The implementation of real time parallel equilibrium reconstruction in EAST PCS

Q.P. Yuan [a], X.N. Yue [b], X.F. Pei [a], Z.P. Luo [a], R.R. Zhang [a], B.J. Xiao [a, b]

[a] *Institute of Plasma Physics, Chinese Academy of Sciences, Hefei, China*

[b] *School of Nuclear Science & Technology, University of Science & Technology of China, Hefei, China*

Abstract The accuracy and efficiency of plasma real time equilibrium reconstruction is essential for plasma shape and kinetic control on advanced tokamak devices. In order to increase the spatial resolution of plasma calculation area and accelerate the reconstruction speed, a parallel code named PEFIT is developed. Using the massively parallel processing cores of graphic processing units (GPU), it took only 0.22 milliseconds for PEFIT to complete an equilibrium reconstruction iteration with grid size 65×65. To apply such real time parallel reconstruction system for EAST plasma control, data interface between PEFIT and EAST PCS was designed. And the exchanging data were transferred through reflective memory board (RFM). Benchmark test was carried out to validate the correctness and robustness of the integrated system using history experimental data. The results are quite encouraging to deploy the PEFIT for plasma control in next EAST operation campaign.

Keywords Parallel equilibrium reconstruction; Graphic processing units; Plasma control system; EAST;

1. Introduction

EAST is the first fully superconducting tokamak with D-shaped poloidal cross section as shown in Figure 1. There are 38 magnetic probes measuring poloidal field and 35 flux loops measuring poloidal flux which are mounted on the vacuum vessel. The plasma current and poloidal coil currents are measured by Rogowski coils. There are 14 poloidal field (PF) coils used for plasma heating and shaping/equilibrium control.

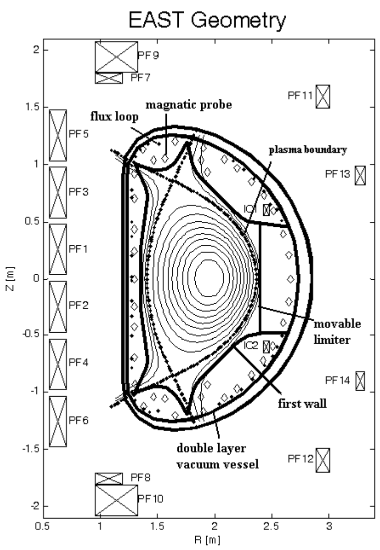

Fig.1. Sectional view of EAST device

To make the best use of the available volume and to ensure good passive stabilization, accurate shape control must be guaranteed. Unfortunately, the plasma shape parameters are not directly measured but can be evaluated from the available diagnostic data. Thus, shape estimation assumes a key role in fulfilling the control requirements.

In EAST, the real time equilibrium reconstruction code (rtEFIT) [1] is applied in the plasma control system (PCS) [2] to obtain plasma shape parameters and generate control errors for shape control algorithms. The rtEFIT is the real time version of the equilibrium reconstruction code EFIT, which approximately solves the Grad-Shafranov equation with some simplifications. To satisfy the fast control needs, the rtEFIT uses lower spatial resolution, in which the 2.4 m×1.4 m rectangle plasma area is divided by 33×33 grids. Although the computation time is less, such rough grid division has bigger discreteness error [3]. In addition, the code is divided into two portions: fast loop and slow loop [1]. The fast loop uses data set from slow loop as the starting point equilibrium and performs least square fit using the most recent diagnostic data to generate control errors in 250 microseconds each cycle. And the slow loop needs 2~3 milliseconds to finish one equilibrium iteration and update the data set for fast loop. Thus, fast loop reuses the

author's email: qpyuan@ipp.ac.cn

data set until new data set is ready. Most of the time, the result is close enough to the well-converged reconstruction. But when plasma shape changes rapidly, the reused starting point equilibrium will have huge difference with the real plasma profile, which affects the accuracy of one iteration result in fast loop and depress the shape control performance.

In order to accelerate real-time reconstruction algorithm with higher spatial resolution, a new parallel equilibrium reconstruction code PEFIT [4] was developed as described in section 2. Through efficiently making use of the massively parallel processing cores of graphic processing units (GPU), PEFIT could complete an equilibrium reconstruction iteration in 220 microseconds with a 65×65 grid number. To apply such real time parallel reconstruction system for plasma shape control in EAST PCS, the hardware integration and data interface between PEFIT and EAST PCS were implemented as reported in section 3. Benchmark test was carried out to validate the correctness and robustness of the integrated system using history experimental data. The results are discussed in section 4.

2. Parallel equilibrium reconstruction

The reconstruction procedure is computational intense and most parts of it can be accelerated with parallel computing. GPU is born with highly parallel structure. Instead of having several powerful processing cores like CPU, one modern GPU could have several hundred simple processing cores. Coupled with its high memory bandwidth, GPU has much stronger floating-point compute capability than CPU. Thus, PEFIT is developed based on a GPU device.

PEFIT has the same basic calculation principle as EFIT. The process is to compute the poloidal flux distribution in the R,Z plane, then get the toroidal current density distribution in the R,Z plane, which can provide a least square best fit to the diagnostic data under the model given by G-S equation.

The parallelism of the calculation process mainly includes parallelization of response matrix calculation, parallelization of the least-square best fit and parallelization of the poloidal flux refreshing. The basic principle of this algorithm is using discrete sine transform to decouple the original problem so that independent part could be solved in parallel on GPU [4]. Through carefully assigning the tasks among every multiprocessor, high acceleration rate has been achieved. Tests were conducted on a workstation running ubuntu 10.10 with a four-core Intel Xeon® E3-1200 and a NVIDIA Tesla C2050 GPU. Currently, with grid size 65×65, the total time needed for PEFIT to complete one complete reconstruction iteration is around 220 microseconds, including 80 microseconds for refreshing the flux, 60 microseconds for computing the response matrix, 30 microseconds for least square best fit and 50 microseconds for the other parts [4].

To prove the correctness of algorithm, especially the feasibility of one iteration strategy in PEFIT, some benchmark tests were carried out [4]. The results illuminated that PEFIT was fast and accurate enough to satisfy the real time shape control requirements.

3. Integrate PEFIT into EAST PCS

In order to apply the new parallel equilibrium reconstruction system for real time plasmas control, data interface between PEFIT and EAST PCS was designed. And the exchanging data were transferred through reflective memory board (RFM), which is an ultra-high speed optic fiber network product. The implementation details will be reported in this section.

3.1 Hardware implementation

EAST PCS is a linux cluster configured with 4 nodes. One works as host and the other three are real time computing nodes as shown in figure 2. RFM has been successfully applied in the system for data output. The distortion free signal transfer has been verified in EAST operation campaigns [5]. Thus, we modified the data input way using RFM network as shown in the figure. The acquired diagnostic data and calculating results from sub systems are transferred to PCS through RFM. In such hardware structure, the workstation running PEFIT was configured with a RFM board and connected to the data input RFM network for exchanging data with EAST PCS.

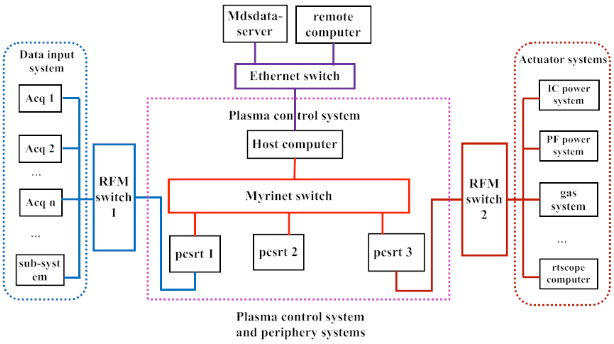

Fig.2. EAST PCS and periphery systems

To the local node, the reflective memory board works like system memory. Data can be written to or read from the memory by any level of software, including the application itself. During the discharge control process, diagnostic data are sent to PEFIT, and control errors generated by PEFIT are sent back to EAST PCS through RFM.

3.2 Data interface and regulations

To avoid overwritten data, the RFM memory was separated into two regions beginning with different offsets. For example, EAST PCS writes setup parameters and diagnostic data starting from OFFSET_1, from where PEFIT can read the data in predefined order and perform parallel calculation in GPU. Then control errors generated by PEFIT are written back to EAST PCS beginning at the other address offset OFFSET_2.

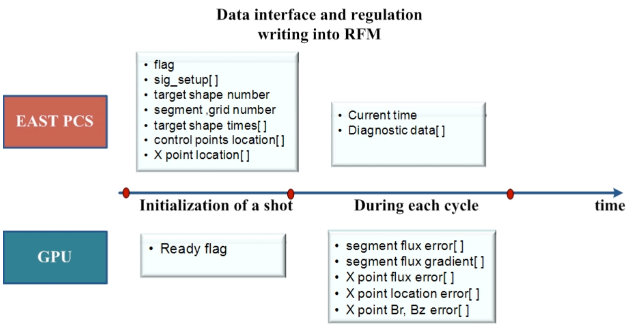

Fig.3. Data interface definition between EAST PCS and PEFIT

As shown in figure 3, the setup parameters including signal setup and target shape information are written once at the beginning of a discharge shot. During each control cycle, only the current time and diagnostic data are written at the offset address after the setup parameters. The control errors written by PEFIT includes flux errors on each segment, flux gradient, flux error of X points, the X point position error, radial and vertical field error. All these data are exchanged through RFM DMA (direct memory access) way without interference to the system CPU. According to the DMA read/write speed [6], about 300 bytes diagnostic data can be written in 14 microseconds and read in 80 microseconds. The data amount of control errors is about 150 bytes with half read/write time. For EAST PCS, the consuming time is acceptable since the control cycle is 50 microseconds. While in PEFIT, some data access strategy must be considered to reduce the executing time. The detail is introduced in section 3.4.

3.3 Algorithm implementation in EAST PCS

An algorithm for integration PEFIT result for real time shape control was implemented in EAST PCS, which provided diagnostic signal setup interface, transferred data with RFM as described in last section, and exchanged data with ISOFLUX control algorithm, such as used segments, control points locations and X point locations of each target shape, and control errors calculated by PEFIT.

In order to increase the control flexibility, a switch is added in the shape control algorithm. Through the switch setting in the operation interface as shown in figure 4, the reconstruction result from rtEFIT and PEFIT can be easily chosen for ISOFLUX control during a shot.

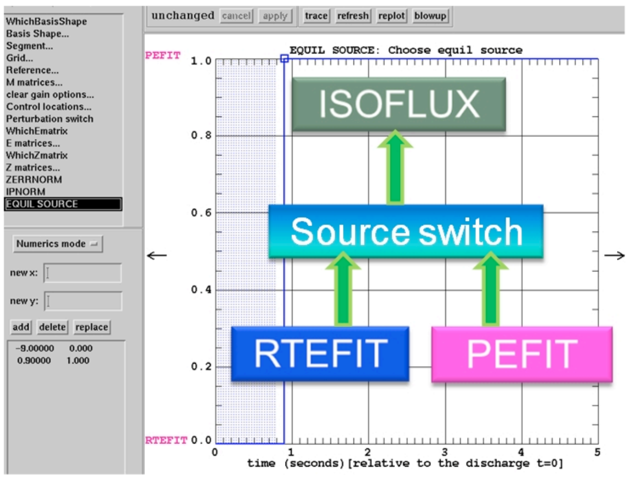

Fig.4. Equilibrium source switch for choosing reconstruction result from rtEFIT or PEFIT for shape control algorithm

3.4 Date interface implementation in PEFIT

The PEFIT algorithm has data interface code running on host CPU and computation intense code running on GPU. The execution flow is shown in figure 5. After reading setup data from RFM, some pre-computing work is done to prepare data for later reconstruction calculation. And the processed data are loaded into GPU memory along with green matrixes. During a discharge, host CPU keeps reading diagnostic data to GPU and sends result to RFM.

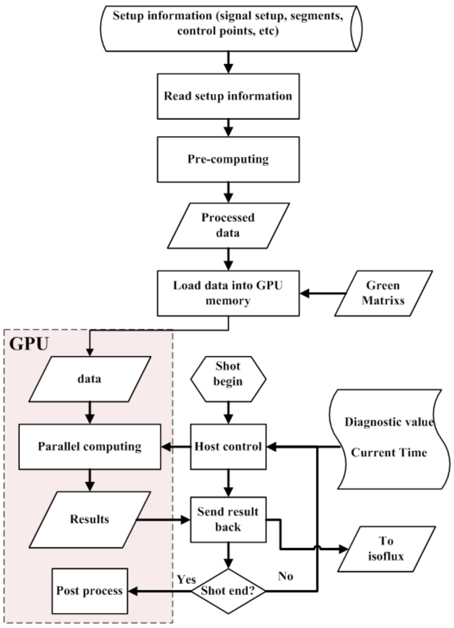

Fig.5. The sketch of PEFIT algorithm execution flow

In order to reduce the effect of data exchanging with RFM to the reconstruction speed, two threads running on two CPUs are used as shown in figure 6. One thread keeps reading data from EAST PCS and sends control errors back when calculation is finished. The data are stored in a pre-defined exchange buffer. The other thread reads data from the exchange buffer and loads them into GPU memory, and writes result back to the exchange buffer. The read/write operation to the exchange buffer is controlled by mutex lock. With such strategy, the RFM DMA data transfer time is converted to the time of exchanging data with CPU memory for PEFIT. Tests showed that only 30 microseconds was increased including data exchanging, control errors calculation and so on.

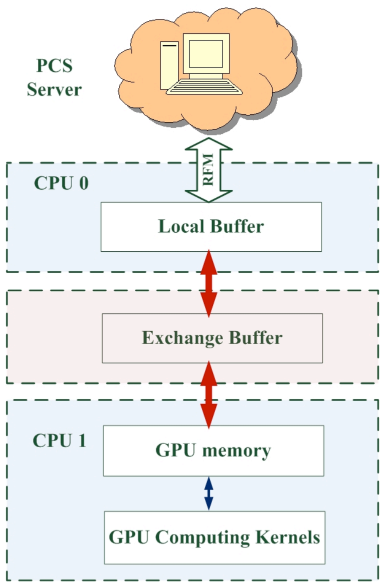

Fig.6. The read/write method used in PEFIT algorithm

4. Benchmark test

In order to verify the correctness of integrating PEFIT into EAST PCS, a real experimental environment simulation test was made. For EAST PCS, there is a running mode called hardware test, in which all the hardware instruments are used and diagnostic data are read from history experimental shot. The benchmark test was based on such hardware test mode. The experimental data were sent to PEFIT each cycle through RFM. And control errors were read back for comparison with rtEFIT result. As shown in figure 7, the boundary difference was obvious at the outer mid-plane. The calculated control errors on the nearby segments such as segment 1, 3, 9 would have corresponding difference.

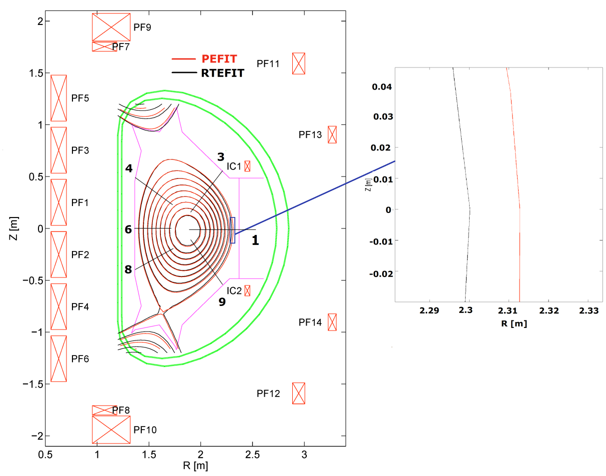

Fig.7. The poloidal flux comparison of PEFIT and rtEFIT simulating EAST experimental shot 43362

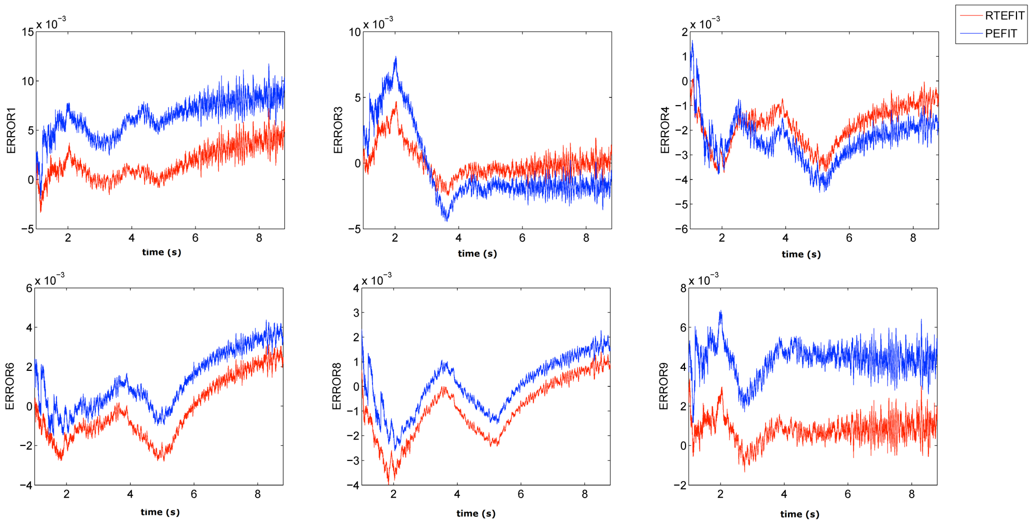

Fig.8. The comparison of flux error on control segments for PEFIT and rtEFIT

As shown in figure 8, there were obvious flux error differences on those segments labeled as ERROR1, ERROR3 and ERROR9. And the value was consistent. Taking segment 1 for instance, the boundary difference was about 1.3 cm and flux gradient was 4×10^{-3} Weber per cm along segment 1, the difference of flux error on segment 1 should be about 5×10^{-3} Weber. The value was verified as shown in the figure. Through such test, the RFM data transfer and the calculation of control errors were proved to be correct.

In EAST shot 40403, we have diagnostic data and rtefit calculation result saved each cycle during 1.0~1.5s. Using these data, another test was carried out to illuminate the reconstruction speed. As shown in figure 9, PEFIT was much faster to have more slices calculated and saved compared to rtEFIT. The number is 1985 slices for PEFIT (0.25 milliseconds per slice) and 94 slices for rtEFIT (5 milliseconds per slice).

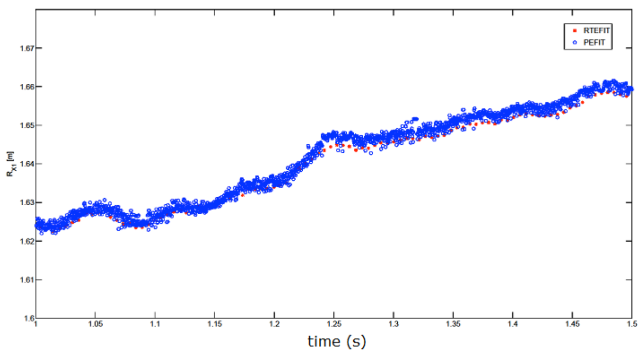

Fig.9. PEFIT has more slices computed and saved than rtEFIT simulating EAST experimental shot 40403

5. Summary

A parallel equilibrium reconstruction code PEFIT running on GPU was integrated into EAST shape control. The implementation of PEFIT into EAST PCS was realized by adding data interface and transferring data through RFM. With higher spatial resolution 65×65 grid size, PEFIT could finish one equilibrium iteration and generate control errors in 250 microseconds. The speed is quite redundant for shape control, which allows more functions being added to the reconstruction program.

Acknowledgement

This work is supported by the National 973 Project with number 2012GB105000, the National Nature Science Foundation of China with contract number 11205200 and 11205191.

References

[1] J.R. Ferron, M.L. Walker and L.L. Lao, Nucl. Fusion, 1998, vol. 38, 1055.

[2] B.J. Xiao, D.A. Humphreys, M.L. Walker, et al., Fusion Engineering and Design, 2008, vol. 83, 181.

[3] Ren, Q., et al., Plasma Phys. Control. Fusion, 2011, 53(9).

[4] X.N. Yue, B.J. Xiao, Z.P. Luo, Y. Guo, Plasma Phys. Control. Fusion, 2013, corrected proof.

[5] Q. P. Yuan, B.J. Xiao, B.G. Penaflor, D.A. Piglowski, Fusion engineering and design, 2010, vol. 85, 474.

[6] Q.P. Yuan, B.J. Xiao, B. Penaflor, et al., Plasma Science and Technology, 2009, Vol. 11, 734.