

Mermaid 3D code in ATLAS applications

A.N.Dubrovin, E.A.Simonov

Budker Institute of Nuclear Physics, Novosibirsk, Russia

S.B. Vorojtsov

Joint Institute for Nuclear Research, Dubna, Russia

Abstract

The paper describes the new periodic boundary condition model of ATLAS detector and changes in the Mermaid 3D code algorithms necessary to calculate it. The parallel processing modification of the Mermaid code to perform the complete Atlas model calculation is presented in comparison with this 1:10 model of the fine TileCal structure.

Introduction

A number of models of complex Atlas geometry were developed for Mermaid 3D code. The last one is a periodic boundary condition model with the 1:10 modeling of the TileCal structure in Z direction.

As small as a few centimeters details should be resolved on the scale of total 25 meters of general Atlas dimensions. So even the huge number of mesh nodes provided by Mermaid code (about 5 millions for available computers) is not enough for input of complete Atlas geometry with high resolution. Hence, the reducing of the problem domain is necessary. As it had been shown [Bergsma, 1995], the minimal 1/16 part of total Atlas geometry is required to reproduce the complete Atlas field map (Figure 1).

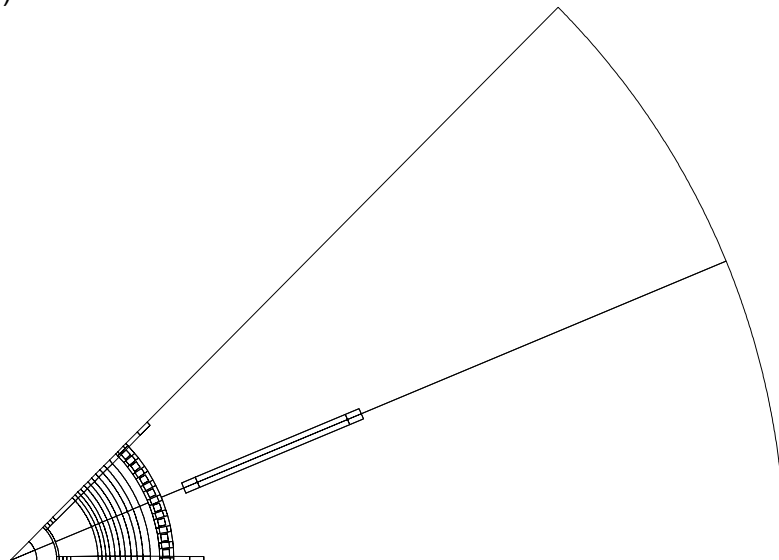


Figure 1 Atlas problem domain in XY plane.

The mesh in Mermaid code is built as a sequence of XY planes covered with nonuniform triangle mesh (Figures 2,3).

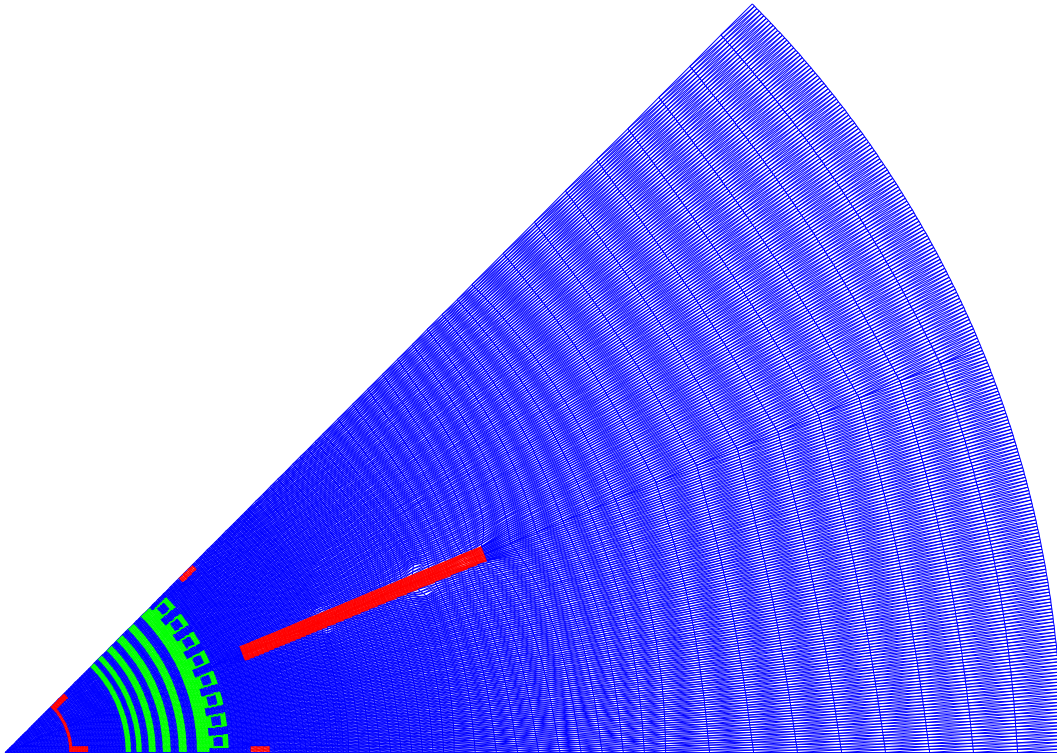


Figure 2 Complete Atlas problem domain mesh in XY plane. The cross-section was covered by 125x125 nonuniform triangle mesh topologically equivalent to unit square mesh (each square is divided into two triangles by a diagonal). The mesh is symmetric about 22.5 deg line.

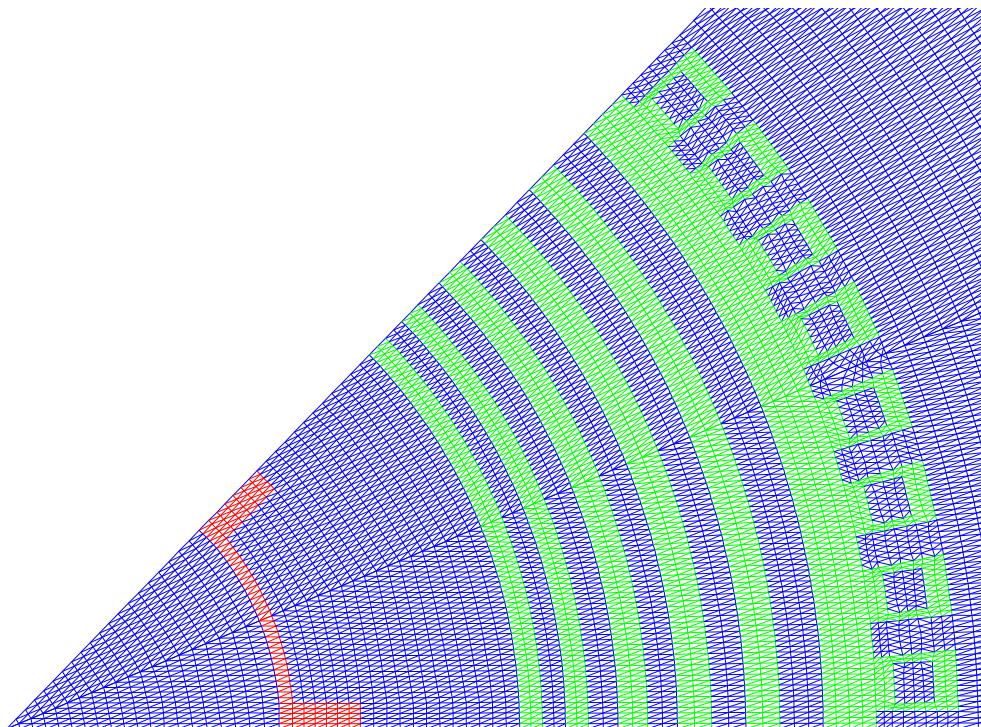


Figure 3 Atlas problem domain mesh in XY in the region of TileCal. Radius step is about 6 cm, angle direction step is about 3 cm.

This two-dimensional XY mesh is repeating in the third Z direction. The steps between XY planes in Z directions can be arbitrary as well as material filling in the

right triangle prisms, which are the finite elements used by Mermaid (Figures 4,5). The Atlas model with 1:10 scale in Z direction of the fine TileCal iron structure was calculated so far because of limitations of the available computers.

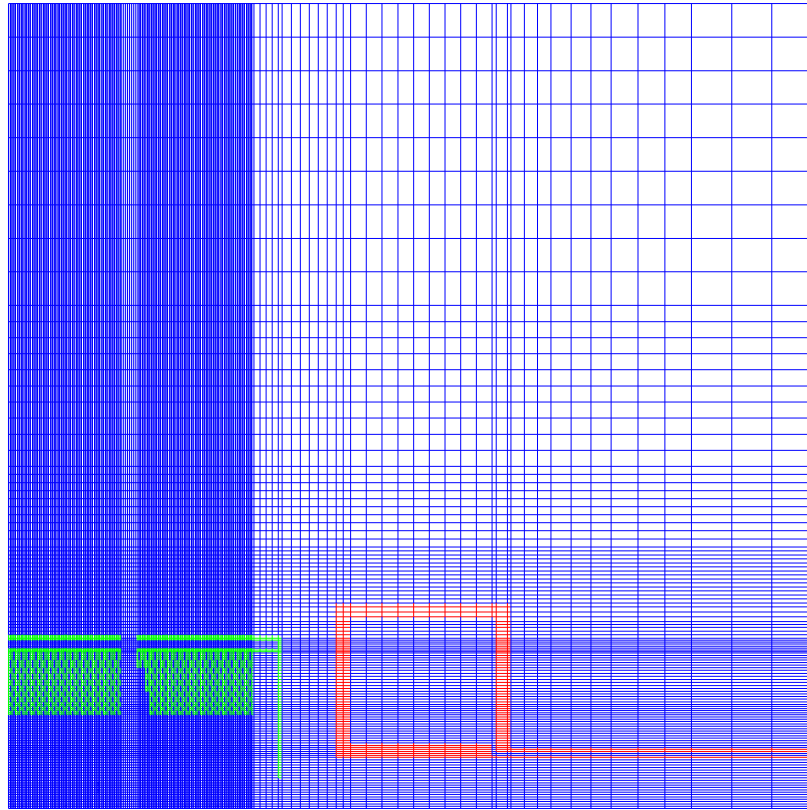


Figure 4 Atlas problem domain mesh in Z direction, 1:10 scale of the fine TileCal structure. 288 of XY planes were expanded in Z direction for total 4.5 millions of mesh nodes.

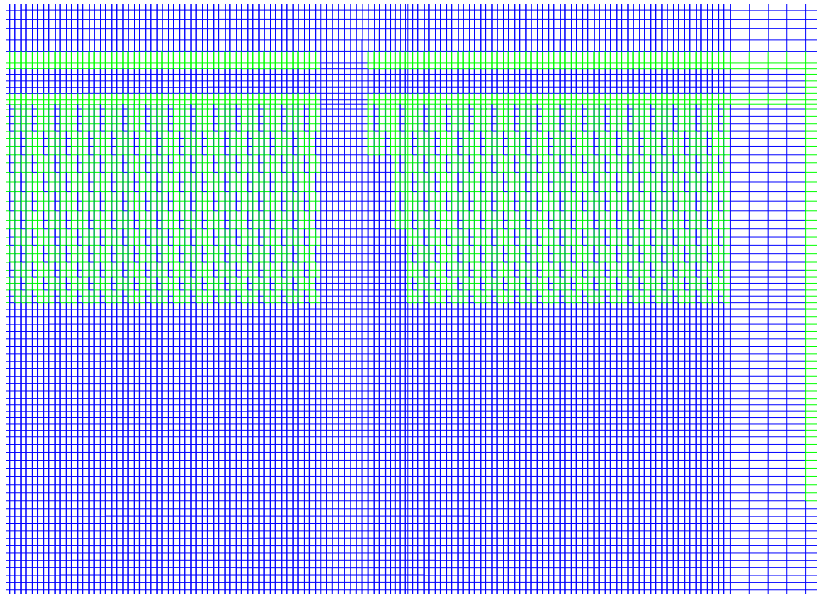


Figure 5 Atlas problem domain mesh in Z direction near the TileCal, 1:10 model of the fine TileCal structure.

New periodic boundary condition solver

Several changes in Mermaid code are required to take into account the complex periodicity of the Atlas detector. The solution of Maxwell's equation in Mermaid is a sum of two terms: $\mathbf{H} = \mathbf{H}' - \nabla\varphi$, where $\text{rot } \mathbf{H}' = \mathbf{j}$ and, consequently, $\text{div}\mu \text{ grad}\varphi = \text{div}\mu\mathbf{H}'$.

We can construct \mathbf{H}' as a sum of two solutions: for the solenoid coil turn on only and for toroids coils turn on only. Traditional Mermaid solver with local boundary condition can find both the solutions:

The field is perpendicular to $y=0$, $x=y$ planes and parallel to $z=0$ plane in toroids on only case;

The field is perpendicular to $z=0$ plane and parallel to $y=0$ and $x=y$ planes in solenoid on only case.

We use very large constant permeability of iron parts to ensure that \mathbf{H}' is zero in them. Special mesh reflection around $\text{Phi}=22.5$ deg line procedure was developed to ensure that the resulting \mathbf{H}' is precisely periodic.

We can call the problem *periodic* if it translates into itself after rotation around some axis on $2\pi/n$ angle. We can call the problem *antiperiodic* if it translates into itself after rotation around some axis on $2\pi/n$ angle with the change of the sign of currents. Atlas is periodic relating to $2\pi/8$ rotation around Z-axis and antiperiodic relating to $2\pi/2$ rotation around $\text{Phi}=22.5$ deg axis.

We will show now that the the requirement $\varphi(0) = \varphi(\pi/4)$ must be fulfilled for corresponding points in periodic problems and $\varphi(\alpha) = -\varphi(\pi/4 - \alpha)$ for antiperiodic problems to expand the magnetostatics field continuously to all the space:

1. The resulting field will satisfy to Maxwell's equations inside the problem domain.
2. The resulting field is continuous on the boundaries.

We use variation formulation of Maxwell's equations for 3D magnetostatics to introduce the magnetic potential.

Variation integral we choose in the following form:

$$I = \int (\mu(\nabla\varphi)^2 / 2 + \rho\varphi) dV ,$$

where $\rho = \text{div}\mu\mathbf{H}'$.

During the first variation over potential, we suppose that μ is not varied, then

$$\delta I = \int (\mu(\nabla\varphi)(\nabla\delta\varphi) + \rho\delta\varphi) dV = 0 ,$$

Using the relation $\mu\nabla\varphi\nabla\delta\varphi = \nabla(\mu\nabla\varphi\delta\varphi) - \nabla(\mu\nabla\varphi)\delta\varphi$, we will have

$$\delta I = \int (\text{div}\mu \text{ grad}\varphi - \text{div}\mu\mathbf{H}')\delta\varphi dV + \int (\mu \text{ grad}\varphi)\delta\varphi d\mathbf{S} = 0 .$$

If we choose the variation $\delta\varphi$ as an arbitrary one *inside* the problem domain, we obtain from the volume integral:

$$\text{div}\mu \text{ grad}\varphi = \text{div}\mu\mathbf{H}' ,$$

so the resulting field will satisfy Maxwell's equations inside the problem domain.

If we choose $\varphi_i = \varphi_{\tilde{i}}$ in any two corresponding nodes i and \tilde{i} for periodic problem then $\delta\varphi_i = \delta\varphi_{\tilde{i}}$ so $\mu \text{ grad}\varphi_i d\mathbf{S} = -\mu \text{ grad}\varphi_{\tilde{i}} d\mathbf{S}$ which corresponds to the continuity of the normal component of \mathbf{B} .

If we choose $\varphi_i = -\varphi_{\tilde{i}}$ in any two corresponding nodes i and \tilde{i} for antiperiodic problem then $\delta\varphi_i = -\delta\varphi_{\tilde{i}}$ so $\mu \text{grad}\varphi_i d\mathbf{S} = \mu \text{grad}\varphi_{\tilde{i}} d\mathbf{S}$ which again corresponds to the continuity of the normal component of \mathbf{B} .

Because of the fact that the requirements $\varphi_i = \varphi_{\tilde{i}}$ and $\varphi_i = -\varphi_{\tilde{i}}$ correspond to the continuity of the tangential component of \mathbf{H} for periodic and antiperiodic problems we proved the continuity of the field on the boundaries.

The simplest way to make $\varphi_i = \varphi_{\tilde{i}}$ (or $\varphi_i = -\varphi_{\tilde{i}}$) on the boundaries is to add $\lambda(\varphi_i - \varphi_{\tilde{i}})^2/2$ (or $\lambda(\varphi_i + \varphi_{\tilde{i}})^2/2$) term with sufficiently large λ parameter to the discrete approximation of the variation integral. This corresponds to the addition of some positive definite Λ matrix to the precondition matrix B of iteration scheme of finding the solution of nonlinear magnetostatics equations:

$$B(\mathbf{y}_{k+1} - \mathbf{y}_k) / \tau + A\mathbf{y}_k = \mathbf{f},$$

where A is the matrix of the equations system, \mathbf{f} is the right hand, \mathbf{y} is the node potential function, B and τ are some solution strategic matrix and parameter, correspondingly.

Because the Λ matrix is positive definite the convergence of nonlinear iterations will be the same (or better) as for local boundary conditions [Samarsky, Nikolaev, 1978].

Let us consider the correction of the cyclic conjugate gradients (CCG) method used by Mermaid for the solution of the linear equations necessary because of the introduction of additional links (coefficients) into the sparse precondition matrix B .

According to the red-black ordering scheme [Young, Hageman, 1981] necessary for the CCG method, we have to distribute the nodes into two sets so that the links inside every set (red or black) should be resolved directly.

In the case of local boundary conditions such a distribution can be achieved by taking nodes on odd lines in Y direction in the first Z plane, even on the second plane and so on as the red set and reminding nodes as the black one. The links inside every set comprise the three-diagonal matrix, which can be simply resolved by the direct method.

In the case of periodic boundary conditions the required ordering can be achieved by using of L shaped lines instead of straight Y lines (Fig 2.). The inversion of the L shaped lines that have not the nodes belonging to the periodical boundary condition set is a simple three-diagonal matrix inversion. Let us consider now the L shaped line, which has nodes belonging to the periodical boundary conditions set. If we reorder the nodes taking them from X and Y planes alternatively we will have five-diagonal matrix, which can be inverted easily (Fig. 2).

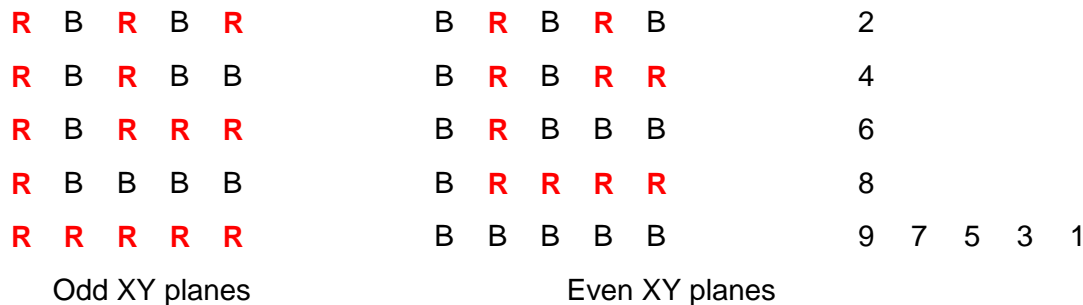


Figure 6 Red-black ordering of unknowns and the ordering of unknowns inside L-line.

Results of calculations

The calculation time for the model described above is 22 hours on Pentium II/300 MHz/0.5 GB of RAM, which is ten times more than an average calculation time for ordinary problems of similar dimension (the number of linear iterations is about 10,000). This can be explained by the fact of fine lamination of the TileCal iron, similar behavior was observed in Babar detector calculations.

The number of nonlinear iterations is equal to 25, which are typical for the problems with moderate level of iron saturation. The accuracy of the results can be estimated as several percents in the TileCal and 10^{-3} in the rest major parts of the detector. Several field map pictures are presented below to illustrate the general level of granularity of the solution [Dubrovin, Vorojtsov, 1999].

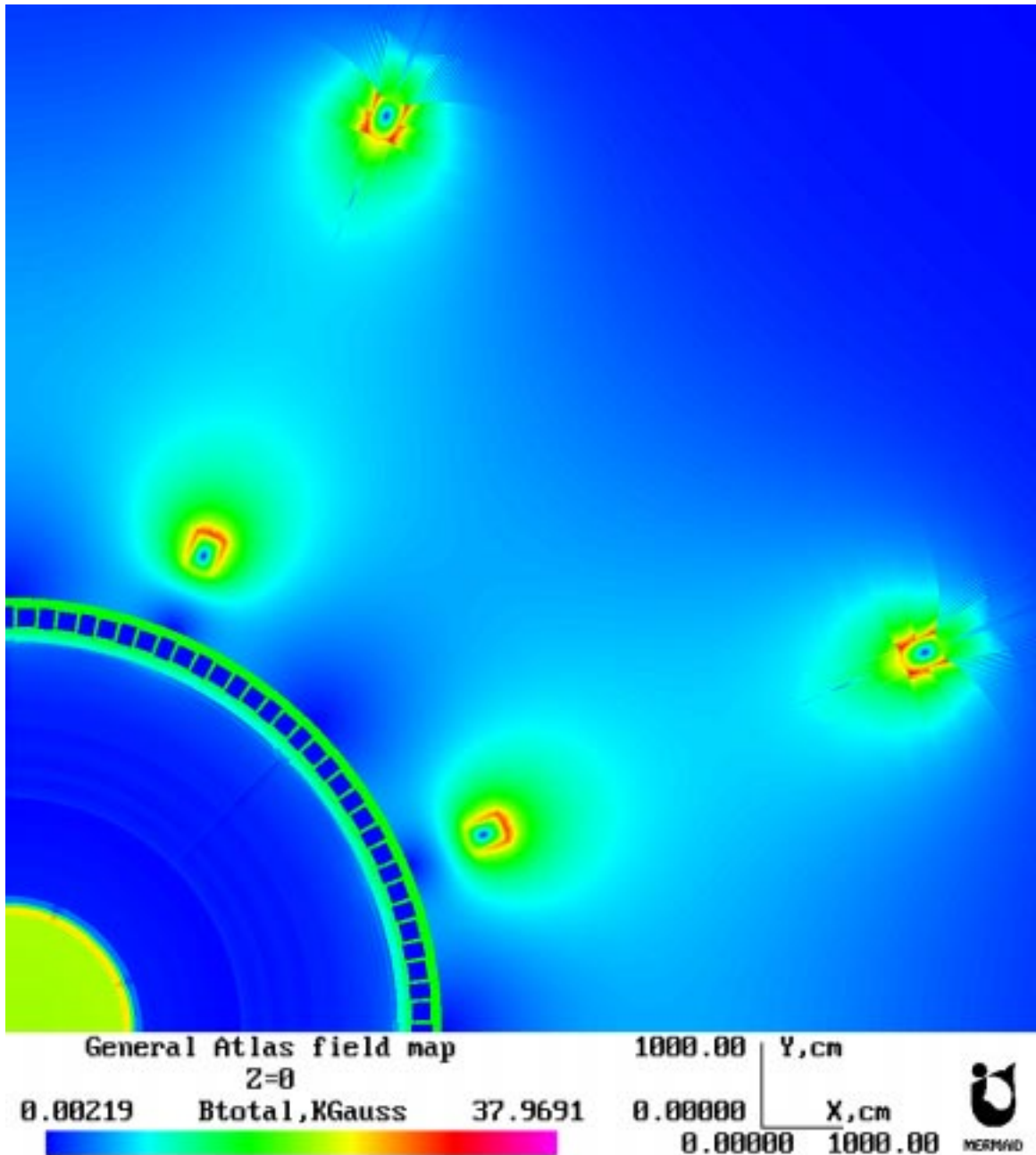


Figure 7 R- ϕ plane, Z=0, $\phi=0^\circ+90^\circ$

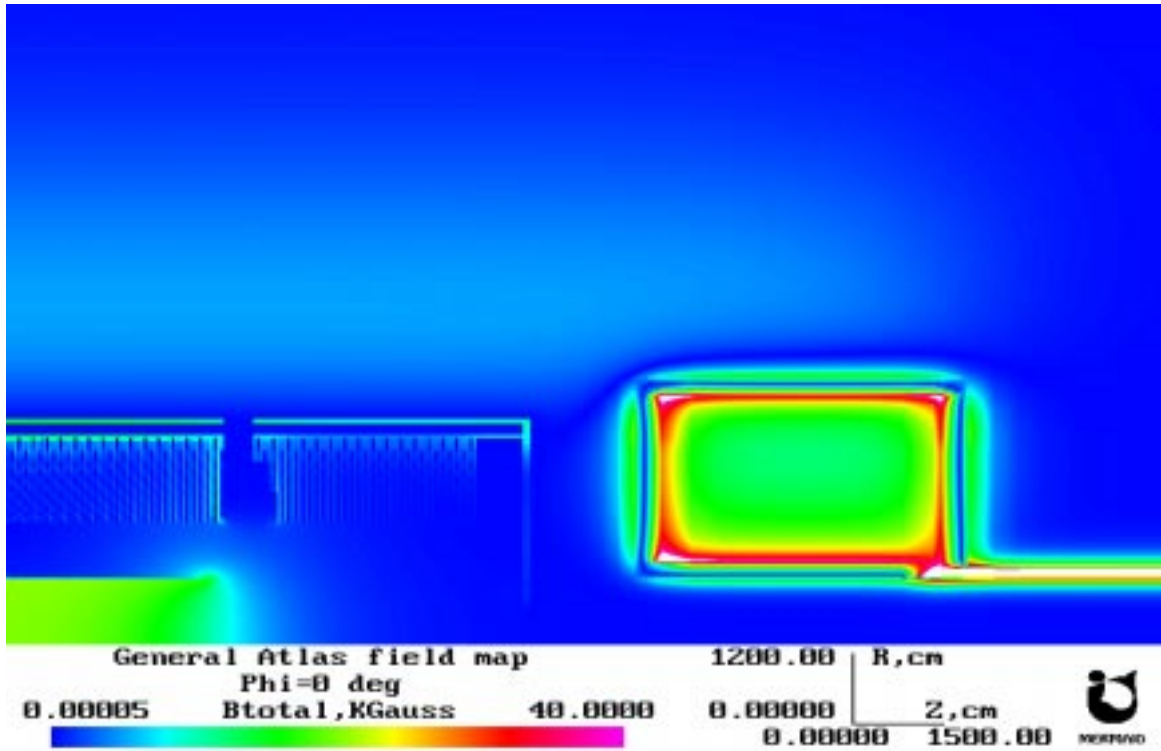


Figure 8 R-Z plane, $\phi=0^\circ$, Bmax=4 Tesla

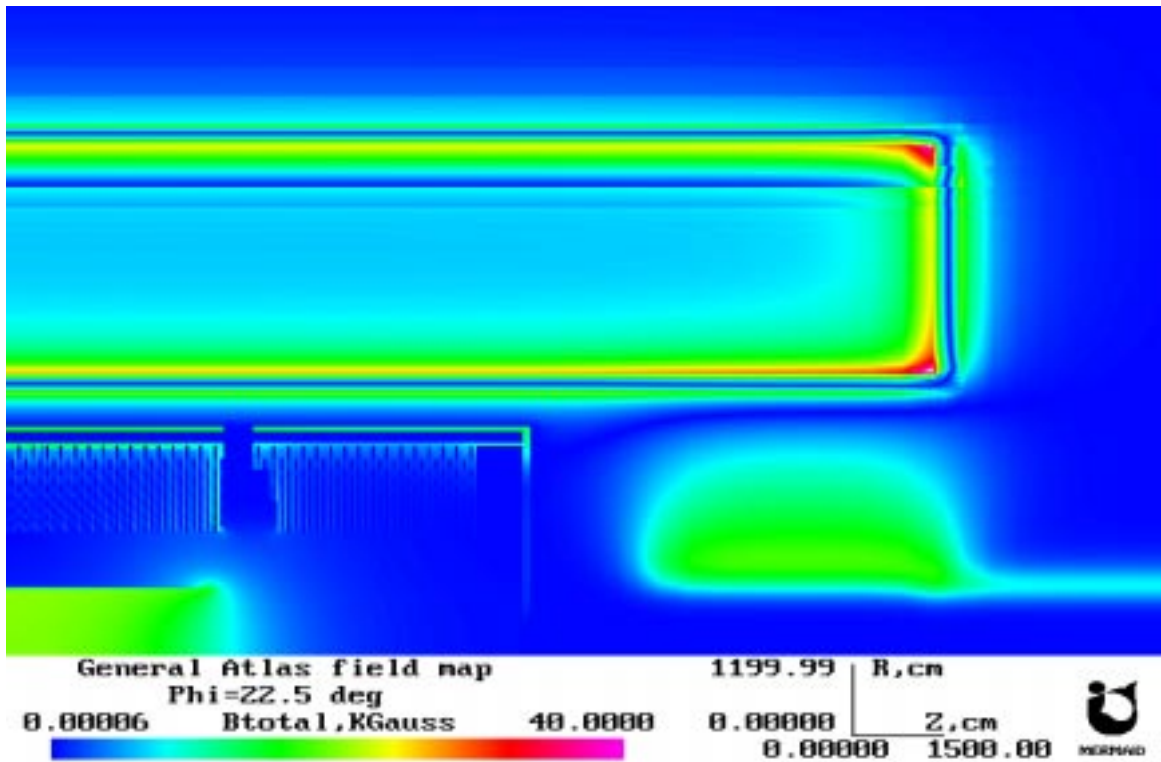


Figure 9 R-Z plane, $\phi=22.5^\circ$, Bmax=4 Tesla

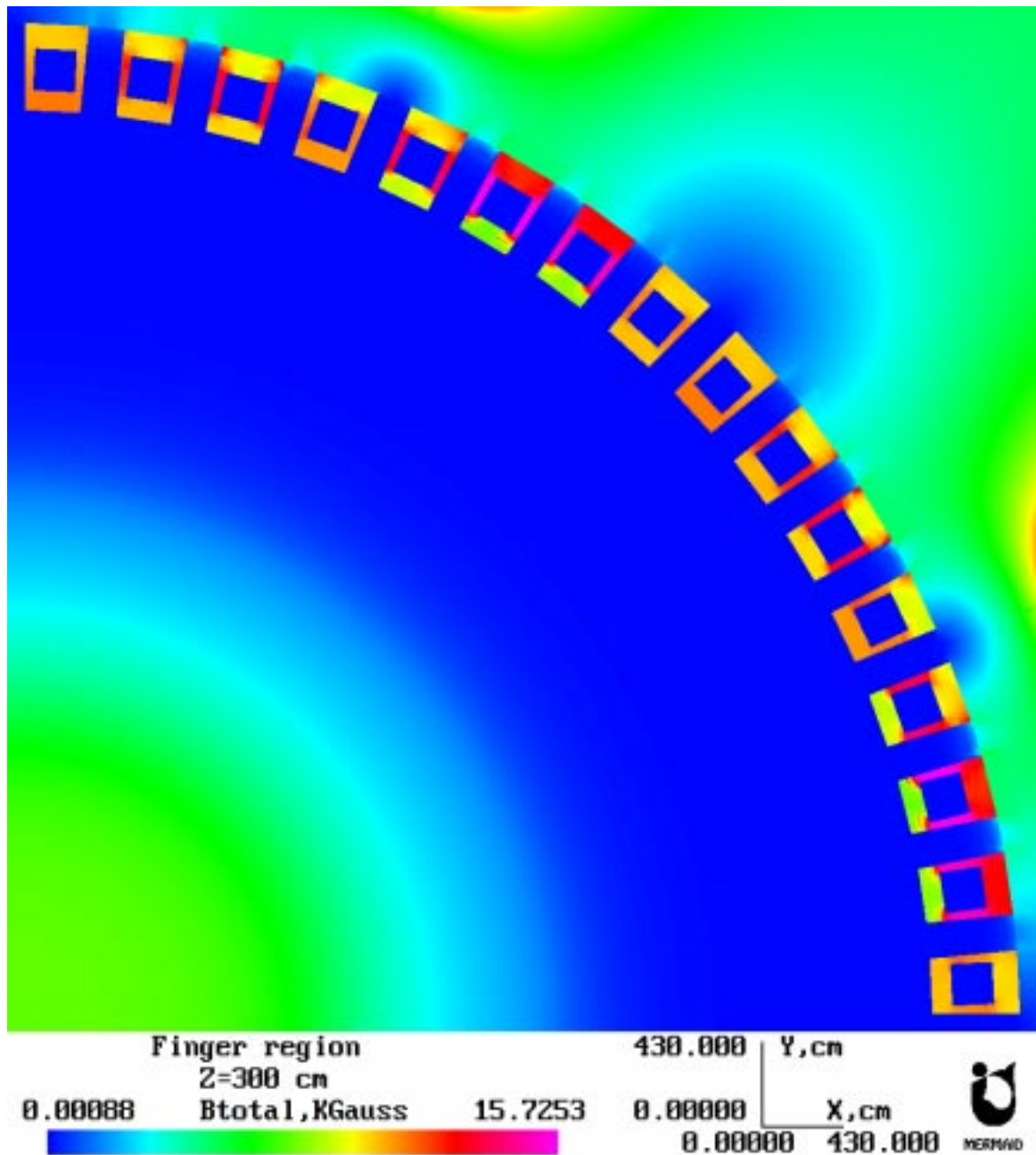


Figure 10 R- ϕ plane, Z=300 cm, $\phi=0^\circ\div 90^\circ$

Parallel processing calculation scheme

Because of highly regular topological structure of the mesh used by Mermaid and cyclic nature of the CCG method the parallel processing scheme of linear iterations is rather simple [Young, Hageman, 1981].

Let us divide the complete set of Z meshes planes into N consecutive subsets where N is the number of processors. Let every processor has the information (the problem matrix and initial values of the potential) only about the part of the problem domain covered by its own subset of Z mesh planes and about two additional Z mesh planes before and after the its subset.

Then all the processors can simultaneously calculate new red node potentials on the base of previous red node potentials and the black ones. After exchange of the information about those newly calculated red node potentials on the boundaries,

all the processors can calculate new black potentials. After the exchange of newly calculated black node potentials on the boundaries and of integrated information about subsets, new iteration parameters can be estimated and all the information is available to all the processors to make the next iteration and so on until the end of the current nonlinear iteration.

It can be seen that the efficiency of this calculation scheme is near 100 percents if the number of processors is smaller at least in 3-5 times than the total number of Z mesh planes and if the speed of the information exchange between the processors is not very poor in comparison with the speed of calculations.

This calculation scheme was checked out on the 1D model with the help of 4-processor computer and showed 4 times increase of the calculation speed.

Realistic Atlas calculations

In the previous section we have described the parallel processing procedure only for the linear part of problem solution. The reason is that the recalculation of the nonlinear iteration matrix is not time-critical for the Atlas case. Indeed the number of nonlinear iterations is in several hundred times less than the number of the linear ones.

The full scale Atlas model calculation requires 10 times increase of the number of mesh nodes for the total of 40 –50 millions and increase in 10-20 times of the speed of calculation to limit the calculation time with 20-40 hours.

Because we need about 50 bytes of information per one mesh node so about 2.5 GB of total RAM is required.

Hence, the minimal computer configuration of 10 processors with 256 MB of RAM available for each processor is required to calculate the full Atlas model.

Conclusion

The new Mermaid 3D periodic boundary condition solver is appeared to be adequate to resolve the fine Atlas TileCal iron structure problems. However, the computation time is large enough to raise a question about the use of more powerful computers with parallel processing and larger memory.

Acknowledgements

The authors are deeply indebted to Dr.M.Nessi for his support and constant interest to the subject of the study, Antokhin E.I. and Sutkin P.Yu for help in input of a number of Atlas models, Vobly P.D. for useful discussions of the results of calculation.

References

1. Bergsma F., Calculation of the Atlas magnetic field, ATLAS Internal Note, TILECAL-NO-054, 1995.
2. Samarsky A.A., Nikolaev E.S., The methods for solving of mesh equations, Nauka, Moscow, 1978.
3. Young D.M., Hageman L.A., Applied Iterative Methods, Academic Press, 1981.
4. Dubrovin A.N., Vorojtsov S.B., ATLAS Magnetic Field Calculations by Mermaid with All the Coils on, ATLAS Internal Note, ATL-TECH-99-001, 1999.