Development of the simulation platform between EAST plasma control system and the tokamak simulation code based on Simulink*

Sen WANG (王森)1, Qiping YUAN (袁旗平)2 and Bingjia XIAO (肖炳甲)1,2

1 School of Nuclear Science and Technology, University of Science and Technology of China, Hefei 230027, People’s Republic of China
2 Institute of Plasma Physics, Chinese Academy of Sciences, Hefei 230031, People’s Republic of China

E-mail: shenwong@mail.ustc.edu.cn

Received 27 July 2016, revised 14 October 2016
Accepted for publication 19 October 2016
Published 21 February 2017

Abstract
Plasma control system (PCS), mainly developed for real-time feedback control calculation, plays a significant part during normal discharges in a magnetic fusion device, while the tokamak simulation code (TSC) is a nonlinear numerical model that studies the time evolution of an axisymmetric magnetized tokamak plasma. The motivation to combine these two codes for an integrated simulation is specified by the facts that the control system module in TSC is relatively simple compared to PCS, and meanwhile, newly-implemented control algorithms in PCS, before applied to experimental validations, require numerical validations against a tokamak plasma simulator that TSC can act as. In this paper, details of establishment of the integrated simulation framework between the EAST PCS and TSC are generically presented, and the poloidal power supply model and data acquisition model that have been implemented in this framework are described as well. In addition, the correctness of data interactions among the EAST PCS, Simulink and TSC is clearly confirmed during an interface test, and in a simulation test, the RZIP control scheme in the EAST PCS is numerically validated using this simulation platform.

Keywords: PCS, TSC, Simulink, EAST, plasma control

(Some figures may appear in colour only in the online journal)

1. Introduction
The plasma control system simulation platform (PCSSP) [1] for the experimental advanced superconducting tokamak (EAST) was limited to the application of a linear rigid plasma response model, receiving commands from the EAST plasma control system (PCS) and calculating simulated diagnostics fed to it for control evaluation, which forms a closed simulation loop primarily used for numerical validation of new or modified control algorithms in EAST PCS before applied to experimental validations [2, 3]. Such a model is indeed rather efficacious for plasma flat-top phase when there exist tiny perturbations on a derived plasma equilibrium. However, it has neglected a great amount of plasma internal features, for example, plasma current profile changes and energy transport, thus it cannot clearly describe time evolved plasma responses [4].

Compared to the linear rigid model in terms of depicting plasma responses, the tokamak simulation code (TSC) [5] occupies obvious superiorities because this code advances the magnetohydrodynamic (MHD) equations along with the flux surface averaged transport models that effectively reflect various properties of burning tokamak plasmas [6], while the linear rigid plasma response model just neglects them and...
assumes plasmas as rigid conducting filaments moving radially and vertically keeping their current profile unchanged [7]. In other words, TSC can produce more physical quantities that are delivered to PCS for various types of control validation, such as $\beta_N$ control validation and current profile control validation, which, however, cannot be achieved using the much simpler linear model. Therefore, the combination of the EAST PCS and TSC to constitute the so-called TSC-based EAST PCSSP seems to be a more effective alternative for numerical validation of various plasma control algorithms.

In this paper, section 1 has compactly presented the motivation for development of a simulation platform that combines the EAST PCS with TSC for integrated control simulation. In section 2, we are dedicated to giving a brief description of the EAST PCS and TSC, showing the techniques for integration of TSC with the EAST PCS, as well as illustration of a simple power supply model (PSM) and data acquisition model (DAM). Section 3 is arranged to verify the effectiveness of these techniques through an interface test, and to numerically validate the RZIP control scheme in EAST PCS with a simulation test. Finally, we make a summary of the whole paper and propose future missions that would probably be undertaken in section 4.

2. Integrated simulation framework

2.1. Overview of the EAST PCS

The EAST PCS, adapted from the DIII-D PCS, is a linux-based real-time control system, composed of one host for remote access and control management, along with three nodes for real-time calculation, each of which is equipped with a dual-core Intel-Xeon 3.2 GHz processor [8]. As shown in figure 1(a), data interactions among these nodes are through the high speed real-time Myrinet network, while communication between the EAST PCS and other associated systems, for example, PF power system, is realized by reflective memory networks. Currently, there are a host of control schemes implemented in the EAST PCS, many of which have already been tested. This code includes provisions for modelling control system, external heating, particle transport, thermal conductivity, radiation and impurities [11].

TSC divides a tokamak poloidal plane into three regions: plasma region, vacuum region and solid conductors. In each region, a modified form of the MHD equations, mainly involving the force balance equation, Ohm’s law and Fara-day’s law, are numerically solved [11]. Special features are in the plasma region where several advanced plasma evolution models are utilized to simulate different aspects of time-evolved plasma behaviours. Specifically, these evolution models mainly include: (a) one-dimensional transport models for the derivation of number and entropy densities; (b) neo-classical corrections to the resistivity; (c) a time-averaged sawtooth model; (d) the Coppi–Tang model for random heat flux evaluation; (e) a surface-averaged radiated power density model; (f) a bootstrap current model. Details of the TSC evolution models can be referenced by [5, 10, 11].

Figure 1. (a) Layout of the EAST PCS and periphery systems; (b) relationships among the EAST PCS, tokamak and Simserver.

(b)
TSC was originally developed by the Princeton Plasma Physics Laboratory and has been benchmarked for EAST discharge simulations for several years [12]. Figure 2 shows the EAST geometrical implementation in TSC. In particular, the yellow domain is equally divided into 46 × 70 grids for poloidal magnetic flux evolution; the inside part of the black dashed frame is confined for plasma evolution; the inside parts of two magenta dashed frames near the divertors indicate search areas of X points.

2.3. Integration of TSC with the EAST PCS

2.3.1. Framework description. To combine the advantages of TSC in the simulation of tokamak plasma evolution with the EAST PCS in the variety of plasma control algorithms, an integrated simulation framework called the TSC-based EAST PCSSP is designed as shown in figure 3. This framework involves three programmes, i.e. the EAST PCS, Simulink and TSC (without its control system module), which play different roles during their combined simulation. Specifically, the EAST PCS acts as a plasma controller in place of the control system module, which play different roles during their combined simulation. Specifically, the EAST PCS acts as a plasma controller in place of the control system module in TSC, while a combination of TSC evolution models are regarded as a virtual EAST tokamak that substitutes for the actual EAST tokamak, directly controlled by the EAST PCS (hereafter in section 2.3 called PCS for simplicity). Originally, TSC neglects various engineering limitations, for example, power saturation and experimental noise, during simulations, whereas this framework is capable of accounting for those effects easily by designing a PSM and DAM in the Simulink infrastructure.

2.3.2. Data interaction. In figure 3, the inside area of the purple dashed frame indicates Simulink infrastructure, which constitutes a closed simulation loop that involves periodic data interactions with TSC and PCS. The basic technique for communication between Simulink and either of the other processes is exactly the Socket/TCP network communication technique that is sufficiently fast and reliable for real-time simulation.

2.3.2.1. Work flow of data interaction. Now that the technique of data interaction between Simulink and either of TSC and PCS is basically similar, as an example we are subsequently dedicated to detailed illustration of that between Simulink and PCS. Figure 4(a) displays work flow of the Simulink Interface to PCS block (SIP) that can clearly describe the process of data exchanges between Simulink and PCS during their combined simulation. Before connection, SIP establishes a socket port waiting for connection from PCS. As soon as SIP has received and confirmed the data request from PCS, the connection between them is established. Subsequently, SIP first receives control commands that are directly passed to the input of PSM from PCS via the Socket/TCP network protocol, and then transmits simulated diagnostics from the output of DAM to PCS using the same protocol, which constitutes a complete data interaction between Simulink and PCS. This scheme continues to work until the pre-set simulation stop time comes when SIP closes its socket port and terminates communication with PCS. Figure 4(b) shows work flow of the Simulink Interface to TSC block (SIT) that intends to couple TSC with Simulink. As the similar
focussed on the techniques of time synchronization among PCS, Simulink and TSC.

A. Techniques for the timestamp synchronization among each process

As one knows, during an integrated simulation, PCS, Simulink and TSC have their own timestamps, i.e. simulated times at the start point of each time step, which need to be synchronized. Basically, five techniques are adopted to ensure the timestamp synchronization among them: (a) timestamps transmitted along with other simulated data during data interactions; (b) a module/process exclusively interacting with the other process/module; (c) a time step representing the same amount of simulated time in each process, e.g. 100 μs; (d) the Socket/TCP data communication technique; (e) careful setup of the simulation start and stop times in each process.

In addition to transmission of diagnostics, commands or actuator data, each data interaction between two processes contains transmission of the associated timestamps, which are subsequently compared with the local timestamps; only the difference between the received and local timestamps less than a particularly small value, e.g. 25 μs, would the local process/module move on to the next step, or it will keep blocked until correct timestamp’s arrival. Indeed, this condition can naturally be satisfied and keep being satisfied if (b)–(e) work properly. Based on (b)–(c), if two separate processes both increase by a time step, the timestamps of them are supposed to increase by the same amount; moreover, supposing that (I) their timestamps are synchronized once and (II) their data exchanges are performed every time when they both complete a time step simulation, theoretically their timestamps would keep being synchronized afterwards. Based on (d), if two processes advance in different speeds, the faster one can be blocked after completing a time step simulation, waiting for the requisite data interaction with the slower one at the end of each time step, which properly satisfies the necessary condition (II). As for (e), we tend to set the simulation start time in Simulink and PCS one or two sample times greater than the TSC counterpart, aimed to avoid missing the only one TSC timestamp that corresponds to them, which properly satisfies the necessary condition (I); the simulation stop time in PCS, Simulink and TSC are set equal to guarantee they finish their simulations simultaneously. A combination of techniques (a)–(e) can properly ensure the timestamp synchronization between PCS/TSC and SIP/SIT, meanwhile, all Simulink modules (including SIP and SIT) self-consistently synchronize their timestamps with each other; therefore, we theoretically conclude that the timestamp synchronization among PCS, Simulink and TSC can be sufficiently guaranteed based on the above techniques plus Simulink infrastructure.

B. Wall-clock timing in each process for the timestamp synchronization

Wall-clock timing in PCS, Simulink and TSC at two consecutive time steps are illustrated in figure 5. From actual time $t[k]$ to $t[k + 1]$, Simulink first carries out its data interaction with PCS via the SIP module that takes $\sim 3431 \mu s$ to receive commands and transfer diagnostics, and then takes
a small amount of time to complete execution of the PSM module, before executing SIT that is blocked for \(\sim 83250 \mu s\), finally around 97 \(\mu s\) is spent in completing the DAM simulation. In terms of PCS, after receiving diagnostics from SIP, it immediately carries out control evaluation which takes \(\sim 100 \mu s\) before entering the mode where it is blocked for a great amount of time (\(\sim 83768 \mu s\)) intended to synchronize its timestamp with SIP. On the TSC side, a time step plasma simulation should be completed to increase its timestamp to \(T_k\) that corresponds to the current SIT timestamp for effective data exchanges; after an 80 \(\mu s\) data interaction with SIT, it continues a new step simulation that would increase its simulated time slowly from \(T_k\) to \(T_k\) + 1. Simply speaking, from time \(t[k]\) to \(t[k+1]\). Simulink and PCS spend a small amount of actual time completing a time step simulation plus data interaction, and waste most of the time waiting for TSC to synchronize their timestamps. We note that the whole actual time for parallel simulation at the time step \(k\) is \(\sim 87299 \mu s\), which is different from that at time \(k+1\) (\(\sim 83430 \mu s\)). It is indeed normal for a dynamic simulation, whose actual time taken at each time step is dynamical and changeable. However, the techniques previously illustrated can make the parallel simulation framework automatically adapt to the characteristics.

C. Data transmission and transformations among each process for closed-loop control simulation

Data transmission and transformations among PCS, Simulink and TSC from actual time \(t[k-2]\) to \(t[k+2]\) are shown in figure 6. At the timestamp \(T_{k-2}\), TSC sends SIT simulated data (\(d[k-2]\), green dashed arrow) that is then handled by DAM to generate diagnostic data for use at \(T_{k-1}\), i.e. \(d[k-1]\), which is made available to PCS at the beginning of the time step \(k\), indicating a time step delay, equivalent to 100 \(\mu s\) plasma time, occurs during data transmission from TSC to PCS. Using the diagnostic data \(d[k-1]\), PCS computes command data for use at \(T_k\), i.e. \(c[k]\), that is sent to SIP and then handled by PSM to generate
actuator data \( q[k] \) at \( T_k \), which is available to TSC at the same timestamp and used for plasma simulation at the time step \( k \), thus indicating a time step delay occurs during data transmission from PCS to TSC. We note that the route of data transmission and transformations illustrated above (see all green arrows) constitutes a time step of closed-loop control simulation, including data acquisition from \( T_{k-1} \) to \( T_k \) and control evaluation from \( T_k \) to \( T_{k+1} \). Similarly, this scheme carries on until the simulation stop time arrives.

2.3.3. Power supply model. Power supply model is devoted to transforming PCS control voltages into active coil currents, which is divided into two sub-models: voltage response sub-model and dynamic response sub-model. Concretely, voltage response sub-model is dedicated to mapping PCS control voltages into simulated power voltages, and dynamic response sub-model is intended for the derivation of active coil currents from power supply voltages by resolving Kirchhoff’s circuit equations. At present, we have implemented the EAST poloidal PSM in Simulink.

2.3.3.1. Voltage response sub-model. On EAST, each poloidal power supplies are totally separated and thus modelled separately, every voltage response sub-model is supposed to comprise of three blocks, namely transport delay, transfer function and saturation, as shown in figure 7(a). Specifically, the transport delay block describes the time delay of control command transmission from PCS to actual poloidal power system; the transfer function block in terms of a 1-pole-1-zero Laplace form converts poloidal power commands that are received from PCS into poloidal power voltages; the saturation block reflects voltage limits of this power system. We note that the transport delay and saturation are normally derived from experimental tests and basically fixed during normal discharges, while the transfer function is numerically fitted against experimental data using the system identification method. Figure 7(b) shows comparison of...
simulated (blue) and experimental (red) poloidal power voltages, which indicates good agreement with each other.

2.3.3.2 Dynamic response sub-model. A dynamic response sub-model is placed directly after the voltage response sub-model in Simulink so as to derive simulated PF coil currents based on the circuit equation \[ \frac{dI}{dt} + RI = V \], where \( M \) and \( R \) represent the mutual inductance matrix and resistance matrix, respectively; \( I \) represents the vector of PF coil currents plus plasma current, which is a known quantity retrieved from TSC; \( V \) represents the vector of poloidal power supply voltages that is the input of this sub-model. By numerically solving the above equation in Simulink, simulated PF coil currents are attained.

2.3.4 Data acquisition model. As one knows, TSC simulated data does not contain any noise. To numerically examine the robustness of PCS control algorithms against experimental noise, a DAM designed to involve that in TSC signals has been implemented in Simulink, which is, in principle, based on experimental measurements. A simple technique, i.e. the fast Fourier transform (FFT), is utilized to analyse experimental signals and design proper white Gaussian noise \( N_{\text{des}} \) for their corresponding TSC signals. Since the technique for different signals is similar, we take the PF1 current for example. Firstly, the experimentally measured PF1 current in shot \#59888 is transformed into frequency domain using FFT; analysis of the magnitude-frequency characteristics indicates components with the frequency larger than 300 Hz normally appear as experimental noise. Secondly, to derive the useful information, coefficients of the Fourier components with the frequency larger than 300 Hz are artificially set to 0; and then the inverse FFT is performed leading to, probably, real data, which is compared with the PF1 signal in figure 8(a). Likewise, the experimental noise can be derived as well, as shown in figure 8(b), which is further analysed to determine its mean (0) and standard deviation (2.3749) that are utilized to design the white Gaussian noise. Figure 8(c) shows this designed noise, indicating relatively good agreement with the experimental noise.

3. Benchmark test

Two benchmark tests have been performed to prove the effectiveness of this integrated simulation framework. One is an interface test to confirm the correctness of data interactions among the EAST PCS, Simulink and TSC. The other is a simulation test which allows the EAST PCS to control TSC simulated plasmas and to physically verify this simulation platform.

3.1 Interface test

In the interface test, PSM and DAM are not involved for convenience of the comparison of exchanged signals between TSC and the EAST PCS. It would not affect test results, since the objective of this test is to verify the effectiveness of data interaction on each interfaces and time synchronization among each process. Figure 9 shows comparisons of the exchanged signals between PCS and TSC. Specifically, figures 9(a) and (b) respectively show global comparisons of commands sent (received) by PCS and diagnostics received (sent) by TSC, indicating data channels are correspondingly defined between them, and roughly demonstrate the effectiveness of data interaction techniques illustrated previously; figure 9(c) shows the timestamp differences between TSC and PCS for the same PS1 points, which implies the PS1 signal is transmitted from PCS to TSC with the average simulated time delay of 111.1 μs. Figure 9(d) shows the timestamp

![Figure 8](image-url). Analysis and design of noise for the PF1 current: (a) comparison of the measured PF1 signal and that with the frequency components less than 300 Hz; (b) experimental noise; (c) designed noise.
differences between PCS and TSC for the same RL1 points, which indicates the RL1 signal is delivered from TSC to PCS with the average simulated time delay of 88.9 $\mu$s. Obviously, the sum of both delays is exactly 200 $\mu$s, conforming to the theoretical derivation. Note that the measured simulated time delays are not fully equal to 100 $\mu$s, since TSC dynamically evolves its simulated time, which cannot be fixed to a specific value at the end of a time step simulation, but can be limited to a reasonably small range. Nevertheless, it would scarcely influence simulation results since during actual discharges the average actual time delay of power supply systems on EAST are normally greater than 500 $\mu$s. Therefore, we conclude that the integrated simulation framework among the EAST PCS, Simulink and TSC has been established with correct data interactions while they are parallel simulating.

3.2. Simulation test

Using this simulation platform, the RZIP control schemes in EAST PCS are engaged in controlling TSC simulated plasmas as a simulation test. As shown in table 1, the numerical experiment comprises of two tasks, the first of which only utilizes feedforward controller while the second uses both controllers, i.e. feedforward and feedback. Figures 10(a) and (b) display comparisons of PCS targets (blue dashed) and TSC evolutions (black), which clearly demonstrates the effectiveness of the RZIP feedback control scheme in the control of TSC plasma current centroid, i.e. $R_c$ and $Z_c$, and the magnitude of TSC plasma current ($I_p$).

4. Summary and perspectives

In this paper, an integrated simulation framework that couples TSC with the EAST PCS, simply called the TSC-based EAST PCSSP, has been developed using Simulink infrastructure and the Socket/TCP communication technique. This integration method can be generally applied to synthesize the advantages of two or more codes for parallel simulation. Meanwhile, a simple linear poloidal PSM that converts PCS control

<table>
<thead>
<tr>
<th>Table 1. Parameter settings for the numerical experiment.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Open loop</strong></td>
</tr>
<tr>
<td>Base shot</td>
</tr>
<tr>
<td>Initial equilibrium</td>
</tr>
<tr>
<td>Control algorithm</td>
</tr>
<tr>
<td>Time interval</td>
</tr>
</tbody>
</table>

Figure 9. (a) Comparisons of commands sent by PCS (black) and received by TSC (red); (b) comparisons of diagnostics sent by TSC (red) and received by PCS (black); (c) illustration of the simulated time delays from PCS to TSC (average: 111.1 $\mu$s, standard deviation: 0.0064) for the same PS1 points; (d) illustration of the simulated time delays from TSC to PCS (average: 88.9 $\mu$s, standard deviation: 0.0064) for the same RL1 points.

Figure 10. Comparisons of PCS targets (blue dashed) and TSC evolutions (black): (a) feedforward only and (b) feedforward + RZIP feedback; from top to bottom (both left and right panels): the radial and vertical position of plasma current centroid and the magnitude of plasma current.
commands into active coil currents has been established to emulate poloidal power supply actuators, as well as a DAM simulating diagnostic sensors. In addition, an interface test has been carried out with the result of correct data interactions and effective time synchronizations among PCS, Simulink and TSC; a simulation test has been carried out that numerically validates the RZIP feedback control scheme in EAST PCS using the TSC-based EAST PCSSP.

Although development of this simulation platform was originally motivated for numerical validation of the EAST PCS control algorithms and enhancement of the TSC control system module, the platform seems to be capable of being applied to resolving complicated real-time plasma control problems, provided TSC is exactly a nonlinear MHD code with several advanced transport models available. In the future, based on this work, we are supposed to carry out two types of specific and in-depth work: (a) improve the TSC-based EAST PCSSP to address specific real-time magnetic plasma control problems; (b) integrate effective nonlinear tokamak plasma transport models, such as the TRANSP code, into this simulation platform aimed to address more complicated real-time kinetic plasma control problems.

Figure 11. Distributions of poloidal magnetic fluxes and plasma boundaries (red cross circles) evolved by TSC in different time slices under the (a)–(c) open-loop and (d)–(i) closed-loop RZIP control schemes. Note that green cross circles indicate plasma boundaries reconstructed by the offline EFIT code.
Acknowledgments

The authors gratefully acknowledge valuable discussions with Drs M J LANCTOT, GUO Yong, LUO Zhengping and QIU Qinglai, and thank the referees for providing useful suggestions for improving the original manuscript.

References

[7] Walker M L, Humphreys D A and Ferron J R 1997 Control of plasma poloidal shape and position in the DIII-D tokamak Proc. 36th Conf. on Decision and Control (San Diego, USA)