AUTHOR(S):

Honda, M.; Aiba, N.; Seto, H.; Narita, E.; Hayashi, N.

Development of a novel integrated model GOTRESS+ for predictions and assessment of JT-60SA operation scenarios including the pedestal

M. Honda[1], N. Aiba[2], H. Seto[3], E. Narita[2] and N. Hayashi[2]

[1] Graduate School of Engineering, Kyoto University, Nishikyo, Kyoto 615-8530, Japan

[2] National Institutes for Quantum and Radiological Science and Technology, Naka, Ibaraki 311-0193, Japan

[3] National Institutes for Quantum and Radiological Science and Technology, Rokkasho, Aomori 039-3212, Japan

E-mail: honda.mitsuru.5c@kyoto-u.ac.jp

Abstract. A novel integrated model GOTRESS+ has been developed, which consists of the iterative transport solver GOTRESS as a kernel of the integrated model, the equilibrium and current profile alignment code ACCOME and the neutral beam heating/current-drive code OFMC. GOTRESS is able to robustly find out an exact solution of the stationary-state transport equations even with a stiff turbulent transport model, taking advantage of global optimization techniques such as a genetic algorithm. GOTRESS+ is then suitable for self-consistently assessing the stationary-state plasma performance of JT-60SA as well as ITER and DEMO or validating their feasibility. Recently GOTRESS+ has been extended to incorporate the in-house EPED1 model exploiting the MHD stability code MARG2D and is now able to predict the plasma profiles even with the pedestal over the entire region from the magnetic axis to the plasma boundary in a self-consistent manner. The two JT-60SA operation scenarios including the ITER-like inductive scenario and the high β fully non-inductively current driven scenario have been assessed by GOTRESS+ with the CDBM turbulent transport model and then were found to be feasible with most of the target dimensionless parameters met.

1. Introduction

As the time to achieve a first plasma of JT-60SA is approaching and the construction of ITER is making great progress, the development of operation scenarios that bring out the plasma performance of these devices is becoming more and more important. Operation scenarios are often stipulated by various target parameters, or sometimes referred to as reference parameters, in a stationary state, such as the normalized beta $\beta_{\mathrm{N}} \equiv \beta/(I_{\mathrm{p}}/(aB_{\mathrm{T}}))$, the H_{H} factor referring to enhancement above the nominal H-mode scaling law and the bootstrap current fraction f_{BS}, and self-consistent predictive simulations using an integrated transport model are essential to investigate whether

these parameters simultaneously satisfy the specified target values. In the definition of β_N, I_p is the plasma current in MA, B_T is the toroidal magnetic field in T and a is the plasma minor radius in m as well as $\beta = \langle p \rangle / (B_\mathrm{T}^2 / (2\mu_0))$ where $\langle p \rangle$ is the mean plasma pressure and μ_0 is the permeability in vacuum. One of the most important factors in predicting high-performance plasmas is the accurate prediction of the pedestal height and width, because an H-mode plasma pedestal accounts for a large fraction of the plasma stored energy. The EPED1 model [1] is considered to be the most successful semi-empirical model in estimating pedestal height and width. The original EPED1 model and its derivatives [2–5] have been successful in accurately predicting pedestals in many tokamaks [1–14]. Several integrated transport simulation codes that incorporate the EPED1 model have been developed, such as OMFIT [15], CORSICA [16], ASTRA [17, 18], JINTRAC [19], IPS-FASTRAN [20] and TRANSP [21], and used to develop operating scenarios.

We have been developing a novel integrated suite of codes, GOTRESS+, with GOTRESS at its core. GOTRESS [22, 23], which is an acronym for Global Optimization version of Transport Equation Stable Solver, is a transport code that finds solutions of the stationary-state transport equations using global optimization techniques such as a genetic algorithm. A stationary-state transport equation is in general a one-dimensional nonlinear elliptic partial differential equation, and it is recognized as being tough to find a solution in a numerically stable fashion. So far, few stationary-state transport codes have been developed in the community. One of the few is TGYRO [24], which is used as a transport module in OMFIT [15]. GOTRESS and TGYRO have the same purpose of finding the solution of the stationary-state transport equations. Both codes are the same in that they do not try to directly discretize the governing equations that will be expressed in (1), but solve them in their volume-integrated form in a way to find the transport flux that matches the target flux, while the numerical methods for that purpose are quite different. In TGYRO, focusing on the dependence of the heat flux on the gradients of the kinetic profiles, both transport and target fluxes are Taylor-expanded with the gradients and the derivatives of the fluxes with respect to the gradients explicitly appear. Then, the Newton-iteration technique is applied to minimize the error between the fluxes by gradually changing the gradients [24]. On the other hand, thanks to the global optimization technique, in GOTRESS it is possible to find a set of temperature and its gradient satisfying the stationary-state transport equation at each grid point. Even though the methodology finding a solution and the algorithms numerically implemented in GOTRESS are detailed in Refs. [22, 23], they will also briefly be explained here to see how GOTRESS works. We here note that the problem we are dealing with is greatly simplified to illustrate the algorithm. A heat flux q_a is the product of the density n_a, the heat diffusivity χ_a and the temperature gradient T_a', where χ_a is usually a function of T_a, T_a' and other quantities and its dependence on these quantities varies depending upon a transport model adopted. In general, it is roughly written as $q_a/n_a \sim -\chi_a(T_a, T_a')T_a'$. Taking account of the fact that the value of q_a/n_a should be known in advance because it should be identical to the known target flux,

all GOTRESS has to do is to find a pair of T_a and T_a' satisfying $q_a/n_a \sim -\chi_a(T_a, T_a')T_a'$. In this process, any relationship between T_a and T_a' is never assumed. Each of a pair is individually and independently determined by the global optimization technique like the genetic algorithm. Finding an appropriate pair is not an easy task due to the nonlinear dependence of χ_a on T_a and T_a'. The genetic algorithm is powerful in robustly finding global optimum solutions in a broad (T_a, T_a') parameter space. Of course there are perhaps countless pairs of candidate solutions found by dealing with this relationship alone. Another equation, which is obtained by cumulatively integrating the heat flux relationship over the volume, must be required to be solved simultaneously. This equation is mainly responsible for determining T_a and the Dirichlet boundary condition for T_a can be imposed on it. The global optimizer finds out a pair simultaneously satisfying both equations, which results in the solution of the transport equation. As one can see from this explanation, the temperature gradient is not computed from the temperature profile using some numerical differentiation method. Turbulent transport fluxes calculated by advanced turbulent transport models having stiff transport nature are sensitive to the gradient of plasma quantities, and a large difference in the fluxes could be caused by a small difference in the gradient especially around a critical gradient. It is not preferable that such important gradients change depending upon the numerical differentiation method the one chooses. One of the main features of GOTRESS is that it can avoid this problem. Even though the disadvantage of global optimization techniques is the slow computational speed, MPI parallelization has alleviated the problem to some extent; GOTRESS runs on the MPMD framework along with MPI-parallelized transport models like TGLF [25, 26] and successfully reproduced the temperature profiles of the JT-60U plasma with TGLF [23]. Since a genetic algorithm makes many trials to find the best answer due to its heuristic nature and produces a large amount of data in the process, GOTRESS is a suitable code for building a neural-network-based surrogate model of the turbulent transport model with deep learning techniques [23].

The explanation of a novel integrated model GOTRESS+ [27] and the workflow thereof will be detailed in the next section, but here its components are briefly introduced. GOTRESS+ mainly consists of GOTRESS, ACCOME [28] and OFMC [29]. ACCOME is a code to calculate a magnetic equilibrium consistent with the current density profile driven by several auxiliary heating systems. It includes the bounce-averaged Fokker-Planck solver and the electron cyclotron heating (ECH) code. Note that the bounce-averaged Fokker-Planck solver does yield a driven current profile only and does not compute the heat deposition profile. This is because the plasma kinetic profiles are given and fixed throughout the calculation in ACCOME and hence the heat deposition profile is not required. The heating profiles, which GOTRESS necessitates, are calculated by OFMC, which is literally an orbit following Monte Carlo code to calculate the profiles of not only heating but also fueling and torque by neutral beam injection (NBI).

We have also been developing another integrated suite of codes, TOPICS [30,31], to analyze JT-60U plasmas and to predict plasmas of JT-60SA, ITER and other tokamaks.

TOPICS solves a set of time-dependent transport equations in its transport module and is thus able to predict temporal evolution of profiles of the plasma quantities like the density, the temperature and the safety factor. It is thus apparent that GOTRESS+ is completely different from TOPICS. In order to evaluate the stationary state of a plasma with time-dependent codes, the calculation has to be continued until the plasma does not nearly vary in time. Therefore, the simulation time could be too long for the purpose of evaluating the plasma performance in a stationary-state current flat-top phase.

We here note that GOTRESS+ is an integrated model for finding the consistent kinetic profiles in a flap-top phase of a scenario with target plasma parameters. Developing operation scenario in the context of this paper means finding such a solution in a stationary state, not modeling an entire scenario of a tokamak discharge such as from current ramp up to ramp down. GOTRESS+ has already been applied to the development of operation scenarios for JT-60SA [27] with the CDBM turbulent transport model used [32]. The CDBM model is widely used for plasma predictions especially with internal transport barriers (ITBs) in JT-60SA as the validity of using the model has been assessed: For ITB plasmas in JT-60U and JET, CDBM demonstrated good reproducibility of temperature profiles among several transport models with TOPICS and CRONOS used [33]. Here, the GOTRESS+ simulation of the scenario #5-1 [34] was focused on: $B_\mathrm{T} = 1.72\,\mathrm{T}$, $I_\mathrm{p} = 2.3\,\mathrm{MA}$, the NBI heating power $P_\mathrm{NBI} = 18.85\,\mathrm{MW}$ and the ECH power $P_\mathrm{ECH} = 7\,\mathrm{MW}$. This scenario has also been assessed by integrated simulations in the previous work [35, 36]. The plasma predicted by GOTRESS+ met the target values: $\beta_\mathrm{N} = 4.42(> 4.3)$, $H_\mathrm{H} = 1.65(> 1.3)$ and the loop voltage $V_\mathrm{loop} = 0.003(\sim 0)$ [27]. Here, we note that the density profiles were prescribed in the simulation and will be in the same way in this paper. The results are in excellent agreement with those by TOPICS, a fact which bolsters the reliability of GOTRESS+. Also, the time to complete the simulation was 6,654 seconds on our PC cluster, which is about six times faster than 42,909 seconds when using TOPICS [27]. We note that OFMC took only about 12 minutes per run due to the assumption of the perfectly axisymmetric magnetic field for the sake of simplicity. The problem with this simulation was that the boundary condition for GOTRESS was set at $\rho = 0.8$ and the MHD stability of the final result was not investigated. Outside $\rho = 0.8$ the temperature profiles were prescribed and fixed. This assumption had to be made due to the fact that neither GOTRESS+ nor TOPICS had the model predicting the pedestal width and height in a self-consistent manner. Considering that the stored energy earned by the pedestal accounts for a significant fraction and that the plasma cannot be established unless it is MHD stable, it must be said that this simulation was not sufficient. In order to resolve these problems, it is essential to incorporate the EPED1 model into GOTRESS+ and to work with the MHD stability code.

The rest of this paper is organized as follows. Section 2 describes the further information on each component of GOTRESS+ and the original workflow of GOTRESS+ in more detail. In section 3, the development of our in-house EPED1 is detailed and then its integration into the new GOTRESS+ workflow is presented.

After that, section 4 is devoted to the development of JT-60SA operation scenarios by the extended GOTRESS+ including EPED1. Finally, conclusions of this paper and discussion are presented in section 5.

2. GOTRESS+ and its components

GOTRESS in essence solves a set of the stationary-state heat transport equations [23],

$$0 = -\frac{\partial}{\partial V}\left(\frac{5}{2}V'T_a\Gamma_a - n_a\langle|\nabla V|^2\rangle\chi_a\frac{\partial T_a}{\partial V}\right) + S_a \tag{1}$$

for electrons and ions, given an equilibrium and heat sources and sinks. Here, V is the volume, Γ is the particle flux, and S denotes the sum of heat sources and sinks. The subscripts a denote species and the prime denotes the ρ-derivative, where ρ is the normalized radial coordinate defined by the square root of the normalized toroidal flux. The heat diffusivity χ_a comprises turbulent and neoclassical contributions: $\chi_a = \chi_a^{\text{turb}} + \chi_a^{\text{neo}}$. The turbulent component χ_a^{turb} is evaluated by the turbulent transport model currently in use, like CDBM or TGLF, and the neoclassical one χ_a^{neo} is calculated by the Matrix Inversion method [37]. While the usual numerical implementation of the code solving the similar equations is to directly discretize this equation, GOTRESS solves the simultaneous equations consisting of the volume-integrated (1) and the one obtained by further integrating the former one over the volume. In other words, a set of T_a and $1/L_{T_a} \equiv -(1/T_a)(\partial T_a/\partial \rho)$ satisfying both equations simultaneously is found out using the global optimization techniques at each radial location [23].

The source and sink term S_a in (1) is detailed here. The source and sink terms for electrons and ions are

$$S_{\text{e}} = S_{\text{aux,e}}^{(\text{f})} + S_{\text{OH,e}}^{(\text{f})} + S_{\text{fus,e}}^{(\text{u})} - S_{\text{ei}}^{(\text{u})} - S_{\text{rad,e}}^{(\text{f,u})}, \tag{2}$$

$$S_{\text{i}} = S_{\text{aux,i}}^{(\text{f})} + S_{\text{fus,i}}^{(\text{u})} + S_{\text{ei}}^{(\text{u})}, \tag{3}$$

respectively, and each term is explained in the following. First of all, the superscript (f) or (u) means that a term is fixed or updated in each iteration in GOTRESS and the term with (f,u) is the one that allows a user to choose (f) or (u). Please note that the term with (f) is in usual updated in the GOTRESS+ procedure.

S_{aux} denotes the auxiliary heating sources, such as NBI and ECH. GOTRESS does not calculate any physics-based auxiliary heating profiles internally except for artificially given functional heating profiles used in a test. They are computed by other codes taking part in GOTRESS+: The heating profile by NBI is computed in OFMC [29], and that by ECH is computed by the EC-Hamamatsu code [38] that is actually invoked from ACCOME. These profiles are read at the beginning of the GOTRESS simulation and fixed throughout the simulation. Joule heating $S_{\text{OH,e}}$, which occurs only for electrons, takes part in S_{e}. It is calculated in ACCOME and is incorporated into GOTRESS in the same manner as S_{aux}. Another heating mechanism is the alpha heating due to fusion reactions, S_{fus}, and is internally computed in GOTRESS in each iteration. The

major heat loss channel is radiation S_{rad}, which is unique for electrons. A user can choose whether its profile is read from a result calculated by other codes, if any, or calculated inside GOTRESS. GOTRESS can currently calculate three mechanisms of the electron radiation loss: Bremsstrahlung [39], the impurity (line) radiation under the coronal equilibrium assumption [40] and the cyclotron radiation by CYTRAN [41]. The radiation loss profiles are calculated and updated in each iteration in GOTRESS if they are chosen to be calculated. The collisional equipartition process also acts as a kind of heat sources and sinks and is calculated in GOTRESS. Here, S_{ei} is defined as the heat transfer from electrons to ions and is actually given by

$$S_{\mathrm{ei}} = \frac{3m_{\mathrm{e}}}{m_{\mathrm{i}}} \frac{n_{\mathrm{e}}}{\tau_{\mathrm{ei}}} (T_{\mathrm{e}} - T_{\mathrm{i}}), \tag{4}$$

where m_a is the mass of species a and τ_{ei} denotes the energy equipartition time between electrons and ions. This process transfers the heat from one species to another, trying to equilibrate the temperatures. It is updated in each iteration in GOTRESS. The actual implementation is detailed in the GOTRESS flowchart shown in Fig. 1 of Ref. [23]. The particle flux Γ_a in (1) is calculated in GOTRESS by integrating the NB particle source S_{NB}^n given from OFMC.

Given the equilibrium and the current density profile, i.e., the safety factor q profile, metrics such as $\langle |\nabla V|^2 \rangle$ and V' are computed in GOTRESS by tracing the magnetic field line at each magnetic surface. Then, GOTRESS finds out the solution as a set of $(T_a, 1/L_{T_a})$ for each species by the method described above. In this sense, the magnetic shear s, which has a large impact on the transport coefficients, is kept fixed during the calculation. It means that GOTRESS is not responsible for calculating the current density profile and the associated quantities like resistivity, while it does calculate the neoclassical transport coefficients using the Matrix Inversion method [37] internally. The equilibrium and the current density profile are computed in ACCOME [28]. The detailed procedure is detailed in Ref. [28], but here the equations solved in ACCOME are briefly summarized. ACCOME has its own free-boundary equilibrium solver. The poloidal flux function ψ is the sum of the poloidal flux function generated by the plasma current, ψ_p, and the vacuum one generated by the external coil current, ψ_v as

$$\psi = \psi_p + \sum_k I_v^k \psi_v^k, \tag{5}$$

where I_v^k and ψ_v^k are the current of the k-th external coil and the poloidal flux function produced by a unit current of the k-th coil, respectively. The Grad-Shafranov equation

$$\Delta^* \psi = -\mu_0 R^2 \frac{\mathrm{d}p}{\mathrm{d}\psi} - F(\psi) \frac{\mathrm{d}F}{\mathrm{d}\psi}, \tag{6}$$

is solved to determine ψ on each grid point of the two-dimensional rectangular domain including the entire plasma and the vacuum region. Here, Δ^* is the Grad-Shafranov operator, the poloidal current function $F(\psi) = RB_t$ and the other variables follow

the convention. The second term on the right-hand side of (6) is determined by the relationship with the parallel current density as

$$-F\frac{\mathrm{d}F}{\mathrm{d}\psi} = \frac{\mu_0}{\langle B^2 \rangle}F\left(F\frac{\mathrm{d}p}{\mathrm{d}\psi} + \langle j_\parallel B \rangle\right), \tag{7}$$

which makes sense inside the plasma, where the plasma current flows. The parallel current density can be decomposed into

$$\langle j_\parallel B \rangle = \langle j_\mathrm{NBCD} B \rangle + \langle j_\mathrm{ECCD} B \rangle + \langle j_\mathrm{BS} B \rangle + \langle j_\mathrm{OH} B \rangle. \tag{8}$$

and all components are updated at each iteration in ACCOME. The bootstrap current $\langle j_\mathrm{BS} B \rangle$ and neoclassical resistivity are evaluated by the Matrix Inversion method. The fast-ion current $\langle j_\mathrm{NBCD} B \rangle$ is calculated by solving the bounce-averaged Fokker-Planck equation and multiplied by the shielding factor [42] to give the NB driven current. We note that the shielding factor model same as the one in ACCOME is adopted in OFMC when the NB driven current computed by OFMC is used in GOTRESS+ instead of that by ACCOME. The electron cyclotron current drive $\langle j_\mathrm{ECCD} B \rangle$ is computed by EC-Hamamatsu [38] invoked in ACCOME. Since ACCOME regards the given total plasma current as a constraint of a calculation, it is necessary to adjust some components of the current density to match it in the end. There are a few options available for this purpose: adjusting $\langle j_\mathrm{NBCD} B \rangle$ or $\langle j_\mathrm{OH} B \rangle$, or adding the so-called additional current. ACCOME is often applied to the case with the finite inductive current, i.e., the finite loop voltage V_loop, albeit in the stationary state of kinetic profiles, but it is in general difficult to know the adequate value or its profile in advance. The ohmic current density $\langle j_\mathrm{OH} B \rangle$ can be calculated internally with V_loop and neoclassical resistivity used if a user can feed a V_loop profile. Otherwise, it is usual to assume $\mathrm{d}V_\mathrm{loop}/\mathrm{d}\rho = 0$, which means the constant V_loop over the profile, and to adjust the $\langle j_\mathrm{OH} B \rangle$ profile or to put a given shape of the additional current, which mimics the spatial variation of V_loop. In such a case, V_loop is an output.

It is GOTRESS+ that needs to be executed to obtain a solution consistent among the transport quantities such as T_a, S_a, q and the equilibrium. The workflow of GOTRESS+ is demonstrated in figure 1 and that of the original one corresponds to the dotted magenta arrow extending from the "GOTRESS" process symbol to the "converge?" decision symbol in the flowchart. The prescribed density and temperature profiles as well as the coil currents supporting an equilibrium and the settings of heating systems are initially given as input of ACCOME. The iterative calculation inside ACCOME yields a consistent solution between the equilibrium and the current profile with fast-ion effects included. The fast-ion pressure is calculated in ACCOME and its effect is reflected in the equilibrium calculation. In an iterative convergence loop of ACCOME, the NB current drive (NBCD) is computed and updated inside ACCOME, whereas the user can alternatively replace it with the NBCD calculated by OFMC at the previous GOTRESS+ iteration. In this case, the NBCD of OFMC used in ACCOME is fixed during ACCOME calculation due to the numerical heaviness of OFMC calculation.
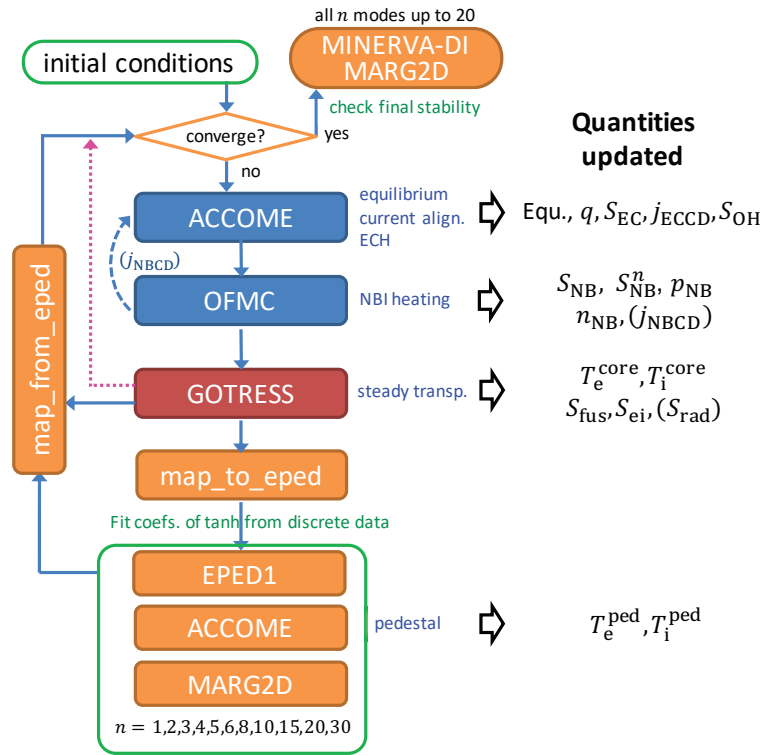
Figure 1. Flowchart of a GOTRESS+ simulation. The workflow in GOTRESS+ is regulated by the Python script and a job scheduler. Whether or not there is a data flow related to the NBCD shown in the dashed arrow from "OFMC" to "ACCOME" depends upon the user's choice. In the original version of GOTRESS+, the temperature profile data is directly transferred from GOTRESS to ACCOME after finishing a GOTRESS simulation and checking the convergence, as shown in the dotted magenta arrow. Main quantities updated in each code, most of which are transferred to other codes for use, are also listed on the right. The quantities with brackets denote the ones that a user can decide to use or not.

This choice is made when the user values the fact that the NBCD calculated by OFMC, which incorporates the finite orbit width effect, is more accurate. With the equilibrium calculated in ACCOME used, OFMC then estimates the heating and current drive profiles by NBI. GOTRESS in turn predicts the temperature profiles, incorporating the equilibrium and the current density profile from ACCOME and the heating profiles and the fast-ion pressure from OFMC. The predicted temperature profiles are transferred to ACCOME, which will be used as input for the next ACCOME calculation. This whole procedure continues until all variables are considered to be sufficiently converged.

An integrated workflow is controlled by a Python script, which controls the execution of all these Fortran codes, the exchange of data between them, and the order of execution, with the aid of a job scheduler.

3. Extension of GOTRESS+ with EPED1

The EPED1 model is a model in which the pedestal height, i.e., the total pressure on top of the pedestal, is determined at the intersection of the peeling-ballooning stability boundary and the semi-empirical pedestal width scaling model that the pedestal width is proportional to the square root of the poloidal beta at the pedestal: $\Delta = 0.076\beta_{p,\mathrm{ped}}^{1/2}$ [1], where $\beta_{p,\mathrm{ped}} = p_{\mathrm{ped}}/(B_p^2/(2\mu_0))$. The numerical factor 0.076 seems to vary device by device and this value was determined based on DIII-D experiments [1]. In JT-60SA plasma predictions, the standard value of 0.076 has usually been used as previously investigated in [43] and will be used in the paper as well. In the original EPED1 model, the MHD stability code ELITE calculates growth rates for $n =$5,6,8,10,15,20 and 30 at each pedestal height to determine the peeling-ballooning stability boundary [1]. Here, n is the toroidal mode number. As the MHD stability code, the in-house EPED1 model exploits MARG2D [44], which realizes a fast stability analysis of ideal external MHD modes from $n = 1$ to high n, a feature which greatly benefits GOTRESS+. A successful benchmark test has confirmed that both ELITE and MARG2D give almost the same growth rates as a function of the toroidal mode number [2]. The MARG2D code has recently incorporates the ion diamagnetic drift effects on the MHD stability, which tends to stabilize the mode [45].

Our EPED1 model mainly consists of five parts: the semi-empirical EPED1 scaling code, MARG2D, ACCOME, the interface programs, and the integrated controller written in Python. The newly implemented parts associated with EPED1 correspond to the orange process symbols in the GOTRESS+ workflow shown in figure 1. In the following, the boundary condition of GOTRESS is in typical set at $\rho = 0.9$, dubbed ρ_b hereafter, indicating that the temperature profiles inside ρ_b are determined by GOTRESS and those outside of it will have to be given by EPED1. GOTRESS runs on the ρ grid, while EPED1 works on the ψ grid, where ψ denotes the normalized poloidal flux. Every time the GOTRESS result is transferred to EPED1, the ψ corresponding to ρ_b, i.e., ψ_b, must be recalculated because the relationship between them differs each time due to the change in an equilibrium.

In our EPED1 model, first the "map_to_eped" interface program is invoked to find the coefficients of the hyperbolic tangent function for the electron density n_e, the electron temperature T_e and the ion temperature T_i by fitting the discrete data given by GOTRESS because EPED1 assumes that the density and temperature profiles are described by the hyperbolic tangent function like

$$f(\psi) = c_0 + c_1 \left[\tanh \frac{2(1 + c_8 - c_2)}{c_3} - \tanh \frac{2(\psi + c_8 - c_2)}{c_3} \right]$$
$$+ c_4 H \left(1 - \frac{\psi}{c_5} \right) \left[1 - \left(\frac{\psi}{c_5} \right)^{c_6} \right]^{c_7}, \tag{9}$$

where f denotes an arbitrary quantity like the density and the temperature as a function of ψ and H is the Heaviside step function. Here, c_0 denotes f at $\psi = 1$, c_1 is the scale factor related to the pedestal height, c_2 denotes the ψ at the center of the pedestal,
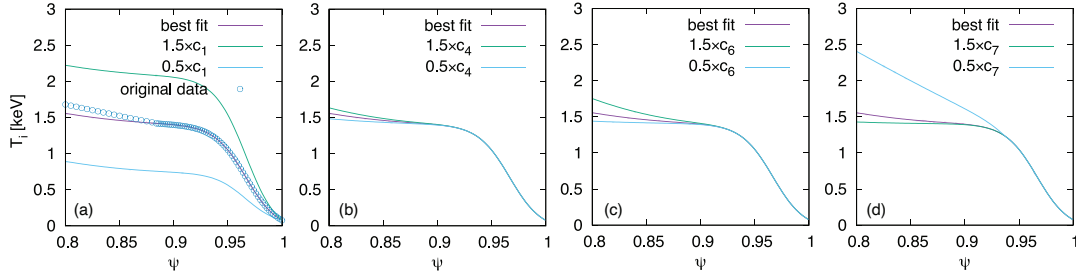
Figure 2. Profiles of the best-fit function of T_i and the functions with (a) c_1, (b) c_4, (c) c_6 and (d) c_7 of the best-fit function multiplied by 1.5 and 0.5, respectively, for pedagogical purposes. The discrete data that was the target of fitting is also displayed as open circles in the subfigure (a).

c_3 is the pedestal width Δ, c_4 is the scale factor related to the core profile, c_5 is the pedestal position ψ_{ped}, c_6 and c_7 are the exponents determining the shape of the core profile, and c_8 denotes the shift of the pedestal ψ_{shift}. This shape of the function f is different from the one in the original model [1] in that c_8 has been introduced, which is the offset of c_2. The coefficients c_2 and c_8 are connected to the other coefficients in the forms: $c_2 = 1 - c_3/2$ and $c_5 = 1 - c_8 - c_3$. The coefficients c_2, c_3 and c_5 are directly linked to the pedestal position and width. The EPED1 model determines the pedestal width c_3, resulting in c_2 and c_5 according to the above relationships. The c_8 is subsequently computed by the above relationship and c_0 is taken from the boundary condition. Therefore, the remaining c_1, c_4, c_6 and c_7 have to be determined such that the resultant hyperbolic tangent function can reproduce the discrete profile data predicted by GOTRESS as much as possible. In general, it is quite difficult to find out these coefficients properly because it is obviously a nonlinear least-squares problem. Fortunately, a genetic algorithm we get used to is a method good at solving this kind of nonlinear least-squares problem [22]. Hence, the genetic algorithm is adopted in the "map_to_eped" program to figure out the coefficients of the function. By weighting, it emphasizes the reproducibility of the profiles by the function in the pedestal region rather than the core.

Here, in order to intuitively understand how the shape of the fitted function changes when the coefficients c_1, c_4, c_6 and c_7 change, graphs are drawn with these coefficients multiplied by 1.5 and 0.5, respectively, from the best-fit case for original discrete data point. For example, the coefficients of the best-fit case of the T_i profile shown in figure 2 (a) are $c_1 \simeq 0.734\,\mathrm{keV}$, $c_4 \simeq 6.63 \times 10^3\,\mathrm{keV}$, $c_6 \simeq 1.42$ and $c_7 \simeq 2.36$, respectively. In this case, the boundary between GOTRESS and EPED1 is located at $\psi_b \simeq 0.882$ and a 10-fold weight was placed on the fitting outside ψ_b, the region in which EPED1 would determine the shape of the pedestal by changing c_2, c_3 and c_5 coefficients afterward. It is clearly found that outside ψ_b the fitted function perfectly reproduced the pedestal profile given by the discrete data. It is straightforward to understand how the function behaves when c_1 changes because it is just the scale factor of the pedestal shape function. Also, the same holds for c_4, which is the scale factor of the core shape function, as shown

in figure 2 (b). The c_6 and c_7 coefficients are the exponents determining the shape of the core profile as well. Figures 2 (c) and (d) demonstrate that they play a role in determining the shape inside the pedestal position, $c_5 \equiv \psi_{\mathrm{ped}} \simeq 0.937$ for this case. Apparently, c_1 affects the shape of the profile over the entire region, while the influence of c_4, c_6 and c_7 is bounded inside ψ_{ped}.

With regard to the density profile, the original EPED1 model assumes $n_{\mathrm{e,sep}} = n_{\mathrm{e,ped}}/4$ [1], where $n_{\mathrm{e,sep}}$ corresponds to c_0, and determines c_1 so as to satisfy this relationship. In this work, the density profile is prescribed without restricting $n_{\mathrm{e,sep}}$ by any relationship with $n_{\mathrm{e,ped}}$. We note that how to determine $n_{\mathrm{e,sep}}$ is very important because it has a great influence on the density gradient and thus the pressure gradient, having an impact on the MHD stability [46]. The density profile prescribed in the simulations shown in the next section can be confirmed in the corresponding figure.

Prior to calling the semi-empirical EPED1 scaling code, some preparation is required in advance. We should recall that the EPED1 modeling handles the total pressure p embedded in the semi-empirical scaling $\Delta = 0.076\beta_{p,\mathrm{ped}}^{1/2}$. It does not distinguish between T_{e} and T_{i}, nor does it distinguish between the density and temperature widths [1]. In the original EPED1 model, T_{i} is taken to be equal to T_{e} due to the minor contribution of T_{i} to the bootstrap current influencing peeling-ballooning stability [1]. Our EPED1 model receives n_{e}, T_{e} and T_{i} separately given as the (9) form. The pedestal width $\Delta(= c_3)$, which is defined to be the average of the n_{e}, T_{e} and T_{i} pedestal widths in ψ space as the original EPED1 model does [1], is evaluated at this point, which is regarded as $\Delta^{(0)}$. ACCOME then constructs an equilibrium based on the coefficients of (9) determined by map_to_eped, which reflects the temperature profiles calculated by GOTRESS.

According to the semi-empirical scaling, the pedestal width $\Delta^{(1)}$ is evaluated at $\psi_{\mathrm{ped}}(= c_5)$, which is given as the average of those for n_{e}, T_{e} and T_{i}. The error is estimated as $\epsilon_\Delta^{(1)} \equiv |\Delta^{(1)} - \Delta^{(0)}|/\Delta^{(0)}$. The results of EPED1 are not considered to converge if $\epsilon_\Delta^{(1)}$ exceeds the designated criterion. In such a case, $\Delta^{(1)}$ is substituted into c_3, and c_2 and c_5 for n_{e}, T_{e} and T_{i} are updated accordingly for the next equilibrium calculation in the EPED1 model. Please note that c_2 and c_3 are common for the density and temperatures because $\Delta(= c_3)$ was averaged out and c_2 is solely dependent on c_3; however, it is possible that c_8 differs for the density and the temperatures, respectively, and c_5 could therefore be different between them. This process is repeated until $\epsilon_\Delta^{(n)}$ at n-th iteration is below the designated criterion.

After convergence, MARG2D scrutinizes its MHD stability for $n =$1-6,8,10,15,20 and 30 with the stabilizing effects by the ion diamagnetic drift taken into account. The integrated controller then scales up/down the temperatures if the plasma is MHD stable/unstable. This whole process corresponds to step 0 in figure 3, which illustrates how our EPED1 model determines the pedestal height in a marginally stable state. In the example shown in figure 3, the plasma is determined to be MHD stable at step 0. Therefore, the temperature profiles are raised to a certain width and then the same thing as was done at step 0 is repeated at step 1. At step 4, the plasma apparently

becomes unstable. At step 5, where the temperature profiles are lowered by half the width increased at the previous step, the plasma is again judged to be unstable. In such a case, the pedestal height at step 3 is considered to be a marginally-stable solution.
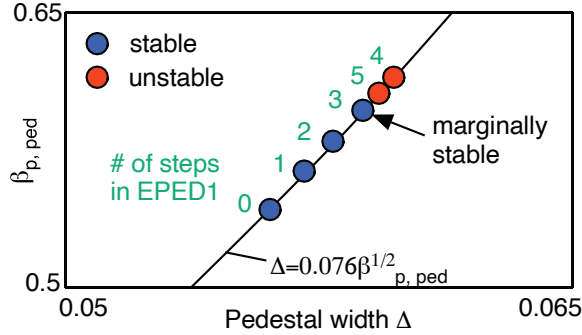

Figure 3. Procedure in our EPED1 model to determine the pedestal height in a marginally stable state.

Now that the pedestal profiles, which are outside of the boundary of GOTRESS, have been determined, the "map_from_eped" interface program provides the mapping of the profiles between ψ and ρ coordinates and also properly joins the pedestal profile with the discrete profile data predicted by GOTRESS. For example, it is assumed that EPED1 predicted temperatures at ρ_b, dubbed $T_a^{\mathrm{E}}(\rho_b)$, different from those before EPED1 had been invoked, $T_a^{\mathrm{G}}(\rho_b)$. Please note that $T_a^{\mathrm{E}}(\rho_b) = T_a^{\mathrm{E}}(\psi_b)$ and $T_a^{\mathrm{G}}(\rho_b) = T_a^{\mathrm{G}}(\psi_b)$. To resolve the gap between $T_a^{\mathrm{E}}(\rho_b)$ and $T_a^{\mathrm{G}}(\rho_b)$ and at the same time keep the core temperature almost unchanged, $(T_a^{\mathrm{E}}(\rho_b)/T_a^{\mathrm{G}}(\rho_b) - 1)(\psi/\psi_b) + 1$ is multiplied by the temperature profiles inside ψ_b in GOTRESS. It does smoothly connect both profiles at ψ_b, while it hardly affects the temperature deep in the core region, making it possible to minimize the change in the β value in this connection process. If the profiles sufficiently converge after the iterations of GOTRESS+ are repeated, $T_a^{\mathrm{E}}(\rho_b)$ and $T_a^{\mathrm{G}}(\rho_b)$ will be nearly identical and thus this connection function will be unity over the entire region. The "map_from_eped" then returns the profiles of n_e, T_e and T_i in the discrete data form back to ACCOME as input for next iteration of the GOTRESS+ outermost iteration loop.

The MHD stability of the final plasma is usually examined by MARG2D or the MINERVA-DI code [47] as post-processing. The inputs for GOTRESS+ should be provided in the TOML format ‡. Apart from it, the Fortran namelist files for GOTRESS and ACCOME are required, but that for OFMC is not needed because ACCOME automatically generates it and then launches OFMC after the end of ACCOME calculation. The workflow in GOTRESS+ is regulated by the Python script, called `autorun.py`, and a job scheduler. GOTRESS+ is basically designed to work without the user being aware of the differences in computer systems on which GOTRESS+ runs. Therefore, for codes that require parallel computation or take a long time to calculate,

‡ https://toml.io/en/

an execution script of the job scheduler installed on the computer system the user is using is automatically issued by the controller of GOTRESS+ and the job is submitted via the scheduler.

Currently, GOTRESS+ is able to run on our PC cluster with 720 cores of Intel Xeon E5-2630 v4 2.20 GHz and on JFRS-1 supercomputer. In the case that GOTRESS+ runs on the PC cluster, it just takes a few minutes for each ACCOME calculation with a single core and 25 minutes for each OFMC run with 120 cores. Depending upon how good the convergence is and a turbulent transport model used, GOTRESS generally takes several minutes with 26 cores. The number of cores and computation time required by each component of the EPED1 model varies. Only MARG2D is parallelized and uses 40 cores. Overall, it takes about 50 minutes to complete the entire EPED1 calculation, depending upon the number of iterations required in EPED1 and how many toroidal mode numbers MARG2D should compute. In total, each GOTRESS+ iteration requires about 80 minutes. Although the computation time for a GOTRESS+ simulation strongly depends upon the number of iterations specified, the simulation for JT-60SA #4-1, which will be presented in the next section, took about 12.5 hours for 10 iterations on our PC cluster.

4. Development of operation scenarios in JT-60SA

In this section, the two operation scenarios for JT-60SA scrutinized by GOTRESS+ will be presented. One is the scenario #4-1, which is called an ITER-like inductive operation scenario, and the other is the scenario #5-1, which is called a high β_{N} fully non-inductively current driven operation scenario [34]. For both scenarios, the preliminary assessment with ACCOME has already been performed by tailoring the kinetic profiles as inputs to ACCOME by hand. The MHD stability and the pedestal feasibility of the resulting prescribed profiles were examined a posteriori using MARG2D and EPED1 to validate the developed scenario. Hence, it is worth performing integrated transport simulations taking advantage of these kinetic profiles as initial profiles for the following GOTRESS+ simulations together with EPED1 self-consistently to assess the comprehensive feasibility of the scenarios. In this work, the density profiles are prescribed as an initial condition and fixed throughout the simulation, even though the shape of the profile would be slightly altered along with the change in an equilibrium in the course of convergence. The Greenwald density fraction f_{GW}, which is one of the target dimensionless parameters, is therefore a designated parameter in the following simulations. The actual shape of the vacuum vessel and the stabilizing plates of JT-60SA, which can be consulted in Fig. 4-2 of [34], is taken into account in evaluating the MHD stability, while resistivity in the conducting wall is currently neglected.

4.1. ITER-like inductive operation scenario

The plasma envisioned in this scenario is an inductive plasma with an ITER-like-shape single-null equilibrium for the purpose of exploring the ITER-relevant plasma regimes
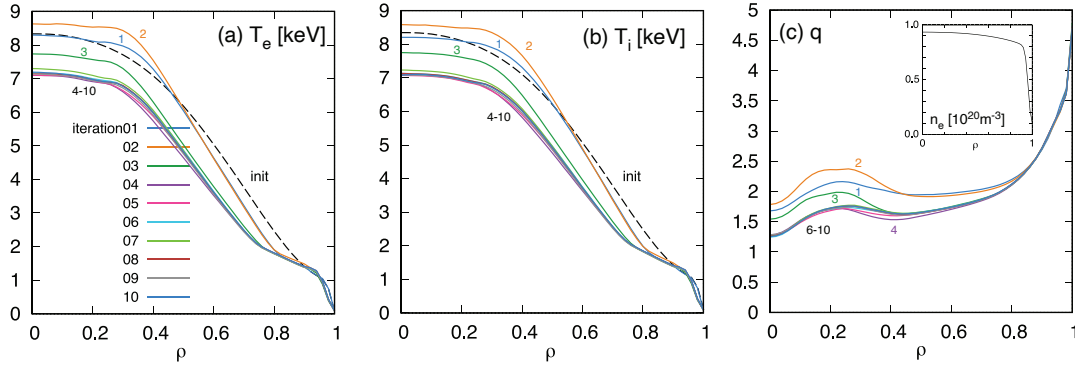
Figure 4. Converging process of (a) T_e, (b) T_i and (c) q profiles in the GOTRESS+ simulation for JT-60SA operation scenario #4-1. The digits in each subfigure denote the iteration number for visibility. The broken lines shown in subfigures (a) and (b) denote the initial profiles. The plot embedded in the subfigure (c) is the n_e profile at tenth iteration.

in terms of the dimensionless parameters. The designated plasma current and toroidal magnetic field are $I_p = 4.6\,\mathrm{MA}$ and $B_T = 2.28\,\mathrm{T}$, respectively [34]. The anticipated equilibrium parameters are: the major radius $R = 2.93\,\mathrm{m}$, $a = 1.14\,\mathrm{m}$, elongation $\kappa = 1.81$, triangularity $\delta = 0.41$, the safety factor at 95% surface $q_{95} = 3.2$ and $V = 122\,\mathrm{m}^3$. The shape factor $S = q_{95}I_p/(aB_T)$, which is the ratio of the safety factors in actual and cylindrical geometry, would be 5.7. The target dimensionless parameters are: $\beta_N = 2.8$, $H_H = 1.1$, $f_{BS} = 0.3$ and $f_{GW} = 0.8$. In the typical scenario of #4-1, NBI of 34 MW is applied to heat a plasma without ECH. The GOTRESS boundary is set at $\rho_b = 0.9$, which is the interface between GOTRESS and EPED1 calculations.

Figure 4 exhibits the converging process of T_e, T_i and q profiles in the GOTRESS+ simulation. The final profile of the electron density n_e is also plotted in figure 4 (c). Not only the temperature profiles but also the q profile are found to be nearly converged after five iterations. In the first few iterations, the q profiles are to some extent reversed in the core region, giving rise to rather strong ITB formation at that moment. At the fifth iteration, the almost flat magnetic shear profile is formed and subsequently the ITB becomes moderate. Afterward, the profiles are almost unchanged over iterations. Focusing on the edge pedestal region, it is found in figure 5 that the shape of the pedestal or the top of the pedestal is somewhat changed iteration by iteration in the first three iterations and after that it is nearly unvaried. It can be seen that in this simulation the change in the pedestal over iterations is small and does not have much influence on convergence. Instead, the strong tie between the ITB formation and the magnetic shear profile affects the converging process. The CDBM model used in the simulation predicts the reduced turbulent heat transport when $s - \alpha$ is almost zero or negative [48], where α is the normalized pressure gradient including the fast-ion pressure. The subsequent steep pressure gradient of the ITB drives the bootstrap current further, facilitating the negative magnetic shear around the peak of the pressure gradient and thus fortifying the ITB. This positive feedback is prone to make the convergence
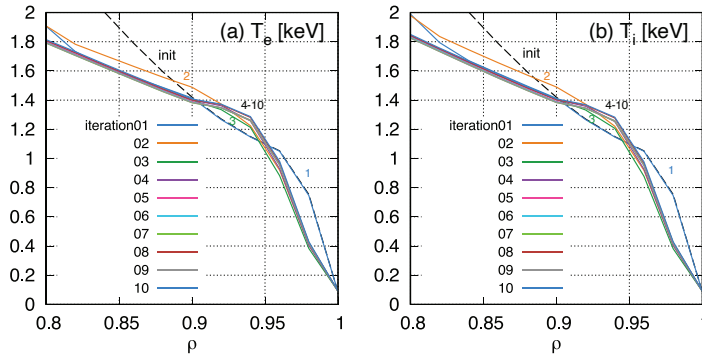
Figure 5. Closeup figures of (a) T_e and (b) T_i near the pedestal. Note that the pedestal profile at initial is identical to that at first iteration because these figures were plotted based on the result of GOTRESS before calling EPED1.

difficult. As apparent in figure 4, GOTRESS+ has succeeded in bringing into a state of convergence within several iterations. The NB heating profiles do not change much. The heating mechanism due to NBI is the energy transfer through collisions between fast neutrals and plasma particles and thus the heat deposition profile is mainly influenced by the density and the temperature of a plasma if an equilibrium does not significantly change. In this simulation, therefore, the change in the NB heat deposition profiles is almost solely dependent on the change in the temperature profiles. In addition, the collisional energy equipartition always functions between electrons and ions, which takes part in the heating mechanisms.

Finally, the converged plasma, which has been confirmed to be MHD stable for $n = 1\text{-}20$ over the entire profile, has $\beta_N = 2.67$, $H_H = 1.06$ and $f_{BS} = 0.23$. These values are slightly less than the aforementioned target values. The β_N on top of the pedestal, $\beta_{N,ped}$, reaches 0.766. Even though it is difficult to directly compare the value without matching the density and temperature at the plasma surface, it is slightly higher or comparable to the previous study for the operation scenarios of ITER [1, 46]. The equilibrium obtained in this simulation is shown in figure 6. The equilibrium parameters are: $R = 2.94\,\mathrm{m}$, $a = 1.14\,\mathrm{m}$, $\kappa = 1.83$, $\delta = 0.490$, $q_{95} = 3.28$, $V = 122\,\mathrm{m}^3$ and $S = 5.83$.

It could be interesting to see what happens if ECH is additionally applied. The dual-frequency-type gyrotrons of $110\,\mathrm{GHz}$ and $138\,\mathrm{GHz}$, the latter frequency of which corresponds to the toroidal magnetic field of $2.25\,\mathrm{T}$, are equipped in JT-60SA. The simulation was performed by adding $2\,\mathrm{MW}$ ECH on the previous case. The position of the ECH absorption depends upon the antenna angle setting, but in this case it was set such that the peak of the heating was around $\rho = 0.42$ and electron cyclotron current drive (ECCD) was almost zero. The converging process can be seen in figure 7, indicating that it takes one extra iteration for the variation of the profiles to converge as compared to the previous case without ECH, but they converge well within 10 iterations. The equilibrium parameters are virtually the same as those in the previous case without ECH. The temperatures at the magnetic axis is higher by about $1\,\mathrm{keV}$, resulting in the better performance of $\beta_N = 2.93$, $H_H = 1.14$ and $f_{BS} = 0.25$. These values meet the
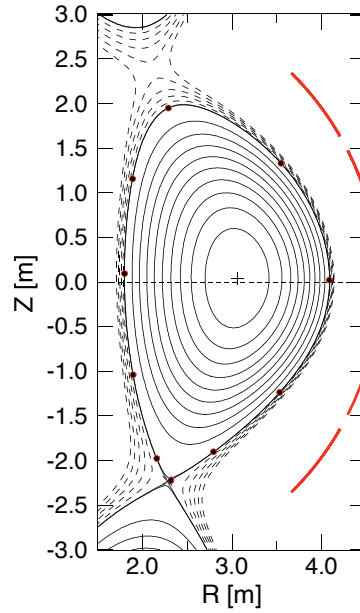
Figure 6. The equilibrium in the GOTRESS+ simulation for JT-60SA operation scenario #4-1. The red arcs express the stabilizing plates.
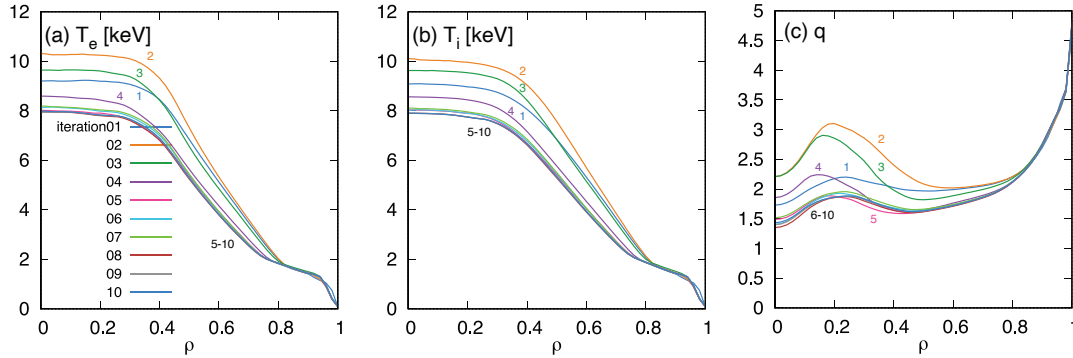

Figure 7. Converging process of (a) T_e, (b) T_i and (c) q profiles in the GOTRESS+ simulation with $138\,$GHz ECH of $2\,$MW. The digits in each subfigure denote the iteration number for visibility.

target ones except for f_{BS}. Based on the fact that the bootstrap current can be varied by manipulating the shape of the density profile, the development of the plasma that meets all the target values is in progress. Finally note that $\beta_{N,ped} = 0.778$, which is almost equivalent to the case without ECH, meaning that the better plasma performance does not stem from the pedestal performance but from the improvement of core confinement.

4.2. High β_N fully non-inductively current driven operation scenario

The plasma envisioned in this scenario is a high β_N fully non-inductive plasma, which is designed to have $I_p = 2.3\,$MA, $B_T = 1.72\,$T, $R = 2.97\,$m, $a = 1.11\,$m, $\kappa = 1.90$, $\delta = 0.47$, $q_{95} = 5.8$, $V = 124\,$m^3 and $S = 7.0$ [34]. The target parameters are: $\beta_N = 4.3$, $H_H = 1.3$,
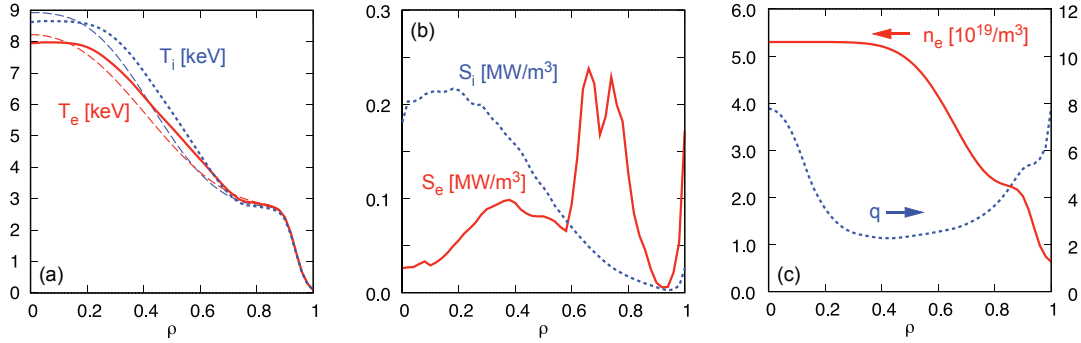
Figure 8. Final state of (a) T_e and T_i, (b) S_e and S_i, and (c) n_e and q profiles in the GOTRESS+ simulation for JT-60SA operation scenario #5-1. S_e and S_i denote the auxiliary heat deposition for electrons and ions. The initial profiles are also plotted in the subfigure (a) as the thin broken lines just for comparison.

$f_{BS} = 0.68$ and $f_{GW} = 0.85$. Unlike the previous one, this scenario is prepared with the goal of developing the DEMO-relevant plasma regimes in a steady state without the inductive current. Self-consistent scenario development would be quite difficult, as all the current must be driven by the current provoked by auxiliary heating systems and the bootstrap current, and the MHD stability must also be met at the same time. The plasma that meets the above conditions ought to naturally have a negative magnetic shear and thus a steep pressure gradient due to the confinement improvement, which generates a significant amount of the bootstrap current. The amount of the current that cannot be covered by the bootstrap current must be fulfilled by the driven current, especially from NBI, which is the main current driver in JT-60SA. However, from the aspect that $f_{GW} = 0.85$ must be met, the steep density pedestal should be sustained, which prevents fast neutrals from penetrating into the core region. It results in the shortage of NBCD in the core. The high plasma density makes it difficult to control the current density profile.

Accordingly, our strategy to attain the fully current driven scenario is to manipulate the density profile and to adjust the number of NBI units activated. The strong beam pressure distributed around $\rho = 0.4$ is likely to destabilize the MHD modes together with the thermal plasma density. It is important to decouple the region of the steep beam pressure gradient and that of the steep thermal plasma density gradient, while keeping f_{GW} at the target value. The GOTRESS boundary is set at $\rho_b = 0.8$ for this simulation. As a result, it has been finally found out in figure 8 (c) that the double barrier structure of the density profile, where the ITB locates around $\rho = 0.4$ and the pedestal resides around $\rho = 0.9$, with $P_{NBI} = 16.01\,\mathrm{MW}$ and $P_{ECH} = 7\,\mathrm{MW}$ with $110\,\mathrm{GHz}$ applied, makes it possible to bring us the scenario that satisfies all the target values and the MHD stability at the same time. Figure 8 (b) indicates that the negative-ion-source NBI generates the high energy fast neutrals that can penetrate the density barriers into the core region. Also, ECH creates the peak of the electron heating around $\rho = 0.7$, which bolsters the higher pedestal temperature. The reversed magnetic shear in
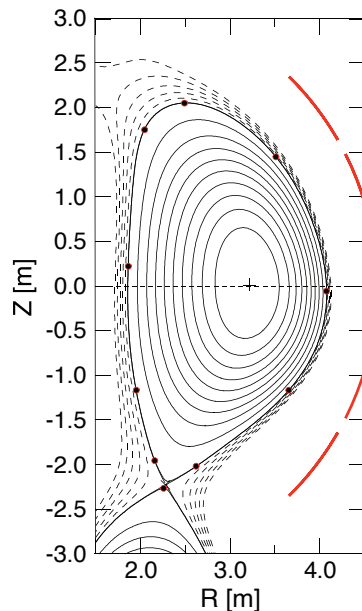
Figure 9. The equilibrium in the GOTRESS+ simulation for JT-60SA operation scenario #5-1.

the core region suppresses turbulent transport and forms the ITBs. Despite the strong ion heating in the core, the collisional energy equipartition attempts to equilibrate the ion temperature with the electron's and thus the resultant temperature profiles do not differ much between electrons and ions, as shown in figure 8 (a). The non-dimensional parameters in question are $\beta_N = 4.33$, $H_H = 1.61$ and $f_{BS} = 0.676$, respectively, which satisfy the target parameters.

The equilibrium at the final state is displayed in figure 9. The parameters associated with the equilibrium results in $R = 2.97\,\text{m}$, $a = 1.11\,\text{m}$, $\kappa = 1.91$, $\delta = 0.503$, $q_{95} = 5.44$, $V = 124\,\text{m}^3$ and $S = 6.59$. The shape factor S is slightly less than the target value by the amount that q_{95} is short of its target value. Still, the high S makes it possible to access the high β_N regime that we achieved in this simulation. The passive stabilizing plates inserted inside the vacuum vessel of course play a critical role in exceeding the no-wall limit. Actually $\beta_{N,\text{ped}} = 1.06$ in this case, comparable to that in the Super H-mode regime plasma [10, 49]. Such high β value could not have been obtained without these stabilizing effects.

As mentioned in section 1, this scenario has already been investigated in the previous work [35, 36]. In the work, the pedestal models different from our modeling have been exploited, where the EPED1 empirical scaling has been employed to estimate the pedestal width only or to get a pedestal prediction double-checked. Garzotti et al have carried out integrated simulations using multiple integrated codes with CDBM as CRONOS, JINTRAC and TOPICS to predict the electron and ion temperature profiles for the reduced power case of 24 MW corresponding to the case in question [36]. In the simulations the density profile was fixed, whereas its shape was quite different from

that used here. The β_N values ranged from 3.5 to 3.9, smaller than the reference value of 4.3. The difference between their results and ours was probably due to the fact that their pedestal heights were lower and the positions of ITBs were more near the core. Romanelli et al have made use of JINTRAC to predict the performance for both full-power and reduced power cases with an emphasis on accounting for SOL/divertor conditions [35]. For the reduced power case, the resultant values of β_N were 3.6 or 3.7, while those of f_{GW} ranged between 0.94 to 0.97, which are much larger than the reference value of 0.85. These results imply the influence of the density profile and the coupling with a SOL/divertor model on performance predictions.

5. Summary and perspectives

The integrated model GOTRESS+, which includes the iterative transport solver GOTRESS at its core, ACCOME and OFMC, has been extended such that it incorporates the in-house EPED1 model with the MHD stability code MARG2D used. One of the advantages of the in-house EPED1 model is the use of MARG2D, which can apply to low to high-n modes. The self-built Python workflow tool makes many Fortran codes, Python scripts and the job scheduler coalesce into one integrated model GOTRESS+. GOTRESS+ is now capable of calculating the plasma profiles from the plasma boundary to the axis self-consistently.

Scrutinizing the validity of JT-60SA operation scenarios provided in the JT-60SA Research Plan [34], it has been used to predict the plasma profiles consistent with heatings, equilibria and the MHD stability and to With the CDBM model used, GOTRESS+ successfully validated the two JT-60SA operation scenarios almost satisfying the pre-defined target values for dimensionless parameters such as β_N, H_H and f_{BS}. Even for the high β_N fully non-inductive plasma, which tends to be MHD unstable, GOTRESS+ could figure out the solution.

In this work, the peak performance in the flat-top phase has been assessed. Especially, it was shown that the double barriers can go together in an MHD stable manner in the scenario #5-1. If the plasma in this scenario will actually exist in a future JT-60SA experiment, edge localized modes (ELMs) may possibly occur. According to the previous experimental findings, it was observed in some JT-60U discharges that the ITB was sustained during the ELMy H-mode, while type-I ELMs typically degrade the ITB and sometimes terminate it unless some measures are taken [50]. It will be important to examine the feasibility of the operation scenario through a time-dependent simulation that considers ELMs.

Resistivity of the conducting wall has been neglected in the simulations. The resistive wall modes (RWMs) would be destabilized if finite resistivity were taken into account [51]. It is well known that toroidal rotation and/or its profile shear is likely to stabilize the RWMs and so are the resonances between the modes and the precessional drifts of thermal [52] and fast particles [53]. Extending GOTRESS+ to compute the evolution of toroidal rotation is an import task for comprehensive predictions.

Some more issues remain to be resolved. OFMC and EPED1, which are time-consuming parts in the GOTRESS+ workflow, may be speeded up by replacing them with their surrogate models, respectively, as demonstrated in the previous study [54,55]. The sensitivity study of the settings like initial conditions is quite interesting and should be performed in the future, as bifurcations in transport have been observed in DIII-D high β plasmas [56].

We ought to assess the feasibility of the density profiles that were given and fixed in this work. Dealing with particle transport to determine the density profiles is of great difficulty as compared to heat transport in the integrated simulation. The understanding of turbulent particle transport is still less mature than heat transport even in the core region, so it is even less near the edge region with the pedestal. To make matters worse, despite the critical importance of estimating the particle source and sink for particle transport, it is very difficult to make a reasonable estimate of it in the edge region, where neutrals play a salient role.

This problem is also linked to the scrape-off-layer plasma physics, which is a key component to determine the boundary condition at the plasma surface, as well as plasma-wall interactions [35,57]. The values at the plasma surface used in this work have been prescribed consistent with the previous work [57,58]. It has been reported that EPED1 predictions with the SOLPS boundary conditions in ITER are different from those with the fixed values used [59], obviously suggesting the importance of the coupling between the core and scrape-off-layer plasma modelings, as also shown in more recent work [18,35]. Implementing particle transport physics and integrating a scrape-off-layer plasma model would therefore be an unavoidable path to make use of GOTRESS+ in reliable predictions for ITER and DEMO plasmas. The mechanisms of the alpha heating and the dilution of helium ash produced by fusion reactions have already been included in GOTRESS, both of which are indispensable to calculations for ITER and DEMO. Also, simulations will plan to be performed with more sophisticated transport models such as TGLF.

Acknowledgments

References

[1] Snyder P. et al. 2009 *Phys. Plasmas* **16** 056118

[2] Snyder P. *et al.* 2011 *Nucl. Fusion* **51** 103016

[3] Merle A., Sauter O. and Medvedev S.Yu. 2017 *Plasma Phys. Control. Fusion* **59** 104001

[4] Saarelma S. *et al.* 2018 *Plasma Phys. Control. Fusion* **60** 014042

[5] Li K. *et al.* 2020 *Plasma Phys. Control. Fusion* **62** 115007

[6] Groebner R.J. *et al.* 2013 *Nucl. Fusion* **53** 093024

[7] Beurskens M.N.A. *et al.* 2014 *Nucl. Fusion* **54** 043001

[8] Aiba N. and Urano H. 2014 *Nucl. Fusion* **54** 114007

[9] Urano H. 2014 *Nucl. Fusion* **54** 116001

[10] Snyder P.B. *et al.* . 2015 *Nucl. Fusion* **55** 083026

[11] Sheikh U.A. *et al.* 2019 *Plasma Phys. Control. Fusion* **61** 014002

[12] Frassinetti L. *et al.* 2017 *Nucl. Fusion* **57** 016012

[13] Frassinetti L. *et al.* 2019 *Nucl. Fusion* **59** 076038

[14] Na Y.S. *et al.* 2020 *Nucl. Fusion* **60** 086006

[15] Meneghini O. *et al.* 2016 *Phys. Plasmas* **23** 042507

[16] Kim S.H. *et al.* 2016 *Nucl. Fusion* **56** 126002

[17] Polevoi A.R. *et al.* 2017 *Nucl. Fusion* **57** 022014

[18] Luda T. *et al.* 2020 *Nucl. Fusion* **60** 036023

[19] Koechl F. *et al.* 2017 *Nucl. Fusion* **57** 086023

[20] Park J.M. *et al.* 2018 *Phys. Plasmas* **25** 012506

[21] Poli F.M. *et al.* 2018 *Nucl. Fusion* **58** 016007

[22] Honda M. 2018 *Comput. Phys. Commun.* **231** 94-106

[23] Honda M. and Narita E. 2019 *Phys. Plasmas* **26** 102307

[24] Candy J. *et al.* 2009 *Phys. Plasmas* **16** 060704

[25] Staebler G.M., Kinsey J.E. and Waltz R.E. 2005 *Phys. Plasmas* **12** 102508

[26] Staebler G.M., Kinsey J.E. and Waltz R.E. 2007 *Phys. Plasmas* **14** 055909

[27] Honda M., Narita E. and Hayashi N. 2019 *Proc. 46th EPS Conf. on Plasma Physics (Milan, 2019)* vol 43C (Geneva: European Physical Society) P1.1083

[28] Tani K., Azumi M., Devoto R.S. 1992 *J. Comput. Phys.* **98** 332-341

[29] Tani K. *et al.* 1981 *J. Phys. Soc. Jpn.* **50** 1726-1737

[30] Hayashi N. and JT-60 Team 2010 *Phys. Plasmas* **17** 056112

[31] Honda M. 2010 *Comput. Phys. Commun.* **181** 1490-1500

[32] Honda M. and Fukuyama A. 2006 *Nucl. Fusion* **46** 580-593

[33] Hayashi N. *et al.* 2017 *Nucl. Fusion* **57** 126037

[34] JT-60SA Research Unit, 'JT-60SA Research Plan. Research Objectives and Strategy', v4.0, Sept. 2018 (http://jt60sa.org/pdfs/JT-60SA_Res_Plan.pdf)

[35] Romanelli M. *et al.* 2017 *Nucl. Fusion* **57** 116010

[36] Garzotti L. *et al.* 2018 *Nucl. Fusion* **58** 026029

[37] Kikuchi M. and Azumi M. 1995 *Plasma Phys. Control. Fusion* **37** 1215

[38] Hamamatsu K. *et al.* 2007 *Plasma Phys. Control. Fusion* **49** 1955

[39] Richardson A.S. 2019 NRL plasma formulary (https://www.nrl.navy.mil/Portals/38/PDFFiles/NRL_Formulary_2019.pdf)

[40] Summers H.P. 1994 *Atomic Data and Analysis Structure Users Manual Report No. JET-IR 06*, JET Joint Undertaking, Abingdon

[41] Tamor S. 1981 *SAIC Report* SAI-023-81-110LJ/LAPS-71

[42] Honda M., Kikuchi M. and Azumi M. 2012 *Nucl. Fusion* **52** 023021

[43] Aiba N. *et al.* 2018 *Plasma Phys. Control. Fusion* **60** 014032

[44] Aiba N. *et al.* 2006 *Comput. Phys. Commun.* **175** 269-289

[45] Aiba N. *et al.* 2017 *Nucl. Fusion* **57** 022011

[46] Medvedev S.Yu. *et al.* 2016 *Plasma Phys. Rep.* **42** 472

[47] Aiba N. 2016 *Plasma Phys. Control. Fusion* **58** 045020

[48] Fukuyama A. *et al.* 1995 *Plasma Phys. Control. Fusion* **37** 611

[49] Knolker M. *et al.* 2020 *Phys. Plasmas* **27** 102506

[50] Fujita T. 2002 *Plasma Phys. Control. Fusion* **44** A19

[51] Pigatto L. *et al.* 2019 *Nucl. Fusion* **59** 106028

[52] Hu B. and Betti R. 2004 *Phys. Rev. Lett.* **93** 105002

[53] Hao G.Z. *et al.* 2011 *Phys. Rev. Lett.* **107** 015001

[54] Meneghini O. *et al.* 2017 *Nucl. Fusion* **57** 086034

[55] Boyer M.D. *et al.* 2019 *Nucl. Fusion* **59** 056008

[56] Buttery R.J. *et al.* 2019 *J. Fusion Energy* **38** 72

[57] Hayashi N. *et al.* 2018 *Nucl. Fusion* **58** 066001

[58] Yamoto S. *et al.* 2020 *Plasma Phys. Control. Fusion* **62** 045006

[59] Polevoi A.R. *et al.* 2015 *Nucl. Fusion* **55** 063019