*27th EPS Conference on Contr. Fusion and Plasma Phys. Budapest, 12-16 June 2000 ECA Vol.* **24B** *(2000) 664 -667*

# Grid Refinement for the B2 plasma edge simulation code

H. Bürbaumer, R. Schneider*, D. Coster*, F. Aumayr, HP. Winter

*Institut f. Allg. Physik, Ass. EURATOM-ÖAW, Wiedner Hauptstr. 8-10, A-1040 Wien, Austria*

**Max-Planck-Institut f. Plasmaphysik, IPP-EURATOM Association, D-85748 Garching*

**Abstract**

The finite volume fluid code package B2, one of the standard scrape-off layer (SOL) transport computer codes, is extended to allow grid refinement/coarsening. To achieve the goal of a flexible, self-adjusting grid mechanism the B2 code package is set on a new numerical basis. First the staggered grid is being transferred into a non-staggered grid, and next a two-mesh structure is introduced to the simulation code with a flexible numerical mesh acting on a very fine basis mesh containing the magnetic flux surface structure of the fusion plasma. The complete two-mesh grid refinement/coarsening method and first tests of the new code are presented.

## 1. Introduction

In the last years sophisticated tools for the computational modeling of the plasma edge region have been developed. Standard SOL-codes like B2([1]) use a finite-volume scheme to describe the Braginskii fluid equations [2] numerically. These codes have reached a high degree of sophistication in describing the physics involved in the edge region [3]. However, some enhancements have to be made to the code for ist future use. Especially in the divertor region of a fusion plasma strongly differing local gradients and sharp thermal fronts have to be treated numerically which requires grid refinement for these regions. Another goal is the extension of the B2-code to the core-region: To achieve an universal code for the whole plasma region the physically 1D core region could be simulated in a 2D fluid code by creating thin core-cells going all around the magnetic O-point defining a quasi 1D cell-structure. For this very reason a special kind of grid coarsening enabling such complex cells would be favorable.

Grid refinement in finite element codes has been investigated widely [4-7], some work has also been done for finite volume [8]. These codes did, however, not have as comprehensive a physics basis as the B2 code and, in the finite volume case, was not capable of creating a quasi-1D core structure by grid coarsening. Section 2 discusses the work necessary for grid refinement and the the method used, section 3 the implementation in B2. Section 4 discusses first tests of the new code.

## 2. The Two-mesh refinement/coarsening method

### a) Preparatory work

B2 uses the finite-volume technique and the SIMPLE algorithm byPatankar [10], which is based on a staggered grid. A staggered grid consists of a separate grid for the velocity terms, which prevents numerical pressure-oscillations. It is much more stable than a normal non-staggered grid but especially in the case of grid refinement the two staggered grids pose a problem. To overcome this problem we adapted a method which has been used up to now mainly in fluid dynamics, the method ofRhie and Chow [9], for our purpose. It consists of a non-staggered grid with a numerical correction term added to the velocity interpolation. In a

first step towards grid refinement/coarsening the B2-code has been transferred to a non-staggered grid and the correction of Rhie and Chow has been added.

**b) Grid refinement/coarsening using a two-mesh-procedure**

To enable grid refinement/coarsening in the new B2-solps5.0 code we use two meshes: A very **fine curvilinear quasi-orthogonal 2D** mesh produced directly from the magnetic flux surface structure of the experimental discharge acts as a **basis grid** (see Fig.1, left side). The main geometric properties as well as magnetic values of these basic grid cells are stored in 2D matrices representing the basic grid.

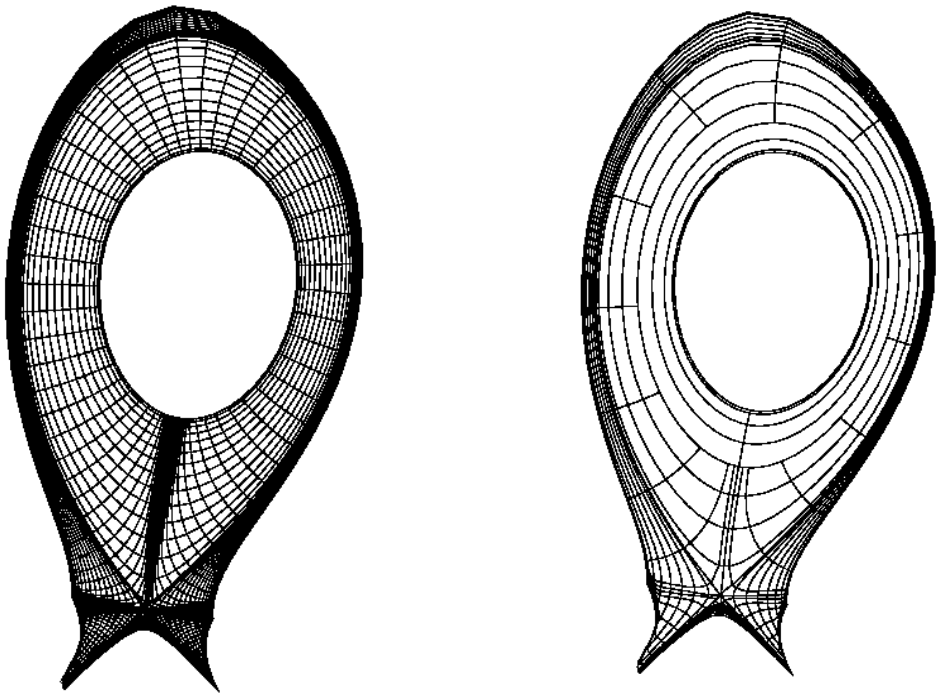

*Fig.1: Basic grid (128x32 cells) (left) and (right) numerical grid with core-cells near the center*

The **numerical grid** on which the main physics of the B2-code is calculated and the fluid equations are solved is a **rough, flexible mesh** that can be modified depending on the physical situation (see Fig. 1 right side). Each cell of the numerical mesh consists of a certain number of grid cells of the basic mesh defined by lower and upper index values in x and y (parallel and perpendicular to the magnetic field) direction. Physical values like densities, fluxes, etc. exist only on the numerical grid and are stored in a 1D-matrix structure. The geometrical information stays in the 2D basic grid structure and can always be recalled. When a numerical grid cell is refined, no interpolation of new cell borders is needed as the underlying basic grid is always present. The refinement level which can be reached is given by the accuracy of the basic grid. Highest refinement is reached locally when a cell of the coarse numerical grid is identical with a cell of the fine basic grid. The number of possible neighbours is given in the code by a parameter *nmax* which defines the abruptness of changes in grid refinement in the code. Following this strategy also round grid cells are possible as the numerical cell may

consist of a large number of basic cells forming curved structures (see Fig.1, right side in the core region). Using quasi-1D cell structures generated by grid refinement/coarsening enables a quasi-1D treatment for the plasma core, one of the major goals for the future B2-code package.

## 3. Realization of the grid-refinement procedure in B2

### a) Matrix solver

The fluid equations [2,11] in the standard B2-code were transformed into the following set of nx (number of grid-cells parallel to the magnetic field) byny (perpendicular) equations with $n_{ij}$ representing the density, temperature or velocity at positioni in the x-direction and j in the y-direction in the old 2D-grid structure [10]:

$$a_{ij}n_{ij-1} + b_{ij}n_{i-1\,j} + c_{ij}n_{ij} + d_{ij}n_{i+1\,j} + e_{ij}n_{ij+1} = r_{ij}$$

This set of equations can be written in a matrix-structure An=r with a square 2D-matrix A with nx by ny side-length, $n_{ij}$ and $r_{ij}$ have the form of a nx by ny vector. This was solved in the old B2-code using standard matrix-solvers. In the new version the variables are represented by a 1D-matrix structure $n_k$ with k as the cell index. Using a 5-point solver every cell-variable is influenced by up to 4 by nmax neighboring cell values resulting in k=1..nn (nn is the number of cells in the grid refined mesh) equations of the form:

$$\sum_{l=bottom} a_{k,l}n_1 + \sum_{l=left} b_{k,l}n_l + c_k n_k + \sum_{l=right} d_{k,l}n_l + \sum_{l=top} e_{k,l}n_l = r_k$$

k stands for the cell index and l is varied over the bottom, left, right or top neighboring cells. This results in a matrix-structure An=r with a square 2D-matrix A with side-lengthnn; $n_k$ and $r_k$ have the form of a nn vector. It can be solved using similar matrix-solvers as above, but as there are more non-zero matrix-elements than in the non-refined code-version it is more time-consuming to be solved.

### b) Interpolation schemes

Two different interpolation schemes can be used within the new B2-solps5.0 code. In the old B2-code, interpolation has been mainly done by volume-averaging. As grid-refinement/coarsening may lead to quite complicated grid structures, the volume of a cell is a bad representation of the distances in the grid. For this very reason two alternative interpolation methods have been introduced into the B2-code. One uses the exact distances between the places on which new interpolated values are located and the neighboring grid cells. The distances of interest within a cell and its neighboring cells are calculated every time grid refinement/coarsening is done and can be used for interpolation without losing extra computer-time. The advantage of more accuracy is outweighed by the problems occurring in quasi-1D cells representing the core plasma. In these cells no cell center is defined at all. For this very reason a simpler, but in the case of core cells preferable interpolation scheme has been used in the first version of the code. Here the value in a neighboring cell is weighted using the inverse product of the cells diameter in the direction to the interpolated value.

**c) Grid-refinement condition and optimization of working memory**

In case the normalized local gradient of one of the main plasma variables as ion density, -velocity, and ion or electron temperature exceeds a certain value, grid refinement takes place in the direction of the gradient. This results in a refined structure in areas with strong local gra-dients. Grid coarsening is done if all of the main plasma gradients lie below a certain value. As each refinement process would enlarge all matrices used in the code, the matrix dimensions in the code are larger than they need to be. When this reserved matrix space is used up, a new allocation of the whole set of matrices in the code takes place enlarging the matrix dimension by a factor given by the user. If grid coarsening takes place and the used matrix-storage space falls below a certain percentage, the allocation process is also started reducing the matrix dimensions to optimize working memory.

**4) Tests of the method within B2-solps5.0**

The convergence of the new non-staggered code has been tested on pure hydrogen plasmas as well as on plasmas with deuterium and carbon. In the case of a pure hydrogen plasma the converged solutions for a long run with the new code are almost identical to the solutions with the standard B2.5 code. In the more complex case of a plasma with impurities, a well converged staggered grid run has been continued with the new non-staggered grid code, the residuals increase at the beginning but the run converges rapidly towards a solution close to the converged solution of the old code. As a whole the convergence of the new code is similar to the one with the old staggered B2.5 code. The grid refinement/coarsening module is working well but the complete transition of the code for grid refinement is not finished.

**Conclusions and Outlook**

The standard SOL plasma fluid code package B2-solps5.0 has been set on a new numerical basis. The staggered grid has been transferred into a non-staggered grid including a correction to prevent pressure oscillations and other numerical problems. A grid refinement/coarsening procedure has been introduced to the code creating two meshes. A fine, curvilinear quasi-orthogonal 2D mesh produced directly from the magnetic flux surface structure of the experimental discharge acts as a basis grid. Using this grid a rough, flexible mesh is refined depending on local density and temperature gradients in the plasma. The method has been especially prepared to create quasi 1D-core cell structures in the core plasma by grid coarsening, an important requirement towards an universal code for the whole plasma.

[1] B.J.Braams, *Computational Studies in Tokamak Equilibrium and transport*, PhD thesis, Rijkuniversiteit, Utrecht, Netherlands, 1986.
[2] S. I. Braginskii, in Reviews of Plasma Physics, Vol. 1, edited by M. A.Leontovich, p. 205
[3] R. Schneider et al., Proceedings of the IAEA Conference, Montreal, Canada 1996
[4] R. Zanino, Max-Planck-Institut Report IPP 5/43 (1991)
[5] R. Zanino, J. Nucl. Mater. 196-198 (1992) 326.
[6] Th. Puetz et al., Contrib. Plasma Phys. 34 (1994) 404.
[7] R. A. Vesey and D. Steiner, J. Comput. Phys. 116 (1995) 300.
[8] F. Subba and R. Zanino, *A 2D fluid model of the scrape-off layer using adaptive unstructured Finite Volumes,* Proceedings of the PSI-14 conference, Rosenheim (Germany), May 2000.
[9] C.M Rhie and W.L. Chow, AIAA J., 21 (11), 1525-1532 (1983)
[10] S.V. Patankar, Numerical Heat Transfer and Fluid Flow, Hemisphere, New York, 1980
[11] B.J.Braams, *Technical Report 68*, Next European Torus, 1987.