GPU-accelerated CarMa code for Resistive Wall Modes analysis

F. Villone[1], A. G. Chiariello[1], Y. Liu[2], S. Mastrostefano[1],

M. Nicolazzo[3], G. Rubinacci[3], S. Ventre[1]

[1]*Ass. EURATOM/ENEA/CREATE, DAEIMI, Università di Cassino, Italy*

[2] *Euratom/CCFE Fusion Association, Culham Science Centre, Abingdon, OX14 3DB, UK*

[3]*Ass. EURATOM/ENEA/CREATE, DIEL, Università "Federico II" di Napoli, Italy*

Abstract

We have accomplished the GPU-acceleration of some key computations required by the CarMa code, used for the analysis of Resistive Wall Modes (RWM) in presence of three dimensional volumetric conductors. Significant accelerations (speed-up factors greater than 10) have been achieved on the computation of specific matrices of the overall model. This computational tool is used to study the effect of thick blanket modules on the n=0 RWM (axisymmetric vertical stability) for a scenario-4 ITER configuration.

Introduction

Resistive Wall Modes (RWM) are ideal MHD instabilities (usually external kink) that often set performance limits to advanced scenarios of present and future fusion devices (e.g. ITER), whose design hence requires reliable and accurate modelling of these modes. With this in mind, in the last years huge efforts have been put by several research groups on crucial aspects of RWM theoretical and numerical modelling, namely, on the one hand, the inclusion of kinetic effects and, on the other hand, a realistic description of the conducting structures surrounding the plasma. The CarMa code [1], which allows the analysis of the effects on RWM of 3D volumetric conducting structures, is one of the computational tools developed with this goal. Unfortunately, accurate numerical models are usually very demanding from the computational point of view, giving rise to so-called huge numerical models, requiring special techniques to be dealt with. For example, state-of-the-art fast and parallel computing techniques have been successfully applied to the CarMa code [2-3], allowing an unprecedented level of accuracy in the realistic description of active and passive conducting structures. In this stream, a promising technical development is the use of a GPU (Graphics Processing Unit) to perform scientific computing. To this purpose, a dedicated cluster has been built at the Università di Cassino, featuring 144 GB RAM, 32 CPUs, 4 GPU units with a total of more than 1500 GPU cores.

The CarMa code and the GPU acceleration

The CarMa model [1] allows the analysis of RWM with 3D structures, including a volumetric description of the conductors in terms of a finite elements mesh. The resulting mathematical model is of the form:

$$\begin{aligned} &\underline{\underline{L}}^{*}\frac{d\underline{I}}{dt}+\underline{\underline{R}}\,\underline{I}=\underline{V} \\ &\underline{y}=\underline{\underline{C}}\,\underline{I} \end{aligned} \qquad (1)$$

where $\underline{I}$ is a vector of discrete 3D currents (both in active conductors and in passive conducting structures), $\underline{\underline{L}}^{*}$ is a modified inductance matrix, which takes into account the presence of plasma, $\underline{\underline{R}}$ is a 3D resistance matrix, $\underline{V}$ is related to external voltages, $\underline{y}$ is an output vector providing the magnetic field perturbation at given points, and $\underline{\underline{C}}$ is a suitable output matrix.

Matrices $\underline{\underline{L}}^{*}$ and $\underline{\underline{C}}$ in (1) can be written as:

$$\begin{aligned} &\underline{\underline{L}}^{*}=\underline{\underline{S}}\,\underline{\underline{Q}} \\ &\underline{\underline{C}}=\underline{\underline{T}}\,\underline{\underline{Q}} \end{aligned} \qquad (2)$$

where $\underline{\underline{S}}$ and $\underline{\underline{T}}$ are matrices comprising the plasma response, while $\underline{\underline{Q}}$ is a matrix providing, through Biot-Savart integral, the magnetic field of each 3D discrete current over a coupling surface (a mathematical surface located in between the plasma and the conducting structures). This magnetic field can be computed at given points of the surface or can be Fourier decomposed along the poloidal angle. The computation of the $\underline{\underline{Q}}$ matrix is one of the most time-consuming computations, at least for small- or medium-size computational models [2]. Hence, we decided to tackle the GPU acceleration of this computation.

The modern programmable GPUs are massive multithreading (up to 512 cores), have a relatively small cache memory compared to central processing units (CPUs), and a very simple control unity. As data-parallel coprocessors, GPUs can be used to solve compute-intensive science and engineering problems. Typically these applications process a large amount of data and/or perform many iterations on the data: they are usually recast into sub-problems that can be safely solved at the same time using the modern GPUs. Usually most applications will use both CPUs and GPUs, in such a way that GPUs complement CPU execution: the sequential parts of the core is performed very well on the CPU, the numerically intensive parts instead on the GPUs.

In the matrix assembly, a serial code would obtain the generic (*i,j*) entry of the $\underline{\underline{Q}}$ matrix for the *j*-th discrete degree of freedom (DoF) and for the *i*-th point of the coupling surface, by implementing two nested loops on mesh elements and on surface points. Conversely, in our implementation, the computational kernel has been designed in order that every thread computes separately a single finite element – point interaction; the partial result is stored in a temporary matrix in the GPU's global memory (3GB). Finally, the CPU assembles all these contributions in the final matrix stored in the host memory. If the number of threads is less than the number of physically available GPU cores, then all the threads are run in parallel; otherwise, the computation is serialized in blocks of suitable dimensions.

Results

We analysed a Scenario-4 ITER configuration (I_p = 9 MA, β_p= 1.92, l_i =0.61), considering the *n*=0 RWM (axisymmetric vertical stability). We assumed three different conducting structures circumventing the plasma:

- Case 1: axisymmetric vessel (toroidally continuous copper cladding)
- Case 2: 3D vessel (with ports extensions and divertor cassette)
- Case 3: 3D vessel + blanket modules (only shielding blocks)

In all cases, we also considered the axisymmetric in-vessel active coils, as current-fed conductors. Fig. 1 reports the mesh of Case 3. Due to the symmetry of the structures and of the plasma, only 20° in the toroidal direction have been discretized, with a mirror symmetry condition plus 9 rotational symmetries.

Table 1 reports the computed growth rates for the unstable *n*=0 RWM for the three cases above, and the speed-up obtained with respect to standard computation with no GPU acceleration. This speed-up refers only to the computation of the $\underline{Q}$ matrix described above.

As expected, with respect to the axisymmetric case (Case 1), the 3D vessel causes a higher growth rate (Case 2), while the insertion of blankets (Case 3) slows down the instability.

Figure 2 reports the transfer function between the up-down antisymmetric current in the axisymmetric in-vessel active coils and the *n*=0 Fourier component of the magnetic field at r=8.916 m, z=0.550 m. The most noticeable effect of 3D structures is a variation of the phase at relatively high frequencies.

This work was supported in part by Italian MIUR under PRIN grant#2008E7J7A3 and by NVIDIA, through the donation of several GPU units.

	Case 1	Case 2	Case 3
γ [s^{-1}]	4.55	5.03	4.86
# dof	994	8705	12861
Speed-up	17.1	11.1	11.4

Table 1. Growth rates (s^{-1}) of the n=0 RWM, number of discrete unknowns and GPU acceleration speed-up.

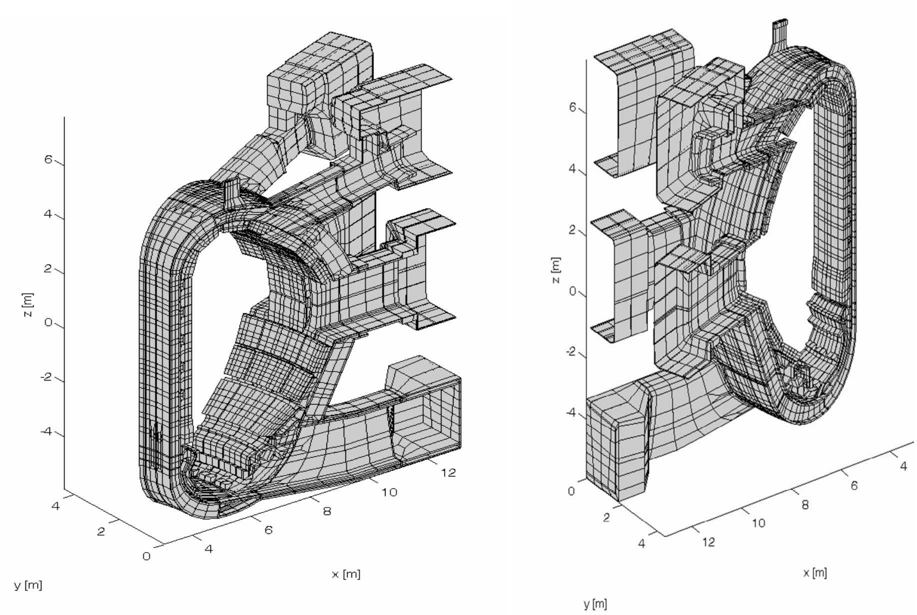
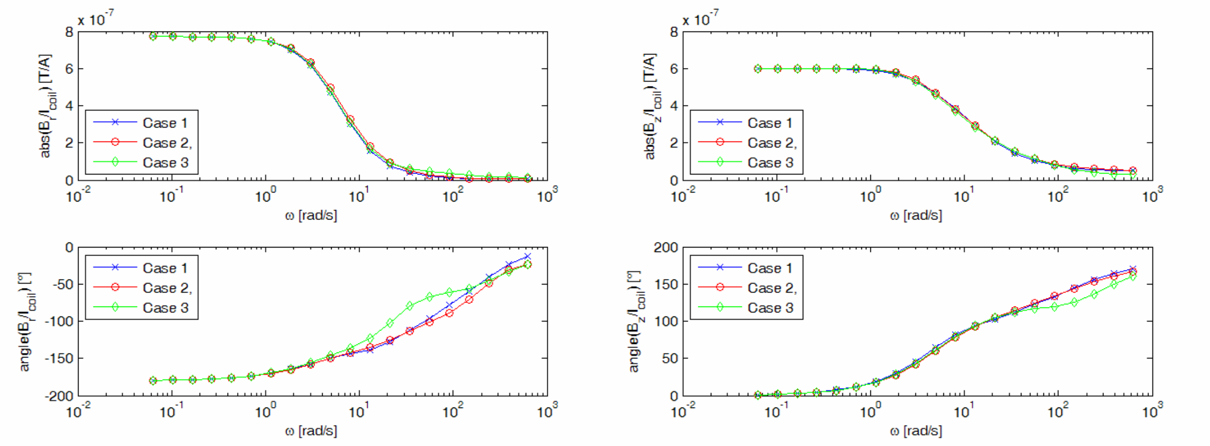


Fig. 1. Mesh for Case 3.



(a) (b)

Fig. 2. Transfer functions: (a) radial magnetic field; (b) vertical magnetic field

[1] A. Portone, F. Villone, Y. Liu, R. Albanese, G. Rubinacci, *Plasma Phys. Control. Fusion* **50** (2008) 085004

[2] G. Rubinacci, S. Ventre, F. Villone, Y. Liu, *J. Comp. Phys.* **228** (2009) pp. 1562-1572

[3] F. Villone, Y. Liu, G. Rubinacci, S. Ventre, *Nucl. Fusion* **50** (2010) 125011