iB_{j,k}x_k - B_{j,k}y_k ,$$

or

$$C_{j,k}z_k = A_{j,k}x_k + iA_{j,k}y_k ,$$

and it is advantageous to transfer vector's multiplication by a matrix **C** to real arithmetic. For the vector **W** with elements $w_j = u_j + iv_j$ this procedure is carried out by the following fragment of the Fortran code:

```
do i=1,N
   u(i)=0
   v(i)=0
   do j=addr_row_A(i),addr_row_A(i+1)-1
      u(i)=u(i)+AiB(j)*x(col_AiB(j))
      v(i)=v(i)+AiB(j)*y(col_AiB(j))
   enddo
   do j=addr_row_B(i),addr_row_B(i+1)-1
      u(i)=u(i)-AiB(j)*y(col(j))
      v(i)=v(i)+AiB(j)*x(col(j))
   enddo
enddo
```

As the result of tricks described above one can use comparably large grid. For illustration properties we present two examples.

2.1 Modeling of the conical defect

In this example we consider the outer conical defect having 50% depth, *16mm* top diameter (on the wall surface) and *2.7mm* bottom diameter. This defect is located inside the uniform grid box of *8* r-cells, *16* θ-cells, and *16* z-cells. Each cell is considered as the "metal" one if it lies strongly outside the cone. The box is a part of the whole non-uniform (r, θ ,z) grid having the volume 56×80×80 cells. In Figure 3 the component H_z of EM field inside the pipe (with *6mm* standoff) near the box (black frame) with defect is shown.

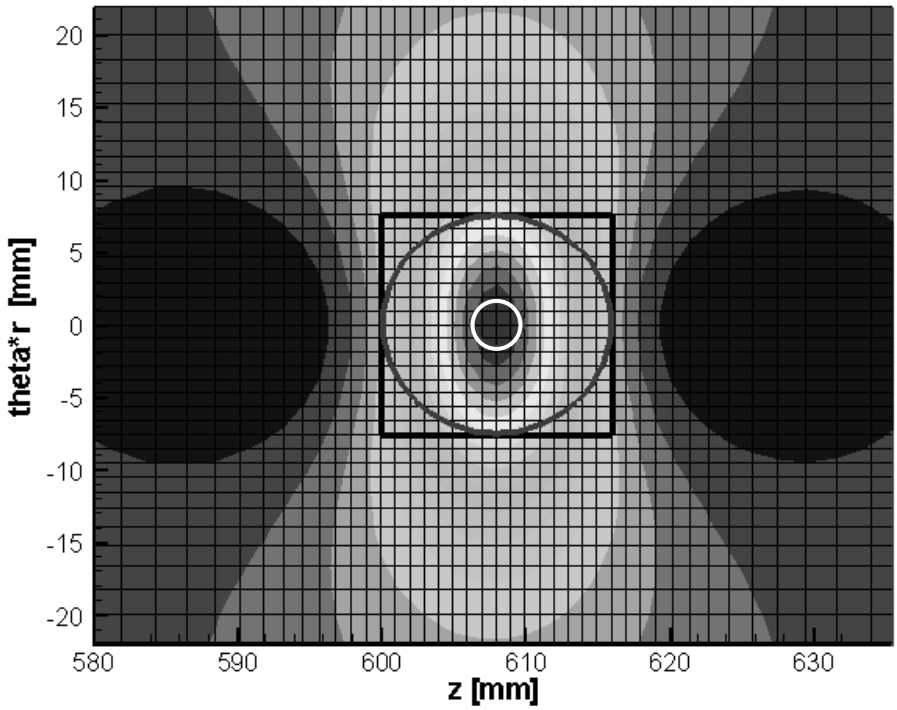


Figure 3. Conical defect: Top and bottom cone circles are drawn by the dark gray color and white color correspondingly; uniform grid box is drawn by the black color. The component H_z of EM field inside the pipe is mapped also.

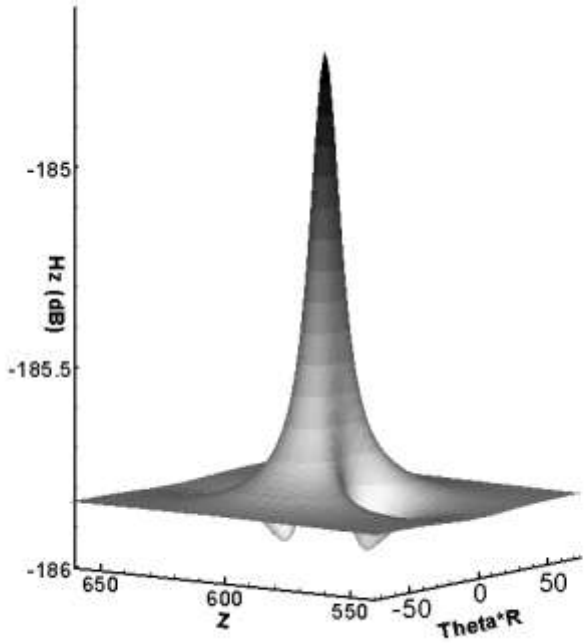


Figure 4. Conical defect: component H_z of EM field inside the pipe is mapped

In Figure 4 the same field H_z as in Figure is presented but as a 3D graph.

In this example we use four grid levels. The graph of the residual history is shown in Figure 5. We stop the iterations after 15 MG cycles. Each MG cycle requires about 50 multiplications of the "main" matrix on the finest grid. One can observe that the convergence rate is sharply drops after 11-th cycle. This is a typical behavior and we can not still overcome this effect. Note that already 10 MG cycles are enough in this example to obtain the solution with required accuracy.

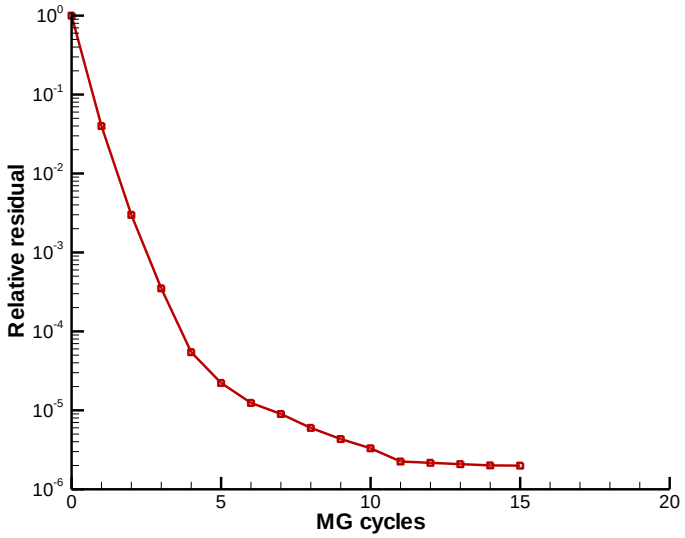


Figure 5. Convergence history for 4 grid levels, the finest grid is 56×80×80

2.2 Modeling of two defects

Here we model the situations with two "square" defects which are rectangular parallelepipeds in coordinates (r,θ,z). Both defects have 50% depth and *16mm* side. We consider two cases of location of the defects, the sequential and the diagonal location, see the scheme in Figure 6. The coordinate $\theta = 0$ is shown by the dashed line. The grid has 60×56×72 cells.

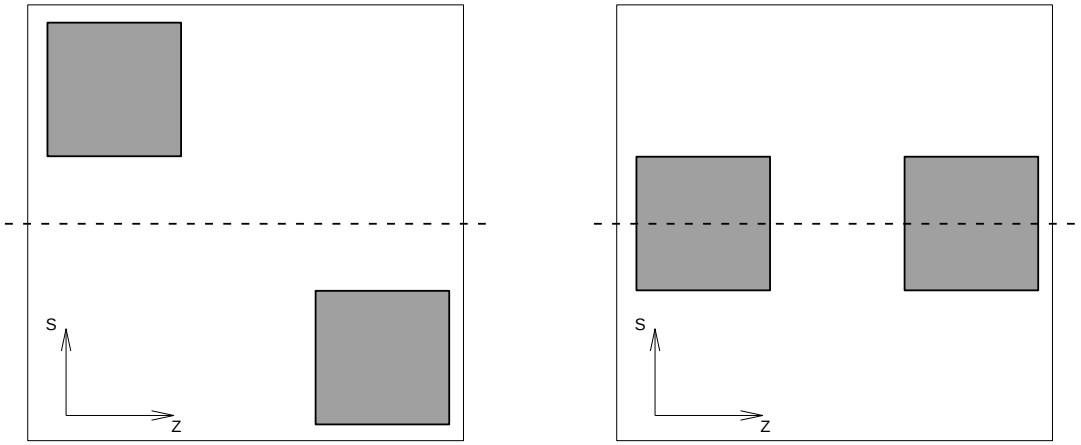
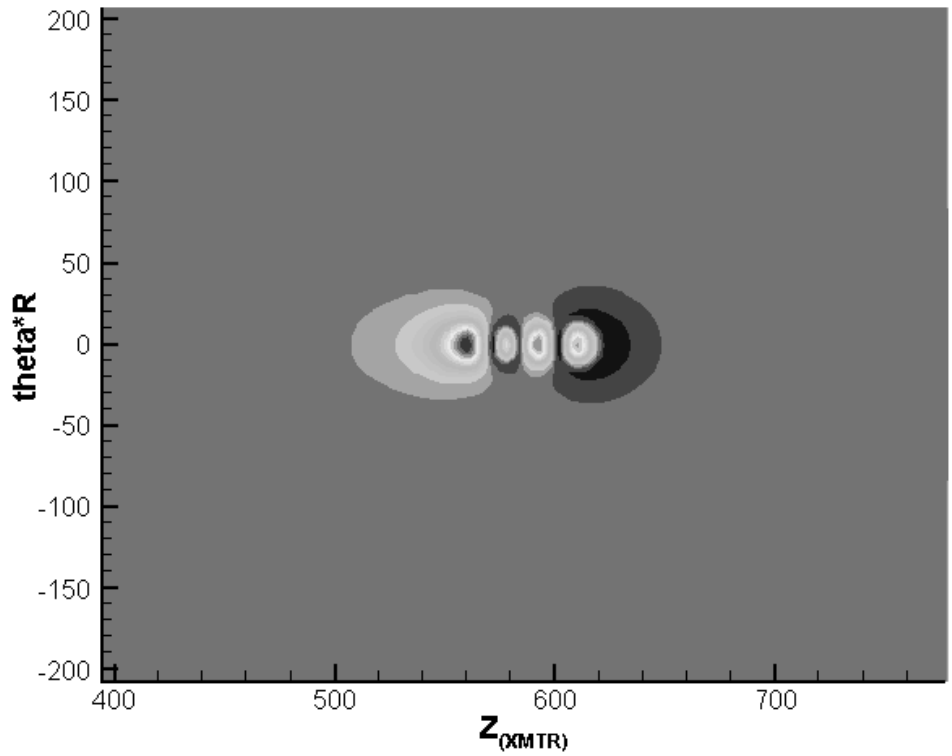


Figure 6. Two *16mm×16mm* defects located in the sequential (left) and diagonal (right) order. Top view on the pipe wall, $S = \theta \times r$



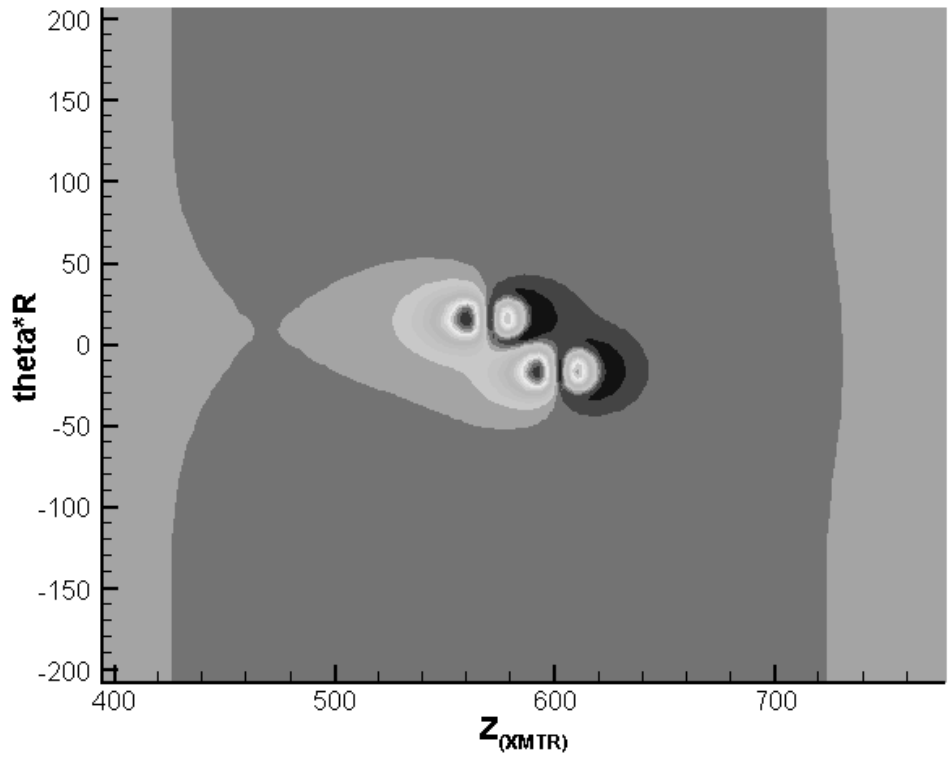
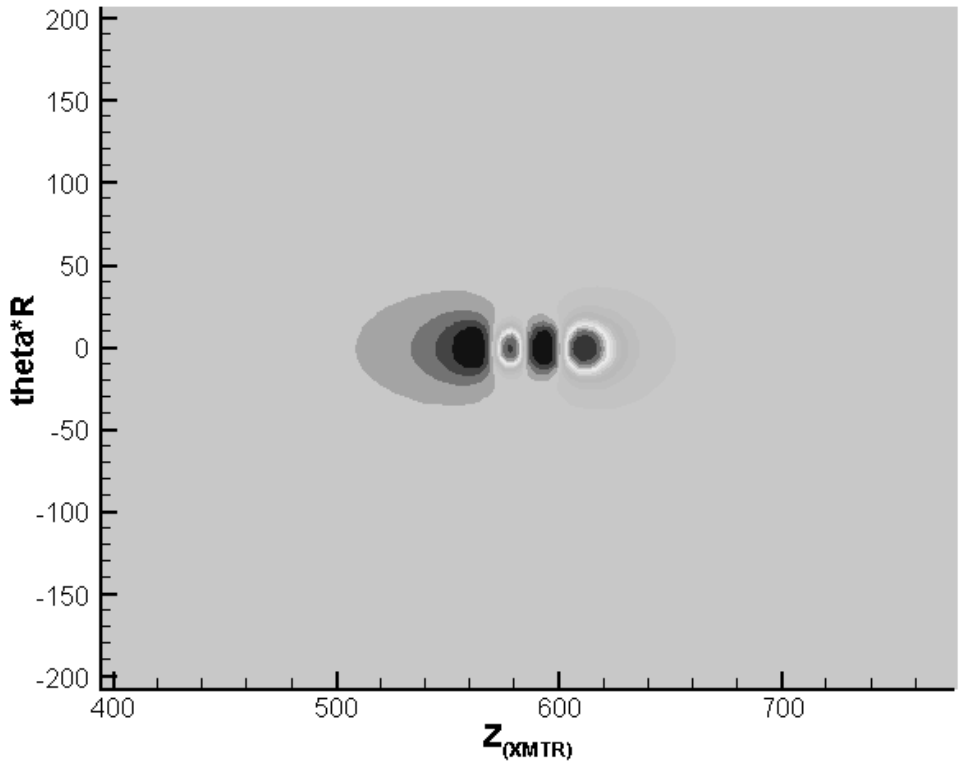


Figure 7. Amplitude of the H_r-component of the magnetic field; sequential (top) and diagonal (bottom) defects



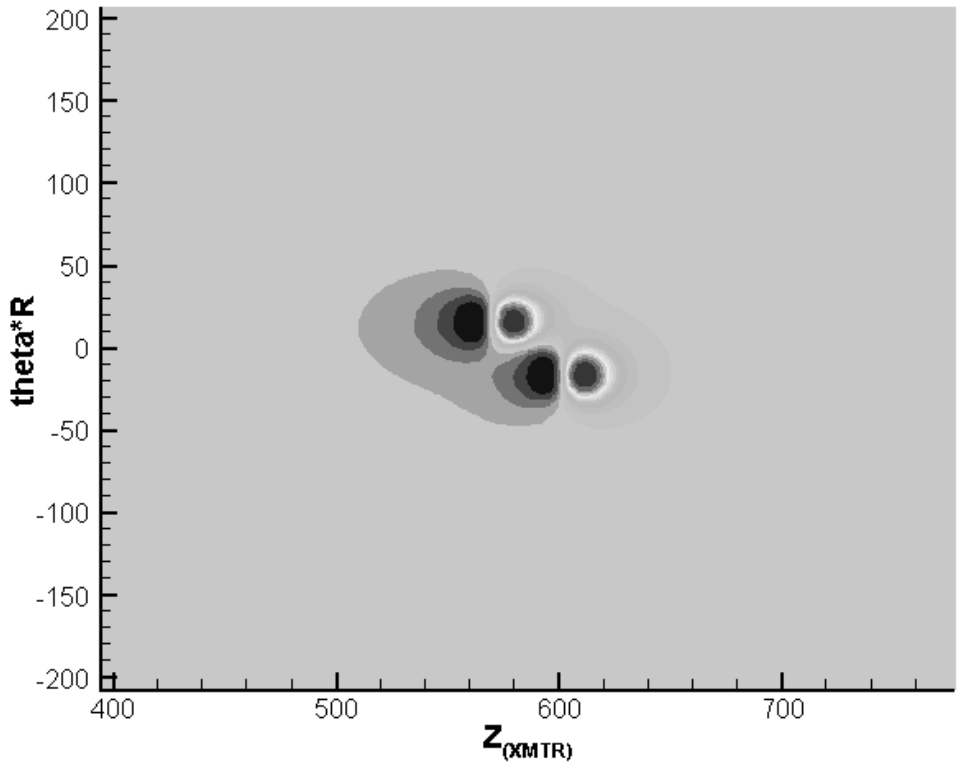


Figure 8. Phase of the H_r-component of the magnetic field; sequential (top) and diagonal (bottom) defects

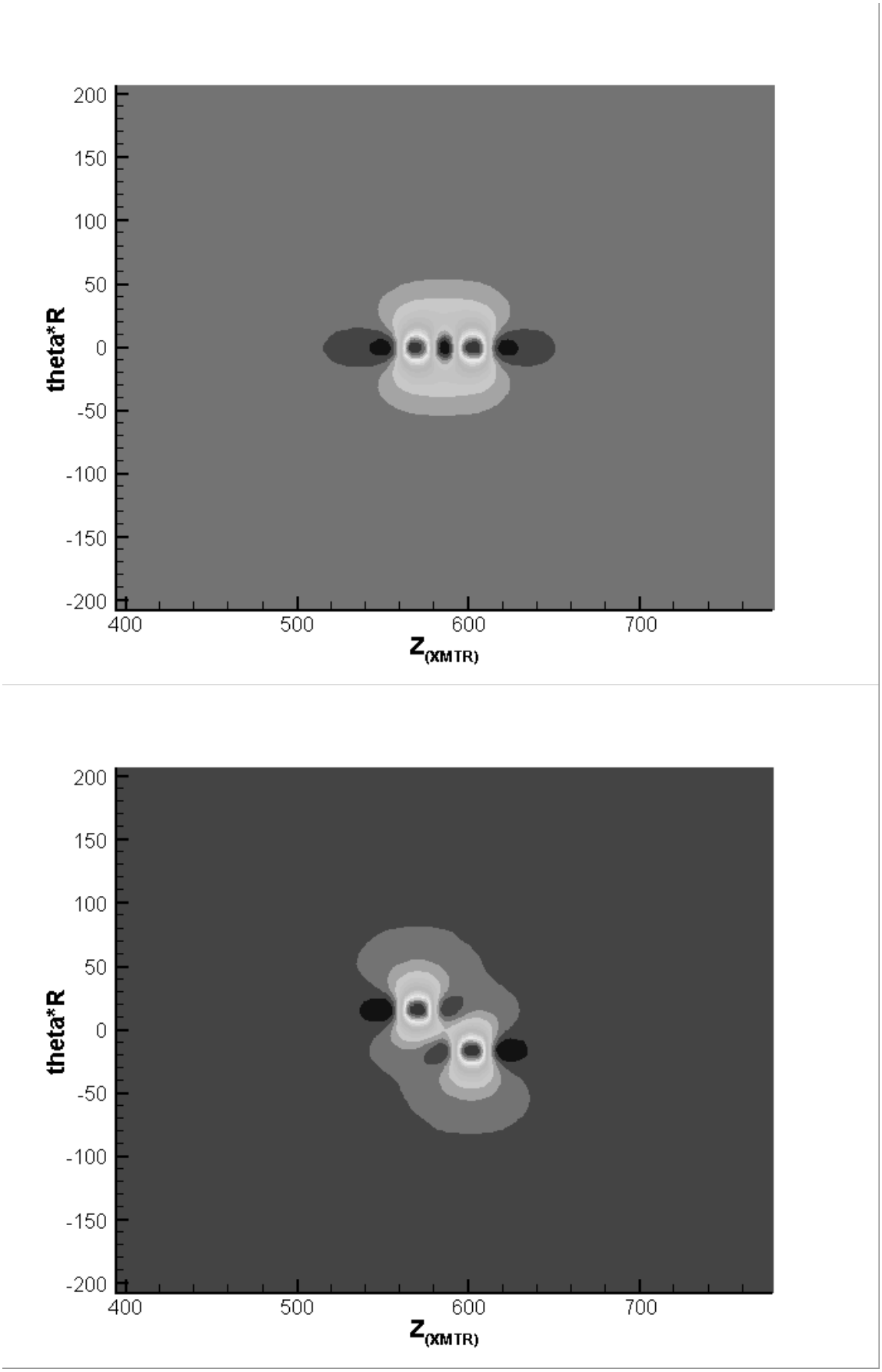


Figure 9. Amplitude of the H_z-component of the magnetic field; sequential (top) and diagonal (bottom) defects

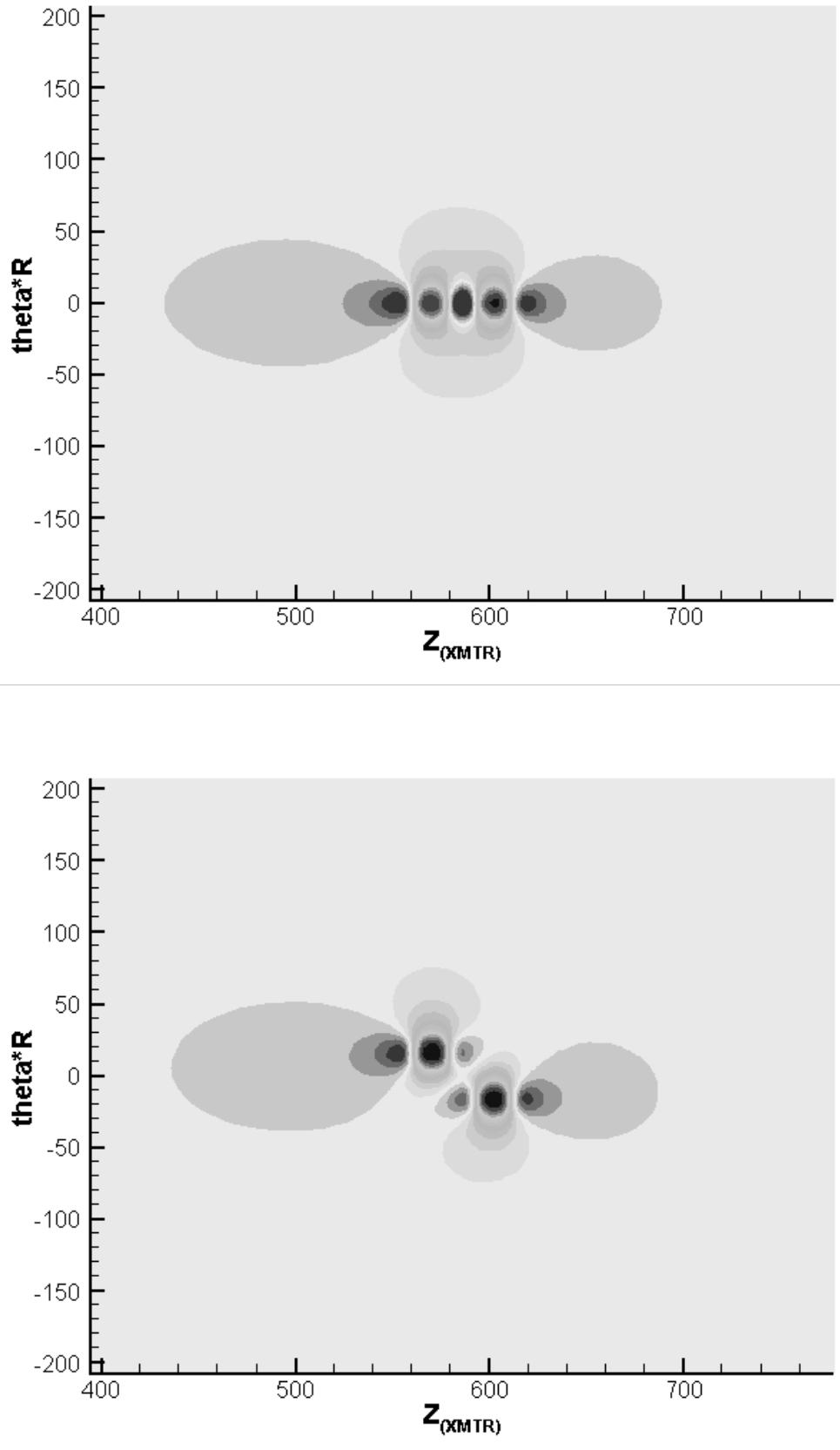


Figure 10. Phase of the H_z-component of the magnetic field; sequential (top) and diagonal (bottom) defects

Figure 7 – Figure 10 show amplitude and phase fields of r- and z-components of the magnetic field for two cases of location of the defects mentioned above.

3 Embedded grids (zoom method)

In our structured computational grid the smallest cells are located near a defect. Far from defect the cells are increased in direction of changing of the correspondent index, i.e. non-uniformly. Therefore there are regions with very thin cells up to the borders of the computational domain. Superfluous computational points essentially increase resources which are necessary to solve tasks (memory and time). But these points do not increase the accuracy of the result. To get over this issue it is proposed to use a zoom method which consists in the following.

First, instead of solving original problem

$$Lu = f$$

on a sufficiently fine grid, we consider and solve a problem

$$L^{(1)}u^{(1)} = f^{(1)}$$

on a coarser grid. It is selected so that global properties of the solution are kept (in particular, on periphery of a computational domain), but cell-size of this grid is not enough to accurately describe the solution in vicinity of defects. Then the second problem

$$L^{(2)}u^{(2)} = f^{(2)}$$

is solved for refinement of the solution in vicinity of defects on a fine grid but in smaller domain, see sketch in Figure 11.

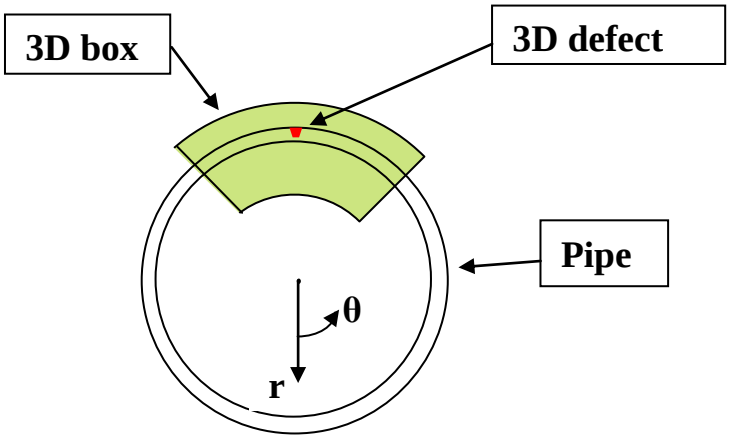


Figure 11. Small 3D computational domain, cross-section at z=const

Dirichlet boundary conditions for this problem are taken from the solution of the first problem. The boundary should be chosen far enough from the defect to be sure that solution of the first problem is accurate enough there.

It is clear that it is possible to make next step and solve a problem

$$L^{(3)}u^{(3)} = f^{(3)}$$

having applied the zoom method to the solution $u^{(2)}$. However, we restrict us by two steps in this paper.

Almost all necessary information for the solution of a problem using the multigrid method is organized as an array of pointers to structures. Each structure describes one multigrid level. It includes geometrical information, environment properties, arrays of unknown quantities ($X^{(1)}, X^{(2)}, X^{(3)}, X^{(4)}$), coefficients of difference equations, right-hand-sides and some additional information as well as auxiliary arrays.

In particular, in order to implement the MG method for discontinuous coefficients we use additional iterations in the vicinity of coefficients' contrasts after procedure of correction prolongation from the coarse grid on the fine one of the previous MG-level. Technically it is carried out as follows. Regions of coefficients' contrasts "are covered" by subregions which form 3-dimensional rectangles in the index space (i, j, k). For each subdomain D_α all interior unknowns X_α are additionally iterated in the smoother, i.e. the corresponding subproblems

$$L_\alpha\left(X_\alpha\right) = f_\alpha$$

are solved more accurately.

The subproblem L_α is formed by those difference equations of the main problem which are corresponded to internal points of the domain D_α. Those unknowns corresponded to an exterior of D_α or its border are not changed during iterative solution and transferred to the right-hand-side.

Data structure of the problem $L^{(2)}$ for the embedded grid is practically the same as described above. This fact prompts us to use the same solver for solving of the embedded problem, as for host one. But recurrence procedure of the MG solver imposes certain limitations on the structure and use of data. Besides, there are some distinctions in setups of embedded and host problems. And it is necessary to take into account these distinctions and to modify procedures of the MG solver to treat both host and embedded problems.

Note three main distinctions: 1) unlike the host problem which is periodic for the angle θ, embedded problem can be both periodic and non-periodic; 2) borders of

the embedded problem should match grid lines of the host problem; 3) different number and structure of regions along r in the embedded and host problems.

We shall give results of two tests to demonstrate zoom method's efficiency.

In the first test the electromagnetic field is calculated for 50% depth outer circular pit having 24 mm diameter. The pipe has internal radius 62.1 *mm* and 7.7 *mm* thickness. The results of two computations are compared. The results of the first computation are obtained for a maximal possible single-grid problem ($96\times80\times96$ cells). The results of the second computation are obtained with the help of zoom method when the host problem of $48\times40\times48$ cells and embedded problem of $40\times40\times40$ cells covering defect are solved. Both modeling computations have the same grid in the vicinity of the pit. In Figure 3 the isolines of the absolute value of the function $X^{(2)}$ are drawn on the plane $(r\cdot\theta, z)$ at radius $r=61.4mm$. The close neighbor lines correspond to first and second computations, respectively.

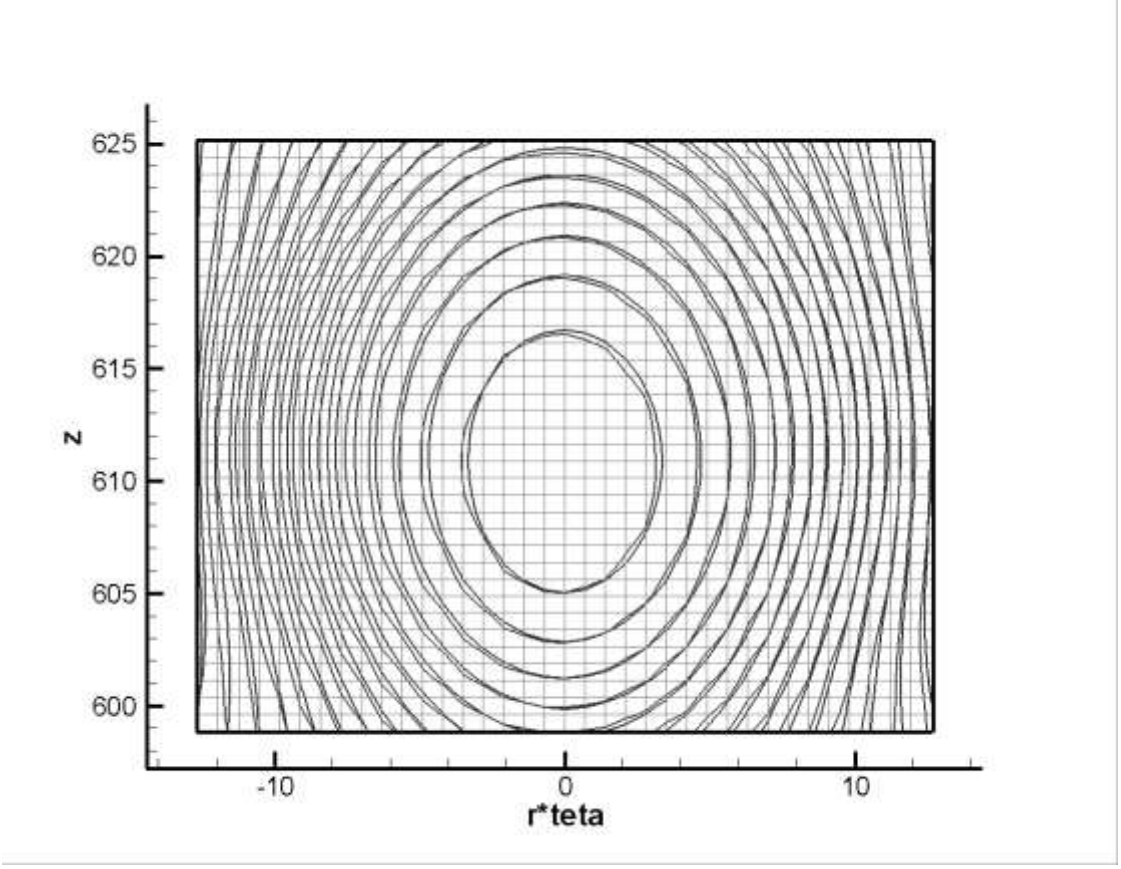


Figure 3. Isolines of the $|X^{(2)}(r,\theta,z)|$ at $r=61.4mm$ for two calculations: single $96\times80\times96$ grid and $48\times40\times48$ host + $40\times40\times40$ embedded grids. Grid lines are also shown.

Comparison shows a good coincidence of results. But the volume of the matrix is 8 times less for the second computation, and CPU time is almost 4 times less (since it was necessary to solve two problems).

A problem with 87.5% depth outer circular pit having diameter 6 mm is solved in the second test. The wall thickness is 8mm (i.e. the remaining wall thickness is 1mm). First the defect is resolved on the single grid of $100\times44\times68$ cells volume. Amplitude of the function $X^{(2)}$ near the defect at the section $\theta=0$ is shown in Figure 4. This is the minimal possible resolution because only four grid cells put on the pit diameter and rest wall (i.e. solution will be too inaccurate if we put less than four grid cells on these thin structures).

It is difficult to estimate the accuracy of the solution since the resources of PC are not enough to double resolution of the grid. However, with the help of an embedded grid it became possible to calculate the defect on a smaller domain for a grid $96\times40\times56$ which has 2 times less step size in each direction. Solutions obtained on the coarse and fine (embedded) grids are shown in Figure 5.

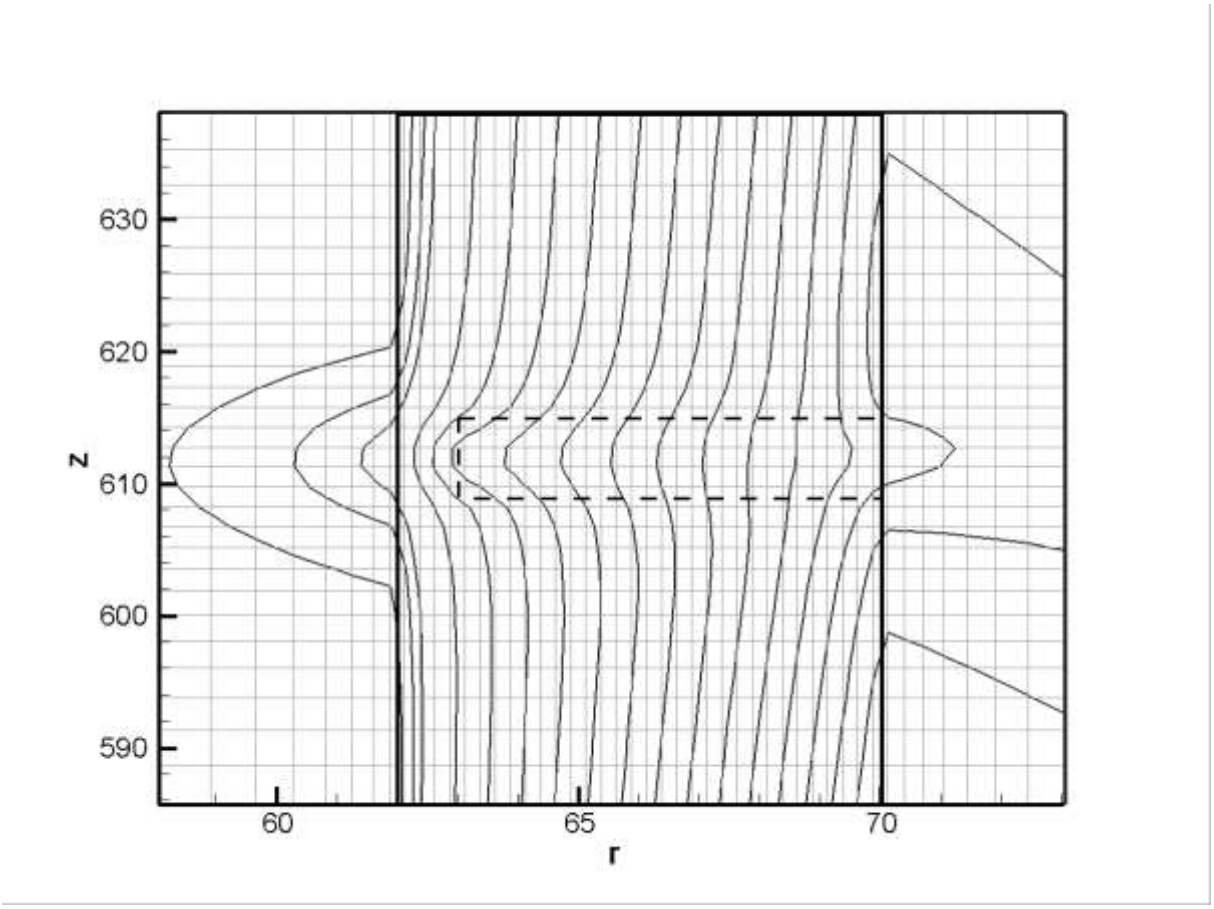


Figure 4. Isolines of $|X^{(2)}|$ for the single $100\times44\times68$ grid solution at the section $\theta=0$. Mesh (grey), pipe (vertical solid) and defect (dashed) are shown as well.

We also present graph of $|X^{(2)}|$ along z-line at $r=60.8mm$, see Figure 6. One can observe that both solutions are close to each other, although finer grid provides a better resolution of course. More precisely, the relative difference of curves is about

4%. For first and second derivatives (which model differential sensors) the relative difference achieves 10% to 20%. Believing that the algorithm has the second order of accuracy, we may conclude that accuracy of the solution on the embedded grid is four times better, i.e. 1% for function and 3% to 5% derivatives. Note that a good agreement of both solutions nearby the defects prompts us to use a smaller domain for the embedded grid, and, therefore, a smaller volume of the latter.

The problem is model, of course, but it illustrates possibilities.

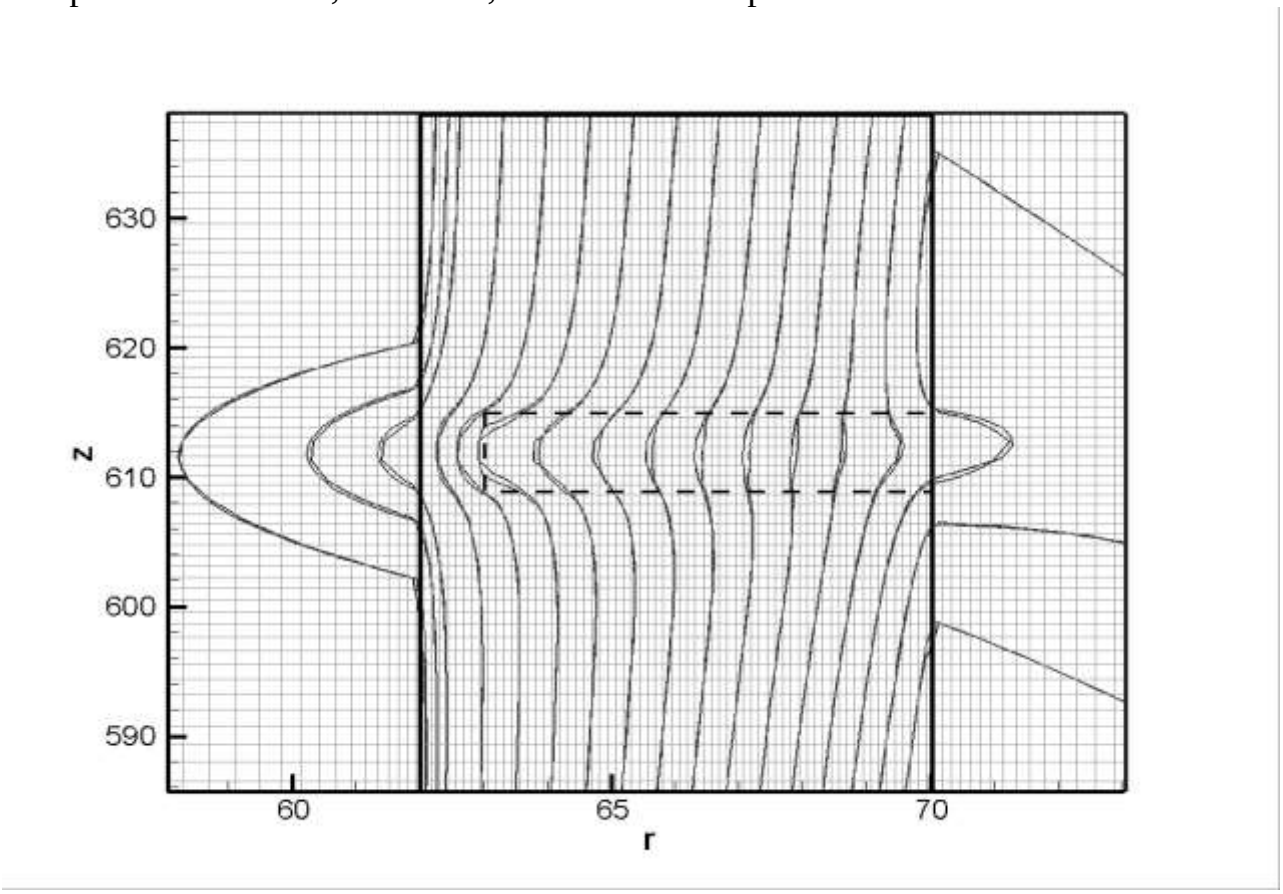


Figure 5. Isolines of $|X^{(2)}|$ for the single $100\times44\times68$ grid solution and embedded grid at the section $\theta=0$. Embedded mesh (grey), pipe (vertical solid) and defect (dashed) are shown as well.

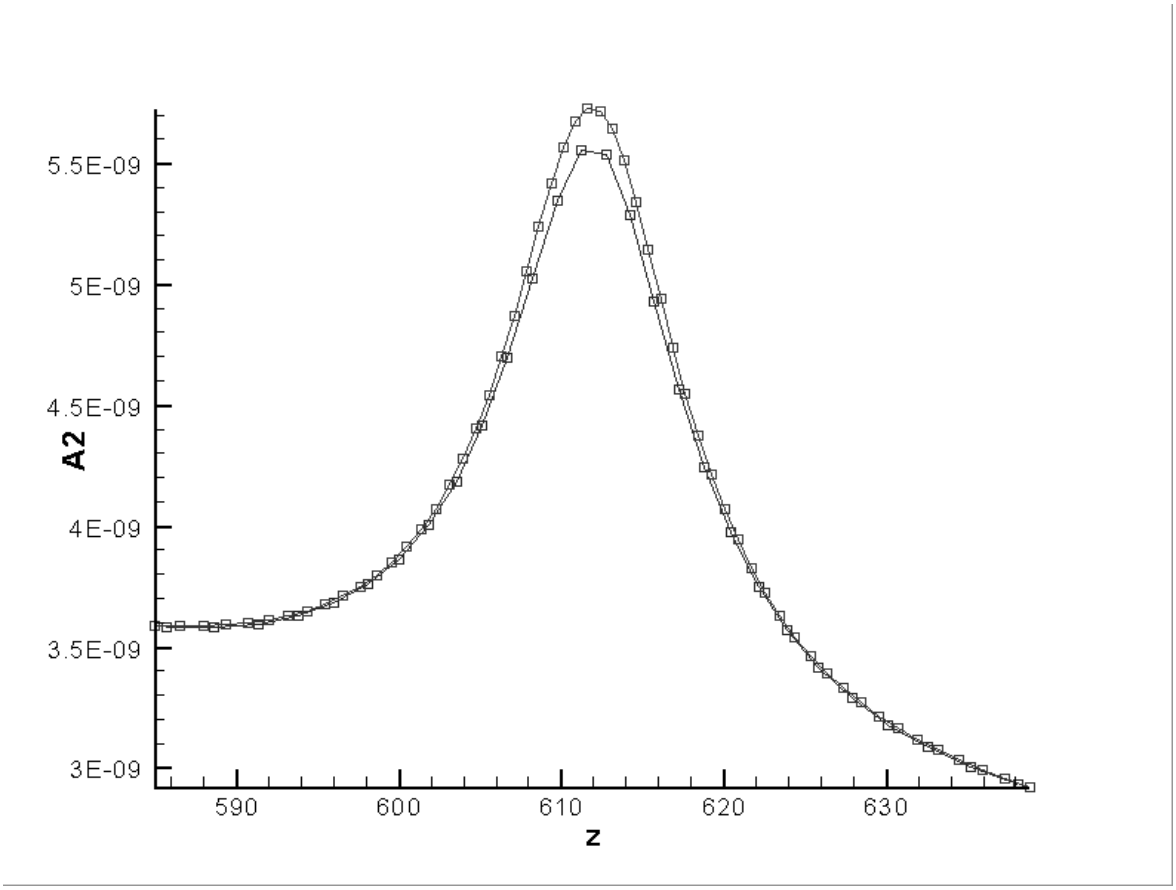


Figure 6. Function $|X^{(2)}|$ for the single $100\times 44\times 68$ grid solution and embedded grid at $\theta = 0$, $r = 60.8mm$.

4 Selfconsistent zoom method

In the previous section we have described the following problem formulation of solution refinement in zones of large gradients (regions with defects, sources). Let us have the solution u_1 of electromagnetic problem

$$L_{h_1} u_1 = 0 \tag{6}$$

on a grid Ω_{h_1} in domain D_1. Dirichlet boundary conditions for u_1 are imposed using solution of 2D problem in a larger domain D_0, $D_1 \subset D_0$.

Then the solution in a subregion $D_2 \subset D_1$ with defect is refined by introduction of a finer grid Ω_{h_2} (see Figure 7). The solution u_2 satisfies the equation

$$L_{h_2} u_2 = 0 \tag{7}$$

with Dirichlet boundary conditions obtained by restriction of the solution u_1 onto the boundary of D_2. The problem (7) looks like (6), but it can be yet either periodical or non-periodical with respect to θ.

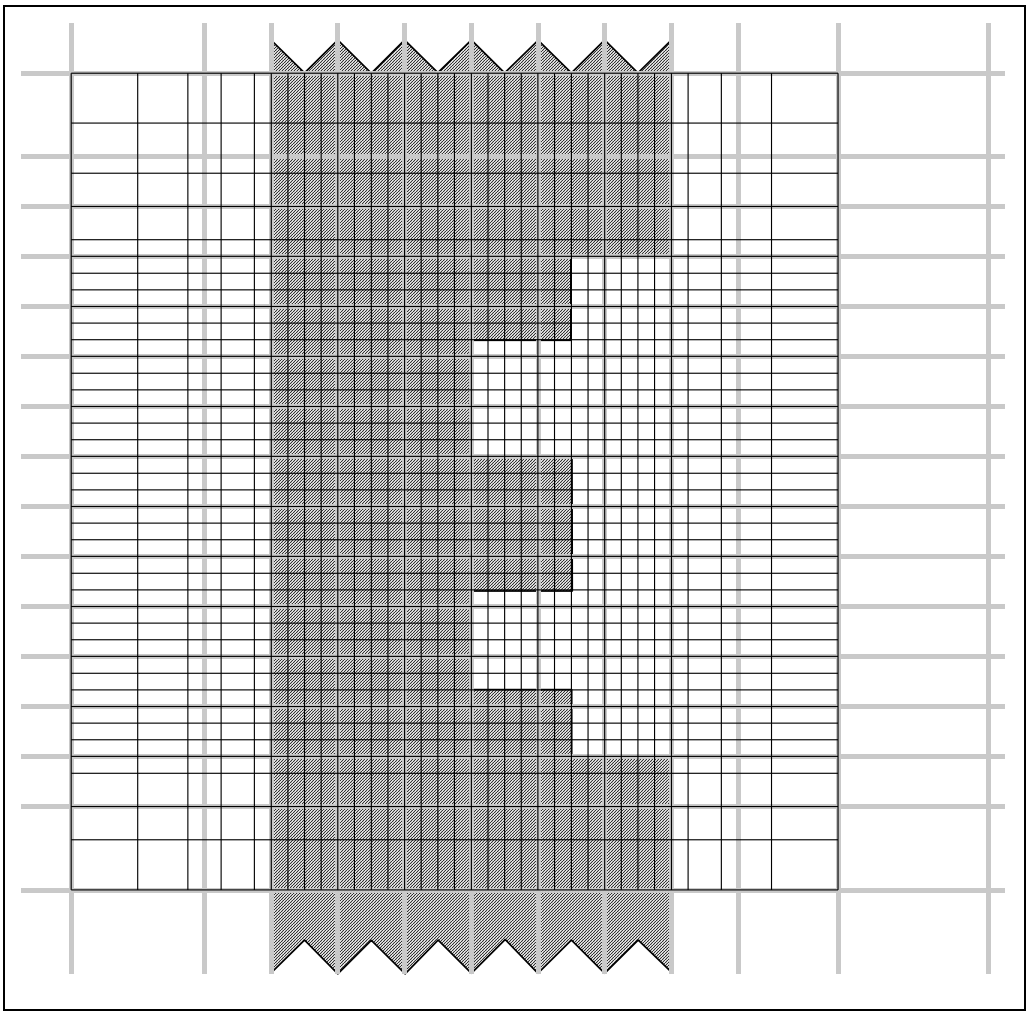


Figure 7. (r,z)**-grids near a pit:** Ω_{h_1} **(gray) and** Ω_{h_2} **(black)**

The algorithm is successfully implemented and it is shown that a more accurate solution can be obtained on two grids $\left(\Omega_{h_1},\Omega_{h_2}\right)$ or it can be obtained faster than on a single finer grid, see the section above.

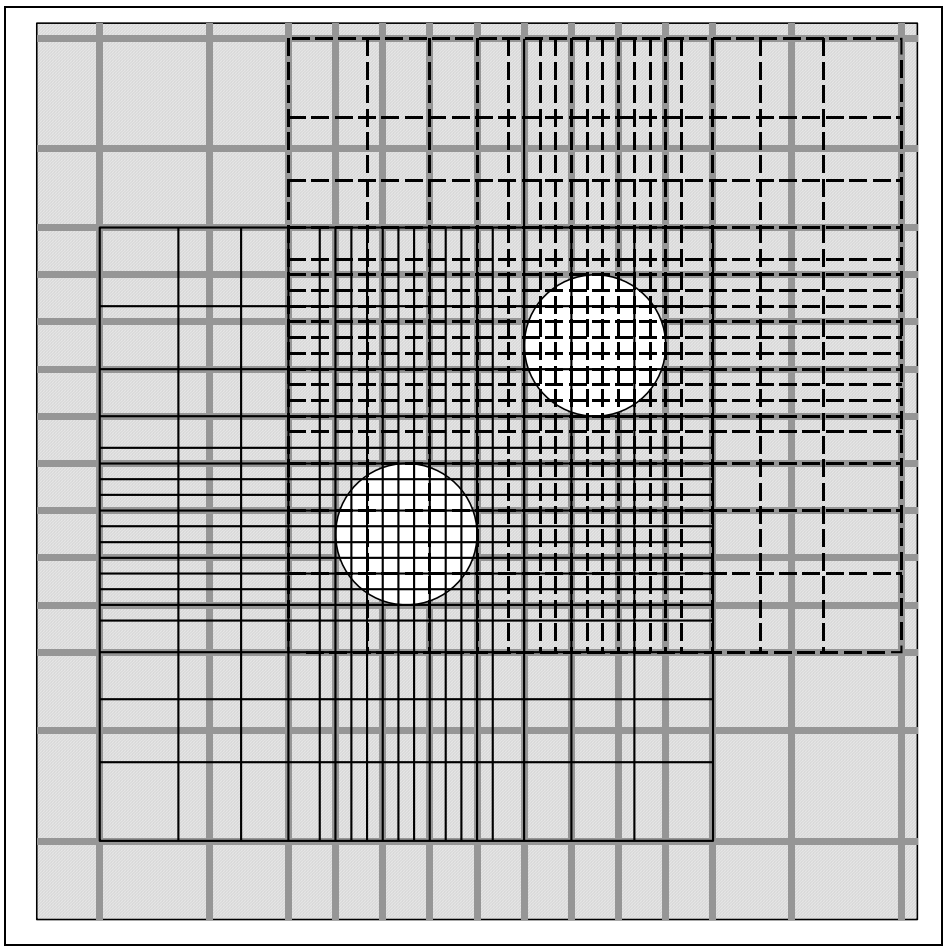


Figure 8. Two embedded grids, (θ, z)-plain section: Ω_{h_1} (fat gray), Ω_{h_2} (thin solid), and Ω_{h_3} (thin dashed)

One can consider, in principle, three and more grids: if, for instance, the casing has two defects the embedded grids D_2 and D_3 could be constructed independently with possible intersection, see Figure 8.

Another option with two embedded grids is the nesting structure: $D_3 \subset D_2 \subset D_1$ with finer and finer grids Ω_{h_2}, Ω_{h_3}.

An evident drawback of the above approach is absence of influence of solution u_2 on solution u_1, what contradicts the elliptical nature of considered problems. That is why we must take large enough size of D_2 in order to diminish this influence on solution near the defect.

A more "intelligent" possible way to reduce the drawback could be consist in organization of the following iterative process:

- step #1 is the inconsistent algorithm described above: we find u_1 and u_2 and denote $u_1^{(0)} := u_1$, $u_2^{(0)} := u_2$;

- step #n (n=2,3,…) solves:
  1. The problem

$$L_{h_1} u_1^{(n)} = f_n \tag{8}$$

in D_1 with a non-zero RHS defined everywhere inside D_2 where the formula

$$f_n := L_{h_1} R u_2^{(n-1)} \tag{9}$$

can be calculated; R is an interpolation operator of $u_2^{(n-1)}$ onto grid points of Ω_{h_1}.

  2. The problem

$$L_{h_2} u_2^{(n)} = 0 \tag{10}$$

with the Dirichlet boundary conditions obtained by restriction operation P of the solution $u_1^{(n)}$ onto the boundary of D_2:

$$u_2^n\big|_{\partial D_2} = P u_1^n . \tag{11}$$

If the iterations converge then one can believe that this the limit solution $\left\{u_1^{(n)}, u_2^{(n)}\right\}$ is actually the solution of above an “ideal” problem with the grid adaptive refinement (by neglecting influence of a small neighborhood of the boundary of domain D_2: one grid level of Ω_{h_1} for both sides of the boundary).

Therefore one can take a smaller domain D_2 comparing the case of an inconsistent problem.

However our first results did not give the expected enhance of the algorithm efficiency. Moreover the iterations diverge sometimes. There are several reasons of the failure, e.g.: a possible bad construction of the operators R and/or P (the cubic Cartesian interpolation); wrong choice of sizes of the embedded domain, etc. We plan to investigate and overcome the failure in oncoming paper.

References

[1] N. Zaitsev and I. Sofronov**. "**An algorithm for solving three-dimensional harmonic Maxwell equations in ferromagnetic pipes" Preprint of Keldysh Institute of Applied Mathematics, No. 128, 2005.

[2] N. Zaitsev and Ivan Sofronov. "Method of calculating low-frequency electromagnetic problems in cylindrical geometry" Preprint of Keldysh Institute of Applied Mathematics, No. 61, 2004.

[3] Haber and U.M. Ascher, Fast finite volume simulation of 3D electromagnetic problems with highly discontinuous coefficients. SIAM J. Sci. Comput. Vol. 22, No. 6, pp. 1943—1961, (2001)